Malay Kumar Kundu
Durga Prasad Mohapatra
Amit Konar
Aruna Chakraborty
Editors

Advanced Computing, Networking and Informatics - Volume 2

Wireless Networks and Security
Proceedings of the Second International Conference on Advanced Computing, Networking and Informatics (ICACNI-2014)

SMART INNOVATION, SYSTEMS AND TECHNOLOGIES 28

KES

Springer

Smart Innovation, Systems and Technologies

Volume 28

For further volumes:
http://www.springer.com/series/8767

About this Series

The Smart Innovation, Systems and Technologies book series encompasses the topics of knowledge, intelligence, innovation and sustainability. The aim of the series is to make available a platform for the publication of books on all aspects of single and multi-disciplinary research on these themes in order to make the latest results available in a readily-accessible form. Volumes on interdisciplinary research combining two or more of these areas is particularly sought.

The series covers systems and paradigms that employ knowledge and intelligence in a broad sense. Its scope is systems having embedded knowledge and intelligence, which may be applied to the solution of world problems in industry, the environment and the community. It also focusses on the knowledge-transfer methodologies and innovation strategies employed to make this happen effectively. The combination of intelligent systems tools and a broad range of applications introduces a need for a synergy of disciplines from science, technology, business and the humanities. The series will include conference proceedings, edited collections, monographs, handbooks, reference books, and other relevant types of book in areas of science and technology where smart systems and technologies can offer innovative solutions.

High quality content is an essential feature for all book proposals accepted for the series. It is expected that editors of all accepted volumes will ensure that contributions are subjected to an appropriate level of reviewing process and adhere to KES quality principles.

Malay Kumar Kundu · Durga Prasad Mohapatra
Amit Konar · Aruna Chakraborty
Editors

Advanced Computing, Networking and Informatics - Volume 2

Wireless Networks and Security Proceedings of the Second International Conference on Advanced Computing, Networking and Informatics (ICACNI-2014)

Springer

Editors
Malay Kumar Kundu
Indian Statistical Institute
Machine Intelligence Unit
Kolkata
India

Durga Prasad Mohapatra
Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela
India

Amit Konar
Department of Electronics and Tele-Communication Engineering
Artificial Intelligence Laboratory
Jadavpur University
Kolkata
India

Aruna Chakraborty
Department of Computer Science and Engineering
St. Thomas' College of Engineering and Technology
West Bengal
India

ISSN 2190-3018 ISSN 2190-3026 (electronic)
ISBN 978-3-319-07349-1 ISBN 978-3-319-07350-7 (eBook)
DOI 10.1007/978-3-319-07350-7
Springer Cham Heidelberg New York Dordrecht London

Library of Congress Control Number: 2014940383

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Foreword

The present volume is an outcome, in the form of proceedings, of the 2nd International Conference on Advanced Computing, Networking and Informatics, St. Thomas' College of Engineering and Technology, Kolkata, India, June 24–26, 2014. As the name of the conference implies, the articles included herein cover a wide span of disciplines ranging, say, from pattern recognition, machine learning, image processing, data mining and knowledge discovery, soft computing, distributed computing, cloud computing, parallel and distributed networks, optical communication, wireless sensor networks, routing protocol and architecture to data privacy preserving, cryptology and data security, and internet computing. Each discipline, itself, has its own challenging problems and issues. Some of them are relatively more matured and advanced in theories with several proven application domains, while others fall in recent thrust areas. Interestingly, there are several articles, as expected, on symbiotic integration of more than one discipline, e.g., in designing intelligent networking and computing systems such as forest fire detection using wireless sensor network, minimizing call routing cost with assigned cell in wireless network, network intrusion detection system, determining load balancing strategy in cloud computing, and side lobe reduction and beam-width control, where the significance of pattern recognition, evolutionary strategy and soft computing has been demonstrated. This kind of interdisciplinary research is likely to grow significantly, and has strong promise in solving real life challenging problems.

The proceedings are logically split in two homogeneous volumes, namely, Advanced Computing and Informatics (vol. 1) and Wireless Networks and Security (vol. 2) with 81 and 67 articles respectively. The volumes fairly represent a state-of-the art of the research mostly being carried out in India in these domains, and are valued-additions to the current era of computing and knowledge mining.

The conference committee, editors, and the publisher deserve congratulations for organizing the event (ICACNI-2014) which is very timely, and bringing out the archival volumes nicely as its output.

Kolkata, April 2014

Sankar K. Pal
Distinguished Scientist and former Director
Indian Statistical Institute

Message from the Honorary General Chair

It gives me great pleasure to introduce the *International Conference on Advanced Computing, Networking and Informatics (ICACNI 2014)* which will be held at St. Thomas' College of Engineering and Technology, Kolkata during June 24–26, 2014. ICACNI is just going to cross its second year, and during this small interval of time it has attracted a large audience. The conference received over 650 submissions of which only 148 papers have been accepted for presentation. I am glad to note that ICACNI involved top researchers from 26 different countries as advisory board members, program committee members and reviewers. It also received papers from 10 different countries.

ICACNI offers an interesting forum for researchers of three apparently diverse disciplines: Advanced Computing, Networking and Informatics, and attempts to focus on engineering applications, covering security, cognitive radio, human-computer interfacing among many others that greatly rely on these cross-disciplinary research outcomes. The accepted papers are categorized into two volumes, of which volume 1 includes all papers on advanced computing and informatics, while volume 2 includes accepted papers on wireless network and security. The volumes will be published by Springer-Verlag.

The conference includes plenary lecture, key-note address and four invited sessions by eminent scientists from top Indian and foreign research/academic institutes. The lectures by these eminent scientists will provide an ideal platform for dissemination of knowledge among researchers, students and practitioners. I take this opportunity to thank all the participants, including the keynote, plenary and invited speakers, reviewers, and the members of different committees in making the event a grand success.

Thanks are also due to the various Universities/Institutes for their active support towards this endeavor, and lastly Springer-Verlag for publishing the proceedings under their prestigious *Smart Innovation, Systems and Technologies (SIST) series*.

Wish the participants an enjoyable and productive stay in Kolkata.

Kolkata, April 2014

Dwijesh Dutta Majumder
Honorary General Chair
ICACNI -2014

Preface

The twenty first century has witnessed a paradigm shift in three major disciplines of knowledge: 1) Advanced/Innovative computing ii) Networking and wireless Communications and iii) informatics. While the first two are complete in themselves by their titles, the last one covers several sub-disciplines involving geo-, bio-, medical and cognitive informatics among many others. Apparently, the above three disciplines of knowledge are complementary and mutually exclusive but their convergence is observed in many real world applications, encompassing cyber-security, internet banking, healthcare, sensor networks, cognitive radio, pervasive computing and many others.

The International Conference on *Advanced Computing, Networking and Informatics* (ICACNI) is aimed at examining the convergence of the above three modern disciplines through interactions among three groups of people. The first group comprises leading international researchers, who have established themselves in one of the above three thrust areas. The plenary, the keynote lecture and the invited talks are organized to disseminate the knowledge of these academic experts among young researchers/practitioners of the respective domain. The invited talks are also expected to inspire young researchers to initiate/orient their research in respective fields. The second group of people comprises Ph.D./research students, working in the cross-disciplinary areas, who might be benefited from the first group and at the same time may help creating interest in the cross-disciplinary research areas among the academic community, including young teachers and practitioners. Lastly, the group comprising undergraduate and master students would be able to test the feasibility of their research through feedback of their oral presentations.

ICACNI is just passing its second birthday. Since its inception, it has attracted a wide audience. This year, for example, the program committee of ICACNI received as many as 646 papers. The acceptance rate is intentionally kept very low to ensure a quality publication by Springer. This year, the program committee accepted only 148 papers from these 646 submitted papers. An accepted paper has essentially received very good recommendation by at least two experts in the respective field.

To maintain a high standard of ICACNI, researchers from top international research laboratories/universities have been included in both the advisory committee and the program committee. The presence of these great personalities has helped the conference to develop its publicity during its infancy and promote it quality through an academic exchange among top researchers and scientific communities.

The conference includes one plenary session, one keynote address and four invited speech sessions. It also includes 3 special sessions and 21 general sessions (altogether 24 sessions) with a structure of 4 parallel sessions over 3 days. To maintain good question-answering sessions and highlight new research results arriving from the sessions, we selected subject experts from specialized domains as session chairs for the conference. ICACNI also involved several persons to nicely organize registration, take care of finance, hospitality of the authors/audience and other supports. To have a closer interaction among the people of the organizing committee, all members of the organizing committee have been selected from St. Thomas' College of Engineering and Technology.

The papers that passed the screening process by at least two reviewers, well-formatted and nicely organized have been considered for publication in the Smart Innovations Systems Technology (SIST) series of Springer. The hard copy proceedings include two volumes, where the first volume is named as *Advanced Computing and Informatics* and the second volume is named as *Wireless Networks and Security*. The two volumes together contain 148 papers of around eight pages each (in Springer LNCS format) and thus the proceedings is expected to have an approximate length of 1184 pages.

The editors gratefully acknowledge the contribution of the authors and the entire program committee without whose active support the proceedings could hardly attain the present standards. They would like to thank the keynote speaker, the plenary speaker, the invited speakers and also the invited session chairs, the organizing chair along with the organizing committee and other delegates for extending their support in various forms to ICACNI-2014. The editors express their deep gratitude to the Honorary General Chair, the General Chair, the Advisory Chair and the Advisory board members for their help and support to ICACNI-2014. The editors are obliged to Prof. Lakhmi C. Jain, the academic series editor of the SIST series, Springer and Dr. Thomas Ditzinger, Senior Editor, Springer, Heidelberg for extending their co-operation in publishing the proceeding in the prestigious SIST series of Springer. They also like to mention the hard efforts of Mr. Indranil Dutta of the Machine Intelligence Unit of ISI Kolkata for the editorial support. The editors also acknowledge the technical support they received from the students of ISI, Kolkata Jadavpur University and also the faculty of NIT Rourkela and St. Thomas' College of Engineering and Technology without which the work could not be completed in right time. Lastly, the editors thank Dr. Sailesh Mukhopadhyay, Prof. Gautam Banerjee and Dr. Subir Chowdhury of St. Thomas' College of Engineering and Technology for their support all the way long to make this conference a success.

Kolkata
April 14, 2014

Malay Kumar Kundu
Durga Prasad Mohapatra
Amit Konar
Aruna Chakraborty

Organization

Advisory Chair

Sankar K. Pal	Distinguished Scientist, Indian Statistical Institute, India

Advisory Board Members

Alex P. James	Nazarbayev University, Republic of Kazakhstan
Anil K. Kaushik	Additional Director, DeitY, Govt. of India, India
Atilla Elci	Aksaray University, Turkey
Atulya K. Nagar	Liverpool Hope University, United Kingdom
Brijesh Verma	Central Queensland University, Australia
Debajyoti Mukhopadhyay	Maharashtra Institute of Technology, India
Debatosh Guha	University of Calcutta, India
George A. Tsihrintzis	University of Piraeus, Greece
Hugo Proenca	University of Beira Interior, Portugal
Jocelyn Chanussot	Grenoble Institute of Technology, France
Kenji Suzuki	The University of Chicago, Chicago
Khalid Saeed	AGH University of Science and Technology, Poland
Klaus David	University of Kassel, Germany
Maode Ma	Nanyang Technological University, Singapore
Massimiliano Rak	Second University of Naples, Italy
Massimo Tistarelli	University of Sassari, Italy
Nishchal K. Verma	Indian Institute of Technology, Kanpur, India
Pascal Lorenz	University of Haute Alsace, France
Phalguni Gupta	Indian Institute of Technology Kanpur, India
Prasant Mohapatra	University of California, USA
Prasenjit Mitra	Pennsylvania State University, USA
Raj Jain	Washington University in St. Louis, USA
Rajesh Siddavatam	Kalinga Institute of Industrial Technology, India

Rajkumar Buyya	The University of Melbourne, Australia
Raouf Boutaba	University of Waterloo, Canada
Sagar Naik	University of Waterloo, Canada
Salvatore Vitabile	University of Palermo, Italy
Sansanee Auephanwiriyakul	Chiang Mai University, Thailand
Subhash Saini	The National Aeronautics and Space Administration (NASA), USA

ICACNI Conference Committee

Chief Patron

Sailesh Mukhopadhyay	St. Thomas' College of Engineering and Technology, Kolkata, India

Patron

Gautam Banerjea	St. Thomas' College of Engineering and Technology, Kolkata, India

Honorary General Chair

Dwijesh Dutta Majumder	Professor Emeritus, Indian Statistical Institute, Kolkata, India Institute of Cybernetics Systems and Information Technology, India Director, Governing Board, World Organization of Systems and Cybernetics (WOSC), Paris

General Chairs

Rajib Sarkar	Central Institute of Technology Raipur, India
Mrithunjoy Bhattacharyya	St. Thomas' College of Engineering and Technology, Kolkata, India

Programme Chairs

Malay Kumar Kundu	Indian Statistical Institute, Kolkata, India
Amit Konar	Jadavpur University, Kolkata, India
Aruna Chakraborty	St. Thomas' College of Engineering and Technology, Kolkata, India

Programme Co-chairs

Asit Kumar Das	Bengal Engineering and Science University, Kolkata, India
Ramjeevan Singh Thakur	Maulana Azad National Institute of Technology, India
Umesh A. Deshpande	Sardar Vallabhbhai National Institute of Technology, India

Organizing Chairs

Ashok K. Turuk	National Institute of Technology Rourkela, Rourkela, India
Rabindranath Ghosh	St. Thomas' College of Engineering and Technology, Kolkata, India

Technical Track Chairs

Joydeb Roychowdhury	Central Mechanical Engineering Research Institute, India
Korra Sathyababu	National Institute of Technology Rourkela, India
Manmath Narayan Sahoo	National Institute of Technology Rourkela, India

Honorary Industrial Chair

G.C. Deka	Ministry of Labour & Employment, Government of India

Industrial Track Chairs

Umesh Chandra Pati	National Institute of Technology Rourkela, India
Bibhudutta Sahoo	National Institute of Technology Rourkela, India

Special Session Chairs

Ashish Agarwal	Boston University, USA
Asutosh Kar	International Institute of Information Technology Bhubaneswar, India
Daya K. Lobiyal	Jawaharlal Nehru University, India
Mahesh Chandra	Birla Institute of Technology, Mesra, India
Mita Nasipuri	Jadavpur University, Kolkata, India
Nandini Mukherjee	Chairman, IEEE Computer Society, Kolkata Chapter Jadavpur University, Kolkata, India
Ram Shringar Rao	Ambedkar Institute of Advanced Communication Technologies & Research, India

Web Chair

Indranil Dutta	Indian Statistical Institute, Kolkata, India

Publication Chair

Sambit Bakshi	National Institute of Technology Rourkela, India

Publicity Chair

Mohammad Ayoub Khan	Center for Development of Advanced Computing, India

Organizing Committee

Amit Kr. Siromoni	St. Thomas' College of Engineering and Technology, Kolkata, India
Anindita Ganguly	St. Thomas' College of Engineering and Technology, Kolkata, India
Arindam Chakravorty	St. Thomas' College of Engineering and Technology, Kolkata, India
Dipak Kumar Kole	St. Thomas' College of Engineering and Technology, Kolkata, India
Prasanta Kumar Sen	St. Thomas' College of Engineering and Technology, Kolkata, India
Ramanath Datta	St. Thomas' College of Engineering and Technology, Kolkata, India
Subarna Bhattacharya	St. Thomas' College of Engineering and Technology, Kolkata, India
Supriya Sengupta	St. Thomas' College of Engineering and Technology, Kolkata, India

Program Committee

Vinay. A	Peoples Education Society Institute of Technology, Bangalore, India
Chunyu Ai	University of South Carolina Upstate, Spartanburg, USA
Rashid Ali	Aligarh Muslim University, Aligarh, India
C.M. Ananda	National Aerospace Laboratories, Bangalore, India
Soumen Bag	International Institution of Information Technology, Bhubaneswar, India
Sanghamitra Bandyopadhyay	Indian Statistical Institute, Kolkata, India
Punam Bedi	University of Delhi, Delhi, India
Dinabandhu Bhandari	Heritage Institute of Technology, Kolkata, India

Paritosh Bhattacharya	National Institute of Technology, Agartala, India
Malay Bhattacharyya	University of Kalyani, Kolkata, India
Sambhunath Biswas	Indian Statistical Institute, Kolkata, India
Darko Brodic	University of Belgrade, Bor, Serbia
Sasthi C. Ghosh	Indian Statistical Institute, Kolkata, India
R.C. Hansdah	Indian Institute of Science, Bangalore, India
Nabendu Chaki	University of Calcutta, Kolkata, India
Goutam Chakraborty	Iwate Prefectural University, Takizawa, Japan
Mihir Chakraborty	Indian Statistical Institute, Kolkata, India
Amita Chatterjee	Jadavpur University, Kolkata, India
Debasis Chaudhuri	Indian Institute of Management, Kolkata, India
Lopamudra Chowdhury	Jadavpur University, Kolkata, India
Nabanita Das	Indian Statistical Institute, Kolkata, India
Soumya Kanti Datta	Institute Eurecom, Sophia Antipolis, France
Rajat De	Indian Statistical Institute, Kolkata, India
Utpal Garain	Indian Statistical Institute, Kolkata, India
Ashis Ghosh	Indian Statistical Institute, Kolkata, India
Mukesh Goswami	Dharmsinh Desai University, Gugrat, India
Yogesh H. Dandawate	Vishwakarma Institute of Information Technology, Pune, India
Biju Issac	Teesside University, Middlesbrough, UK
Lakhmi C. Jain	University of South Australia, Adelaide, Australia
R. Janarthanan	T. J. S. Engineering College, Chennai, India
Boleslaw K. Szymanski	Rensselaer Polytechnic Institute, New York, USA
Tienfuan Kerh	National Pingtung University of Science and Technology, Pingtung, Taiwan
Zahoor Khan	Dalhouise University, Halifax, Canada
Dakshina Ranjan Kisku	Asansole Engineering College, Asansole, India
Sotiris Kotsiantis	University of Patras, Hellas, Greece
Dipak Kr. Kole	St. Thomas' College of Engineering and Technology, Kolkata, India
Aswani Kumar Cherukuri	VIT University, Vellore, India
Swapan Kumar Parui	Indian Statistical Institute, Kolkata, India
Prasanta Kumar Pradhan	St. Thomas' College fo Engineering and Technology, Kolkata, India
B. Narendra Kumar Rao	Sree Vidyanikethan Engineering College, Tirupati, India
Flavio Lombardi	Roma Tre University of Rome, Rome, Italy
Subhamoy Maitra	Indian Statistical Institute, Kolkata, India
Pradipta Maji	Indian Statistical Institute, Kolkata, India
Raghvendra Mall	University of Leuven, Leuven, Belgium
Amiya Kr. Mallick	St. Thomas' College fo Engineering and Technology, Kolkata, India
Debaprasad Mandal	Indian Statistical Institute, Kolkata, India
Ujjwal Maulik	Jadavpur University, Kolkata, India
Pabitra Mitra	Indian Institute of Technology, Kharagpur, India

Azadeh Ghandehari	Islamic Azad University, Tehran, Iran
Ch Aswani Kumar	Vellore Institute of Technology, India
Cristinel Ababei	University at Buffalo, USA
Dilip Singh Sisodia	National Institute of Technology Raipur, India
Jamuna Kanta Sing	Jadavpur University, Kolkata, India
Krishnan Nallaperumal	Sundaranar University, India
Manu Pratap Singh	Dr. B.R. Ambedkar University, Agra, India
Narayan C. Debnath	Winona State University, USA
Naveen Kumar	Indira Gandhi National Open University, India
Nidul Sinha	National Institute of Technology Silchar, India
Sanjay Kumar Soni	Delhi Technological University, India
Sanjoy Das	Galgotias University, India
Subir Chowdhury	St. Thomas' College of Engineering and Technology, Kolkata, India
Syed Rizvi	The Pennsylvania State University, USA
Sushil Kumar	Jawaharlal Nehru University, India
Anupam Sukhla	Indian Institute of Information Technology, Gwalior, India

Additional Reviewers

A.M., Chandrashekhar
Acharya, Anal
Agarwal, Shalabh
B.S., Mahanand
Bandyopadhyay, Oishila
Barpanda, Soubhagya Sankar
Basu, Srinka
Battula, Ramesh Babu
Bhattacharjee, Sourodeep
Bhattacharjee, Subarna
Bhattacharya, Indrajit
Bhattacharya, Nilanjana
Bhattacharyya, Saugat
Bhowmik, Deepayan
Biswal, Pradyut
Biswas, Rajib
Bose, Subrata
Chakrabarti, Prasun
Chakraborty, Debashis
Chakraborty, Jayasree
Chandra, Helen
Chandra, Mahesh
Chatterjee, Aditi
Chatterjee, Sujoy
Chowdhury, Archana
Chowdhury, Manish
Dalai, Asish
Darbari, Manuj
Das, Asit Kumar
Das, Debaprasad
Das, Nachiketa
Das, Sudeb
Datta, Biswajita
Datta, Shreyasi
De, Debashis
Dhabal, Supriya
Dhara, Bibhas Chandra
Duvvuru, Rajesh
Gaidhane, Vilas
Ganguly, Anindita
Garg, Akhil
Ghosh Dastidar, Jayati
Ghosh, Arka
Ghosh, Lidia
Ghosh, Madhumala
Ghosh, Partha
Ghosh, Rabindranath
Ghosh, Soumyadeep
Ghoshal, Ranjit
Goswami, Mukesh
Goyal, Lalit
Gupta, Partha Sarathi
Gupta, Savita
Halder, Amiya
Halder, Santanu
Herrera Lara, Roberto
Jaganathan, Ramkumar
Kakarla, Jagadeesh
Kar, Mahesh
Kar, Reshma
Khasnobish, Anwesha
Kole, Dipak Kumar
Kule, Malay
Kumar, Raghvendra
Lanka, Swathi
Maruthi, Padmaja
Mishra, Dheerendra
Mishra, Manu
Misra, Anuranjan
Mohanty, Ram
Maitra, Subhamoy
Mondal, Jaydeb
Mondal, Tapabrata
Mukherjee, Nabanita

Mukhopadhyay, Debajyoti
Mukhopadhyay, Debapriyay
Munir, Kashif
Nasim Hazarika, Saharriyar Zia
Nasipuri, Mita
Neogy, Sarmistha
Pal, Monalisa
Pal, Tamaltaru
Palodhi, Kanik
Panigrahi, Ranjit
Pati, Soumen Kumar
Patil, Hemprasad
Patra, Braja Gopal
Pattanayak, Sandhya
Paul, Amit
Paul, Partha Sarathi
Phadikar, Amit
Phadikar, Santanu
Poddar, Soumyajit
Prakash, Neeraj
Rakshit, Pratyusha
Raman, Rahul
Ray, Sumanta
Roy, Pranab
Roy, Souvik
Roy, Swapnoneel
Rup, Suvendu
Sadhu, Arup Kumar
Saha Ray, Sanchita
Saha, Anuradha
Saha, Anushri
Saha, Dibakar
Saha, Indrajit
Saha, Sriparna
Sahoo, Manmath N.
Sahu, Beeren
Sanyal, Atri
Sardar, Abdur
Sarkar, Apurba
Sarkar, Dhrubasish
Sarkar, Ushnish
Sen, Sonali
Sen, Soumya
Sethi, Geetika
Sharma, Anuj
Tomar, Namrata
Umapathy, Latha
Upadhyay, Anjana
Wankhade, Kapil

Contents

Vehicular, *Ad hoc* and Sensor Networks

Communication Technologies

Network Routing

Data Hiding and Cryptography

Cloud Computing

Efficient Architecture and Computing

Innovative Technologies and Applications

Errata

Parallel Processing Concept Based Vehicular Bridge Traffic Problem

Debasis Das[1] and Rajiv Misra[2]

[1] Department of Computer Science and Engineering,
NIIT University, Neemrana, Rajasthan-301705, India
[2] Department of Computer Science and Engineering,
Indian Institute of Technology, Patna
Bihar-800013, India
Debasis.Das@niituniversity.in,
rajivm@iitp.ac.in

Abstract. A Cellular Automata (CA) is a computing model of complex System using simple rules. In this paper the problem space is divided into number of cells and each cell can be constituted of one or several final state. Cells are affected by neighbors with the application of simple rule. Cellular Automata are highly parallel and discrete dynamical systems, whose behavior is completely specified in terms of local relation. In this paper CA is applied to solve a bridge traffic control problem. Vehicular travel which demands on the concurrent operations and parallel activities is used to control bridge traffic based on Cellular Automata technique.

Keywords: Cellular Automata, Simple Rule, Cell, Bridge Traffic Problem.

1 Introduction

Due to the rapid development of our economy and society, more emphasis is required on urban traffic research [8]. Within urban traffic research, the formation and dispersion of traffic congestion is one of the important aspects. Transportation research has the goal to optimize transportation flow of people and goods. As the number of road users constantly increases, and resources provided by current infrastructures are limited, intelligent control of traffic will become a very important issue in the future. Optimal control of traffic lights using sophisticated sensors and intelligent optimization algorithms might therefore be very beneficial. Optimization of traffic light switching increases road capacity and traffic flow, and can prevent traffic congestions. In the recent years there were strong attempts to develop a theoretical framework of traffic science among the physics community. Consequentially, a nearly completed description of highway traffic, e.g., the “Three Phase Traffic” theory, was developed. This describes the different traffic states occurring on highways as well as the transitions among them. Also the concepts for modeling vehicular traffic [8], [20] are well developed. Most of the models introduced in the recent years are formulated using the language of cellular automata [1], [18], [19]. Unfortunately, no comparable framework for the description of traffic states in city networks is present.

M.K. Kundu et al. (eds.), *Advanced Computing, Networking and Informatics - Volume 2*,
Smart Innovation, Systems and Technologies 28,
DOI: 10.1007/978-3-319-07350-7_1,

2 Cellular Automata

A cellular automaton [1] is a decentralized computing model providing an excellent platform for performing complex computation with the help of only local information. Researchers, scientists and practitioners from different fields have exploited the CA paradigm of local information, decentralized control and universal computation for modeling different applications [7], [19].CA is an array of sites (cells) where each site is in any one of the permissible states. At each discrete time step (clock cycle) the evolution of a site value depends on some rule (the combinational logic), which is a function of the present state *k* of its neighbors for a k-neighborhood CA. Cellular automata (CA) are a collection of cells such that each adapts one of the finite number of states. Single cells change their states by following a local rule that depends on the environment of the cell. The environment of a cell is usually taken to be a small number of neighboring cells. Fig.1 shows two typical neighborhood [1] options (a) Von Neumann Neighborhood (b) Moore Neighborhood.

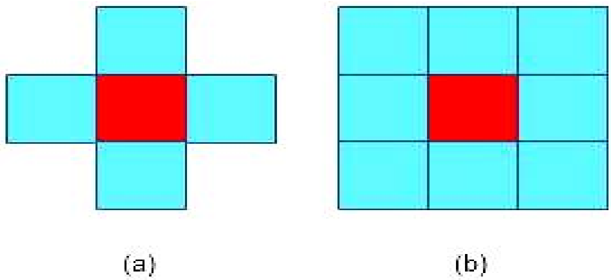


Fig. 1. (a) Von Neumann Neighborhood (b) Moore Neighborhood of the cell is taken to be the cell itself and some or all of the immediately adjacent cells

Typically, a cellular automaton consists of a graph where each node is in a finite state automaton (FSA) or cell. This graph is usually in the form of a two dimensional lattice whose cells evolve according to a global update function applied uniformly over all the cells. As arguments, this update function takes the cell's present state and the states of the cells in its neighborhood as shown in Fig. 2.

3 Related Work

The basic one-dimensional Cellular Automata model for highway traffic flow is based on the rules of the Cellular Automata. The highway is divided into number of cells. Each cell can either be empty or occupied by one car. If a cell is occupied by a car then it is denoted by 1 or otherwise 0.All cars have the same length of cell. In this section we discuss the following CA models [2-10].

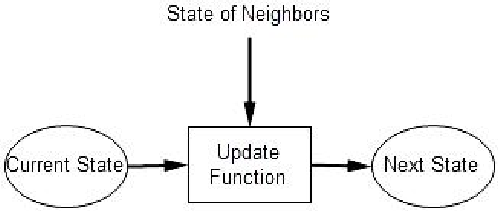


Fig. 2. State Transition Depend on Neighborhood State

3.1 Wolfram's Rule 184 (CA-184) Model

In1983, S. Wolfram's [1], [10] proposed first one dimensional Cellular Automata model with binary state. Here rule 184 is used for traffic flow so, it is called Wolfram's rule 184.For the CA-184 model [2] , we have the following two rules:

(R1) Acceleration and Braking

$v_i(t) \leftarrow \min\{g_{si}(t-1),1\}$

(R2) Vehicle Movement

$x_i(t) \leftarrow x_i(t-1) + v_i(t)$

Rule 1 (R1) is the speed of ith vehicle; this is the current updated configuration of cell. Rule 2 (R2) is for the Vehicle Movement.

3.2 Nagel – Schreckenberg CA Model (NS Model)

In 1992, NS model [11] was framed based on the4 rules of the CA. The rules are:

1. if (V < V max) then V = V+1, V is the speed of the car.

If the present speed is smaller than the desired maximum speed, the vehicle is accelerated.

2. if (V > gap) then V = gap, .

If present speed id is greater than the gap in the front, set v=gap.

This rule **avoids** the collision between vehicles.

3. if (V > 0) then V = V-1 with Pbreak.

This rule introduced a random element into the model.

4. X = X + V.

The present position on the road is moved forward by V.

According to these rules the speed and the acceleration/deceleration ratio of vehicles are independent of the speed of the other vehicles at any time.

3.3 BJH Model

In 1996, another model developed by Benjamin, Johnson, and Hui (BJH model) [12] is similar to the NS model, but with 'slow-to-start' rule. The slow-to-start rule allows a stopped vehicle to move again with this slow-to-start probability 1-Ps. If the vehicle doesn't move, then it tries to move again but this time with the probability Ps. The authors used this model to study the effect of junctions on highways, finding that, setting a speed limit near junctions on single lane roads can greatly decrease the queue length to enter the road.

3.4 Fukui-Ishibashi CA Model

In 1996,Fukui and Ishibashi [13] constructed a generalization of prototypical CA-184 CA model. This model has two different categories: 1). Stochastic Fukui--Ishibashi 2). Deterministic Fukui—Ishibashi.

They assume a gradual acceleration of cell per time step and followed the modified rule:

(R1): Acceleration and Braking

$v_i(t) \leftarrow \min\{v_i(t-1) + 1, gs_i(t-1), vmax\}$

These experimental observations have indicated that there is no difference in global system dynamics with respect to either adopting gradual or instantaneous vehicle acceleration.

3.5 The New Time Oriented Cellular Automata (TOCA)

In 1999, The New Time Oriented Cellular Automata (TOCA)[14] model was framed in which the threshold of changing speed is equal to the minimum time headway t. Thus the time headway between two vehicles can never be smaller than the threshold of changing speed.

The TOCA rules can be rewritten as:

(1) if (gap > v·tH) and (v < vmax) then v = v + 1 with pac

The speed is increased by 1 with the probability pac if the time headway to the vehicle in the front is larger than tH. An average acceleration ratio with the value pac is resulted.

(2) if (v > gap) then v = gap

(3) if (gap < v·tH) and (v > 0) then v = v – 1 with pdc

The speed is reduced by 1 with the probability pdc if the time headway to the vehicle in the front is smaller than tH. An average deceleration ratio with the value pdc is resulted.

(4) x = x + v

3.6 GE Hong Xia, DONG Li-yun, LEI Li, DAI Si-Qiang Model

A modified cellular automata model for traffic flow was proposed by GE Hong Xia, DONG Li-yun , LEI Li and DAI Si-Qiang (2003)[15]. In this model a changeable security gap is introduced. This model is a modified version of the NS model. Discrete random valuesarefrom 0 to Vmax (Vmax = 5). The rules of the extended NS model are set as follows:

---

(1)Acceleration: vn→min(vn + 1, vmax);

(2) Deceleration: vn→min(vn ,$d_n^{(eff)}$) with $d(v_n, d_n^{(eff)}) = d_n + \max(\text{Vanti} - \text{gap}_{Security}, 0)$, denoting the effective gap, where Vanti = min(d_{n+1}, v_{n+1}) is the anticipated velocity of the leading vehicle in the next time step, $\text{gap}_{Security}$ = round (tv_n) is the security gap while considering the the velocity of its following car;

(3) Randomization: vn→max (vn - 1,0) with the probability p .

(4) Update of position: $x_n \rightarrow x_n + v_n$.

The velocity of vehicles is determined by Steps (1) - (3). Finally, the position of the car is shifted in accordance with the calculated velocity in Step (4). r is a parameter determined by the simulation.

3.7 M. Namekawa, F. Ueda, Y. Hioki, Y. Ueda and A. Satoh Model

In 2004, M. Namekawa, F. Ueda, Y. Hioki, Y. Ueda and A. Satoh [16] proposed a model that improves the NS model.

The Simulation time per clock is 0.1 seconds and the speed is 5 km/h. Modified rule of the proposed mode.

---

(1) Acceleration: (v+1)*a<=g and v < vmax →v=v+1

(2) slow down: v*a < g→v =g/a

The distance between the vehicles that was suitable for speed with value to be decided at the minimum distance between the vehicles v*a.

3.8 Clarridge and Salomaa Model

In 2009, Clarridge and Salomaa [17] proposed a Modified version of **BJH model.**

In this model, the cars' velocities are adjusted at each time step according to the following rules. Recall that d is the distance to the next car, v is the velocity of the current car, vnext is the velocity of the next car, pslow is the probability that the slow-to-start rule is applied, and pfault is the probability that the car slows down randomly. We fix vmax =5.

1. Slow-to-Start**:** As in the BJH rule, if v =0and d> 1 then with probability (1 – p_{slow}), the car accelerates normally (this step is ignored), and with probability p_{slow} the car stays at velocity 0 at this time step (does not move) and accelerates to v =1atthe next time step.

2. Deceleration (when the next car is near): if d<= v and either v<next or v<= 2, then the next car is either very close or going at a faster speed, and we prevent a collision by setting v ← d– 1, but do not slow down more than is necessary. Otherwise, if d<= v, v>= vnext, and v> 2weset v ← min(d−1,v −2) in order to possibly decelerate slightly more, since the car ahead is slower or of the same speed and the velocity of the current car is substantial.

3. Deceleration (when the next car is farther): if v<d<=2v,thenif v>= vnext+4, decelerate by 2 (v ← v−2). Otherwise, if vnext+2 <= v<= vnext+3 then decelerate by 1 (v ← v – 1).

4. Acceleration: if the speed has not been modified yet by one of rules 1-3 andv<vmax and d>v +1, then v ← v +1.

5. Randomization: if v> 0, with probability pfault, velocity decreases by one (v ← v – 1).

6. Motion: the car advances v cells.

This model performs an iteration of cars moving on a road in O(L) time , where L is the length of the road and a parallel implementation based on the constant time.

4 Proposed Model for Vehicular Bridge Traffic Control

It is assumed that the bridge Traffic is caused by cars moving from a dual-lane [21-25] road to a single-lane road, as shown in Fig. 3, where Lane1 and Lane2 are two parallel lanes. These lanes are connected with a bridge lane Bridge at B1. No matter what the time step is, only one car can enter the bridge. After passing through the bridge, at B2 another lanes are Lane3 and Lane4 (as shown in Fig. 3), these two are opposite direction parallels lanes, car can choose Lane 3 and Lane 4 to move to the opposite direction.

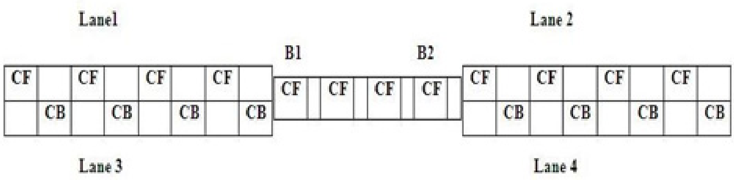


Fig. 3. The model of the Vehicular Bridge Traffic Control System, CF represents cars with forward direction at the instance of time (t) and CB Represent cars with Backward direction at the instance of time (t+1)

Table 1. The Rules that Updated the Next State of the Cellular Automata Cells

Rule	111	110	101	100	011	010	001	000
170	1	0	1	0	1	0	1	0
85	0	1	0	1	0	1	0	1

Table 1 focuses on the situation of the parallel lanes: Lane1, Lane 2, Lane 3, and Lane 4 in all of which no car changes lanes and cars can cross the Bridge only in the forward or backward direction but not in both direction. Cars in each lane simultaneously evolve according to the cellular automata simple rules [1]. Wolfram [1] has investigated cellular automata using empirical observations and simulations. For 2-state 3-neighborhood CA, the evolution of the ith cell can be represented as a function of the present states of (i−1)th, (i)th, and (i+1)th cells as: $x_i(t+1) = f(x_{i-1}(t), x_i(t), x_{i+1}(t))$ *where f*, represents the combinational logic. For a 2-state 3-neighborhood cellular automaton there are 2^3=8 distinct neighborhood configurations or cells are from 0(000) to 7(111) and 2^8=256 distinct mappings from all these neighborhood configurations to the next state, each mapping representing a CA rule ,means 256 CA rules are available, from rule 0 to rule 255 . The next state of the i_{th} cell depends on the present states of its left and right neighbors and on its own present state.

The Bridge Traffic problem[26-28] present in this paper is constructed using cellular automata based on rules 170 and 85.The combinational logic of the rules 170 and 85 [1] for the cellular automata can be expressed as follows:

$$\text{Rule 170: } a_i(t+1) = (a_{(i+1)}(t)) \quad (1)$$

$$\text{Rule 85: } a_i(t+1) = (1 \oplus a_{(i+1)}(t)) \quad (2)$$

i means the cells position and t means time at any instance. The rule specifies the evolution of the cellular automata from the neighborhood configuration to the next state and these are then represented in Table1.

If “1” Represents a car which move the forward or backward direction at any instance of time (t) and “0” represents no car at any instance of time (t) . The rule 170 and rule 85 models for car moving in each lane is shown by Table1.

5 Conclusion

This paper presents a bridge traffic problem based on cellular automata approach. In this paper, we are trying to solve the bridge traffic problem to reduce the number of accident based on cellular automata simple rules(the rules being 170 and 85) with respect to the increasing vehicles. The above knowledge and the proposed model can be taken in planning and controlling the bridge traffic problem in vehicular networks. The processing technique is more effective and efficient for solve the bridge traffic problem in vehicular network.

References

1. Wolfram, S.: A new kind of science Wolfram Media (2002)
2. Adamatzky, A., Lawniczak, R.A.-S.A., Martinez, G.J., Morita, K.: AUTOMATA-2008 Theory and Application of Cellular Automata. Thomas Worsch Editors (2008)
3. Nishinari, K., Schadschneider, A., Chowdhury, D.: Traffic of Ants on a Trail: A Stochastic Modelling and Zero Range Process. In: Sloot, P.M.A., Chopard, B., Hoekstra, A.G. (eds.) ACRI 2004. LNCS, vol. 3305, pp. 192–201. Springer, Heidelberg (2004)
4. Campari, E.G., Levi, G., Maniezzo, V.: Cellular Automata and Roundabout Traffic Simulation. In: Sloot, P.M.A., Chopard, B., Hoekstra, A.G. (eds.) ACRI 2004. LNCS, vol. 3305, pp. 202–210. Springer, Heidelberg (2004)
5. Shoufeng, L., Ximin, L.: Based on Hybrid Genetic Algorithm and Cellular Automata Combined Traffic Signal Control and Route Guidance. In: Chinese Control Conference, pp. 53–57 (2007)
6. Wei, J., Wang, A., Du, N.: Study of Self-Organizing Control of Traffic Signalsin an Urban Network based on Cellular Automata. IEEE Transactions on Vehicular Technology 54(2), 744–748 (2005)
7. Wolfram, S. (ed.): Theory and applications of CA. WorldScientific (1986)
8. Traffic and Granular Flow. Springer (1996,1998,2000,2001)
9. Olariu, S., Weigle, M.C.: Vehicular Networks, From Theory to Practice. CRC Press, Taylor &Francis Group (2009)
10. Wolfram, S. (ed.): Theory and applications of CA. WorldScientific (1986)
11. Nagel, K., Schreckenberg, M.: A cellular automaton model for freeway traffic. J. PhysicsI France 2, 2221–2229 (1992)
12. Benjamin, S.C., Johnson, N.F., Hui, P.M.: Cellular automata models of traffic flow along a highway containing a junction. Journal of Physics A: Mathematical and General 29(12), 3119–3127 (1996)
13. Fukui, M., Ishibasi, Y.: Traffic flow in 1D Cellular Automation Model including cars moving with high speed. Journal of Physics, 1868–1870 (1996)
14. Maerivoet, S., Moor, B.D.: Cellular automata model of road traffic. Physics Reports 419(1), 1–64 (2005)
15. Ge, H.-X., Dong, L.-Y., Lei, L., Dai, S.-Q.: A Modified Cellular Automata Model for traffic flow. Journal of Sanghai Universiy 8(1), 1–3 (2004)
16. Namekawa, M., Ueda, F., Hioki, Y., Ueda, Y., Satoh, A.: General purpose road traffic simulation with cell automation model, pp. 3002–3008. Modelling and Simulation Society of Australia and New Zealand Inc. (2005)
17. Clarridge, A., Salomaa, K.: A Cellular Automaton Model for Car Traffic with a Slow-to-Stop Rule. In: Maneth, S. (ed.) CIAA 2009. LNCS, vol. 5642, pp. 44–53. Springer, Heidelberg (2009)
18. Das, D., Ray, A.: A Parallel Encryption Algorithm for Block Ciphers Based on Reversible ProgrammableCellular Automata. Journal of Computer Science And Engineering 1(1), 82–90 (2010)
19. Das, D.: A Survey on Cellular Automata and Its Applications. In: Krishna, P.V., Babu, M.R., Ariwa, E. (eds.) ObCom 2011, Part I. CCIS, vol. 269, pp. 753–762. Springer, Heidelberg (2012)
20. Das, D., Misra, R.: Parallel Processing Concept Based Road Traffic Model. In: 2nd International Conference on Computer, Communication, Control and Information Technology, pp. 267–271 (2012)

21. Rickert, M., Nagel, K., Schreckenberg, M., Latour, A.: Two lane traffic simulations using cellular automata. Physica A: Statistical Mechanics and its Applications 231(4), 534–550 (1996)
22. Wolf, D.E.: Cellular automata for traffic simulations. Physica A: Statistical Mechanics and its Application 263(1-4), 438–451 (1999)
23. Takayasu, M., Takayasu, H.: 1/F Noise in A Traffic Model. World Scientific (1993)
24. Das, S.: A cellular automata based model for traffic in congested city. In: IEEE International Conference on Systems, Man and Cybernetics, pp. 2397–2402 (2009)
25. Bham, G.H., Benekohal, R.F.: A high fidelity traffic simulation model based on cellular automata and car-following concepts. Transportation Research Part C: Emerging Technologie 12(1), 1–32 (2004)
26. Xiao, S., Kong, L., Liu, M.: A cellular automaton model for a bridge traffic bottleneck. Acta Mechanica Sinic 21(3), 305–309 (2005)
27. Han, Y.S., Ko, S.K.: Analysis of a cellular automaton model for car traffic with a junction. Journal of Theoretical Computer Scienc 450, 54–67 (2012)
28. Esser, J., Schreckenberg, M.: Microscopic Simulation of Urban Traffic Based on Cellular Automata. International Journal of Modern Physics 8(5) (1997)

Analysis of GPS Based Vehicle Trajectory Data for Road Traffic Congestion Learning

Swathi Lanka and Sanjay Kumar Jena

Department of Computer Science and Engineering
National Institute of Technology, Rourkela - 769008, India
swathivanet@gmail.com, skjena@nitrkl.ac.in

Abstract. Successful developments of effective real-time traffic management and information systems demand high quality real time traffic information. In the era of intelligent transportation convergence, traffic monitoring requires traffic sensory technologies. We tabulate various realistic traffic sensors which aim to address the technicalities of both point and mobile sensors and also increase the scope to prefer an optimal sensor for real time traffic data collection. The present analysis extracted data from Mobile Century experiment. The data obtained in the experiment was pre-processed successfully by applying data mining pre-processing techniques such as data transformation, normalization and integration. Finally as a result of the availability of pre-processed Global Position System (GPS) sensors trace data a road map has been generated.

Keywords: Traffic sensor, Traffic flow, GPS probe, Data fusion, Floating car, Fleet management.

1 Introduction

Vehicular Ad-Hoc Network (VANET) is one of the key enabling technologies which can provide the communication between the vehicles which are connected through wireless links [1]. VANET is a component of Intelligent Transportation System (ITS) which can brin0g a noticeable improvement in transportation system towards decreasing congestion and improving safety and traveler convenience. ITS is used to design a smart vehicle. Developing Advanced Driving Assistance Systems (ADAS) aiming to alert drivers about road situation, traffic conditions, and possible traffic congestion with other vehicles has attracted a lot of attention recently [2].

The Advanced Traveler Information System (ATIS) is one of the six components of ITS. ATIS provides solutions for intelligent transportation related applications. It implements emerging computer, communication and information technologies to provide vital information to the users of a system regarding traffic regulation, route and location guidance, hazardous situations and safety advisory and warning messages. ATIS requires a large amount of data for processing, analysis, and storage for effective dissemination of traveler information [3].

M.K. Kundu et al. (eds.), *Advanced Computing, Networking and Informatics - Volume 2*,
Smart Innovation, Systems and Technologies 28,
DOI: 10.1007/978-3-319-07350-7_2,

Traffic congestion has a significant negative impact on social and economic activities around many cities in the world. Road traffic monitoring aims to determine traffic conditions of different road links, which is an essential step toward active congestion control. Many tasks, such as trip planning, traffic management, road engineering, and infra-structure planning, can benefit from traffic estimation [4].Traditional approaches for traffic monitoring rely on the use of point traffic sensors, which can mount at a fixed location along the roadway and sense the traffic parameters at the particular location [5], [6]. After traditional approaches, with the increasing growth of mobile technology mobile sensors has got attention, will be placed in a vehicle can collect vehicle related data [6], [7]. Recently in the era of mobile internet services, with the shrinking cost and increased accuracy of GPS, and increasing penetration of mobile phones in the population makes Global Position System (GPS) with Floating Car Data (FCD) as an attractive traffic sensor[8], [9]. Table 1 shows particulars of commercially available traffic sensors.

With the growing prevalence of GPS receivers embedded in vehicles and smart-phones, there have been increasing interests in using their location updates or trajectories for monitoring traffic [10]. Even though GPS is becoming more and more used and affordable, so far only a limited number of cars are equipped with this system, typically fleet management services. Traffic data obtained from private vehicles or trucks is more suitable for estimating traffic under motorways and rural areas [11]. In case of urban traffic, taxi fleets are particularly useful due to their high number and their on-board communication systems already in place. Currently, GPS probe data are widely used as a source of real-time information by many service providers [12].

Existing Conventional traffic congestion detection systems used location based data for congestion detection. However, quantifying congestion is generally carried out using traffic density which is a spatial parameter. Hence spatial data such as travel time helps to detect congestion with a less delay. In our work, we have collected spatio temporal data from mobile based GPS receivers which are attached with each vehicle travelling on the freeway. In this paper, we are particularly interested to collect spatio temporal data and make the raw data set more suitable for efficient congestion learning. The data set is pre-processed in to a human, machine understandable format. The resultant data set can able to improve the effectiveness and the performance of the data mining algorithms and machine learning techniques whenever it applies on the dataset.

The paper is organized as follows: Technicality of various traffic sensors are discussed in Section1. Section 2 designed a three-level structure vehicle activity database format. Mobile Century Data set has discussed in Section 3. This is followed by data pre-processing methods, resultant datasets and realistic road map in Section 4. Section 5 presents conclusions and future work.

Table 1. Technicalities of commercially available traffic sensors

Technology	Sensing parameters	Strengths	Weakness	Suitable applications
Inductive loops [5] (Point sensor)	Vehicle volume, occupancy, time, speed	Conventional standard can obtain accurate occupancy measurements, flexible design can satisfy large variety of applications, adoptable and less sensitive for all weather and lighting conditions	Installation is intrusive to traffic, maintenance and installation cost is more, gives less detection accuracy when large number of vehicles are involved, reinstallation is needed whenever road is repaved.	Traffic flow detection, congestion detection, traffic-density detection
Pneumatic tubes [6] (Point sensor)	Speed, direction of flow, time, volume	Ideal for short term engineering studies, less maintenance and installation cost, portable device can be reused in many locations.	Has limited lane coverage, intrusive to traffic, system damage causes to inaccurate data collection	Vehicle count, traffic flow detection
Video Image Processors [6], (Point sensor)	Road vehicle images, video streams of traffic	Rich array of data collection, can monitor multiple lanes and detection zones with minimum installation and maintenance, insertion and deletion of detection zones is easy	Performance may be affected by weather, vehicle shadows, vehicle projections, occlusions, strong winds, day-night transitions and water, dust on the camera lens. Setup cost is high.	Traffic count, vehicle speed detection, vehicle classification
Acoustic/Ultrasonic Sensors [6] (Mobile sensor)	Occupancy, count, speed	Multiple lane operation is possible, capable to detect high occupancy vehicle with high accuracy, in sensitive to precipitation	Environmental conditions may affect the performance, cold temperature may affect vehicle count accuracy, and occupancy measurement accuracy may be degraded when vehicle travelling with high speed.	Vehicle parking assistance, vehicle detection, pedestrian count
Active/Passive Infrared Sensors [6] (Mobile sensor)	Vehicle position, speed, count,	Can be operated both day and night, multiple lane operation is possible, usage of sophisticated signal processing algorithms gains better accuracy	Sensitive to inclement weather conditions and ambient light, installation and maintenance cost is more	Road obstacle detection, distance measurement

Table 1. (*continued*)

RFID Sensors [7] (Point & Mobile sensor)	Vehicle ID, time	In expensive Less installation and maintenance cost non intrusive to traffic	Only detect equipped vehicles Collect poor array of data Privacy concerns and actors interest is required	Automatic Vehicle Identification, E-Z pass, Electronic Toll Collection
Microwave Radar [6] (Mobile sensor)	Speed, occupancy, presence	A single detector can cover multiple lanes, usage of efficient signal processing techniques increases detection accuracy	Multi path coverage causes redundant vehicle detection, false detection sometimes, unable to detect stopped vehicle	Calculates vehicle speed, vehicle detection
Magnetometer [6] (Point sensor)	Vehicle count, time	Can be used where a point or small-area location of a vehicle is necessary, can be used where loops are not feasible (e.g, bridge decks), insensitive to weather conditions	Installation requires pavement cut, requires multiple units for full lane detection, maintenance cost is more	Vehicle presence detection, vehicle passage detection
GPS with FCD [8],[9] (Mobile sensor)	Longitude, latitude, time, speed	In vehicle sensor simple to install and operate, less maintenance cost, easy penetration due to raped increase of mobile phones, works under all weather and lighting conditions, never suffer with energy consumption problem since GPS will be equipped in a moving vehicle, collects on road real-time information, non intrusive to traffic.	GPS signals may be obstructed by tall buildings and trees, actor interest is required and Signal strength may be degraded under bad weather conditions.	Congestion detection, collusion detection, intersection safety, Road safety

2 Data Base Design and Data Conversion

The initial task of this research paper is to develop a common database format for vehicle activity data, followed by conversion of dataset in to human understandable format. Three-level structure vehicle activity database format is designed and illustrated in Table 2.

The top level of the database lists a program which helps to collect the vehicle activity data. It contains fields such as name of the dataset, the dates of the program, number of vehicles tested, total testing time duration and parameters that are collected(e.g., longitude, latitude, etc). Each entry in this level has a pointer to the second level of the database.

The second level of the database listing data trips for overall program. When a vehicle is travelling on the road, it collects data would correspond to a single trip entry in the second layer of the database. For each trip, various parameters are listed including date, starting and ending times and testing duration per day. Each entry in this level has a pointer to the third level of the database.

The third level represents spatio temporal time series data. The time series data contains the time sequence of position (longitude, latitude), speed and time. Further in this layer of the database, additional parameters may be derived from the existing data for the determination of congestion level of the roadway.

Table 2. Database Topology

Data set title	Date	No of vehicles	No of seconds	Parameters
Mobile Century	2 Days: 8 February 2008, 9 February 2008	100	28,800	longitude, latitude, speed, time
-------------	--------------	--------	-----------	--------------

Run	Date	Starting time	Ending time	Duration(sec)
1	8/2/2008	19:00:00	23:59:58	17,998
2	9/2/2008	00:00:01	02:59:59	10,799

Date	Time	Vehicle ID	Latitude	Longitude	Speed
8/02/2008	19:00:02	1	37.600	-122.064	0.009
9/02/2008	02:59:58	100	37.6002	-122.063	0.013

3 The Data Set

Vehicle trajectories are typically collected from GPS equipped vehicle based mobile phone from Mobile Century experiment [13] took place on February 8th, 2008. It consisted in deploying 100 GPS- equipped Nokia N95 cell phones on a freeway in 100 vehicles during 8 hours (from 8 February at 19:00:00 pm to 9 February at 03:00:00 am). The experiment was conducted on Highway I-880, near Union City, California; between Winton Ave. to the North and Stevenson Blvd. to the South. This 10-mile long section was selected for field experiment. Data has collected on four lane road with a regular time interval of 3 seconds.

4 Data Pre-processing

Several conversion and filtering steps are often necessary for mobile century data. Pre-processing may include (1). Conversion of date and time from Unix Time Zone to local date and time. (2). Conversion of latitude and longitude in to decimal degrees. (3). Constructing new attributes such as vehicle ID. Data fusion is also necessary which can integrate hundred vehicle activity data in to one unified dataset includes all of the data points and time steps from the input data sets.

Pre-processing has done by using data mining pre-processing techniques such as normalization and attribute construction. Normalization technique used unit conversion method. Unit conversion method converted Unix time in to local time and date. Attribute construction must be replacing or adding new attributes inferred by existing attributes. It is necessary to create new attributes that can capture the important information in a data set more effectively than the original ones [14]. In our system vehicle ID is newly constructed attribute. Among hypothesis-driven and data driven methods, data driven method is particularly used for Attribute construction in the present work. The new attributes are then evaluated according to a given attribute quality measure. Table 3 shows pre-processed data for vehicle ID 1 . Finally data from multiple sources have fused and placed in a single data set by using data fusion technique. In order to analyze the road position along with vehicle motion the entire dataset is sorted with respective time has shown in the Table 4. Experimented roadmap has generated with resultant dataset has shown in Fig. 1.

5 Pre-processing Results

Table 3. Pre-processed data for Vehicle 1

V ID	Date & Time	Latitude	Longitude	Speed
1	08-02-08 19:00:02	37.60043	-122.064	0.009
1	08-02-08 19:00:06	37.60043	-122.064	0.01
1	08-02-08 19:00:09	37.60043	-122.064	0.013
1	08-02-08 19:00:12	37.60043	-122.064	0.015
1	08-02-08 19:00:16	37.60043	-122.064	0.016
1	08-02-08 19:00:20	37.60043	-122.064	0.017
1	08-02-08 19:00:24	37.60043	-122.064	0.017
1	08-02-08 19:00:27	37.60043	-122.064	0.015

Table 4. Pre-processed Road based vehicle moment data

V ID	Date & Time	Latitude	Longitude	Speed
1	08-02-08 19:00:08	37.6105	-122.069	5.002
1	08-02-08 19:00:08	37.6220	-122.078	67.776
1	08-02-08 19:00:09	37.6004	-122.064	0.013
1	08-02-08 19:00:09	37.6430	-122.092	52.402
1	08-02-08 19:00:09	37.6141	-122.072	3.143
1	08-02-08 19:00:09	37.6087	-122.068	65.229
1	08-02-08 19:00:09	37.5934	-122.057	68.11
1	08-02-08 19:00:09	37.6005	-122.062	66.612

Fig. 1. Experimented four lane roads on high way I-880, CA. The symbol represents vehicles. Vehicles are positioned on the road by processing on resultant pre-processed Mobile Century data.

6 Conclusion and Future Work

Real-world traffic data is highly susceptible to noise, redundancy and inconsistent data due to their huge size, heterogeneous sources and type of sensory technologies. Low quality traffic data will lead to low quality results processing. Often low quality information leads to in complete control and management. This paper presents a pre-processed Mobile Century data set and a realistic four lane highway I-880 roadmap with positioned vehicles. Roadmap has generated by taking the input as resultant pre-processed dataset. In future work, the resultant data set will be used for road traffic congestion learning for efficient intelligent congestion control under heterogeneous traffic conditions. Future work in our research program will take advantage of this work, and will focus on congestion prediction and detection.

References

1. Hartenstein, H., Kenneth, P.L.: A Tutorial Survey on Vehicular Ad Hoc Networks. IEEE Communication Magazine 46(6), 164–171 (2008)
2. Zhang, J., Fei, Y.W.: Data Driven in Intelligent Transportation System: A Survey. IEEE Transactions on Intelligent Transportation System 12(4) (2011)
3. Kumar, P., Varun, S.: Advanced Traveller Information for Hyderabad City. IEEE Transactions on Intelligent Transportation System 6(1) (2005)
4. Agarwal, V., Venkata, M.: A Cost-Effective Ultrasonic Sensor-Based Driver-Assistance System For Congestion Traffic Conditions. IEEE Transactions on Intelligent Transportation System 10(3) (2009)
5. Ali, S.M., George, B., Vanajakshi, L.: An Efficient Multiple-Loop Sensor Configuration Applicable for Undisciplined Traffic. IEEE Transactions on Intelligent Transportation System 14(3) (2013)
6. Leduc, G.: Road Traffic Data: Collection Methods and Applications. JRC Technical Notes, Joint Research Centre, European Commission (2008)
7. Cheng, W., Cheng, X., Song, M., Chen, B., Zhao, W.W.: On the Design and Deployment of RFID Assisted Navigation Systems for VANET. IEEE Transaction on Parallel and Distributed System (2011)
8. Yong, Z., Zuo, X., Zhang, L., Chen, Z.: Traffic Congestion Detection based on GPS Floating-Car Data. Procedia Engineering 15, 5541–5546 (2011)
9. Vanajakshi, L., Subramanian, S.C.: Travel time prediction under heterogeneous traffic conditions using global positioning system data from buses. IET Intelligent Transportation Systems (2008)
10. Sun, Z., Xuegang, B.: Vehicle Classification Using GPS Data. Transportation Research Record Part C 37, 102–117 (2013)
11. Messelodi, S., Carla, M.M.: Intelligent Extended Floating Car Data Collection. Expert systems with applications 36, 4213–4227 (2009)
12. Zhu, Y., Li, Z.: A compressive sensing approach to urban traffic estimation with probe vehicles. IEEE Transactions on Mobile Computing 12(11) (2013)
13. Herrera, J.C., Daniel, B.W.: Evaluation Of Traffic Data Obtained Via GPS-Enabled Mobile Phones: The Mobile Century Field Experiment. Transportation Research Record Part C 18, 568–583 (2011)
14. Jiawei, H., Micheline, K.: Data mining: Concepts and Techniques, 2nd edn. Morgan Kaufmann Publishers (2006)

Fast Estimation of Coverage Area in a Pervasive Computing Environment

Dibakar Saha, Nabanita Das, and Bhargab B. Bhattacharya

Advanced Computing and Microelectronics Unit
Indian Statistical Institute, Kolkata, India
dibakar.saha10@gmail.com,
ndas,bhargab@isical.ac.in

Abstract. In many applications of pervasive computing and communication, it is often mandatory that a certain service area be fully covered by a given deployment of nodes or access points. Hence, a fast and accurate method of estimating the coverage area is needed. However, in a scenario with a limited computation and communication capability as in self-organized mobile networks, where the nodes are not static, computation-intensive algorithms are not suitable. In this paper, we have presented a simple algorithm for estimating the area covered by a set of nodes randomly deployed over a 2-D region. We assume that the nodes are identical and each of them covers a circular area. For fast estimation of the collective coverage of n such circles, we approximate each real circle by the tightest square that encloses it as well as by the largest square that is inscribed within it, and present an $O(n \log n)$ time algorithm for computation. We study the variation of the estimated area between these two bounds, for random deployment of nodes. In comparison with an accurate digital circle based method, the proposed algorithms estimate the area coverage with only 10% deviation, while reducing the complexity of area computation significantly. Moreover, for an over-deployed network, the estimation provides an almost exact measure of the covered area.

Keywords: Pervasive Computing, Wireless Sensor Networks (WSN), Coverage, Digital Circle, Range.

1 Introduction

In a pervasive computing environment, for tetherless computing and communication, it is often required to place the computing nodes or access points to offer services over a predefined area. In many cases, like vehicular networks, ad-hoc networks, mobile health-care services, surveillance, wireless sensor networks, the nodes are often mobile and have limited power, limited storage and limited computation and communication capabilities. These networks are often self-organized, and can take decision based on their local information only.

A typical wireless sensor network (WSN) consists of spatially distributed autonomous sensor nodes to monitor physical or environmental conditions, such as temperature, sound, pressure etc. Each sensor node has the ability to collect,

M.K. Kundu et al. (eds.), *Advanced Computing, Networking and Informatics - Volume 2,* Smart Innovation, Systems and Technologies 28,
DOI: 10.1007/978-3-319-07350-7_3,

process, and route the sensed data. The streams of sensed data from each node are forwarded cooperatively through several intermediate nodes to finally arrive at the sink node. Sensor networks are used in many applications such as habitat and ecosystem monitoring, weather forecasting, smart health-care technologies, precision agriculture, homeland security and surveillance. For all these applications, the live nodes are required to cover the area to be monitored. Therefore, the classical problems of covering an area with a specific kind of shape such as circle, square, or rectangle, are recently being revisited for modeling and analysis of such networks. In this paper, we address the problem of estimating the area covered by a set of nodes distributed randomly over a 2-D plane. We assume that the nodes are homogeneous, and each of them covers a circular area with a fixed radius. Since estimation of the area covered by an arbitrary set of circles is computation-intensive, it may not be feasible to perform it in real time where the nodes have limited power, storage and computational ability, as in a pervasive computing environment. The area-coverage of the square meshes, hexagonal meshes and honeycomb meshes was studied earlier by Luo *et al.* [1]. Some related theoretical and algorithmic issues concerning rectangle intersection problems were revisited by Six and Wood [2]. Bentley and Wood [3] proposed an optimal algorithm for reporting intersections of n rectangles. In these two papers, the authors proposed an $O(n \log n + k)$ algorithm where k is the number of intersecting pairs. An $O(n\log^2 n)$ algorithm in [4] can be used to construct a generalized Voronoi diagram for a set of n circular discs and to compute the coverage area in terms of circular sectors and quadrangles. A more efficient algorithm with $O(n \log n)$ time complexity and $O(n)$ space complexity for circle intersection/union using a particular generalization of Voronoi diagram called power diagrams was reported later [5]. However, all these algorithms require complex data structures and rigorous computation. In order to reduce computational effort, in literature often it is assumed that the monitoring area is composed of a number of elementary areas or unit square grids [6–9]. In [10], [11], the authors investigated random and coordinated coverage algorithms. Some authors used partitioning techniques to decompose the query region into square grid blocks and studied the coverage of each block by sensor nodes [12–14]. For a more realistic estimation of the covered area, a new $O(n \log n)$ algorithm based on digital geometry is proposed in [15], where a real circle is replaced by a digital circle and discrete domain computation is applied. Though the complexity of the algorithm remains the same as it is in [5], the former uses simple data structures and primitive arithmetic operations only. In this paper, we propose a simpler and faster method of coverage estimation. Given a random distribution of n nodes over a 2-D plane, the circular area covered by each node is approximated within two bounds: (*i*) by the smallest enclosing square providing an upper approximation, and (*ii*) by the largest inscribing square providing a lower approximation. For these cases, simpler algorithms have been proposed to estimate the covered area. It is evident that in the first case the algorithm produces an overestimate of the area covered, whereas, by the second approach an underestimate is achieved always. However, it is interesting to study the variation of areas for several random deployment of nodes by varying the number of nodes and the radius. Simulation experiments show that in comparison with an earlier work [15], the proposed algorithms are capable of estimating the coverage area with no more than 10% error. Moreover, for bounded areas, in over-deployed networks, it may provide an almost exact result.

The rest of the paper is organized as follows: Section 2 presents the problem formulation. Section 3 describes the algorithms for finding the intersection points and the area covered by squares. Section 4 shows the simulation results and finally Section 5 concludes the paper.

2 Problem Formulation

Let a set of n nodes $\mathcal{S} = \{s_1, s_2, \ldots, s_n\}$ be deployed randomly over a 2-D region $\mathcal{A}$. Each node covers a circular area with radius r. The problem is to decide whether the region $\mathcal{A}$ is fully covered by the nodes. Note that a computation considering real circles in Euclidean geometry is rather complex. To alleviate this problem, we approximate a circular region as i) the largest square inscribed within it called *inside square*, and ii) as the smallest square enclosing the circle, referred to as *outside square*. With this model, instead of real circular area πr^2 we estimate the covered area in terms of inside squares with area $2r^2$ and outside squares with area $4r^2$ as shown in Fig. 1 and Fig. 2 respectively. Therefore, the problem of measuring the area covered by a set of circles now reduces to the problem of finding the area covered by a set of squares distributed randomly on a 2D plane. To solve this problem, firstly the intersection points among the squares are to be identified. Then the covered area can be represented in terms of some monotone isothetic objects [4], and hence can be computed in linear time only. It is evident that the measured covered area is always underestimated in case-i) and overestimated in case-ii) respectively. However given any random deployment of nodes due to arbitrary overlapping of covered areas it is interesting to study the dependence of the deviations of the estimated areas from the exact covered area with number of nodes and the radius respectively.

The following section describes the details of the area estimation procedure and the respective algorithms.

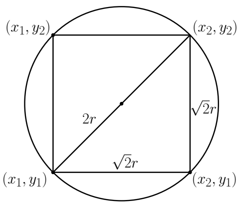


Fig. 1. Maximum square inscribed within a circle

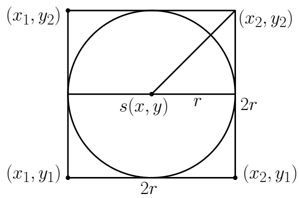


Fig. 2. Minimum square enclosing a circle

3 Area Coverage by Squares

3.1 Intersection of Two Squares

We assume that each square s is defined as a quadruple $s = (x_1, x_2, y_1, y_2)$ where $(x_1, y_1), (x_1, y_2), (x_2, y_1), (x_2, y_2)$ are the bottom-left, top-left, bottom-right and top-right corner points of s respectively, and all co-ordinates are integers.

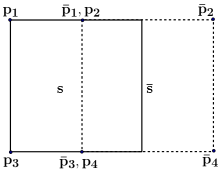


Fig. 3. $x_1 \leq \bar{x}_1$, $y_1 = \bar{y_1}$

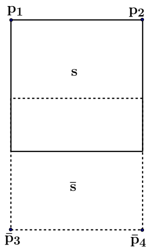


Fig. 4. $x_1 = \bar{x_1}$, $y_1 > \bar{y_1}$

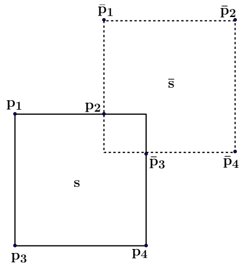


Fig. 5. $x_1 \leq \bar{x}_1$, $y_1 < \bar{y_1}$

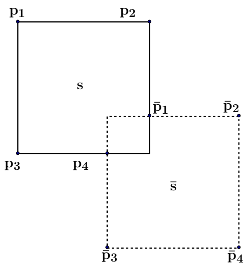


Fig. 6. $x_1 \leq \bar{x}_1$, $y_1 > \bar{y_1}$

Two squares $s = (x_1, x_2, y_1, y_2)$ and $\bar{s} = (\bar{x}_1, \bar{x}_2, \bar{y}_1, \bar{y}_2)$ are said to intersect if and only if the intervals

1) $[\bar{x}_1, \bar{x}_2]$ and $[x_1, x_2]$ overlap, and
2) $[\bar{y}_1, \bar{y}_2]$ and $[y_1, y_2]$ overlap,

where, $[x_1, x_2]$ and $[y_1, y_2]$ define the closed intervals given by the projection of s on the x-axis and the y-axis respectively. Also, each square s maintains a list of four integers, $P(s) : \{p_1, p_2, p_3, p_4\}$. Initially, $p_1 = x_1, p_2 = x_2, p_3 = x_1$ and $p_4 = x_2$ respectively. To find the pair of intersection points between the squares s and $\bar{s}$, without loss of generality, we assume that $x_1 \leq \bar{x}_1$. If the two squares intersect, depending on the relative positions of s and $\bar{s}$ the values of $P(s)$ and $P(\bar{s})$ will change in four different ways as shown in Fig. 3, Fig. 4, Fig. 5, and Fig. 6.

Given two squares s and $\bar{s}$, the procedure to find the intersection points and to update $P(s)$ and $P(\bar{s})$ appropriately is presented in Algorithm 1.

3.2 Area Covered by a Set of Squares

To compute the area covered by a set of n squares distributed randomly over a 2-D plane, here we propose an iterative procedure Algorithm 2 based on the strategy proposed in [4] that finds the intersection among a set of monotone objects. We start with a list L_x of the projections (a_i, b_i) of the squares s_i sorted along the x-axis as shown in Fig. 7. Next we scan L_x and include s_i (if $L_x = a_i$), or delete s_i (if $L_x = b_i$) in L and compute intersections among the newly adjacent

Algorithm 1. Intersection(s,$\bar{s}$, $P(s)$, $P(\bar{s})$)

Input: $s = (x_1, x_2, y_1, y_2)$, $\bar{s} = (\bar{x}_1, \bar{x}_2, \bar{y}_1, \bar{y}_2)$, $P(s) : (p_1, p_2, p_3, p_4)$, $P(\bar{s}) : (\bar{p}_1, \bar{p}_2, \bar{p}_3, \bar{p}_4)$
Output: $P(s), P(\bar{s})$
if $y_1 == \bar{y}_1$ **then**
  if $p_1 \le \bar{x}_1$ and $p_2 \ge \bar{x}_1$ **then** $p_2 = \bar{x}_1$;
  if $p_3 \le \bar{x}_1$ and $p_4 \ge \bar{x}_1$ **then** $p_4 = \bar{x}_1$;
if $x_1 == \bar{x}_1$ **then**
  if $y_1 > \bar{y}_1$ **then**
    $\bar{p}_1 = \bar{p}_2 = p_3 = p_4 = null$;
  else
    $p_1 = p_2 = \bar{p}_3 = \bar{p}_4 = null$;
if $\bar{y}_1 > y_1$ **then**
  if $p_1 \le \bar{x}_1$ and $p_2 \ge \bar{x}_1$ **then** $p_2 = \bar{x}_1$;
  if $\bar{p}_3 \le x_2$ and $\bar{p}_4 \ge x_2$ **then** $\bar{p}_3 = x_2$;
else
  if $p_3 \le \bar{x}_1$ and $p_4 \ge \bar{x}_1$ **then** $p_4 = \bar{x}_1$;
  if $\bar{p}_1 \le x_2$ and $\bar{p}_2 \ge x_2$ **then** $\bar{p}_1 = x_2$;

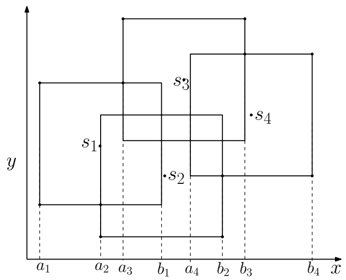


Fig. 7. Projections of squares for four nodes distributed over an area

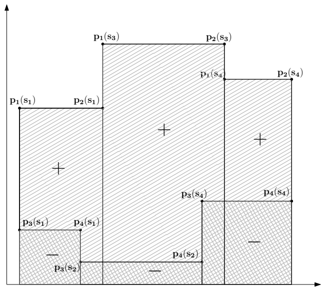


Fig. 8. Area computation by Algorithm 2 for the node distribution in Fig. 7

pairs of squares in L, and update the lists $P(s)$ of relevant squares by Algorithm 1 appropriately. Finally, given $P(s_i)$ for each square s_i, Algorithm 2 computes the area traversing along the closed intervals defined by $P(s_i)$'s appropriately. Fig. 8 shows the area computed by Algorithm 2 for the node distribution shown in Fig. 7.

Complexity Analysis: It is evident from Algorithm 1 that the intersection points between two squares can be computed in constant time. In Algorithm 2, to sort the given set of squares, $O(n \log n)$ time is required, and a linear traversal along the list $P(i)$'s to find the area will require $O(n)$ time. Hence the total complexity of area computation is $O(n \log n)$. Moreover, the procedure requires primitive arithmetic operations such as addition and multiplication of integers, with simple data structures only.

4 Simulation Studies

In order to study the error in the estimated area, we may compute it more accurately assuming that each real circle is approximated by a digital circle on

Algorithm 2. Area computation of a set of squares

Input: Squares $S := \{s_1, s_2, \ldots, s_n\}$
Output: Area total : A_{tot}
Step 1: for *each square* $s_i \in S$ **do**
  Compute a_i and b_i and include in L_x in sorted order along x-axis;
Step 2: for $i = 1$ *to* $2n$ **do**
  if $L_x(i) = a_j$ **then**
    include s_j in L in sorted order along y-axis ;
    By *Algorithm 1*, update $P(s)$ for the newly adjacent pairs of squares in L;
  if $L_x(i) = a_j$ *or* b_j *and* $s_j = L(k)$ *and if both* $L(k-1)$ *and* $L(k+1)$ *exist in* L **then**
    check if intersection points between squares of any pair $(L(k-1), L(k), L(k+1))$ is included within the third one.
  Update the $P(s)$ lists;
  if $L_x(i) = b_j$ **then** delete $L(k)$ from L;
Step 3: for *each square* s_i **do**
  if $p_1(s_i)$ or $p_2(s_i) \neq$ *null* **then** $A_{tot} \leftarrow A_{tot} + (|p_1(s_i) - p_2(s_i)| * (y_2(s_i))$;
  if $p_3(s_i)$ or $p_4(s_i) \neq$ *null* **then** $A_{tot} \leftarrow A_{tot} - (|p_3(s_i) - p_4(s_i)| * (y_1(s_i))$;

an integer grid [15]. To avoid the rigorous computation involved in finding the area covered by a set of real circles, the problem is mapped to digital circles [16]. With a large radius (i.e., on a dense grid), a digital circle can represent the real circle closely in respect of covered area as shown in Fig. 9. For completeness, a brief outline of the procedure for computing the area covered by a given set of digital circles is given below.

4.1 Area Covered by Digital Circles

The area covered by a set of n digital circles is basically an isothetic cover, and can be computed in terms of vertical strips, as shown in Fig. 9. To compute the covered area, an iterative procedure is proposed in [15]. In each iteration, for a pair of digital circles, following digital geometry based concepts, the intersection points, as shown in Fig. 10, are computed in constant time. Next, the covered area is represented in terms of a sequence of intersection points defining the boundary of that area. For each circle, a circular list of its intersection points

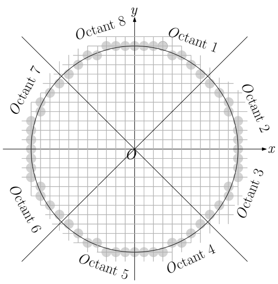


Fig. 9. A digital circle

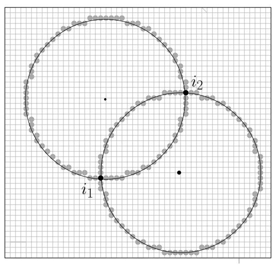


Fig. 10. Intersection between two digital circles

is maintained. The area of a digital circle is computed by traversing the list of intersection points in a cyclic order. In fact, any pair of intersection points i_1 and i_2 defines an arc of a digital circle. The area bounded by the appropriate arcs can be computed by traversing the vertical strips from i_1 to i_2, within the circle. This algorithm also runs in $O(n \log n)$ time; however, the algorithm proposed here using two square approximations has a smaller constant term in the asymptotic complexity, and hence, it needs much less CPU time and memory for practical problems as it involves fewer computations and simpler data structures.

4.2 Results and Discussions

In our simulation study, we assume that n nodes, $5 \leq n \leq 100$, are distributed randomly over a 100×100 grid area. Fig. 11 and Fig. 12 show how the area covered by inside squares, outside squares, digital circles and real circles increases with radius for a single node when the area to be monitored is bounded and unbounded respectively. Fig. 13 and Fig. 14 show how the covered area increases with n for inside, outside squares and digital circles in bounded and unbounded areas respectively. As expected, the outside square grid always overestimates the area and inside squares underestimate the area; the difference is always observed to lie within ± 10 % from that estimated by digital circles. Also for $n \geq 25$, all three shapes estimate the same area in bounded case. It also reveals the fact that the area estimated by outside squares is closer to the real area compared to that achieved by inside squares. Fig. 15 shows the variation of the estimated area with radius. It also shows that the outside square scheme is better compared to the inside square. From the simulation results, it is evident that for unbounded areas, this estimation strategy performs poorly as the radius or as the number of nodes grows. But, for all practical purposes the area to be monitored is a bounded one. Also, it is interesting to observe that for bounded areas, the estimated area can achieve exact results for sufficiently high radius or number of nodes.

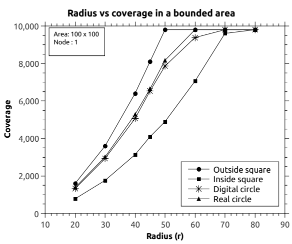


Fig. 11. Radius vs area coverage in a bounded area

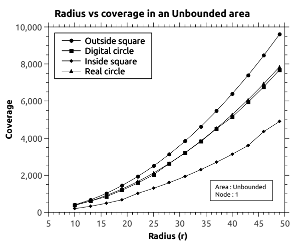


Fig. 12. Radius vs area coverage in an unbounded area

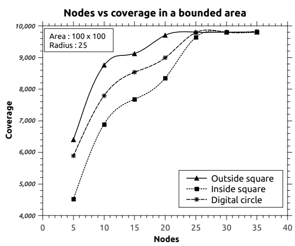


Fig. 13. Nodes vs area coverage in a bounded area

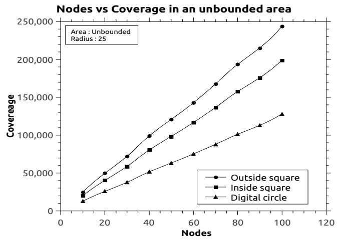


Fig. 14. Nodes vs area coverage in an unbounded area

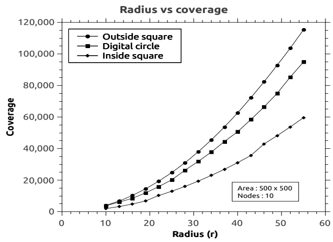


Fig. 15. Radius vs area coverage with multiple nodes

5 Conclusion

In this paper, we have addressed the problem of estimating the area covered by a set of nodes randomly deployed over a 2-D area. It is assumed that each node covers a circular area with a fixed radius r. An upper (lower) approximation of each circular area is provided by the smallest enclosing (largest inscribed) square. Our algorithm provides a fast and simple method of estimating a nearly-accurate area coverage, which has very small computational overhead. Therefore, it will be highly suitable for a low-energy pervasive environment, where dynamic coverage estimation is frequently needed. Experimental results indicate its potential in several promising application areas.

References

1. Luo, C.J., Tang, B., Zhou, M.T., Cao, Z.: Analysis of the wireless sensor networks efficient coverage. In: Proc. International Conference on Apperceiving Computing and Intelligence Analysis (ICACIA), pp. 194–197 (2010)
2. Six, H.W., Wood, D.: The rectangle intersection problem revisited. BIT Numerical Mathematics 20, 426–433 (1980)
3. Bentley, J.L., Wood, D.: An optimal worst case algorithm for reporting intersections of rectangles. IEEE Transactions on Computers C-29, 571–577 (1980)
4. Sharir, M.: Intersection and closest-pair problems for a set of planar discs. SIAM Journal on Computing 14, 448–468 (1985)
5. Aurenhammer, F.: Improved algorithms for discs and balls using power diagrams. Journal of Algorithms 9, 151–161 (1988)
6. Gallais, A., Carle, J., Simplot-ryl, D., Stojmenovic, I.: Localized sensor area coverage with low communication overhead. IEEE Transactions on Mobile Copmuting, 661–672 (2008)
7. Slijepcevic, S., Potkonjak, M.: Power efficient organization of wireless sensor networks. In: Proc. IEEE International Conference on Communications, vol. 2, pp. 472–476 (2001)
8. Pervin, N., Layek, D., Das, N.: Localized algorithm for connected set cover partitioning in wireless sensor networks. In: Proc. 1st International Conference on Parallel Distributed and Grid Computing (PDGC), pp. 229–234 (2010)

9. Huang, C.F., Tseng, Y.C.: The coverage problem in a wireless sensor network. In: Proc. 2nd ACM International Conference on Wireless Sensor Networks and Applications, pp. 115–121. ACM (2003)
10. Hsin, C., Liu, M.: Network coverage using low duty-cycled sensors: Random coordinated sleep algorithms. In: Proc. Third International Symposium on Information Processing in Sensor Networks, pp. 433–442 (2004)
11. Sheu, J.P., Yu, C.H., Tu, S.C.: A distributed protocol for query execution in sensor networks. In: Proc. IEEE Wireless Communications and Networking Conference, vol. 3, pp. 1824–1829 (2005)
12. Saha, D., Das, N.: Distributed area coverage by connected set cover partitioning in wireless sensor networks. In: Proc. First International Workshop on Sustainable Monitoring through Cyber-Physical Systems (SuMo-CPS). ICDCN, Mumbai (2013)
13. Saha, D., Das, N.: A fast fault tolerant partitioning algorithm for wireless sensor networks. In: Proc. Third International Conference on Advances in Computing and Information Technology (ACITY), vol. 3, pp. 227–237 (2013)
14. Ke, W., Liu, B., Tsai, M.: The critical-square-grid coverage problem in wireless sensor networks is NP-complete. Journal of Computer Networks, 2209–2220 (2010)
15. Saha, D., Das, N., Pal, S.: A digital-geometric approach for computing area coverage in wireless sensor networks. In: Natarajan, R. (ed.) ICDCIT 2014. LNCS, vol. 8337, pp. 134–145. Springer, Heidelberg (2014)
16. Bhowmick, P., Bhattacharya, B.B.: Number-theoretic interpretation and construction of a digital circle. Discrete Applied Mathematics 156, 2381–2399 (2008)

A Fault Tolerance Approach for Mobile Agents

Munshi Navid Anjum, Chandreyee Chowdhury, and Sarmistha Neogy

Department of Computer Science and Engineering
Jadavpur University, Kolkata, India
sarmisthaneogy@gmail.com

Abstract. Mobile agent is a program that can migrate autonomously from one environment to another. Many factors affect execution of mobile agents during its life cycle. Errors may occur on the server, or during communication. Error probability further increases with longer path. In mobile agent computing environment any component of the network - node, link, or agent may fail at any time, thus preventing them from continuing their execution. Therefore, fault-tolerance is a vital issue for the deployment of mobile agent systems. Here we propose a scheme to tolerate faults caused by malicious node behavior and link failure using agent cloning. The strategy is shown to prevent the agents from getting lost at irrational nodes (nodes that behave maliciously). The scheme is simulated using IBM Aglet platform and is found to be scalable when the no. of irrational nodes is fixed. Performance improves with more no. of agents.

1 Introduction

A mobile agent is a program that migrates from one environment to another, and is capable of performing appropriately in the new environment [1]. An agent consists of three components: the program which implements it, the execution state of the program and the data [2]. The owner (the node that spawns agents) can decide the route of the mobile agent or the agent itself can decide its next hop destination dynamically depending on context. The migration is similar to remote procedure call (RPC) [3] methods. For instance when a user directs an Internet browser to "visit" a website the browser merely downloads a copy of the site or one version of it in case of dynamic web sites. Similarly, a mobile agent accomplishes a migration attempt through data replication. When a mobile agent decides to migrate, it saves its own state (serialization may be used in case of Java based agents), transports this saved state to the new host, and resumes execution from the saved state. This is called weak migration.

But a mobile agent may also migrate in another way called strong migration [4]. A strong migration occurs when the mobile agent carries out its migration between different hosts while conserving its data, state and code. The platform is the environment of execution. The platform makes it possible to create mobile agents; it offers the necessary elements required by them to perform their tasks such as execution, migration towards other platforms and so on.

M.K. Kundu et al. (eds.), *Advanced Computing, Networking and Informatics - Volume 2*,
Smart Innovation, Systems and Technologies 28,
DOI: 10.1007/978-3-319-07350-7_4,

Typical benefits of using mobile agents include [4], [5] reducing the network load, overcoming network latency, executing asynchronously and autonomously, etc.
Since mobile agents migrate from one node to the other collecting and/or spreading meaningful information, loss of a mobile agent results in more data loss as compared to loss of a message. The longer the trail an agent needs to visit, the chance of errors is more. Chances of migrating to a malicious node are even higher. In this paper, a fault tolerance scheme is proposed for the agents in order to prevent them from getting lost while en'route. The scheme uses the concept of cloning. The clone is a copy of a mobile agent, with no critical code and data. Measures are taken so that cloning does not become a significant performance overhead. Security issues are also considered.

The rest of the paper is outlined as follows: Section 2 discusses state of the art. Our work is introduced in Section 3. Implementation of the scheme using Aglet is discussed in Section 4. Finally Section 5 concludes.

2 Related Work

In mobile agent computing environment either the agent may fail due to software failure or the nodes or links in the underlying environment may fail. In either case the agents could not make successful migration. Therefore, fault-tolerance measures should be taken while deploying the mobile agents. Fault tolerance schemes for mobile agents to survive agent server crash failures are complex as execution of agent code at the remote hosts could not be controlled. There are [6], [7] several ideas presented to implement fault-tolerance in mobile agents. We are trying to summarize some of them below:

In witness agent, a failure detection and recovery mechanism is used named witness agent passing [8]. This is done by the virtue of inter agent communication. This communication can be done using two methods, one direct message and other indirect message. Direct messages are passed when (the system assumes that) the agent is at the last visited node. In other cases (when the agent is not present at immediately previous node) indirect messages are used. In this method there is a mailbox at each owner that keeps those unattended messages. This process needs lot of resources and with the increase in the traverse path, more witness agents need to be created, hence consuming even more resources.

CAMA frameworks support fault tolerance at application level. This schema handles faults by introducing three types of operations over the exception namely raise, check and wait [9]. These exceptions are handled through inter agent communication. The advantage of this approach is that the exception handling allows fast operation. It also allows elective error detection and recovery mechanism. Its drawback is that execution of the process can be stalled if any agent raises any exceptions and malfunctions.

Adaptive mobile agents can adapt themselves to the environment. The rules it follows are dependent on the current environment and the working also changes accordingly. Two or more adaptive mobile agents should communicate with each other to acquire the correct role suited for the environment [10]. The roles are also specified about access or restriction to a resource. This control strategy is called Role Based Access Control (RBAC) [11]. The advantage of this technique is that as the mobile agents already resides within the system the communication overhead for inter agent communication is less for that the time required to respond is less for a mobile agent. Increase of routing of adaptive mobile agents, node, link failure or topological changes may produce errors.

Transient errors can be detected before an agent start executing at a node. It is done by comparison of the states of the agent. This technique has the capability to detect and correct more than one error [12]. The time and space overhead are minimal. One issue might be there that if bit errors are not corrected by any duplicates this may block the process.

Unexpected faults may arise in unreliable networks. Then it is not a good approach to create fault tolerance mechanism for every one of them. A unique solution may be created an adaptive mechanism to deal with several types of faults. Chameleon is one of such mechanism for fault tolerance [13]. Flexibility of chameleon gives it its advantage in the case of unreliable networks. The disadvantage of this mechanism is that it becomes blocking if execution at any node fails.

Exactly once protocol guarantees fault less execution in case where the agent needs to be executed only once to yield a correct result, executing more than once may lead to errors. For the execution of each elementary part of an agent a set of resources are needed. The states of these resources are changed for every such part. Resource manager keeps track of these changes [13]. The disadvantage of this technique is that the agents underlying process become atomic; multiple commit or rollback operations increase the complexity.

3 Our Work

In this paper we propose a fault tolerance technique for the mobile agents that can protect the agents from getting lost while en'route. Here we define a mobile agent clone as the copy of the agent which has the same code and data as the original agent at some state. Here it is assumed that the clone is similar to its original copy and hence carries with it critical code and data when it migrates to the next node.

3.1 Problem Description

A node in the network having mobile agent platform spawns mobile agents if it needs to collect (spread) information from (to) N (<= total no. of nodes in the network) hosts in the network. The owner may send an agent with N nodes in its trail or may divide the job to several agents depending on agent performance and network conditions. After a mobile agent visits all the hosts mentioned in its trail and gets required services, it retracts back to the owner. During computation on the hosts, the mobile agent may crash due to hardware/software failures at the hosts or software error at the agent itself. Also the agent may get lost due to link failure or irrational behavior of the intermediate nodes. Any undesirable change of agent code can be detected if the agent's code is digitally signed as in [1].

So, our focus in this work is to protect mobile agents during its life cycle and to minimize the amount of information lost due to failure of an agent.

3.2 The Scheme

In this scheme, we define a mobile agent clone to be a replica of the original. So this clone protects original agent from getting lost in a network during migration and hence protects mobile agents from irrational nodes (hosts). Here a node deploys an agent and assigns its task for which the agent is asked to visit a number of nodes (the owner is interested in) according to some policy. Before migrating to an unknown host, an agent creates its replica. The cloned agent migrates to a number of unknown hosts before a counter (say, SKIP) expires. Here SKIP holds a value that signifies how many nodes (from its trail) the agent will traverse before creating another replica. The value of the counter can be fixed, decided by the underlying application or can be tuned according to agent performance. The variable SKIP is initialized to that value. SKIP is decremented by the cloned agent whenever the cloned agent migrates to a new node, and executes its task successfully. When the counter reaches 0, the cloned agent sends an acknowledgement back to the original agent that was kept at some previously visited host site (owner, for the first time SKIP decrements to 0). That agent upon receiving such acknowledgement kills itself. The agent then creates

```
AgentCode()
  1. Move to a remote host site according to the
     trail given by the owner
     1.1.  To migrate to a remote host site a replica
           agent need to be created.
  2. If a reply from its replica is received then
     2.1.  This replica kills itself.
     2.2.  Return
  3. If time-out occurs then
    //reply from replica is not received in due time
     3.1.  The agent retracts back to owner.
     3.2.  Reports to the owner about missing reply
           from the previously visited host site.
  4. Decrement SKIP by 1.
  5. Execute its task.
  6. Verify signature of the code.
  7. If SKIP< 0 then
     7.1.  Send a reply back to the node where a copy
           is kept/owner
     7.2.  Create a replica in the present site.
     7.3.  Reset SKIP to the default value.
  8. Otherwise
           Migrate to a new node according to its
           policy
```

another replica and keeps it in the current node and moves on to the next unvisited nodes from its trail. The value of SKIP is reset to the old value. This continues until the trail finishes. If there is no reply from the replica (due to irrational node behavior and agent getting lost) then the agent residing at a previous host site may retract back to owner directly. SKIP is decremented by one at the owner for the next agent that will visit the same trail. Thus more frequently agent cloning happens in order to cope with increasingly hostile/faulty network. Moreover, if an agent retracts back to owner successfully after visiting all the hosts it was asked to, then SKIP is incremented by 1 for the agent that will have the same trail.

In this scheme the replica is kept saved until a reply (that it has executed successfully) from the agent residing at some other node is received. On receipt of a successful reply the replica destroys itself. Let us take an example to describe the situation. Say agent X from node 1 (owner of X) has node nos. 2, 3, 4, 6, 7, 8 in its trail with SKIP=2. So clone X will send a reply to the original X in node 1 after it has reached node 3 whereby SKIP becomes 0. X now keeps itself here and creates a clone, resets SKIP to 2 and resumes its journey.

3.3 Our Scheme to Protect Mobile Agents

This scheme has a fault tolerance strategy. As the replica is kept in one or several nodes, there is a backup even when a link failure occurs or the agent is lost due to irrational nodes. If an agent comes back and reports about a missing reply (according to step 3.2 in AgentCode()), the owner may take necessary steps to inquire about that part of the network and also decrease the default value of SKIP by 1 so that the agents it spawns further may cope with increasingly noisy or hostile behavior of the network. Otherwise it may increase the default value of SKIP by 1 to signify stable environment.

As clones are not created at every node by default, the no. of replies is reduced. Hence the system works reasonably faster.

4 Results

The simulation is carried out in IBM Aglet platform [15]. It runs on Tahiti server. Since it deploys java based mobile agents, it is readily portable to any platform. We have installed Aglet on Linux (Fedora 16) operating system.

Here each node that an agent visits is designated as a port. For example the default port no. 4434 can work as an owner and spawn an agent. Ports can be created by the programmer. These ports are host platforms (nodes) forming a network. Any port can spawn any no. of agents.

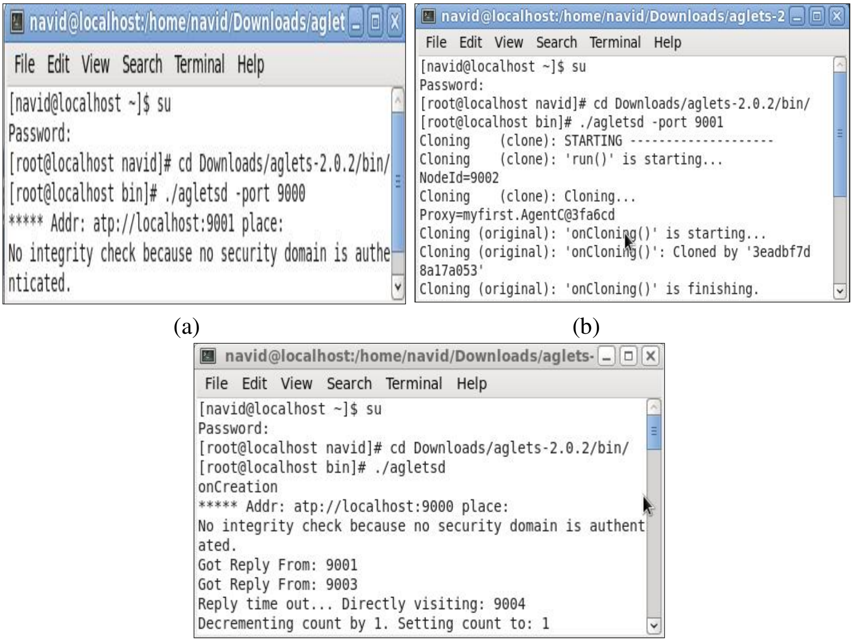
Here each node that an agent visits is designated as a port. In our example we have shown an agent spawned by port 4434 visiting a no. of nodes (ports) according to its policy. Fig. 1 shows an example of our implementation on Aglet. The route of an agent is given in Fig. 1. Fig. 1(a) shows that an agent spawned by node 4434 sends an agent to node 9000, and executes its task. In port 9000, no agent copy is kept according to steps 6-7 of AgentCode() and forward to the next node that is port 9001. The counter strucks 0 at port 9001, it is found to be a trusted one as the signature of agent code is not modified and hence a replica is created and stored and the counter is reset as shown in Fig. 1(b). Then the agent visits port 9003 and finally returns to the owner after visiting all the nodes and collecting the relevant information which the owner displays (shown in Fig. 1(c)). This shows rough journey of an agent. It can be influenced by factors like link failure, malicious behavior of both the nodes that an agent is expected to visit and the attacks in-transit etc.

A series of experiments was carried out in Aglet. Some of the results are listed in Fig. 2. Two metric are introduced to measure the performance of the agent based system. One is ratio of successful agents and the other one is ratio of irrational nodes. The first one is defined as follows

$$Ratio\ of\ Successful\ Agents = \frac{No.of\ agent\ ssuccessfully\ returned}{No.of Agents\ Deployed} \quad (1)$$

The ratio of irrational nodes is defined by

$$Ratio\ of\ Irrational\ Nodes = \frac{No.of\ Irrational\ Nodes}{Total\ no\ .of\ Nodes} \quad (2)$$



(c)

Fig. 1. (a) Aglet window showing that the host (port 9001) visited by an agent and there is no cloning;(b) Aglet window showing that the host (port 9002) visited by an agent and the agent cloned by itself;(c) Aglet window showing that the agent returns to its owner (port 4434)

In Fig. 2(a) agent success is measured in an increasingly hostile network. Here total no. of nodes in the network is taken to be 20. It can be observed that application performance with cloning (our scheme) is better than without cloning. When all nodes are irrational, it is obvious that no agents can perform its job. But even when 90% of the nodes behave irrationally, that is 18 out of 20 nodes are irrational then also the agents can show some progress with our scheme. However if no fault tolerance measure is applied to the agents, the performance drops to 0 when 60% of the nodes behave irrationally in a network.

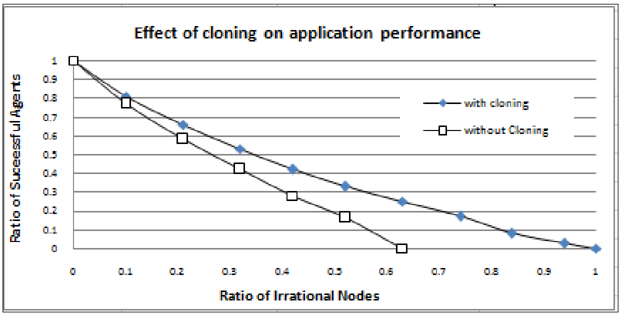


Fig. 2. (a) Graph showing that application performance with cloning and without cloning

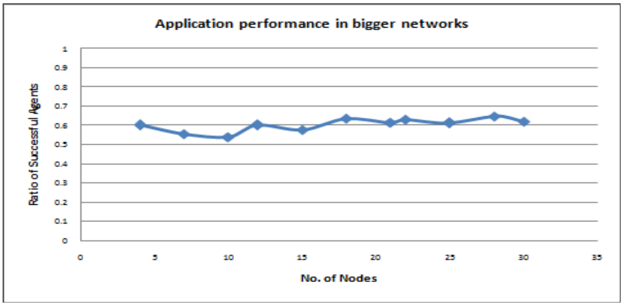


Fig. 3. (b) Graph showing that application performance when the no. of nodes are increased

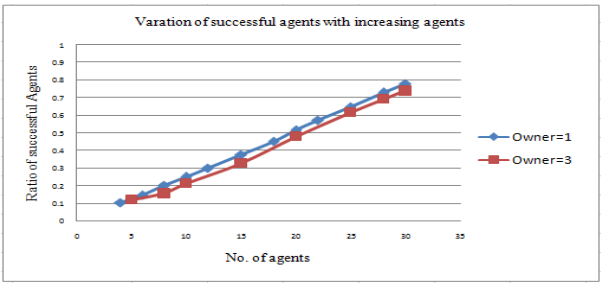


Fig. 4. (c) Graph showing application performance when the no. of agents are increased but no. of irrational nodes are fixed

In the next experiment application performance is measured in increasingly larger network. Here ratio of irrational nodes is taken to be 0.21. The graph shown in Fig. 2(b) indicates that application performance changes a little as the network grows in size. Thus the fault tolerance scheme is found to be scalable even when the underlying network is as big as having 30 nodes.

Finally in a network of 20 nodes where no. of irrational nodes is fixed to 5, agent performance is measured with increasing no. of agents. Here it is assumed that the extra agents do not pose significant bandwidth overhead and hence no rational node kills an agent when it is en'route. In Fig. 2(c) the blue line indicates a single agent group that is only a node is spawning agents with similar characteristics. The red one indicates the situation where three nodes in the network are spawning agents for some purpose. The figure indicates that performance improves almost linearly with no. of agents in the system. As different applications running at various nodes spawn agents with differing characteristics, overall performance still improves linearly with total no. of agents in the network. It can be observed from the figure that agent heterogeneity does not much affect overall performance of the applications.

5 Conclusion

In this paper, we propose a fault tolerance scheme for mobile agents working in hostile networks where an agent halts due to the nodes, links, or agent software failure in the network. The fault tolerance scheme presented not only ensures minimum data loss upon failure but also protects the mobile agents from getting lost at the malicious hosts (nodes) and disconnected nodes, using agent cloning. The scheme is tested in IBM Aglet platform. The results indicate that our scheme improves application performance in an increasingly hostile network. The scheme is also found to be scalable.

References

1. Anjum, M.N., Chowdhury, C., Neogy, S.: Securing Network using Mobile Agents. In: Proceedings of 2012 International Conference on Communication, Devices and Intelligent System (CODIS), pp. 266–269 (2012)
2. Chowdhury, C., Neogy, S.: Mobile Agent Security Based on Trust Model in MANET. In: Abraham, A., Lloret Mauri, J., Buford, J.F., Suzuki, J., Thampi, S.M. (eds.) ACC 2011, Part I. CCIS, vol. 190, pp. 129–140. Springer, Heidelberg (2011)
3. Pleisch, S.: State of the Art of Mobile Agent Computing - Security, Fault Tolerance, and Transaction Support. Research Report, IBM Research (1999)
4. Hans, R., Kaur, R.: Fault Tolerance Approach in Mobile Agents for Information Retrieval Applications using Check Points. International Journal of Computer Science & Communication Network 2(3) (2012)
5. Hans, R.: Fault Tolerance Techniques In Mobile Agents: A Survey. In: International Conference on Computing and Control Engineering (2012)

6. Rostami, A., Rashidi, H., Zahraie, M.S.: Fault Tolerance Mobile Agent System Using Witness Agent in 2-Dimensional Mesh Network. International Journal of Computer Science Issue 7(5), 153–158 (2010)
7. Budi, A., Alexei, I., Alexander, R.: On using the CAMA framework for developing open mobile fault tolerant agent systems. In: Proceedings of 2006 International Workshop on Software Engineering for Large-Scale Multi-Agent Systems (2006)
8. Marikkannu, P., Jovin, J.J.A., Purusothaman, T.: Fault-Tolerant Adaptive Mobile Agent System using Dynamic Role based Access Control. International Journal of Computer Applications 20(2) (2011)
9. Ferraiolo, D.F., Kuhn, D.R., Chandramouli, R.: Role-based Access Control. Artech House, Inc. (2007)
10. Kumar, S.G.: Transient Fault Tolerance in Mobile Agent Based Computing. INFOCOMP Journal of Computer Scienc 4(4), 1–11 (2005)
11. Bagchi, S., Whisnant, K., Kalbarczyk, Z., Iyer, R.K.: Chameleon: Adaptive Fault Tolerance Using Reliable, Mobile Agents. In: Proceedings of 16th Symposium on Reliable Distributed Systems (1997)
12. Rothermel, K., Strasser, M.: A fault-Tolerant Protocol for Providing the Exactly-Once Property of Mobile Agents. In: Proceedings ofof 17th IEEE Symposium on Reliable Distributed Systems (1998)
13. Lange, D.B., Oshima, M.: Programming and Developing Java Mobile Agents with Aglets. Addison Wesley (1998)

Path Discovery for Sinks Mobility in Obstacle Resisting WSNs

Prasenjit Chanak and Indrajit Banerjee

Department of Information Technology
Indian Institute of Engineering Science and Technology (IIEST)
Shibpur, Howrah-711103, India
prasenjit.chanak@gmail.com, ibenerjee@it.becs.ac.in

Abstract. Sink mobility achieves great success in network life time improvement in wireless sensor networks. In mobile sink movement, mobile sink moves through random path or optimized path in obstacle free area. Fixed or constrained path is used in obstacle resisting environment. In fixed or constrained path, sink move through a predefined path, which is designed by the end user. In these strategies, only fixed obstacles are avoided which are previously present in the monitoring area. These strategies are unable to avoid those obstacles which are randomly entered in the monitoring area. In this paper, we propose a quad-tree based online path detection strategy that detects any type of obstacles which enter within the network life time and design a shortest mobile sink movement path avoiding detected obstacles. The proposed scheme divides whole network into different small size region and detects region wise obstacle. On the other hand, proposed scheme constructs region wise shortest path for mobile sink data collection. Simulation results are presented to verify our proposed scheme.

Keywords: Wireless sensor networks, sink mobility; obstacle, quad tree, network life span, energy efficiency.

1 Introduction

Wireless sensor networks provide reliable monitoring from long distance without any human interference. The main requirements of these networks are highly fault tolerant, long life time and low–latency data transfers [1], [2]. The primary goal of this network is to gather relevant data from surrounding and transmits to base station (BS). Deployed sensor nodes are energy constraints, limited computation and low storage capacities. Therefore, energy efficient routing is a very challenging issue in network life time enhancements. Multi-hop communication achieves great success in energy efficient routing strategy design. Nowadays, sink mobility gives better performance in WSN life time enhancements. In mobile sink based data routing techniques, mobile sinks are moved in different paths and collect data from static sensor nodes. WSNs have enabled numerousadvanced monitoring and control application in environmental, biomedical, and numerous other applications.

M.K. Kundu et al. (eds.), *Advanced Computing, Networking and Informatics - Volume 2*,
Smart Innovation, Systems and Technologies 28,
DOI: 10.1007/978-3-319-07350-7_5,

In rural application, different types of obstacles are present in the monitoring area [3]. These obstacles are like mountains, buildings etc. Obstacles prevent communication between the nodes. On the basis of movements, obstacles can be classified onto two categories: a) static obstacle and b) moving obstacle. The static obstacles are unable to move any other place i.e. static obstacles are fixed. Once static obstacles have been detected it cannot change position. Therefore, in presence of static obstacle, obstacle moves into a fixed and predefined path easily. On the other hand, moving obstacles are changed its location with respect to random time interval. In mobile sink movement strategy, current obstacle position detection is very important for optimal sink mobility path design. Due to presence of moving obstacle, optimal path construction is a challenging issue in WSNs. Some works have been done on sink movement in obstacle residing. These works only designed a fixed and constant path for sink movements on the basis of static obstacle position. In fixed and constant path movement, sink is unable to avoid moving obstacle and newly introduces obstacle in random time interval in the network monitoring area. In our proposed scheme, according to obstacle position and movement information sink makes its own movement strategy. The optimal sink movement increases network performance.

In mobile sink based data routing scheme, selected number of mobile sinks move in different locations of the network and collect data from static sensor nodes. Therefore, static nodes' energy loss is decreased and increase life time. In sink mobility based data routing scheme, path constriction for mobile sink movement is very important issue in WSN. Luo*et al.* [4] consider a WSN with a mobile base station. The moving base station repeatedly relocates to change the bottleneck nodes closer to the base station. In this technique various types of predetermined strategies are used to search base station movement path and data routing. Somasundara*et al.* [1] approach a cluster based data routing scheme. In this technique sensor nodes are arranged in different cluster. Cluster head collects data from cluster member nodes. Cluster head transmits data to mobile node when it passes by. On the other hand, Bi *et al.*[5] proposed an autonomous moving strategy for mobile sinks in data –gathering sensor networks. In this paper, authors consider a WSN with one mobile sink. Mobile sink moves proactively towards the node that has the highest residual energy in the network. When mobile sink reaches a new location, it broadcasts a message for sensor node data collection. Sensor nodes transmit data to mobile nodes by multi-hop communication.

On the basis of movement strategy, sink mobility can be classified into three categories: a) Random sink mobility, b) Optimized sink mobility, c) Fixed or constrained sink mobility.In random sink mobility strategy, mobile sink randomly moves in arbitrary length and direction paths and collects data from static sensor nodes. In random sink mobility strategy, sink speed is also arbitrary [6], [7], [8]. In optimized sink mobility strategy, mobile sink move through optimal path for data collection. The optimal movement path is designed on the basis of a particular network variable. The sink movement path continuously regulated to ensure optimal network performance [5], [9], [10]. In fixed or constrained sink mobility [11], mobile sink moves through a predefined path. According to obstacle position, end user designs a mobile sink movement path. Mobile sink follows this path and collects data from the static sensor nodes. If any new obstacle entered within the network