

# International Journal on

# Advances in Networks and Services



2016 vol. 9 nr. 1&2

The *International Journal on Advances in Networks and Services* is published by IARIA.

ISSN: 1942-2644

journals site: <http://www.ariajournals.org>

contact: [petre@aria.org](mailto:petre@aria.org)

Responsibility for the contents rests upon the authors and not upon IARIA, nor on IARIA volunteers, staff, or contractors.

IARIA is the owner of the publication and of editorial aspects. IARIA reserves the right to update the content for quality improvements.

Abstracting is permitted with credit to the source. Libraries are permitted to photocopy or print, providing the reference is mentioned and that the resulting material is made available at no cost.

Reference should mention:

*International Journal on Advances in Networks and Services, issn 1942-2644*  
vol. 9, no. 1 & 2, year 2016, [http://www.ariajournals.org/networks\\_and\\_services/](http://www.ariajournals.org/networks_and_services/)

The copyright for each included paper belongs to the authors. Republishing of same material, by authors or persons or organizations, is not allowed. Reprint rights can be granted by IARIA or by the authors, and must include proper reference.

Reference to an article in the journal is as follows:

<Author list>, "<Article title>"  
*International Journal on Advances in Networks and Services, issn 1942-2644*  
vol. 9, no. 1 & 2, year 2016, <start page>:<end page>, [http://www.ariajournals.org/networks\\_and\\_services/](http://www.ariajournals.org/networks_and_services/)

IARIA journals are made available for free, proving the appropriate references are made when their content is used.

Sponsored by IARIA

[www.aria.org](http://www.aria.org)

Copyright © 2016 IARIA

**Editor-in-Chief**

Tibor Gyires, Illinois State University, USA

**Editorial Advisory Board**

Mario Freire, University of Beira Interior, Portugal  
Carlos Becker Westphall, Federal University of Santa Catarina, Brazil  
Rainer Falk, Siemens AG - Corporate Technology, Germany  
Cristian Anghel, University Politehnica of Bucharest, Romania  
Rui L. Aguiar, Universidade de Aveiro, Portugal  
Jemal Abawajy, Deakin University, Australia  
Zoubir Mammeri, IRIT - Paul Sabatier University - Toulouse, France

**Editorial Board**

Ryma Abassi, Higher Institute of Communication Studies of Tunis (Iset'Com) / Digital Security Unit, Tunisia  
Majid Bayani Abbasy, Universidad Nacional de Costa Rica, Costa Rica  
Jemal Abawajy, Deakin University, Australia  
Javier M. Aguiar Pérez, Universidad de Valladolid, Spain  
Rui L. Aguiar, Universidade de Aveiro, Portugal  
Ali H. Al-Bayati, De Montfort Uni. (DMU), UK  
Giuseppe Amato, Consiglio Nazionale delle Ricerche, Istituto di Scienza e Tecnologie dell'Informazione (CNR-ISTI), Italy  
Mario Anzures-García, Benemérita Universidad Autónoma de Puebla, México ]  
Pedro Andrés Aranda Gutiérrez, Telefónica I+D - Madrid, Spain  
Cristian Anghel, University Politehnica of Bucharest, Romania  
Miguel Ardid, Universitat Politècnica de València, Spain  
Valentina Baljak, National Institute of Informatics & University of Tokyo, Japan  
Alvaro Barradas, University of Algarve, Portugal  
Mostafa Bassiouni, University of Central Florida, USA  
Michael Bauer, The University of Western Ontario, Canada  
Carlos Becker Westphall, Federal University of Santa Catarina, Brazil  
Zdenek Becvar, Czech Technical University in Prague, Czech Republic  
Francisco J. Bellido Outeiriño, University of Cordoba, Spain  
Djamel Benferhat, University Of South Brittany, France  
Jalel Ben-Othman, Université de Paris 13, France  
Mathilde Benveniste, En-aerion, USA  
Luis Bernardo, Universidade Nova of Lisboa, Portugal  
Alex Bikfalvi, Universidad Carlos III de Madrid, Spain  
Thomas Michael Bohnert, Zurich University of Applied Sciences, Switzerland  
Eugen Borgoci, University "Politehnica" of Bucharest (UPB), Romania  
Fernando Boronat Seguí, Universidad Politecnica de Valencia, Spain  
Christos Bouras, University of Patras, Greece  
Mahmoud Brahimi, University of Msila, Algeria  
Marco Bruti, Telecom Italia Sparkle S.p.A., Italy  
Dumitru Burdescu, University of Craiova, Romania

Diletta Romana Cacciagrano, University of Camerino, Italy  
Maria-Dolores Cano, Universidad Politécnica de Cartagena, Spain  
Juan-Vicente Capella-Hernández, Universitat Politècnica de València, Spain  
Eduardo Cerqueira, Federal University of Para, Brazil  
Bruno Chatras, Orange Labs, France  
Marc Cheboldaeff, T-Systems International GmbH, Germany  
Kong Cheng, Vencore Labs, USA  
Dickson Chiu, Dickson Computer Systems, Hong Kong  
Andrzej Chydzinski, Silesian University of Technology, Poland  
Hugo Coll Ferri, Polytechnic University of Valencia, Spain  
Noelia Correia, University of the Algarve, Portugal  
Noël Crespi, Institut Telecom, Telecom SudParis, France  
Paulo da Fonseca Pinto, Universidade Nova de Lisboa, Portugal  
Orhan Dagdeviren, International Computer Institute/Ege University, Turkey  
Philip Davies, Bournemouth and Poole College / Bournemouth University, UK  
Carlton Davis, École Polytechnique de Montréal, Canada  
Claudio de Castro Monteiro, Federal Institute of Education, Science and Technology of Tocantins, Brazil  
João Henrique de Souza Pereira, University of São Paulo, Brazil  
Javier Del Ser, Tecnalia Research & Innovation, Spain  
Behnam Dezfouli, Universiti Teknologi Malaysia (UTM), Malaysia  
Daniela Dragomirescu, LAAS-CNRS, University of Toulouse, France  
Jean-Michel Dricot, Université Libre de Bruxelles, Belgium  
Wan Du, Nanyang Technological University (NTU), Singapore  
Matthias Ehmann, Universität Bayreuth, Germany  
Wael M El-Medany, University Of Bahrain, Bahrain  
Imad H. Elhadj, American University of Beirut, Lebanon  
Gledson Elias, Federal University of Paraíba, Brazil  
Joshua Ellul, University of Malta, Malta  
Rainer Falk, Siemens AG - Corporate Technology, Germany  
Károly Farkas, Budapest University of Technology and Economics, Hungary  
Huei-Wen Ferng, National Taiwan University of Science and Technology - Taipei, Taiwan  
Gianluigi Ferrari, University of Parma, Italy  
Mário F. S. Ferreira, University of Aveiro, Portugal  
Bruno Filipe Marques, Polytechnic Institute of Viseu, Portugal  
Ulrich Flegel, HFT Stuttgart, Germany  
Juan J. Flores, Universidad Michoacana, Mexico  
Ingo Friese, Deutsche Telekom AG - Berlin, Germany  
Sebastian Fudickar, University of Potsdam, Germany  
Stefania Galizia, Innova S.p.A., Italy  
Ivan Ganchev, University of Limerick, Ireland  
Miguel Garcia, Universitat Politecnica de Valencia, Spain  
Emiliano Garcia-Palacios, Queens University Belfast, UK  
Marc Gilg, University of Haute-Alsace, France  
Debasis Giri, Haldia Institute of Technology, India  
Markus Goldstein, Kyushu University, Japan  
Luis Gomes, Universidade Nova Lisboa, Portugal  
Anahita Gouya, Solution Architect, France  
Mohamed Graiet, Institut Supérieur d'Informatique et de Mathématique de Monastir, Tunisie  
Christos Grecos, University of West of Scotland, UK  
Vic Grout, Glyndwr University, UK  
Yi Gu, Middle Tennessee State University, USA  
Angela Guercio, Kent State University, USA  
Xiang Gui, Massey University, New Zealand

Mina S. Guirguis, Texas State University - San Marcos, USA  
Tibor Gyires, School of Information Technology, Illinois State University, USA  
Keijo Haataja, University of Eastern Finland, Finland  
Gerhard Hancke, Royal Holloway / University of London, UK  
R. Hariprakash, Arulmigu Meenakshi Amman College of Engineering, Chennai, India  
Go Hasegawa, Osaka University, Japan  
Eva Hladká, CESNET & Masaryk University, Czech Republic  
Hans-Joachim Hof, Munich University of Applied Sciences, Germany  
Razib Iqbal, Amdocs, Canada  
Abhaya Induruwa, Canterbury Christ Church University, UK  
Muhammad Ismail, University of Waterloo, Canada  
Vasanth Iyer, Florida International University, Miami, USA  
Peter Janacik, Heinz Nixdorf Institute, University of Paderborn, Germany  
Imad Jawhar, United Arab Emirates University, UAE  
Aravind Kailas, University of North Carolina at Charlotte, USA  
Mohamed Abd rabou Ahmed Kalil, Ilmenau University of Technology, Germany  
Kyoung-Don Kang, State University of New York at Binghamton, USA  
Sarfraz Khokhar, Cisco Systems Inc., USA  
Vitaly Klyuev, University of Aizu, Japan  
Jarkko Knecht, Nokia Research Center, Finland  
Dan Komosny, Brno University of Technology, Czech Republic  
Ilker Korkmaz, Izmir University of Economics, Turkey  
Tomas Koutny, University of West Bohemia, Czech Republic  
Evangelos Kranakis, Carleton University - Ottawa, Canada  
Lars Krueger, T-Systems International GmbH, Germany  
Kae Hsiang Kwong, MIMOS Berhad, Malaysia  
KP Lam, University of Keele, UK  
Birger Lantow, University of Rostock, Germany  
Hadi Larijani, Glasgow Caledonian Univ., UK  
Annett Laube-Rosenpflanzner, Bern University of Applied Sciences, Switzerland  
Gyu Myoung Lee, Institut Telecom, Telecom SudParis, France  
Shiguo Lian, Orange Labs Beijing, China  
Chiu-Kuo Liang, Chung Hua University, Hsinchu, Taiwan  
Wei-Ming Lin, University of Texas at San Antonio, USA  
David Lizcano, Universidad a Distancia de Madrid, Spain  
Chengnian Long, Shanghai Jiao Tong University, China  
Jonathan Loo, Middlesex University, UK  
Pascal Lorenz, University of Haute Alsace, France  
Albert A. Lysko, Council for Scientific and Industrial Research (CSIR), South Africa  
Pavel Mach, Czech Technical University in Prague, Czech Republic  
Elsa María Macías López, University of Las Palmas de Gran Canaria, Spain  
Damien Magoni, University of Bordeaux, France  
Ahmed Mahdy, Texas A&M University-Corpus Christi, USA  
Zoubir Mammeri, IRIT - Paul Sabatier University - Toulouse, France  
Gianfranco Manes, University of Florence, Italy  
Sathiamoorthy Manoharan, University of Auckland, New Zealand  
Moshe Timothy Masonta, Council for Scientific and Industrial Research (CSIR), Pretoria, South Africa  
Hamid Menouar, QU Wireless Innovations Center - Doha, Qatar  
Guowang Miao, KTH, The Royal Institute of Technology, Sweden  
Mohssen Mohammed, University of Cape Town, South Africa  
Miklos Molnar, University Montpellier 2, France  
Lorenzo Mossucca, Istituto Superiore Mario Boella, Italy  
Jogesh K. Muppala, The Hong Kong University of Science and Technology, Hong Kong

Katsuhiko Naito, Mie University, Japan  
Deok Hee Nam, Wilberforce University, USA  
Sarmistha Neogy, Jadavpur University- Kolkata, India  
Rui Neto Marinheiro, Instituto Universitário de Lisboa (ISCTE-IUL), Instituto de Telecomunicações, Portugal  
David Newell, Bournemouth University - Bournemouth, UK  
Armando Nolasco Pinto, Universidade de Aveiro / Instituto de Telecomunicações, Portugal  
Jason R.C. Nurse, University of Oxford, UK  
Kazuya Odagiri, Yamaguchi University, Japan  
Máirtín O'Droma, University of Limerick, Ireland  
Rainer Oechsle, University of Applied Science, Trier, Germany  
Henning Olesen, Aalborg University Copenhagen, Denmark  
Jose Oscar Fajardo, University of the Basque Country, Spain  
Constantin Paleologu, University Politehnica of Bucharest, Romania  
Eleni Patouni, National & Kapodistrian University of Athens, Greece  
Harry Perros, NC State University, USA  
Miodrag Potkonjak, University of California - Los Angeles, USA  
Yusnita Rahayu, Universiti Malaysia Pahang (UMP), Malaysia  
Yenumula B. Reddy, Grambling State University, USA  
Oliviero Riganelli, University of Milano Bicocca, Italy  
Teng Rui, National Institute of Information and Communication Technology, Japan  
Antonio Ruiz Martinez, University of Murcia, Spain  
George S. Oreku, TIRDO / North West University, Tanzania/ South Africa  
Sattar B. Sadkhan, Chairman of IEEE IRAQ Section, Iraq  
Husnain Saeed, National University of Sciences & Technology (NUST), Pakistan  
Addisson Salazar, Universidad Politecnica de Valencia, Spain  
Sébastien Salva, University of Auvergne, France  
Ioakeim Samaras, Aristotle University of Thessaloniki, Greece  
Luz A. Sánchez-Gálvez, Benemérita Universidad Autónoma de Puebla, México  
Teerapat Sanguankotchakorn, Asian Institute of Technology, Thailand  
José Santa, University Centre of Defence at the Spanish Air Force Academy, Spain  
Rajarshi Sanyal, Belgacom International Carrier Services, Belgium  
Mohamad Sayed Hassan, Orange Labs, France  
Thomas C. Schmidt, HAW Hamburg, Germany  
Hans Scholten, Pervasive Systems / University of Twente, The Netherlands  
Véronique Sebastien, University of Reunion Island, France  
Jean-Pierre Seifert, Technische Universität Berlin & Telekom Innovation Laboratories, Germany  
Sandra Sendra Compte, Polytechnic University of Valencia, Spain  
Dimitrios Serpanos, Univ. of Patras and ISI/RC ATHENA, Greece  
Roman Y. Shtykh, Rakuten, Inc., Japan  
Salman Ijaz Institute of Systems and Robotics, University of Algarve, Portugal  
Adão Silva, University of Aveiro / Institute of Telecommunications, Portugal  
Florian Skopik, AIT Austrian Institute of Technology, Austria  
Karel Slavicek, Masaryk University, Czech Republic  
Vahid Solouk, Urmia University of Technology, Iran  
Peter Soreanu, ORT Braude College, Israel  
Pedro Sousa, University of Minho, Portugal  
Vladimir Stantchev, SRH University Berlin, Germany  
Radu Stoleru, Texas A&M University - College Station, USA  
Lars Strand, Nofas, Norway  
Stefan Strauß, Austrian Academy of Sciences, Austria  
Álvaro Suárez Sarmiento, University of Las Palmas de Gran Canaria, Spain  
Masashi Sugano, School of Knowledge and Information Systems, Osaka Prefecture University, Japan  
Young-Joo Suh, POSTECH (Pohang University of Science and Technology), Korea

Junzhao Sun, University of Oulu, Finland  
David R. Surma, Indiana University South Bend, USA  
Yongning Tang, School of Information Technology, Illinois State University, USA  
Yoshiaki Taniguchi, Kindai University, Japan  
Anel Tanovic, BH Telecom d.d. Sarajevo, Bosnia and Herzegovina  
Olivier Terzo, Istituto Superiore Mario Boella - Torino, Italy  
Tzu-Chieh Tsai, National Chengchi University, Taiwan  
Samyr Vale, Federal University of Maranhão - UFMA, Brazil  
Dario Vieira, EFREI, France  
Lukas Vojtech, Czech Technical University in Prague, Czech Republic  
Michael von Riegen, University of Hamburg, Germany  
You-Chiun Wang, National Sun Yat-Sen University, Taiwan  
Gary R. Weckman, Ohio University, USA  
Chih-Yu Wen, National Chung Hsing University, Taichung, Taiwan  
Michelle Wetterwald, HeNetBot, France  
Feng Xia, Dalian University of Technology, China  
Kaiping Xue, USTC - Hefei, China  
Mark Yampolskiy, Vanderbilt University, USA  
Dongfang Yang, National Research Council, Canada  
Qimin Yang, Harvey Mudd College, USA  
Beytullah Yildiz, TOBB Economics and Technology University, Turkey  
Anastasiya Yurchyshyna, University of Geneva, Switzerland  
Sergey Y. Yurish, IFSA, Spain  
Jelena Zdravkovic, Stockholm University, Sweden  
Yuanyuan Zeng, Wuhan University, China  
Weiliang Zhao, Macquarie University, Australia  
Wenbing Zhao, Cleveland State University, USA  
Zibin Zheng, The Chinese University of Hong Kong, China  
Yongxin Zhu, Shanghai Jiao Tong University, China  
Zuqing Zhu, University of Science and Technology of China, China  
Martin Zimmermann, University of Applied Sciences Offenburg, Germany

**CONTENTS**

*pages: 1 - 10*

**Practical Design of Network Coded Multicast over Satellite**

Paresh Saxena, AnsuR, Spain

Maria Angeles Vazquez-Castro, Universitat Autònoma de Barcelona, Spain

*pages: 11 - 19*

**A Method for Transforming Movement Paths in Wireless Mobile Network Simulation**

Hawra Alseef, University of New Brunswick, Canada

John DeDourek, University of New Brunswick, Canada

Przemyslaw Pochec, University of New Brunswick, Canada

*pages: 20 - 29*

**Multi-Target Data Association in Binary Sensor Networks for Ambulant Care Support**

Sebastian Matthias Müller, OFFIS Institute for Information Technology, Germany

Andreas Hein, School of Medicine and Health Sciences, University of Oldenburg, Germany

*pages: 30 - 37*

**Application Server Availability; Measurement and Analysis of the NorNet Core**

Sune Jakobsson, NTNU, Norway

## Practical Design of Network Coded Multicast over Satellite

Paresh Saxena\* and M. A. Vázquez-Castro\*\*

\*AnsuR Solutions Barcelona, Advanced Industry Park,  
c/Marie Curie 8-14, 08042 Barcelona, Spain.

\*\*Dept. of Telecommunications and Systems Engineering,  
Universitat Autònoma de Barcelona, Barcelona, Spain.

Email: paresh.saxena@ansur.es, angeles.vazquez@uab.es

**Abstract**—In this paper, we present the practical design of systematic random network coding (SNC) for multicast over satellite. In particular, the satellite coverage is over a large geographical area that consists of several users. These users may face different channel conditions and undergo different packet loss rates. In this work, we show two regions of transmission. First, the general multicast region where all users subscribed to the multicast channel can recover all the data packets and second, we identify the multi-unicast region where only users with good channel conditions can recover all the data packets but those with bad channel accept transmission losses. Further, we present the architectural and encapsulation feasibility of SNC at satellite-dependent and satellite-independent layers. We derive theoretically and by simulation the benefits of SNC against state-of-the-art end-to-end coding at these layers. Our results show that: (i) SNC at link layer can achieve up to 26.90% and 24.26% higher maximum achievable rates for multicast and multi-unicast, respectively and (ii) SNC at application layer requires up to 6.66% and 30.02% smaller available network bandwidth for multicast and multi-unicast respectively.

**Index Terms**—Network coding; Multicast; Satellite communication; Achievable rates.

### I. INTRODUCTION

This paper extends our earlier work [1]. In this paper, we consider more detail analysis of system model and introduces the results and analysis for multicast and multi-unicast using network coding at both satellite-dependent and satellite-independent layers.

Network coding (NC) extends traditional network operations from routing and store-and-forward to more powerful operations that allow for coding information at intermediate nodes. NC was first introduced in [2], where it is shown that NC can achieve the min-cut capacity of the error-free networks by allowing coding at intermediate nodes. In subsequent work [3], it is also shown that the linear encoding and decoding operations at intermediate nodes are sufficient to achieve the min-cut max-flow performance. These results lead to the investigation of efficient linear encoding and decoding strategies for network coding. As a result, it is shown that random linear network coding (RNC) [4], where information packets transmitted in the network are random linear combinations of the original data packets, is asymptotically capacity achieving.

The main philosophy of using RNC as a capacity-achieving network coding scheme is that it allows the practical application of network coding in a distributed manner. For example, there is no need of a centralized architecture to take care of coding coefficients used at all the nodes. The coding

coefficients are generated at random and they are sent within the packet headers [5]. Although RNC has several theoretical benefits, it has three main limitations for the practical applications of network coding: high complexity, high delay, and high overhead. The high decoding complexity is due to the use of Gaussian elimination (GE) algorithm. In order to solve a system of linear equations, GE needs to solve a densely filled decoding matrix. This results into the high decoding complexity. The high delay is due to the time the receiver waits for the arrival of the complete block in order to start the decoding process and the high overhead is due to the coding coefficients, which are attached as a side information with the coded packets [5] in practical network coding schemes. Therefore, RNC is not the most efficient form of network coding and because of all the aforementioned constraints, it can adversely effect the applications like real-time video streaming, where there can be impairments in video quality due to high complexity and delay.

Recently, systematic network coding (SNC) [6]-[14] has been investigated as a powerful practical network coding solution. If systematic coding is used, the sinks can receive both uncoded and coded packets. There are three main benefits of using systematic coding. Firstly, the sink has to decode only the packets, which have not arrived in their original form. Hence, some rows of the decoding matrix are singleton and contain only one non-zero element. In this case, the decoding is done over a sparse decoding matrix that contains several zero elements that reduces the decoding complexity significantly. Secondly, the sink does not have to wait for complete block to start recovering packets. The packets that are received in their original form, are recovered instantly, which decreases the overall per-packet delay. Finally, the systematic packets do not have overhead of coding coefficients, as these are not the encoded packets. This reduces the overall overhead significantly. Therefore, SNC can overcome all the limitations imposed by RNC. In this work, we explore the benefits of using SNC for transmission over satellite.

#### A. Network coding over satellite

In the literature, there are specifically two directions of work on the use of network coding for transmission over satellite. The first direction focuses on throughput improvement. In [15][16], network coding is shown to enhance throughput by load balancing and allocating coded packets across different beams in multi-beam satellite systems. Further, works in [17][18] take advantage of orthogonal transmission available

using multi-link reception in multi-beam satellite systems. In [17][18], network coding is shown to provide benefits in terms of higher throughputs subject to the location of user terminals. In [19][20], network coding is shown to improve end-user quality-of-experience for video streaming application over relay satellites.

A different direction looks at the application of network coding to counteract packet losses and guarantee higher reliability. It has been shown that network coding together with congestion control algorithm can provide many-fold improvements than existing transport layer protocols [21]. In addition, it has been shown in [22], that unequal-protection aware overlapping network coding together with congestion control algorithms can provide improvements in quality-of-experience (QoE) of video streaming. Further, network coding implementation at link layer has shown to provide several advantages in terms of reliability, complexity, delay, etc. for unicasting over satellite networks with several intermediate nodes [11]-[13]. In [23], it is also shown that it can be used to cope with packet losses, thus counteracting prediction failures of the handover procedures in smart gateway diversity satellite systems.

### B. Our contributions

Existing results on the use of SNC has dealt mainly with unicast with a source and a sink connected via several intermediate nodes. In this paper, we extend existing results and investigate the use of SNC for multicast and multi-unicast over satellite. We identify two regions of transmission, one for multicast and another for multi-unicast with one in-network (re)encoding node to increase the achievable rates and the reliability of satellite networks. The one in-network re-encoding node can be a satellite gateway or other node at satellite ground station.

Further, we also present the application of network coding in two different sets of layers of a satellite network protocol architecture. These are satellite-dependent layers (link and physical layers) and satellite-independent layers (application, transport and IP layers). In particular, the satellite-dependent layers are of interest for the system operators while the satellite-independent layers are of interest for application developers who have access to the data flowing in these layers but not to the system operational kernel. Specifically, we present first the implementation of network coding at the link layer of the satellite systems and second, the implementation of network coding at the application layer for better internet communication over satellite systems. The overall architectural design is also presented showing the practical feasibility of SNC over different layers.

The rest of the paper is organized as follows. In Section II, we formalize the system model. In Section III, we describe the proposed SNC scheme for multicast and multi-unicast transmission. In Section IV, we present the theoretical analysis and derivation of theoretical expressions for the reliability and the achievable rate. Section V and Section VI present the implementation of SNC at the link layer and application layer respectively. Section VII presents the simulation results and Section VIII concludes this paper.

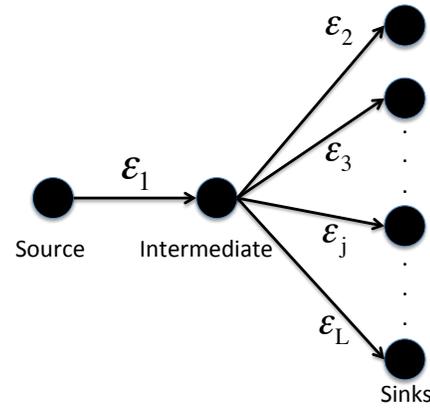


Figure. 1: System model.

## II. SYSTEM MODEL

We consider a system topology (Figure 1), where a source is connected to all the sinks via an intermediate node. This system topology is a relevant case in satellite systems where there is one intermediate node, which could be a gateway (or others) and there can be several sink nodes which are the users distributed in a large geographical area undergoing different packet loss rates. Our theoretical derivations and simulation results on the reliability and the achievable rates show the benefits of network coding with respect to state-of-the-art end-to-end forward erasure codes (FEC) codes like Reed-Solomon (RS) codes for both multicast and multi-unicast.

Consider that a source node has  $K$  data packets to send to  $L - 1$  sink nodes. Each packet is a column vector of length  $M$  over a finite field  $\mathbb{F}_q$ . The set of data packets in matrix notation is  $\mathbf{S} = [\mathbf{s}_1 \ \mathbf{s}_2 \ \dots \ \mathbf{s}_K]$ , where  $\mathbf{s}_t$  is the  $t^{\text{th}}$  data packet. The source is connected to all the sinks via an intermediate node as shown in Figure 1. All the links are modeled as memoryless erasure channels. There are  $L$  links in the network. The erasure probability from the source to the intermediate node is denoted by  $\epsilon_1$  and the erasure probability from the intermediate node to the sink node  $j$  is denoted by  $\epsilon_j, j = 2, \dots, L$ .

We assume there is no feedback from the sinks (or from the intermediate node) due to the inherent large latency of satellite systems. We also assume that packet transmissions occur at discrete time slots such that each node can transmit one packet per time slot. We will also assume that the coding schemes run for a total of  $N$  time slots ( $N$  is larger than or equal to  $K$ ) and every node (except the sinks) transmits a packet in each time slot  $t = 1, 2, \dots, N$ .

## III. SYSTEMATIC NETWORK CODING FOR MULTICAST AND MULTI-UNICAST

### A. Encoding at the source node

The SNC encoder sends  $K$  data packets in the first  $K$  time slots (systematic phase) followed by  $N - K$  random linear combinations of data packets in the next  $N - K$  time slots (non-systematic phase). Let,

$$\mathbf{X} = \mathbf{S}\mathbf{G}$$

represent  $K$  systematic packets and  $N - K$  coded packets transmitted by the SNC encoder during  $N$  consecutive time slots. The generator matrix

$$\mathbf{G} = [ \mathbf{I}_K \quad \mathbf{C} ]$$

consists of the identity matrix  $\mathbf{I}_K$  of dimension  $K$  and  $\mathbf{C} \in \mathbb{F}_q^{K \times N-K}$  with elements chosen randomly from a finite field  $\mathbb{F}_q$ . The code rate is given by  $\rho = \frac{K}{N}$ .

### B. Re-encoding at the intermediate node

The SNC re-encoder performs re-encoding operations at every time slot and sends  $N$  packets to the sink nodes. Let,

$$\mathbf{X}_I = \mathbf{X}\mathbf{D}_1\mathbf{T}$$

represent  $N$  packets transmitted by the SNC re-encoder during  $N$  consecutive time slots, where  $\mathbf{D}_1 \in \mathbb{F}_q^{N \times N}$  represents erasures from the source node to the intermediate node and  $\mathbf{T} \in \mathbb{F}_q^{N \times N}$  represents the re-encoding operations at the intermediate node.

The erasure matrix  $\mathbf{D}_1$  is an  $N \times N$  diagonal matrix with every diagonal component zero with probability  $\epsilon_1$  and one with probability  $1 - \epsilon_1$ .

The re-encoding matrix  $\mathbf{T}$  is modeled as an upper triangular matrix. The non-zero elements of  $\mathbf{T}$  are selected as follows. During the systematic phase, if a packet  $\mathbf{s}_t$  is lost, i.e.,  $\mathbf{D}_1(t, t) = 0$  then the non-zero elements of the  $t^{\text{th}}$  column of matrix  $\mathbf{T}$  are randomly selected from  $\mathbb{F}_q$ . This represents that if the systematic packet is lost from the source node to the intermediate node, then the intermediate node transmits a random linear combination of the packets stored in its buffer. If a packet  $\mathbf{s}_t$  is not lost, i.e.,  $\mathbf{D}_1(t, t) = 1$  then the  $t^{\text{th}}$  column of matrix  $\mathbf{T}$  is the same as the  $t^{\text{th}}$  column of identity matrix  $\mathbf{I}_N$ . This represents that the intermediate node forwards this systematic packet to the sinks. During the non-systematic phase, the intermediate node sends a random linear combination of the packets stored in its buffer and all the non-zero elements of last  $N - K$  columns of  $\mathbf{T}$  are chosen randomly from the finite field  $\mathbb{F}_q$ .

### C. Decoding at the sink nodes

Let  $\mathbf{Y}_j = \mathbf{X}_I\mathbf{D}_j$ ,  $j = 2, 3, \dots, L$  represents  $N$  packets received by the sink node  $j$  where  $\mathbf{D}_j$  represents erasures from the intermediate node to the sink node  $j$ .  $\mathbf{D}_j$  is  $N \times N$  diagonal matrix of the same type as  $\mathbf{D}_1$  but with erasure probability  $\epsilon_j$ . If the sink node  $j$  does not receive any packet in time slot  $t$  then the  $t^{\text{th}}$  column of  $\mathbf{Y}_j$  is a zero column.

The overall SNC coding strategy can be expressed using a linear operation channel (LOC) model, where the output at the sink node  $j$  is  $\mathbf{Y}_j = \mathbf{S}\mathbf{G}\mathbf{H}_j$  where  $\mathbf{H}_j = \mathbf{D}_1\mathbf{T}\mathbf{D}_j$  represents the transfer matrix from the source to the sink  $j$ . We assume that the coding vectors are attached in the packet headers so that the matrix  $\mathbf{G}\mathbf{H}_j$  is known at the sink  $j$ . However, the overhead, due to the attached coding vectors, is

kept low due to the use of systematic coding (coding vectors are not attached with the systematic packets). The decoding is progressive using the Gaussian Jordan algorithm as in [11]. All the  $K$  data packets are recovered when  $K$  innovative packets are received at the sink  $j$ , i.e.,  $\text{rank}(\mathbf{G}\mathbf{H}_j) = K$ .

## IV. THEORETICAL ANALYSIS

In this section, we present the theoretical expressions of the average reliability and the average achievable rate of the considered topology. First, let us note that the capacity of the topology is the min-cut of the network, which is given by  $\min_j (1 - \epsilon_j)$ .

Let us now define  $\eta$  as the residual erasure rate of any link that could be achieved after the overall coding and decoding operations. The reliability of the link is given by  $(1 - \eta)$ . Based on the definitions of the residual erasure rate and using the definition of achievable rate from [25], we define the average achievable rate of the considered topology as,

$$R_{av} = \rho(1 - \eta_{av}) \quad (1)$$

with,

$$\eta_{av} = \frac{1}{L-1} \sum_{j=2}^L [1 - (1 - \eta_1)(1 - \eta_j)] \quad (2)$$

as the average reliability of the considered topology where  $\eta_1$  is the residual erasure rate from the source node to the intermediate node and  $\eta_j$  is the residual erasure rate from the intermediate node to the sink node  $j$ . Here,

$$\eta_l = \epsilon_l(\phi_{l1} + \phi_{l2}), l = 1, 2, \dots, L. \quad (3)$$

where  $\epsilon_l$  is the probability by which a systematic packet is lost at link  $l$ .

Let us now derive the residual erasure rate  $\eta_l$ . Note that the residual erasure rate is zero if the decoding is successful and all the data packets are recovered. Further, a decoding failure will result into a finite value of residual erasure rate. The decoding failure can be due to two reasons. First, when the number of total packets received is less than  $K$ . The probability of this event is represented as,

$$\phi_{l1} = Pr(A < K - 1) \quad (4)$$

where  $A$  is the total number of packets received. Second, when the number of total packets received is more than  $K$  but they are not linearly independent to have successful decoding. This event corresponds to a not full rank decoding matrix and its probability is represented as,

$$\phi_{l2} = Pr(A \geq K - 1)Pr(\text{rank}(\mathbf{G}\mathbf{H}_j) < K) \quad (5)$$

Assuming Bernoulli distributed erasures and using the probability mass function  $\alpha(f, v, p) = \binom{v}{f} (1-p)^f (p)^{v-f}$ , we have,

$$\phi_{l1} = \sum_{f=0}^{K-1} \alpha(f, N-1, \epsilon_l), \quad (6)$$

$$\phi_{t2} = \sum_{f_1=0}^{K-1} \alpha(f_1, K, \epsilon_l) \sum_{f_2=K-f_1}^{N-K} \alpha(f_2, N-K, \epsilon_l) (1 - \pi(f_2, K-f_1)). \quad (7)$$

where  $f_1, f_2$  are the number of systematic and coded packets received, respectively, and  $\pi(f_2, K-f_1)$  is the probability of  $f_2$  coded packets having  $K-f_1$  degrees of freedom. When there are  $f_1$  degrees of freedom from  $f_1$  systematic packets and  $K-f_1$  degrees of freedom from  $f_2$  coded packets then  $\text{rank}(\mathbf{GH}) = K$ . Using (equation 7, [26]), we have the exact expression for  $\pi(f_2, K-f_1) = \prod_{f_3=0}^{K-f_1-1} 1 - q^{f_3-f_2}$ . From the above expressions, we can obtain the average residual erasure rate and average achievable rate using (2).

In the next sections, we will present the practical application of network coding at different layers of the satellite systems. In particular, we consider the application at link and application layers of the satellite systems.

## V. IMPLEMENTATION IN LINK LAYER OF SATELLITE SYSTEMS

### A. State-of-the-art link layer protocols in satellite systems

The current state-of-the-art link layer protocols in the satellite systems provide efficient encapsulation of network layer (IP) protocol data units (PDUs) over the physical layer frames. For example, generic stream encapsulation (GSE) protocol [27] in digital video broadcasting by satellite - second generation (DVB-S2) based systems is used as a link layer protocol to encapsulate network layer IP packets.

The existing link layer forward erasure correction (LL-FEC) frameworks in the satellite systems are mainly based on RS or Raptor codes [28]. However, the main limitation of the existing frameworks is that they operate only in end-to-end fashion and do not utilize the coding opportunities at the intermediate node. In this section, we will present an architectural and encapsulation framework to enable link layer systematic network coding (LL-SNC) at the source and at the intermediate node of the satellite systems.

### B. LL-SNC architecture and encapsulation

In Figure 2, we present the complete information flow with LL-SNC architecture and LL-SNC encapsulation, where IP packets are transmitted from the source and recovered at the sinks. This figure represents the case when there is only one sink in the network. When there are several sinks, the same LL PDUs are transmitted from the intermediate node to all the sink nodes.

At the source, the network layer IP packets are encapsulated into an LL-SNC frame. The LL-SNC frame consists of an application data table (ADT) to store IP PDUs, a network coding data table (NCDT) to store network coded packets and a coefficient data table (CDT) to store coding coefficients. The data from the LL-SNC frame is then encapsulated into LL

PDUs. The LL PDUs are then encapsulated into the physical (PHY) frames.

At the intermediate node, the payload of correctly received LL PDUs is stored in the LL-SNC frame. The IP PDUs are stored in the ADT of the intermediate node. The coded packets and the corresponding coefficients are stored in NCDT and CDT of the intermediate node. When the intermediate node receives LL PDU without error, it sends the LL PDU to the sink node and also stores it in the LL-SNC frame. When the intermediate node receives LL PDU with errors, it discards the LL PDU and generates new coded packet and coding coefficients as explained in Section III. These new coded packets and the corresponding coefficients are stored in NCDT and CDT of the intermediate node.

At the sink node, the correctly received LL PDUs are stored in the LL-SNC frame. The IP PDUs are stored in the ADT of the sink node. The coded packets and the coding coefficients are stored in NCDT and CDT, respectively. The progressive decoding is performed and the lost IP PDUs are recovered. These IP PDUs are then passed to the upper layers.

The encapsulation process is similar to the encapsulation of LL-FEC over GSE [28] but with some modifications to accommodate network coding parameters. A source block (ADT) consists of  $K$  columns and stores ADUs. Now, ADUs are arranged column wise starting from the upper left corner. If an ADU does not fit in one column, it continues at the top of the following column and so on. If the ADT is not completely filled then the zero-padding bytes are inserted in last column to fill it completely. Each ADU is then encapsulated in a single or multiple RTP packets. The FEC block (NCDT) contains  $N-K$  columns with  $N-K$  coded packets and the coefficient block (CDT) contains  $N-K$  columns with  $N-K$  set of coding coefficients. Each coded packet from NCDT and the corresponding coding coefficients from CDT are encapsulated in one RTP packet. The first  $K$  bytes of RTP payload contain  $K$  coding coefficients followed by the corresponding NCDT column. The value of  $K$  is signaled through the RTP header of the RTP packet. Finally, CRC-32 is added with every RTP packet to detect errors in RTP packets at the receiving end. Now, if an ADU is lost, then the corresponding part of the column or the complete column is also lost. The progressive decoding is performed and lost columns (or lost part of columns) in ADT are filled with the recovered data.

## VI. IMPLEMENTATION IN APPLICATION LAYER OF SATELLITE SYSTEMS

### A. State-of-the-art application layer protocols in satellite systems

The current state-of-the-art application/transport layer protocols primarily include transmission control protocol (TCP) based schemes, which are most commonly used to guarantee reliability. TCP used retransmission mechanism to recover from packet losses. However, TCP and TCP variants have limited performance over satellite due to long round-trip times (RTT) [31]. In addition, for the real-time multimedia delivery, specifically for audio and video transmission, TCP is not

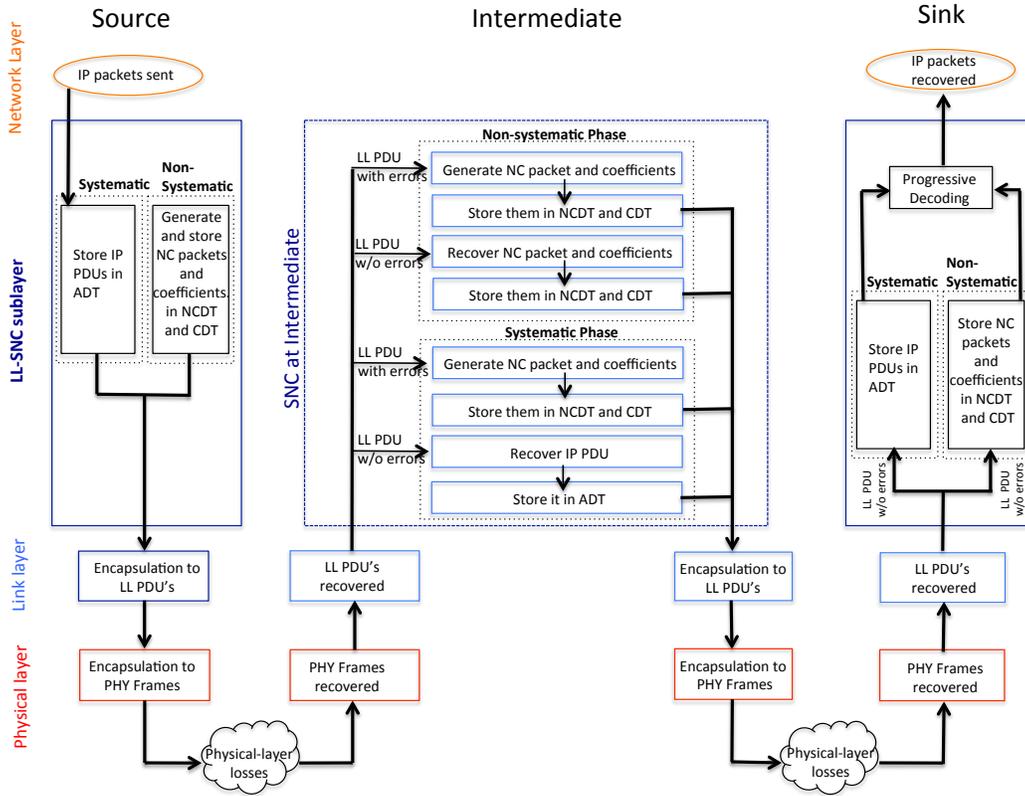


Figure. 2: Flow diagram with LL-SNC architecture and LL-SNC encapsulation in satellite system.

suitable for timely delivery as it requires heavy feedback and acknowledgements [32], [33], [34]. Another disadvantage of TCP is its compatibility with the multicast and broadcast communication. There may exist parallel TCP connections with the sender to different receivers, but each TCP connection can require transmission of different lost packets that will result into a resource-intensive solution.

Therefore, most of the real-time protocols for the multimedia delivery are built on the top of user datagram protocol (UDP). As the reliability is not guaranteed with UDP, forward erasure correction (FEC) codes are used to recover from packet losses. Specifically, Reed-Solomon (RS) code, which is a maximum distance separable (MDS) code, is optimal in terms of erasure correction performance and mainly used as a start-of-the-art FEC scheme. However, there are mainly three limitations of using RS codes. Firstly, a construction of the RS code is based on a finite algebraic arithmetic, therefore, the receiver has to wait for all the packets to start the decoding process. Secondly, the constraints on the RS coding parameters, make its incompatible with adaptive solutions and finally, when there are multiple hops where intermediate nodes can perform coding, they have to first decode the original packets and then re-encode to generate the coded packets. This results into an additional delay because of decoding and then re-encoding the original packets at every node. In this section, we will present an architectural and encapsulation framework to enable application layer systematic network coding (AL-SNC) at the source and at the intermediate node of the satellite systems.

### B. AL-SNC architecture and encapsulation

The encapsulation process is similar to the encapsulation of RS codes over real-time transport protocol (RTP) [30]. In addition, there are some modifications to accommodate network coding parameters. In Figure 3, we present the modified encapsulation process to be used for the network coding. An application layer source block (AL-ADT) consists of  $K$  ADUs in  $K$  columns. The number of rows in the source block is  $M = E + 2$ , where  $E$  is the length of the largest ADU. The columns, which do not have the largest ADU, are filled with zeros to be completely filled. Each column can be considered as a data packet. The first two bytes of each column in the source block contain the length of the corresponding ADU. ADUs are then encapsulated into RTP packets. The first two bytes and the zero paddings are not sent over the network. The application layer FEC block (AL-NCDT) contains  $N - K$  columns with  $N - K$  coded packets and the application layer coefficient block (AL-CDT) contains  $N - K$  columns with  $N - K$  set of coding coefficients. FEC packets and coefficients are then encapsulated into RTP packets. Each RTP packet contains RTP payload, RTP header and FEC payload ID. This FEC payload ID is used for signaling the coding parameters like source block ID, FEC packet ID, values of  $K$  and  $N$ , etc. The CRC-32 is added with every RTP packet to detect errors in RTP packets at the receiving end. At the receiver, the values of coding parameters are extracted from the FEC payload ID. Now, if ADUs are lost then the complete columns are lost. So, if FEC decoding succeeds, the receiver recovers

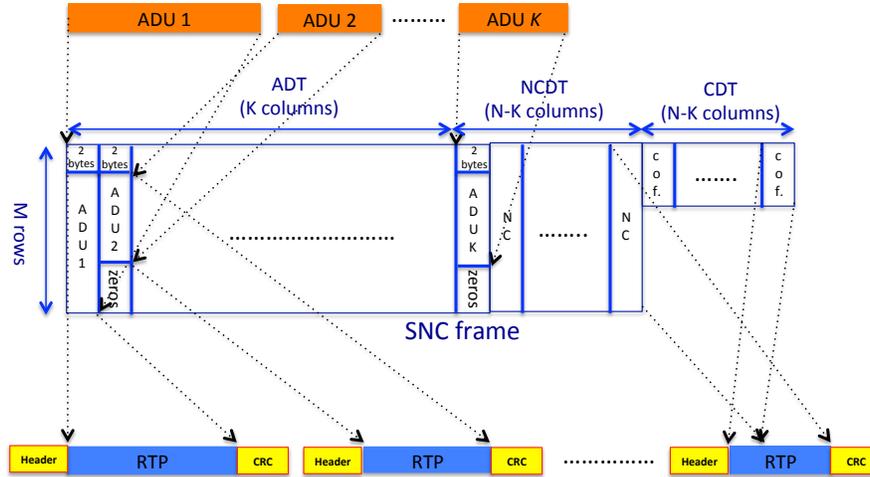


Figure. 3: Encapsulation of application layer data units over RTP with network coding.

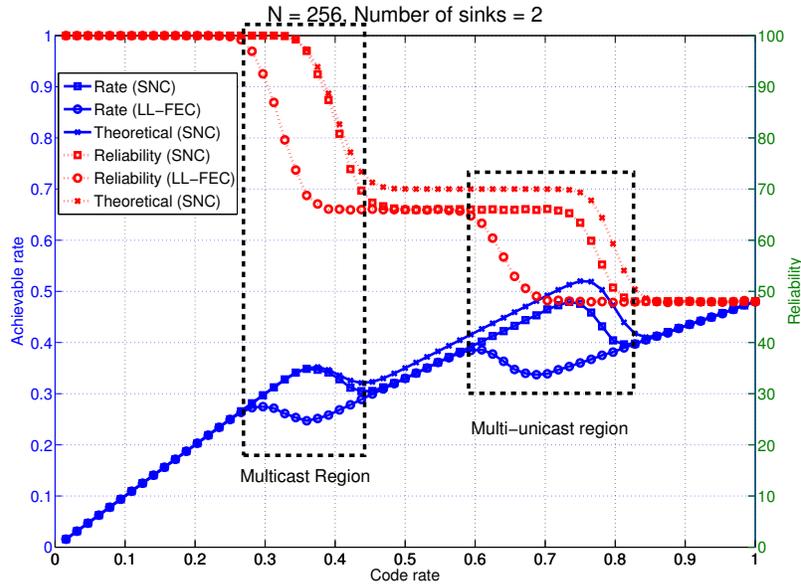


Figure. 4: LL-SNC multicast and multi-unicast region with two sinks.

ADUs by filling the erased columns. The initial two bytes are used to remove zero padding from the data packets to recover the ADUs.

## VII. SIMULATION RESULTS

### A. Performance metrics

1) *Achievable rates and reliability*: The theoretical expressions of the average achievable rate and the average reliability are derived in Section IV. In this section, we will present the simulation results on the average achievable rate and the average reliability. In the results, we also compare the simulation results with the theoretical expressions derived in Section IV.

2) *Average delay per-packet*: If a packet  $s_t$  is transmitted by the source at time  $t_j$  and it is recovered at the sink at time  $t_r$  then packet  $s_t$  incurs a delay  $\delta_t$  where,  $\delta_t = t_r - t_j$ . For

TABLE I: MAXIMUM ACHIEVABLE RATES FOR MULTI-CAST AND MULTI-UNICAST

N	Application	Sinks	LL-SNC	LL-FEC	Gain
256	Multicast	2	0.3486	0.2747	26.90%
256	Multi-unicast	2	0.4794	0.3858	24.26%
256	Multicast	10	0.3575	0.3043	17.48%
256	Multi-unicast	10	0.6745	0.5404	24.81%
50	Multicast	2	0.3035	0.2553	18.88%
50	Multi-unicast	2	0.3852	0.3579	7.63%
50	Multicast	10	0.3107	0.2751	12.94%
50	Multi-unicast	10	0.5406	0.4990	8.34%

the block of  $K$  packets, the average delay per-packet is given as,  $\Delta = \frac{\sum_{t=1}^K \delta_t}{K}$ . Note that the delay is evaluated only for the packets that are recovered at the sink.

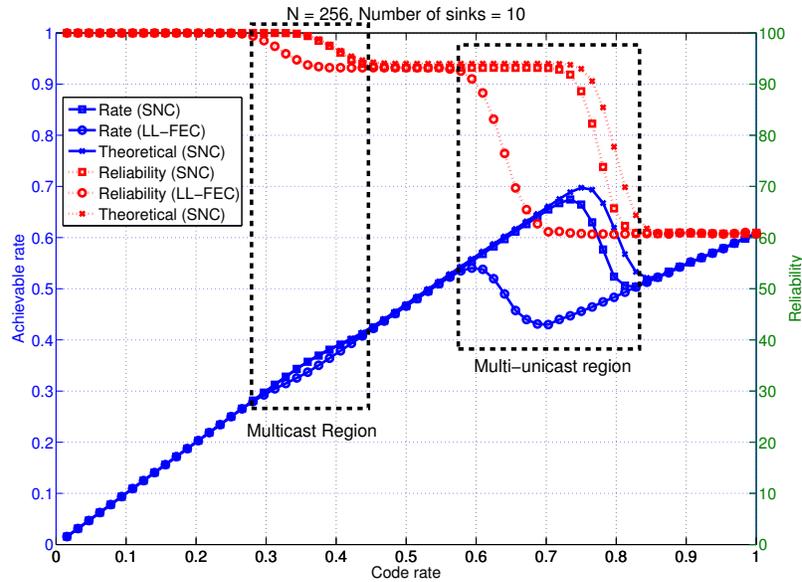


Figure 5: LL-SNC multicast and multi-unicast region with two sinks.

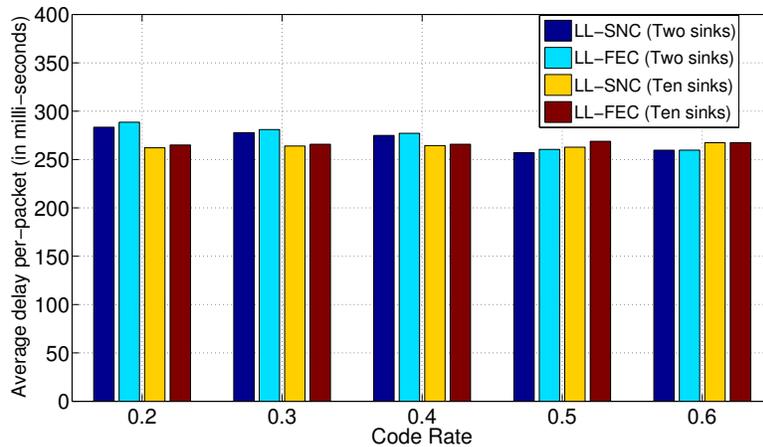


Figure 6: Average delay per-packet

### B. Results for the implementation at link layer

1) *Simulation setup:* In our simulation setup, first we consider network coding implementation at the link layer. We consider realistic satellite transmission scenarios with links having light rainfall (erasure rate of 0.2) and/or heavy rainfall (erasure rate of 0.6) [29]. These erasures correspond to the loss of LL-PDUs. In each case, we compare LL-SNC with LL-FEC. We assume IP PDUs of length 1500 bytes. Each IP PDU is mapped to a column of the ADTs of consecutive LL-SNC frames. Two LL-SNC frame lengths,  $N \in \{50, 256\}$  and several values of code rates are considered for comparison. The size of ADT, i.e.,  $K$  changes with the code rate. We set the physical layer symbol rate of  $B_s = 27.5$  Mbaud/s,  $\zeta = 2$  as the modulation constellation and  $r_{phy} = 1/2$  as the physical coding rate such that the bit rate is  $B_s \zeta r_{phy} = 27.5$  Mbps. The transmission delay is set to be 250 ms. In each case, we average over 1000 experiments for every performance metric.

The number of erasures per-frame varies (according to the random erasure rate) between 1000 experiments.

2) *Results:* In Figure 4, we show the results on achievable rates and reliability when there are two sinks in the network. The multicast capacity of the network is limited by one of the links joining the intermediate node to the sink node. We consider the following erasure probabilities:  $\epsilon_1 = 0.2$ ,  $\epsilon_2 = 0.2$  and  $\epsilon_3 = 0.6$ . In Figure 5, we show the results on achievable rates and reliability when there are ten sinks in the network. The multicast capacity of the network is limited by one of the links joining the intermediate node to the sink node. We consider the following erasure probabilities:  $\epsilon_1 = 0.2$ ,  $\epsilon_j = 0.2, j = 1, 2, \dots, 9$  and  $\epsilon_{10} = 0.6$ . Furthermore, we present the maximum achievable rates for both multicast and multi-unicast in Table I. Following are the key conclusions from these results:

- We have identified two regions in these graphs: one for

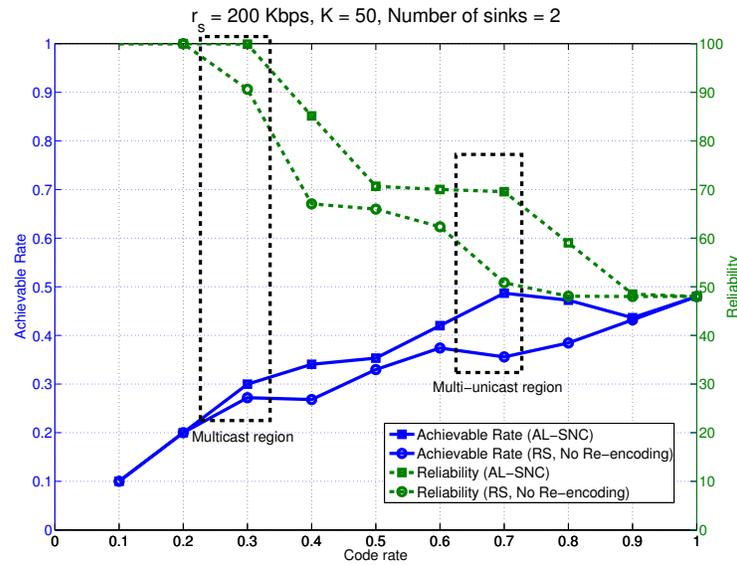
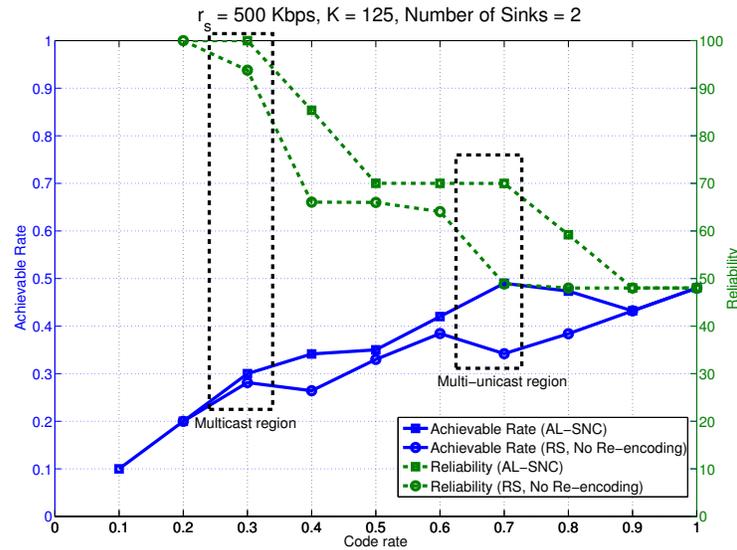

 (a) AL-SNC multicast and multi-unicast regions with video codec rate  $r_s = 200$  Kbps.

 (b) AL-SNC multicast and multi-unicast regions with video codec rate  $r_s = 500$  Kbps.

Figure. 7: AL-SNC multicast and multi-unicast regions with different video codec rates.

multicast, and another for multi-unicast (represented by dashed boxes). The region for multicast is corresponding to the case when the average reliability approaches 100%. This would mean that all the sinks in the network are able to recover all the data packets. We have also identified the multi-unicast region where the sinks with better channel recover all the data packets and the sink(s) with bad channel still suffer from some losses. The benefit of multi-unicast over multicast is that one can achieve overall higher transmission rates by not sacrificing the rate due to the bottleneck sink (link with higher erasure rate). Hence, based on the requirements of the users, our results provide optimal usage of available bandwidth for transmission.

- The multicast is feasible only when the code rate is smaller than the capacity of the network, which is  $\min_j(1 - \epsilon_j) = 0.4$ . However, when the code rate is higher than the multicast capacity, multi-unicast is feasible. This is because the capacity of the sinks with good channel is different and higher than the capacity of the network (in our example it is 0.8). Therefore, when the code rate is smaller than 0.8, the sinks with good channel can recover all the data packets making multi-unicast feasible.
- LL-SNC provides higher transmission rates and higher reliability than LL-FEC in all the cases. When the number of sink increases, LL-SNC can provide close to 100% reliability in the multi-unicast region itself. This is because there is only one bottleneck link in the network and only

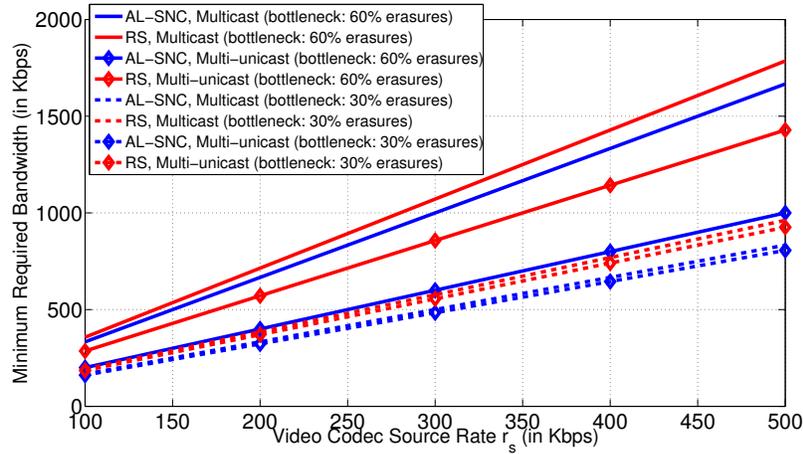


Figure. 8: Minimum network bandwidth requirement for multicast and multi-unicast with AL-SNC.

one sink suffers from the bad channel. In this case, it would be efficient to transmit in the multi-unicast region such that the higher transmission rates are achieved and almost all the sinks (except the one with the bad channel) are able to recover all the data packets. Furthermore, LL-SNC also provides higher maximum achievable rates than LL-FEC for both multicast and multi-unicast. The maximum achievable rate increases as the frame length increases or the number of sinks increases. Our results (Table I) show that LL-SNC can achieve up to 26.90% and 24.26% higher maximum achievable rates than LL-FEC for multicast and multi-unicast respectively.

In Figure 6, our simulation results also show that LL-SNC provides smaller average delay per-packet than LL-FEC. This is because of the following two reasons. First, the progressive decoding in LL-SNC allows the sinks to start decoding and recovering as soon as it receives the first packet. Second, the re-encoding in LL-SNC helps the sink to receive  $K$  degrees of freedom and complete the decoding process in fewer time slots than LL-FEC. The overall delay includes the inherent transmission delay of 250 ms of the satellite systems. Note that the average per-packet delay for LL-SNC and LL-FEC are very close. This is due to the fact that the transmission delay majorly contributes to the overall delay and per-packet delay is smaller due to the higher rates. However, at the lower rates, when per-packet delay is higher, LL-SNC is expected to provide higher advantage than LL-FEC.

### C. Results for the implementation at application layer

1) *Simulation setup:* For the case of network coding implementation at application layer, we also consider different cases with erasure rates of 0.2 and 0.6. These erasures correspond to the loss of RTP packets. We assume that the source generates packets of length  $M = 1500$  symbols over Galois Field with size  $q = 2^8$ . We consider state-of-the-art video codecs with video frames grouped into Groups of Pictures (GoPs). Each GoP contains several packets. The codec outputs each GoP in a fixed time  $T_{GoP}$  such that the source block is  $K = \left\lfloor \frac{r_s \times T_{GoP}}{M \times 8} \right\rfloor$  where  $r_s$  is the video codec source rate. Several values of code

rates are considered for comparison. The size of coded block, i.e.,  $N$  varies with the change in code rate. In each case, we average over 1000 experiments for every performance metrics.

2) *Results:* Figure 7 shows the achievable rates for the application of network coding at application layer. Two configurations of codecs are considered with  $r_s = 200$  kbps and  $r_s = 500$  kbps with  $T_{GoP} = 3$  seconds. The source blocks corresponding to these configurations are  $K = 50$  and  $K = 125$ , respectively. We show the multicast and multi-unicast region of transmission in these figures.

The main difference between the simulation of link and application layer network coding implementation is the variation of block sizes and frame sizes. In case of link layer implementation, the frame size ( $N$ ) is constant ( $K$  changes with the code rate). In case of application layer implementation, the block size ( $K$ ) is constant ( $N$  changes with the code rate). Here, the block size depends on the codec rate, GoP time and packet sizes. In this paper, we have presented results with  $K$  changing with codec rates. However, the conclusions remain the same with the variations in packet sizes and GoP time.

Finally, we also show the minimum network bandwidth required for enabling multicasting and multi-unicasting at application layer. The minimum network bandwidth required is calculated as the ratio of video codec source rate and achievable rate. We consider several video codec source rates ranging from 100 Kbps to 500 Kbps. We consider two erasure rates for the bottleneck (worst) link: 30% erasures and 60% erasures. It is shown that to achieve 500 Kbps video codec rate, it is required to have at least 1.8 Mbps network bandwidth for multicasting (100% average reliability). Note that 500 Kbps is used for the video code source rate, and the rest of the available network bandwidth is used for the network coding rate to counteract packet losses and to guarantee 100% reliability. In addition, with the help of SNC, the required network bandwidth can be reduced from 1.786 Mbps to 1.667 Mbps in case of multicasting. This is equivalent to save around 6.66% in network bandwidth when SNC is used instead of RS coding. In case of multi-unicasting, the gain is even higher where SNC helps to reduce the network bandwidth requirement from 1.429

Mbps to 1 Mbps (around 30.02% save in bandwidth).

### VIII. CONCLUSIONS

In this paper, we have focussed on the use of SNC for multicast and multi-unicast over satellite. We have identified the transmission regions for multicast and multi-unicast over satellite by characterizing the reliability and achievable rates offered by SNC in these two different regions. We have derived the theoretical expressions for the average reliability and the average achievable rate of the considered topology. Our theoretical and simulation analysis present the benefits of SNC over end-to-end coding for both multicast and multi-unicast. Our results have shown that a higher rate is achievable for the multi-unicast however, not all the users in multi-unicast can recover all the data packets. Therefore, based on the requirements from different users, the transmission region can be chosen for the optimal usage of available bandwidth. Finally, we have explored the benefits of network coding at different layers of the satellite network protocol stack. We have shown the encapsulation and architecture feasibility of network coding application in the satellite-dependent layers and satellite-independent layers. Future work includes the investigation of SNC on more complex networks such as network with multiple sources. Several other factors like bursty erasures, video codec characteristics, processing complexity, etc should be taken into account while analyzing the performance of network coding schemes for multicasting.

### REFERENCES

- [1] P. Saxena and M. A. Vázquez-Castro, "Network coding multicast and multi-unicast over satellite," in the Seventh International Conference on Advances in Satellite and Space Communication (SPACOMM), 2015, pp. 40-45.
- [2] R. Ahlswede, N. Cai, S. Yen Robert Li, and R.W. Yeung, "Network information flow," IEEE Transactions on Information Theory, vol. 46, no. 4, 2000, pp. 1204-1216.
- [3] S. Yen Li, R. W. Yeung, and N. Cai, "Linear network coding," IEEE Transactions on Information Theory, vol. 49, 2003, pp. 371-381.
- [4] T. Ho et al., "A random linear network coding approach to multicast," Information Theory, IEEE Transactions on, vol. 52, no. 10, 2006, pp. 4413- 4430.
- [5] P. A. Chou, Y. Wu, and K. Jain, "Practical network coding," 41st Annu. Allerton Conf. Communication, Control and Computing, 2003.
- [6] J. Heide, M. Pedersen, F. H. P. Fitzek, and T. Larsen, "Network coding for mobile devices - systematic binary random rateless codes," in Communications Workshops, 2009. ICC Workshops 2009. IEEE International Conference on, 2009, pp. 1-6.
- [7] Y. Li, P. Vingelmann, M. Pedersen, and E. Soljanin, "Round-robin streaming with generations," in Network Coding (NetCod), 2012 International Symposium on, 2012, pp. 143-148.
- [8] B. Shrader and N. Jones, "Systematic wireless network coding," in Military Communications Conference, 2009. MILCOM 2009. IEEE, 2009, pp. 1-7.
- [9] D. Vukobratovic, C. Khirallah, V. Stankovic, and J. Thompson, "Random network coding for multimedia delivery services in LTE/LTE-Advanced," Multimedia, IEEE Transactions on, vol. 16, 2014, pp. 277-282.
- [10] S. Teerapittayanon et al., "Network Coding as a WiMAX Link Reliability Mechanism," in Multiple Access Communications, vol. 7642 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2012, pp. 1-12.
- [11] P. Saxena and M. A. Vázquez-Castro, "Network coding advantage over MDS codes for multimedia transmission via erasure satellite channels," in lecture notes of the institute for computer sciences, social informatics and telecommunications engineering, (Springer 2013), Volume 123, 2013, pp. 199-210.
- [12] P. Saxena and M. A. Vázquez-Castro, "DARE: DoF-Aided Random Encoding for network coding over lossy line networks," IEEE communication letters, vol. 19, no. 8, 2015, pp. 1374-1377.
- [13] P. Saxena and M. A. Vázquez-Castro, "Link layer random network coding for DVB- S2X/RCS," IEEE Communication Letters, vol. 19, no. 7, 2015, pp. 1161-1164.
- [14] M. A. Vázquez-Castro and P. Saxena, "Network coding over satellite: from theory to design and performance," in proceedings of the 7th EAI International Conference on Wireless and Satellite Systems, Bradford (Britain), 2015.
- [15] F. Vieira, S. Shintre, and J. Barros, "How feasible is network coding in current satellite systems ?" in ASMS conf. and SPSC Workshop, 2010, pp. 31-37.
- [16] F. Vieira, D. Lucani, and N. Alagha, "Load-aware soft-handovers for multibeam satellites: A network coding perspective," in ASMS conf. and SPSC Workshop, 2012, pp. 189-196.
- [17] R. Alegre-Godoy, N. Alagha, and M. A. Vazquez-Castro, "Offered capacity optimization mechanisms for multi-beam satellite systems," in IEEE ICC, 2012, pp. 3180-3184.
- [18] M. A. Vazquez-Castro, "Graph model and network coding gain of multibeam satellite communications," in IEEE ICC, 2013, pp. 4293-4297.
- [19] S. Gupta and M. A. Vazquez-Castro, "Location-adaptive network-coded video transmission for improved quality-of-experience," in 31st AIAA International Communications Satellite Systems Conference (ICSSC), 2013.
- [20] S. Gupta, M. A. Pimentel-Niño and M. A. Vazquez-Castro, "Joint network coded/cross layer optimized video streaming over relay satellite channel," in 3rd international conference on wireless communications and mobile computing (MIC-WCMC), 2013.
- [21] J. Cloud, D. Leith and M. Medard, "Network Coded TCP (CTCP) Performance over Satellite Networks," in International conference on advances in satellite and space communications (SPACOMM), 2014, pp. 53-556.
- [22] M. A. Pimentel-Niño, P. Saxena, and M. A. Vazquez-Castro, "QoE driven adaptive video with overlapping network coding for best effort erasure satellite links," in 31st AIAA international communications satellite systems conference (ICSSC), 2013.
- [23] M. Muhammad, G. Giambene and T. De Cola, "Channel prediction and network coding for smart gateway diversity in terabit satellite networks," in GLOBECOMM, 2014, pp. 3549-3554.
- [24] D. S. Lun, M. Medard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks," Physical Communication, vol. 1, no. 1, 2008, pp. 3 - 20.
- [25] S. Yang and R. Yeung, "Coding for a network coded fountain," in Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on, 2011, pp. 2647-2651.
- [26] O. Trullols-Cruces, J. Barcelo-Ordinas, and M. Fiore, "Exact decoding probability under random linear network coding," IEEE Communications Letters, vol. 15, 2011, pp. 67-69.
- [27] "ETSI TS 102 606 V1.1.1, Digital Video Broadcasting (DVB); Generic Stream Encapsulation (GSE) Protocol," 2007.
- [28] "DVB BlueBook a155-2, Digital Video Broadcasting (DVB); Second Generation DVB Interactive Satellite System (DVB-RCS2); Part 2: Lower Layers for Satellite standard," 2013.
- [29] F. de Belleville, L. Dairaine, J. Lacan, and C. Fraboul, "Reliable multicast transport by satellite: A hybrid satellite/terrestrial solution with erasure codes," in High Speed Networks and Multimedia Communications, vol. 3079, Springer Berlin Heidelberg, 2004, pp. 436-445.
- [30] S. Galanos, O. Peck, and V. Roca, "RTP payload format for Reed Solomon FEC," Internet-Draft, 2011.
- [31] A. Pirovano and F. Garcia, "A new survey on improving tcp performances over geostationary satellite link," Network and Communication technologies, 2, 2013.
- [32] S. Floyd, M. Handly, J. Padhye, and J. Widmer. TCP Friendly Rate Control (TFRC): Protocol Specification, IETF RFC 5348, 2008.
- [33] H. Seferoglu and A. Markopoulou. Video-aware opportunistic network coding over wireless network. Journal of Selected Areas in Telecommunications (JSAT), 27:713-728, 2009.
- [34] H. P. Shiang and M. van der Schaar. A quality-centric TCP-Friendly congestion control for multimedia transmission. Multimedia, IEEE Transactions on, 14(3):896-909, 2012.

# A Method for Transforming Movement Paths in Wireless Mobile Network Simulation

Hawra Aloseef, John DeDourek, Przemyslaw Pochec  
 Faculty of Computer Science  
 University of New Brunswick  
 Fredericton, Canada  
 e-mail: {hawra.alseef, dedourek, pochec}@unb.ca

**Abstract**—Mobility pattern of nodes significantly affects performance of MANETs. We propose a new investigative approach for mobile network performance evaluation based on modifying the mobile node movement path between the waypoints. We implemented the modifications to the movements generated using the standard *setdest* utility for the ns2 simulator, replacing line movements between the waypoints with the new node movements along the curved paths generated using simple fractals. Applying the new path modification approach in a study of a sample MANET showed that only the node speed significantly affected MANET performance, and not the shape of the individual path segments between the waypoints.

**Keywords**—movement generator; network simulation; ns2; fractal path; MANET

## I. INTRODUCTION

Mobility [1] is an important aspect of operation of any Mobile Ad hoc Network (MANET), where mobile devices cooperate with each other by exchanging messages and forwarding data [2][3]. Mobile devices are linked together through wireless connections without infrastructure and can change locations and reconfigure network connections. During the lifetime of the network, nodes are free to move around within the network and node mobility plays a very important role in mobile ad hoc network performance. Mobility of mobile nodes significantly affects the performance of a MANET [3].

Mobile Medium, or Mobile Medium Ad hoc Network (M2ANET) [4], is a particular type of mobile network that affords a significant flexibility in choosing a movement pattern for its nodes. M2ANET consists of two types of nodes: a small number of user nodes that actively send data one to another, and a large cloud of mobile nodes that act as relays and forward the data between the user nodes along multi hop paths. The mobile nodes act as a mobile infrastructure and can be set up to provide the best service for the user nodes. When setting up M2ANET, the administrator chooses the number of nodes required, the routing/forwarding protocol to be used by the Mobile Medium nodes and the mobile behavior of the forwarding nodes. The mobile nodes for M2ANET can be implemented with autonomous aerial drones carrying radio transceivers. M2ANETS are not yet common and experimenting with

such a network would require controlling both physical objects in space (aerial drones) as well as data circulating in the network.

For preliminary evaluation of properties on the new M2ANETS network simulation can be used very effectively. Simulation allows for modelling existing networks as we as future networks. Using simulation, different mobile network configurations with mobile nodes following different movement patterns, working under different traffic load conditions and using different routing protocols, can be quickly and easily modelled and evaluated. For mobile networks, if no other design constraints are present, random motion of the nodes is usually used. Using a common and specific random motion model allows to create the base condition for comparison between different network evaluations.

ns2 is an open source simulator well suited for modelling wired and wireless networks [5]. It includes a motion scenario generator *setdest* designed to automatically generate random motion paths for a large number of nodes. This tool generates a random motion path for each node by selecting a random destination for the node and then moving the node towards this destination along a straight line. Once this destination is reached by the node, a new destination is randomly selected and, after an optional pause time, the node starts moving again to the new destination.

In this paper, we propose to use the random motion generated by the ns2 *setdest* utility [5] to create a new trajectory for the mobile nodes. The waypoints are kept the same but the path followed by the node between two waypoints is no longer defined by one straight line segment, and is replaced by a fractal curve composed of a number of shorter line segments. The manipulation of the node movement path for mobile networks has never been reported before. Incorporating this approach into mobile network simulation would allow to experiment with performance of similar mobile networks, checking how the performance would be affected by modifying the trajectories of mobile nodes, while keeping the waypoints the same. (The scenario is analogous to changing the flight paths of the aeroplanes while keeping the departure and destination cities the same.) The proposed approach, which is based on application of fractals, is particularly suitable for implementation in a discrete event simulator like ns2.

In Section II, we review different random movement models commonly used in simulation. Section III introduces properties of the Koch fractal. Section IV describes the new movement generator based on generation of the node movement along a fractal path. In Section V we introduce our simulation environment: the ns2 simulator. Section VI presents a study of two mobile networks: one with the conventional random movement and the other with the fractal movement. Conclusion is presented in Section VII.

## II. STATE OF THE ART

Any model of a MANET requires a mobility model specifying the movement pattern of the nodes [6]. The most realistic models are trace driven but cannot be always applied because of their *a posteriori* nature. On the other hand, the synthetic models [6][7] are not trace driven but instead rely on assumptions about the node movement mode. The synthetic models attempt to generate the next node movement based on a goal driven scenario (e.g., reaching a particular destination) and some physical constraints on the movement of the node. The constraints may be presented in terms of geographical restrictions. The most common example of these is the movement area defined in the simulator typically as a rectangular region with the nodes not able to breach the boundary. This restriction is particularly important to consider when nodes are designed to follow curved path between waypoints; the simulation must assure that the curved path falls within the simulated region. In more complex scenarios the node movement planning would also have to avoid obstacles. Another type of movement restriction is a temporal dependency: the next move of a node is affected by the past movement. Typically, this restriction incorporates the constraints from the real world object dynamics, where physical systems cannot accelerate, or change direction, at an arbitrary rate, and allows for the implementation of more realistic simulation scenarios. The synthetic models may also incorporate special dependencies: the movement of a node may depend on the movement of the other nodes around it. Two obvious scenarios could be considered: collision avoidance and group mobility. Group mobility is particularly attractive for MANETS where, in order for the nodes to maintain the communication links with each other, the nodes need to stay within the communication range of each other's transmitter.

The synthetic model most often used mobile network proof concept simulation is the random mobility model where the nodes move randomly and without restrictions and where the destination and the speed are also chosen randomly. A random mobility model typically implements a rudimentary geographical restriction and keeps all the nodes within a designated simulation region, typically a rectangle.

There are many different types of random mobility models that are used in MANETs. The main ones are the Random Walk, the Random Waypoint, and the Random

Direction. The Random Walk model [7] mimics the Brownian motion of particles found in nature. Each node travels in a straight direction for a specified time interval before randomly changing the speed and the direction, and then continuing for another time interval. In the Random Waypoint model [8], each node selects a destination within the simulation area and then follows a straight path to it; once the destination is reached the node may pause and then select a new destination (waypoint). In the Random Direction model [9], instead of selecting a random destination, the node selects a random direction and then moves along this direction until it reaches the simulation area boundary where, possibly after a pause, it selects a new direction for the next move.

The ns2 *setdest* utility generates the node movements following the Random Waypoint algorithm [5]. In the Random Waypoint Movement (RWP), each node moves from its randomly selected initial starting position towards the randomly selected at a randomly selected speed. Once at the target destination the node may pause for a randomly selected time, and then start the next random move. This process will be repeated until the end of the simulation by the ns2. One notable aspect of the RWP movement is that the nodes following this pattern tend to concentrate in the center region of the deployment area [10][11].

While most studies use straight line piecewise motion for modelling mobile network node movements, investigating curved motion trajectories in mobile networks was a subject of a very few studies. Wang et al. [12] investigates the impact of the shape of the movement path on the efficiency of intrusion detection in a (battlefield) sensor network. An intruder can invade the network following a curved path, or even a random walk, in order to improve its network attacking probability. Wang's research describes the effects of different paths taken by the intruder, on the intrusion detection probability in an arbitrary wireless sensor network. Wang's study of the performance of a wireless mobile network would be the kind of investigation that could potentially benefit from the new approach described in our paper for transforming the existing movement paths into new ones.

## III. THE KOCH FRACTAL

Fractal objects were first mentioned in 17<sup>th</sup> century and referred to as "fractional exponents" by Gottfried Leibniz when he explored the concepts of recursion and self-similarity [13]. In 1872, Karl Weierstrass proposed the definition of a curve, based on a function defined on the sum of Fourier series, that is everywhere continuous but nowhere differentiable, and could be characterised as a fractal. In 1883, Georg Cantor introduced the Cantor sets, which are examples of subsets of the real line that have unusual properties and are also considered fractals. In the last part of that century, Felix Klein and Henri Poincaré discovered a number of fractal patterns that known as "self-inverse" fractals. In 1904, Helge Von Koch, introduced the

famous fractal, the Von Koch curve. The actual term 'fractal' was introduced only in 1975 by B. Mandelbrot [14].

Fractals are complex patterns exhibiting self-similarity at different scales. Two most common types of fractals are complex number fractals and iterative function system (ITF) fractals. Mandelbrot and Julia sets are examples of fractals that are generated by iterating a recursive complex number formula. Koch snowflake and Sierpinski triangle are the examples of iterative function system fractals. The ITF fractals, when constructed in two dimensions, are of particular interest here, as they involve a transformation of an initial pattern on a plane. The initial pattern can be anything: a point, a line, a triangle etc., as well as the path taken by a mobile node in a wireless communication system.

We propose to use fractals for the movement generation for mobile network simulation based on the RWP model. Instead of moving the nodes along a straight line between the waypoints, the nodes are moved along a fractal path. We selected the Koch fractal because, like a line segment, it has a defined starting and ending points, and because of the simplicity of its generating algorithm [15][16].

#### A. Construction of the Koch curve

The construction of the Koch starts with a straight line that is then converted to the Koch fractal curve, Figure 1.



Figure 1. Step 1.



Figure 2. Step 2.



Figure 3. Step 3.

This process is then repeated for each of the 4 segments generated at the first iteration, leading to the curve shown in Figure 3. These steps can be applied repeatedly and eventually result in a complex shape. When the Koch curve generating algorithm is applied to an equilateral triangle it results in a closed curve called the Koch snowflake [16].

#### B. Properties of the Koch snowflake

Number of Sides (n): for each iteration, every segment of the curve from the previous iteration will be converted to four segments in the following iteration. Since we begin

with three sides, the formula for the number of sides in the Koch curve is:

$$n = 3 * 4^a \quad (1)$$

where a indicates the number of iterations. For iterations 0, 1, 2, and 3, the numbers of sides are 3, 12, 48, and 192 respectively.

Length of Sides (L): In every iteration, the length of a side is 1/3 the length of a side from the previous iteration. If we begin with an equilateral triangle with side length x, then the length of a side in iteration a is:

$$L = x * 3^{-a} \quad (2)$$

For iterations 0 to 3, length = x, x/3, x/9, and x/27.

Perimeter (p): The key features of the Koch curve lies in having the same length of all sides in each iteration, this leads to a perimeter, which is simply the number of sides multiplied by the length of a side:

$$p = n * L \quad (3)$$

For the snowflake, from the previous formulas, we get:

$$p = (3 * 4^a) * (x * 3^{-a}) \quad (4)$$

In the same manner, for the first 4 iterations (0 to 3) the perimeter is 3x, 4x, 16x/3, and 64x/9. We notice that, the perimeter increases by 4/3 times for each iteration, so we can rewrite the formula as

$$p = (4/3)^a * 3x \quad (5)$$

#### IV. CUSTOM MOVEMENT GENERATION WITH FRACTALS

The main objective of this research is to propose and implement a new method for movement generation in MANET simulation in ns2. Indeed, the standard way for movement generation is to use the setdest utility that generates a set of setdest commands that are then "executed" in the ns2 simulator. setdest commands generate a movement along a straight line between the current location and the designated destination point. This research aims at providing a new tool for modifying the simulation environment by modeling motion in wireless network simulations, specifically for generating movement files for ns2 simulation that specify the motion along curved (fractal) paths. Typically, defining the node movements needs to be done ahead of the ns2 simulation. In general a curved path can be approximated by a series of short line segments, which determine the final shape of the curve. Therefore, a Java program was implemented that reads the movement file with random movements generated, for example, by the setdest utility. Then, as each movement in the movement file is specified by a separate setdest command, we will replace each one of these setdest commands, each

specifying a movement along a straight line, with a series of setdest commands specifying the movement along a curved path (fractal). Once the new movement file is generated the ns2 simulation can proceed in a standard way.



Figure 4. The result of fractal transformation of a line segment AB.

Let us consider the original setdest command for the direct movement from A to B (Figure 4):

```
$ns_ at T "$node_(#) setdest XB YB S"
```

where T indicates the starting time at which the node starts moving towards the destination  $X_B, Y_B$  at the specified speed S. While splitting the initial path (line segment AB) into four segments (AP, PQ, QR and RB) and defining the destination of each of the four moves is a simple geometry, the other setdest command parameters require careful consideration. More precisely, the need of updating the time and speed in the setdest commands arises when applying the fractal transformation. In order to make the fractal movements arrive at the final destination (point B) at the same time that the original straight movement would have arrived, we need to do the following modifications:

```
$ns_ at TP "$node_(#) setdest XP YP Snew "  
$ns_ at TQ "$node_(#) setdest XQ YQ Snew "  
$ns_ at TR "$node_(#) setdest XR YR Snew "  
$ns_ at TB "$node_(#) setdest XB YB Snew "
```

The four (fractal) movements should proceed sequentially, each having a starting time after the previous movement ends. To calculate the precise time of each move and the new speed we need to determine the new speed and the new starting time for each of the four new setdest commands. First, we need to calculate the time the node would take to travel from A to B at speed S along the original straight line path AB:

$$t_{AB} = \sqrt{(X_B - X_A)^2 + (Y_B - Y_A)^2} / S \quad (6)$$

then the start times for each move are calculated as:

$$\begin{aligned} T_P &= T \\ T_Q &= T_P + t_{AB}/4 \\ T_R &= T_Q + t_{AB}/4 \\ T_B &= T_R + t_{AB}/4 \end{aligned} \quad (7)$$

and the new speed, due to the distance travelled increased by 1/3, is:

$$S_{new} = 4 * S / 3 \quad (8)$$

For example, let us consider the following setdest statement taken from a movement file generated by the setdest utility:

```
$ns_ at 0.0 "$node_(1) setdest 900.0 0.0 10.0"
```

This line specifies that at time 0.0 s, node 1 starts to move from the starting point (0,0) towards the destination (900,0) at a speed of 10.0 m/s (this is one single movement in a straight line). When our fractal transformation is applied once to the movement path, this single command in the movement file is then replaced by four new commands generating the movement along the path corresponding to the shape of the Koch fractal (i.e., one iteration of the Koch fractal generation algorithm). The four movements are listed below:

```
$ns_ at 0.0 "$node_(0) setdest 300.0 0.0 13.3"  
$ns_ at 22.5 "$node_(0) setdest 450.0 260.0 13.3"  
$ns_ at 45.0 "$node_(0) setdest 600.0 0.0 13.3"  
$ns_ at 67.5 "$node_(0) setdest 900.0 0.0 13.3"
```

Applying the fractal transformation to these four new movements for the second time, results in 16 new movements:

```
$ns_ at 0.0 "$node_(0) setdest 100.0 0.0 15.029"  
$ns_ at 5.625 "$node_(0) setdest 150.0 87.0 15.029"  
$ns_ at 11.25 "$node_(0) setdest 200.0 0.0 15.029"  
$ns_ at 16.875 "$node_(0) setdest 300.0 0.0 15.029"  
  
$ns_ at 22.5 "$node_(0) setdest 350.0 87.0 15.029"  
$ns_ at 28.125 "$node_(0) setdest 300.0 173.0 15.029"  
$ns_ at 33.75 "$node_(0) setdest 400.0 173.0 15.029"  
$ns_ at 39.375 "$node_(0) setdest 450.0 260.0 15.029"  
  
$ns_ at 45.0 "$node_(0) setdest 500.0 173.0 15.029"  
$ns_ at 50.625 "$node_(0) setdest 600.0 173.0 15.029"  
$ns_ at 56.25 "$node_(0) setdest 550.0 87.0 15.029"  
$ns_ at 61.875 "$node_(0) setdest 600.0 0.0 15.029"  
  
$ns_ at 67.5 "$node_(0) setdest 700.0 0.0 15.029"  
$ns_ at 73.125 "$node_(0) setdest 750.0 87.0 15.029"  
$ns_ at 78.75 "$node_(0) setdest 800.0 0.0 15.029"  
$ns_ at 84.375 "$node_(0) setdest 900.0 0.0 15.029"
```

Please note that, in the case of the Koch curve generation, standard geographical restrictions on the mobile node movements apply. When the calculated position of the intermediate point Q of the Koch fractal would fall outside the predefined simulation region (Figure 5), then the corresponding segment of the fractal is not generated, and the original straight line segment remains, Figure 6.

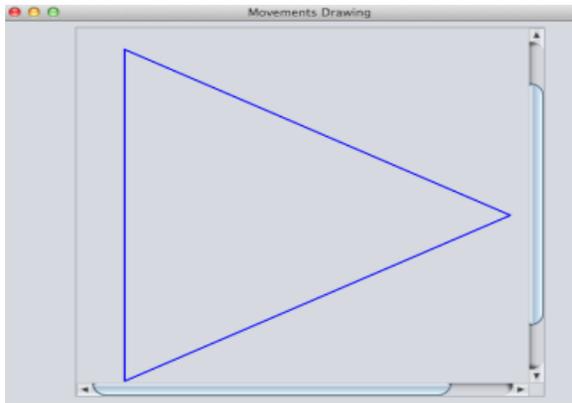


Figure 5. Trace of sample simple node movement.

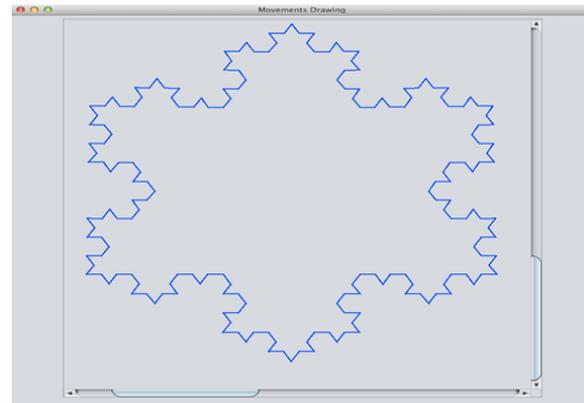


Figure 8. The same movement as in Figure 6 after three Koch fractal steps.

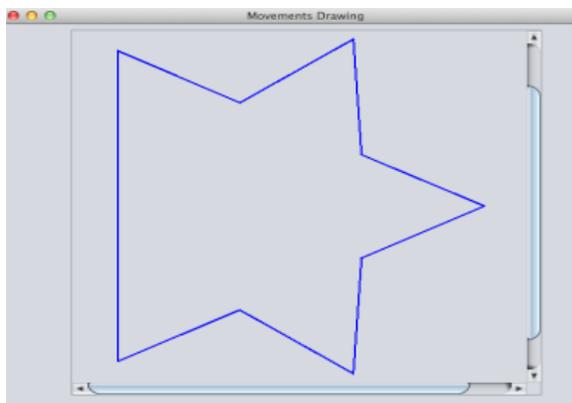


Figure 6. Fractal movement generated from Figure 5.

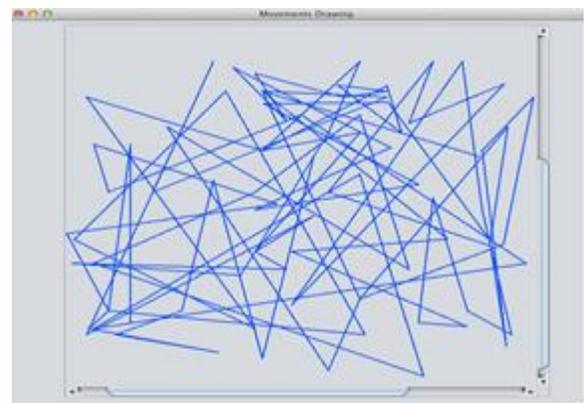


Figure 9. Trace of complex random movement.

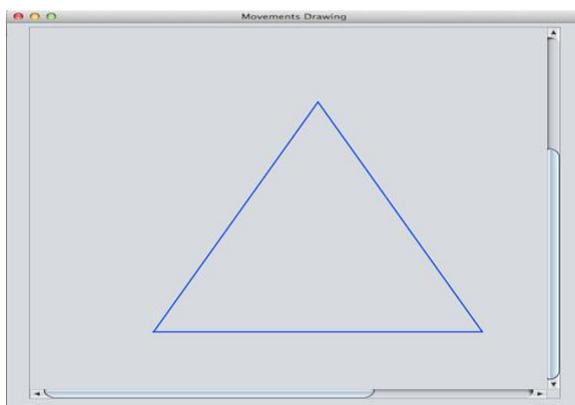


Figure 7. Sample original movement along the edges of a triangle.

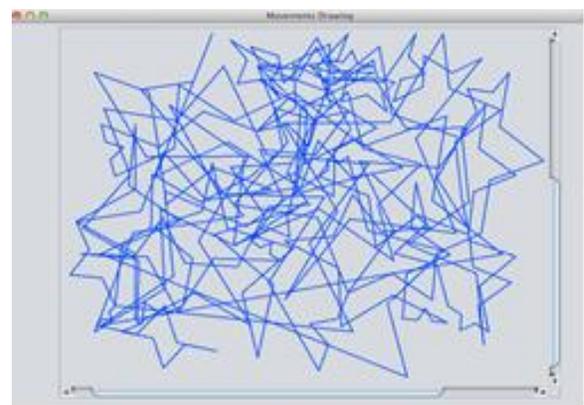


Figure 10. Fractal movement generated from Figure 8.

In a more complex scenario, the effects of the new fractal path modification algorithm can be best illustrated graphically: when the transformation is applied three times to the movement file with three original movements defined along the sides of an equilateral triangle, the result is a curved movement defined with 192 straight line movements in the shape of the Koch snowflake (Figures 7 and 8).

Finally, when the transformation is applied to a sample movement file generated with ns2 setdest utility, the original random movements, each along a straight line, Figure 9, are turned into fractal movements, four times as many, covering the same locations (waypoints) as in the original movement file (Figure 10).

## V. EXPERIMENTAL ENVIRONMENT: NS2 SIMULATOR

The network simulator ns-2 is a popular open source simulation environment [4]. In a typical configuration, it consists of the simulator module ns2, the network animator module nam, the stand alone utility for generating wireless scenarios setdest and the plotting module xgraph. ns2 is an object oriented discrete event simulator written in C++ and OTcl. The event scheduler and the basic network component libraries are written in C++ for greater efficiency. OTcl is used for setting up and controlling the simulation. In a typical simulation session the user sets up the network nodes, links, transport protocols and traffic generators using an Otcl script and then runs the simulation. In case of simulating wireless mobile network, the user also specifies the node characteristics, the routing protocol and the node movements. To make it more convenient to define more complex wireless scenarios with a large number of nodes making many moves over an extended period of time (each move requires a separate command), ns2 installation includes the setdest tool. This tool automatically generates a wireless scenario, with a large number of individual node movements specified, based on a small set of user defined parameters. Setdest is particularly suitable for creating base case scenarios with a large number of nodes moving randomly over a rectangular regions and predefined range of speeds.

The simulation results are written into a trace files recording all the network events. The network animator tool nam displays the animation of the network's events from the trace file, showing the movement of nodes and packets in the network. The results can be plotted with xgraph as well. The trace files generated by ns2 are in the form of text files that can be viewed directly, or processed easily, to extract the desired network performance metrics.

ns2 includes a library of the most common transport and, in case of wireless mobile network simulation, routing protocols. Various versions of the Transport Control Protocol (TCP) and the User Datagram Protocol (UDP) are available for experimentation. For experimenting with the ad hoc mobile networks, the Destination Sequence Distance Vector (DSDV), Dynamic Source Routing (DSR), Temporally Ordered Routing Algorithm (TORA) and Ad hoc On-demand Distance Vector (AODV) are supported.

The AODV protocol [17][18] used in our experiments, is an ad hoc network reactive routing protocol. The node, wishing to transmit to another node, first broadcasts a route request (RREQ) message to the neighbouring nodes. This process continues until a RREQ arrives at the destination (or at a node that possesses a current route to the destination). As the RREQ traverses the network towards the destination, all the nodes in its path set up the reverse path back to the source. Once the destination is reached, a route reply packet (RREP) is sent back following the reverse path, signalling the establishment of the route to the source node. Under some conditions, AODV offers superior performance when compared to other established routing protocols [18].

## VI. EVALUATION OF MANET PERFORMANCE UNDER FRACTAL MOVEMENT

We evaluated the performance of a sample MANET under different motion generation conditions. A MANET with the number of nodes ranging from 5 to 80 was simulated over the area of 800 by 800 meters. The MANET in the experiment was set up as the Mobile Medium Ad Hoc Network (M2ANET) [4]. Mobile Medium networks are a special case of MANETs, where the mobile network nodes are divide into two categories: the forwarding only nodes (shown in black in Figure 11) and the communicating nodes (shown in red) that use the rest of the Mobile Medium for multi hop communication. In our experiment, the M2ANET was set up with two stationary communicating stations located at (100,500) and (700, 500), Figure 11. Constant Bit Rate (CBR) traffic was generated over UDP and routed with the AODV protocol (Table I). Standard RWP movement was generated with setdest and then the standard movement was converted to the fractal movement using one step of Koch generating algorithm, as shown in Figures 9 and 10.

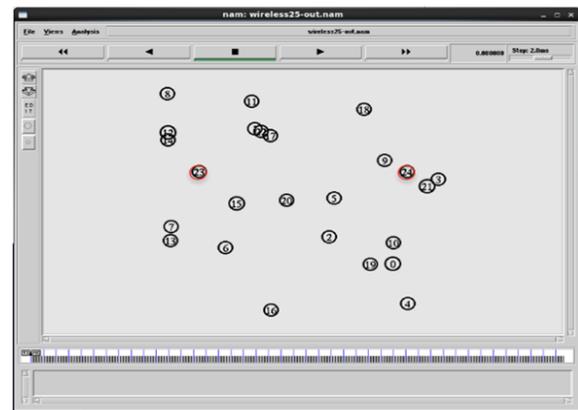


Figure 11. Screen shot of ns2 network animation in the nam utility, showing the mobile nodes and two fixed communication stations.

TABLE I. SIMULATION PARAMETERS

Parameters	
Simulator	NS-2.33
Channel Type	Channel / Wireless Channel
Network Interface Type	Phy/WirelessPhy
Mac Type	Mac/802.11
Radio-Propagation Type	Propagation/Two-ray ground
Interface Queue Type	Queue/Drop Tail
Link Layer Type	LL
Antenna	Antenna/Omni Antenna
Maximum Packet in ifq	50
Area (n * n)	800 x 800
Source Type	(UDP) CBR
Simulation Time	100s
Routing Protocol	AODV

Two scenarios were investigated: (i) low speed (10m/s), and (ii) high speed (30m/s).

Figure 12 illustrates the difference in the number of packets received at the destination when using the original movement and the new fractal movement at low speed. It shows that most of the time the packet delivery for the fractal movement is higher than the original linear movement. Although the speed of the fractal path is higher than the original (because of the increased path length along the fractal curve between the original waypoints), we observed a higher number of packets delivered at the destination for the fractal movement at speed of 13m/s. However, applying the t-test for the comparison of two paired means representing the packets received in the linear motion and the fractal motion experiments with 25 nodes gives 8%, which indicates that the observed difference is not statistically significant. Also, comparing the average packet delivery across all node densities does not show a significant difference (t-test value 49%).

Figure 13 shows the packet delivery for linear and fractal motions at high speed. This time we observe a lower packet delivery for fractal motion recorded in most of the experiments.

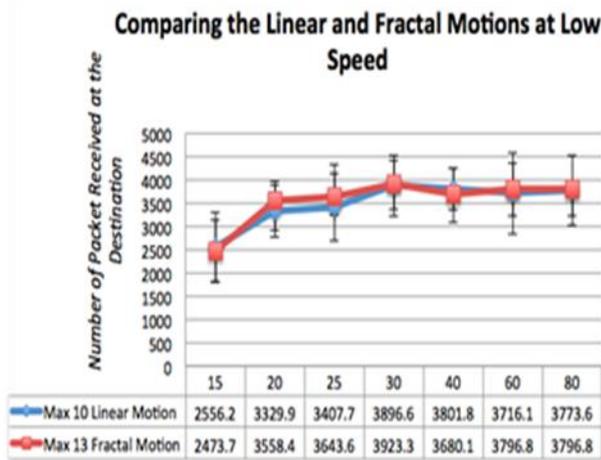


Figure 12. Throughput comparison at low speed

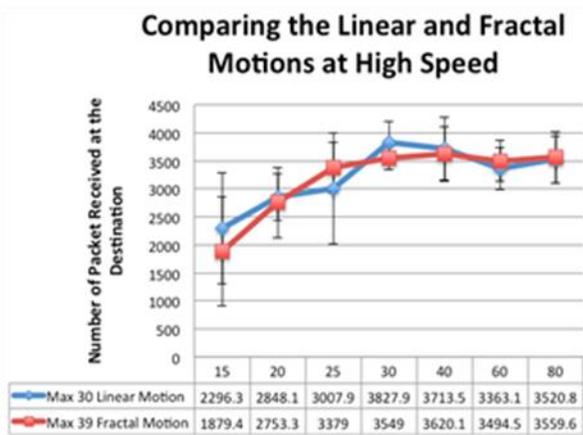


Figure 13. Throughput comparison at high speed.

One possible explanation of lower performance with fractal motion is that the increase in movement speed of 10m/s, from 30 to 40, results in more frequent link disconnections and consequently lower packet delivery. Applying the t-test for the comparison of two paired means representing the packets received in the linear motion and the fractal motion experiments with 25 nodes gives 32%, which indicates that the observed difference is not statistically significant. Also, comparing the average packet delivery across all node densities does not show a significant difference (t-test value 60%).

Figure 14 illustrates the advantage of using lower speed in a network with linear motion. It shows that the packet delivery is consistently higher at low speed for almost all node densities. Applying the t-test for the comparison of two means representing the packets received in the linear motion and the fractal motion experiments with 20 nodes gives 4%, which indicates that the observed difference is statistically significant. The average packet delivery for all node densities is 3176 at high speed and 3497 at low speed, and this difference in performance is statistically significant (t-test value 0.076%). This result is consistent with the performance of ad hoc routing protocols, like AODV, in a high mobility network [19]. At the network layer, packets are buffered and eventually dropped if the valid route to the destination node is not known at the forwarding node. At the MAC layer, the packets are dropped when the routing information is obsolete and the next hop node is out of range.

Similarly, Figure 15 illustrates the advantage of using lower speed in a network with the fractal motion. The packet delivery is consistently higher at low speed for all node densities. Applying the t-test for the comparison of two means representing the packets received in the linear motion and the fractal motion experiments with 20 nodes gives 1%, which indicates that the observed difference is statistically significant. The average packet delivery for all node densities is 3186 at high speed and 3553 at low speed, and this difference in performance is statistically significant (t-test value 1.7%).

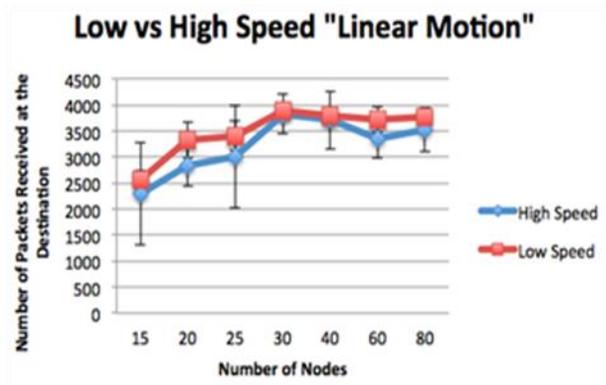


Figure 14. Throughput comparison for linear motion

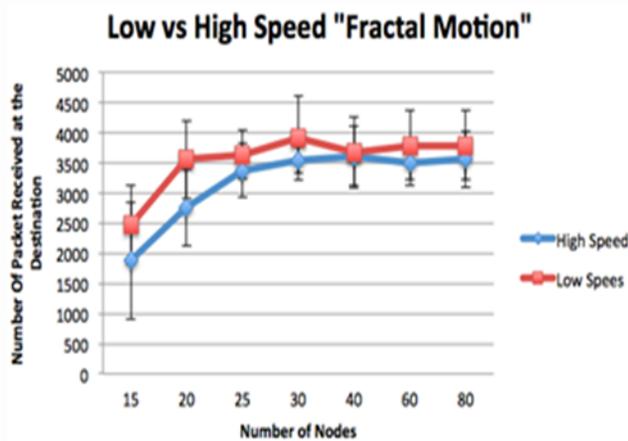


Figure 15. Throughput comparison for fractal motion.

## VII. CONCLUSION AND FUTURE WORK

In this paper, we presented a new paradigm for investigating mobile wireless networks using simulation. While conventional simulation approaches focus on using different protocols, data generators and different movement patterns, our new paradigm focusses on manipulation of the existing movement files. The existing movement files can either represent the actual node movements in a real network (e.g., trace based) or be automatically generated (e.g., random models). The new investigative paradigm aims at examining the changes in the network performance resulting from replacing the existing node movement pattern with a new one, that is obtained as a result of a transformation of the old movement pattern.

We presented a tool for transforming linear movements into fractal movements based on the Koch curve. The new tool reads a standard ns2 movement file, decodes each movement, and replaces it with a series of new movements forming a fractal curve, and then outputs a new movement file. The newly generated movement file satisfies the ns2 specifications and can be used in the ns2 simulator. Both, standard movement files generated with *setdest* and new movement files generated with the new fractal tool were used in simulating a MANET with varying number of nodes (i.e., with different node densities). The fractal transformation can be applied to any ns2 movement files, including the ones automatically generated with the *setdest* utility.

We used the new fractal transformation in ns2 simulation and compared the MANET performance in terms of packet delivery under two different motion scenarios and at different speeds. In the experiments, we observed marginally higher performance of MANET with fractal motion at low movement speeds, suggesting that using a curved path between the waypoints (rather than straight line) would offer an advantage in terms of network performance. However, the statistical tests show that the

difference observed in our limited experiments is not significant. We also observed that the packet delivery is lower at higher speeds for both motion types, and after the application of the t-test for the difference of the means, we concluded that the observed lower packet delivery at higher speed is statistically significant. From our results, we conclude that, in the scenarios investigated, only the node speed significantly affects the MANET performance, and not the shape of the path taken by a node.

The work presented in this paper successfully demonstrated the new experimental approach for investigating performance of mobile networks, i.e., applying transformations to the node movement paths. The future work on transforming the node movement paths might include using more than one iteration of the generating function of the Koch fractal, calibrating the node speed when it starts moving on the new curved path and testing if the new path generators reduce the tendency observed in the RWP model of clustering the nodes towards the center of the experimental area. In general, the proposed new paradigm for mobile network simulation, that involves transformation of mobile nodes paths, could be applied in a study of mobile network intrusion detection, similar to Wang et al. [11]. In such a simulation study, we could specify only the destination of the rogue node penetrating a mobile network, and then use the approach similar to the one described in our paper, to generate multiple different paths to this destination, and then investigate the performance of intrusion detection under these different, and automatically generated, experimental scenarios.

## ACKNOWLEDGMENT

This work is sponsored and funded by the Ministry of Higher Education of Saudi Arabia through the Saudi Arabian Cultural Bureau in Canada.

## REFERENCES

- [1] A. Aelsef, J. DeDoureik and P. Pochech, "A Method for Custom Movement Generation in Wireless Mobile Network Simulation", The Seventh International Conference on Emerging Networks and Systems Intelligence EMERGING 2015, Nice, France, July 19, 2015, pp. 27-32.
- [2] S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic (Eds.), *Mobile Ad Hoc Networking*, New York, Wiley-IEEE Press, 2001.
- [3] F. Bei and A. Helmy, *A survey of mobility models in wireless Ad hoc Networks*, University of California, USA, 2004.
- [4] J. DeDoureik and P. Pochech, "M2ANET: a Mobile Medium Ad Hoc Network", *Wireless Sensor Networks: Theory and Practice*, WSN 2011, Paris, France, Feb. 2011, pp. 1-4.
- [5] H. Ekram and T. Issariyakul, *Introduction to Network Simulator NS2*, Springer, 2009.
- [6] N. Aschenbruck, E. G. Padilla, and P. Martini, "A survey on mobility models for performance analysis in tactical mobile networks", *Journal of Telecommunications and Information Technology*, vol. 2, 2008, pp. 54-61.
- [7] T. Camp, J. Boleng, and V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research", *Wireless Communication and Mobile Computing (WCWC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, vol. 2, no. 5, 2002, pp. 483-502.

- [8] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols", Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom98), ACM, October 1998, pp. 85-97.
- [9] E. M. Royer, P. M. Melliar-Smith, and L. E. Moser, "An Analysis of the Optimum Node Density for Ad hoc Mobile Networks", Proceedings of the IEEE International Conference on Communications (ICC), Helsinki, Finland, June 2001, pp. 857-861.
- [10] C. Bettstetter, "Mobility Modeling in Wireless Networks: Categorization, Smooth Movement, and Border Effects", ACM Mobile Computing and Communications Review, vol. 5, no. 3, July 2001, pp. 55-67.
- [11] R. Alghamdi, Movement Generator for Mobile Network Simulation. Master's Report, Faculty of Computer Science, University of New Brunswick, Fredericton, Canada, 2012.
- [12] Y. Wang, Y. K. Leow, and J. Yin, "Is Straight-line Path Always the Best for Intrusion Detection in Wireless Sensor Networks?", ICPADS '09, Proceedings of the 2009 15th International Conference on Parallel and Distributed Systems, IEEE Computer Society Washington, DC, USA, Dec. 2009, pp.564-571.
- [13] B. B. Mandelbrot, The Fractal Geometry of Nature, Freeman 1982.
- [14] B. B. Mandelbrot, Les Objets Fractals: Forme, Hasard et Dimension, Flammarion, Paris, 1975.
- [15] G. Edgar, Measure, Topology, and Fractal Geometry, 2<sup>nd</sup> Ed., Springer 2008
- [16] Koch's Snowflake, online, [http://en.wikipedia.org/wiki/Koch\\_snowflake](http://en.wikipedia.org/wiki/Koch_snowflake), retrieved: May 2015.
- [17] C. Perkins, E. Belding-Royer, and D. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing", IETF, RFC 3561, <https://tools.ietf.org/html/rfc3561>, retrieved 2016-04-28.
- [18] A. Tuteja, R. Gujral, and S. Thalia, "Comparative Performance Analysis of DSDV, AODV and DSR Routing Protocols in MANET Using NS2", ACE 2010, International Conference on Advances in Computer Engineering, 2010, pp. 330-333.
- [19] Q. Zhao and H. Zhu, "An Optimized AODV Protocol in Mobile Ad Hoc Network", 4th International Conference on Wireless Communications, Networking and Mobile Computing, Dalian, China, Oct. 2008, pp. 1-4.

# Multi-Target Data Association in Binary Sensor Networks for Ambulant Care Support

Sebastian Matthias Müller

OFFIS – Institute for  
Information Technology  
Oldenburg, Germany

Email: [sebastian.mueller@offis.de](mailto:sebastian.mueller@offis.de)

Andreas Hein

School of Medicine and Health Sciences  
University of Oldenburg  
Oldenburg, Germany

Email: [andreas.hein@uni-oldenburg.de](mailto:andreas.hein@uni-oldenburg.de)

**Abstract**—Numerous applications of home automation, security and ambulant medical care use binary sensors such as passive infrared motion sensors or light barriers to monitor activity in the house. While the data of individual sensors does not facilitate the recognition and separation of the presence of more than one person, there exist multi-target tracking algorithms that allow for at least a partial separation of activity in data from multiple persons. While many tracking algorithms demonstrate good performance across various sensing modalities and sensor setups, little research has been done to determine the impact of placement and varying density of sensors for tracking performance. This paper presents the results of two evaluations of a Bayesian multi-hypothesis multi-target tracking algorithm on data of residents monitored by a network of binary sensors. The algorithm's performance is evaluated across varying quantity and placement. It is shown that the approach outperforms other approaches in low-resolution setups using data collected in a home lab and further demonstrate its applicability in a field trial across three households. While tracking performance naturally decreases with the number of sensors, it also strongly varies by sensor positioning.

**Keywords**—Multi-target tracking; Assisted living; Wireless sensor networks; home automation; smart homes; telemedicine.

## I. INTRODUCTION

This article is an extended version of an article originally submitted to the International Conference on Ambient Computing, Applications, Services and Technologies (AMBIENT 2015 [1]).

The emergence of research on technical support systems for ambulant care and support for patients and elderly stem from numerous recent societal developments as well as changes in demographic structure. First, the coincidence of prolonged life expectancy [2], [3] and the atomization of households [4], [5] puts an increasing care demand into the hands of third parties, especially in Western countries with low birth rates: US census data shows that the average household size in the United States declined from 4.5 persons per household in 1900 to 2.6 in 2000 [6]. According to the German Federal Statistical Office, the number of single households will increase sixfold relative to the population between 2010 and 2030. At the same time, the ratio between care personnel supply and demand will cut in half [7]. Second, increasing life expectancy, in combination with improved medical care and "modern lifestyles and behavior" causes an increase in the proportion of population living with chronic diseases, thus furthering demand for ambulant care [8]. Third, there is a general trend towards outpatient

care by hospitals. According to the Avalere Health analysis of American Hospital Association Annual Survey, the percentage of revenue for community hospitals in the United States has increased from 25% to approximately 44% in 20 years between 1992 and 2012 [9]. The average length of stay of a patient after surgery dropped from 7.0 in 1993 to 5.4 days in 2013.

These developments drive the research on technical support systems in home and care environments. Applications for such include automated assessments such as mobility measurements [10], activity monitoring such as the detection of Activities of Daily Living (ADLs) [11] or fall detection for automated emergency calls [12]. Existing literature shows the importance of ADLs such as bathing and eating and Instrumental Activities of Daily Living (IADLs) as indicators of physical and cognitive abilities of elderly individuals [13]. ADL performance has also been shown to improve technology design for patients suffering from Alzheimer disease [14]. Overall, ADLs and IADLs are considered the "gold standard" for measurement of functional ability. Thanks to the increasing availability of sensors to detect motion (passive infrared motion sensors) and activities (door contact sensors, acceleration sensors, RFID chips and sensors), there is a strong hope to eventually perform such tests automatically. At the same time, such sensors can also provide data for other useful services to increase safety and comfort of people in their home [10], [15], [16]. Most of these, however, have only been tested in one-person households.

To preserve a maximum of privacy and comfort while at the same time collecting data necessary for the application, many approaches include the use of ambient sensors such as motion sensors and light barriers. Since the data collected from these sensors does not carry identifying information, use of any such application in settings where more than one person – the patient – moves or resides becomes difficult. Complex sensors, such as cameras and microphones are usually considered invasive and are thus rarely accepted in living spaces. Body-worn sensors are often forgotten or ignored due to discomfort. Binary sensors such as light barriers and motion sensors are easy to retrofit, have relatively little power consumption and can be installed unobtrusively. A no-requirements sensor model also enables us to install more complex sensors (such as laser scanners or depth-finding cameras) as required. The necessary information can be extracted from their data by partitioning the sensors' range and converting activity in each partition to a binary signal.

To separate data from multiple persons moving in a space monitored by binary sensors, a multi-target tracking algorithm using Bayesian estimation and multi-hypothesis tracking is presented. This algorithm makes no assumptions on the selection and placement of sensors or sensing technology. Tracking takes place on a graph of the sensors and their spatial relation. It is thus not helpful in determining the precise location of a present person, but at (or below) room-level accuracy. More importantly, however, it can help determining when there is more than one person present, and helps to separate the activity data. This algorithm performs particularly well on low-resolution data, such as when only few binary sensors are used. To study its precision, the algorithm is tested across various sets of sensors, varying by placement and number. The data was recorded at a home lab and was labelled and verified using video recordings. A decreasing number of sensors will likely have an impact on the tracking accuracy, but is important in regard to energy consumption, costs and user acceptance. It is shown that data from two residents in an apartment can be separated with high (>90%) accuracy, and that the selection and placement of sensors can play a significant role in tracking accuracy. To test its applicability, as a second evaluation a field trial is conducted in which three two-person households are equipped with sensors and an implementation of the algorithm, running on a small-form-factor PC, determines the number of people present.

The remainder of this article is structured as follows: Section II summarizes related works on multi-target tracking and activity monitoring in the home using binary sensors. Section III describes the algorithm and the theoretical principles surrounding data association and multi-hypothesis tracking for single- and multi-target tracking. Section IV describes how the algorithm and its implementation were evaluated, including data preparation, the sensor placement concept and the field trial setup. The results of the evaluation are presented in Section V. Section VI summarizes the article and Section VII gives an outlook on future work.

## II. RELATED WORK

Prior work has shown that data collected from sensor networks allow for the deduction of information used in activity monitoring, care assessments and behavior modeling. Target tracking, in particular multi-target tracking, is a task often applied to visual data such as video feeds and images. The practical application of multi-target tracking in binary or low-resolution home sensor networks has been subject to little research.

### A. Target Tracking in Home Sensor Networks

Target tracking in the home is a fundamental problem of ubiquitous computing, and proposed solutions span a variety of sensors, including cameras, laser scanners, RFID (Radio frequency identification) and infrared or ultrasound badges [17], [18], [19].

Wilson and Atkeson describe an algorithm for tracking of multiple persons and their activity status in a binary sensor network [17]. Similar to the proposed use of a weighted graph in this article, the authors use a transition matrix representing transition probabilities between sensors. By keeping track of the targets' identities, personal motion models emerge. The data association is achieved using a particle filter. During a

five-day experiment in a house instrumented with 49 sensors (contact switches, motion sensors), data during two-person scenarios was correctly assigned 82.1% of the time. While the approach solves both the data association as well as the identification, it is based on individual motion models and thus relies on data that is commonly unavailable.

Krüger et al. use a particle filter and *action plans* to assign sensor events from motion sensors and light switches to tracks and simultaneously identify the target [20]. Action plans describe action sequences in terms of sensor data. These plans can be synthesized or learned from historic data. For the evaluation, an office corridor was equipped with six light switches and six motion sensors. The mean squared error across time and all targets is reported as approximately 0.26 for two-person scenarios. The work shows how – similar to trained motion models – previous knowledge of a person's plans can help tracking individuals in binary sensor networks.

Oh and Sastry perform tracking on data of binary sensor networks and passage connectivity graphs [21]. The graphs are calculated from transition probability matrices. A tracking algorithm, derived from the Viterbi algorithm, pruning strategies and multiple target tracking extensions are presented. No evaluation on real world data is conducted.

Marinakakis et al. derive the topology of a sensor network in terms of transition times and probabilities from data of unspecified sensors [22]. The authors use Monte Carlo Expectation Maximization to assign activity to agents (people present) in order to build a graph of the sensor network. 95% of the topology of simulated node graphs is recovered correctly. The results for a trial using a network of cameras and photocell-based sensors are not reported.

### B. Activity Monitoring in Home Sensor Networks

The aim of this work is to test the performance of a multi-target tracking algorithm on data collected by binary sensors such as light barriers and motion sensors. The motivation is to *a)* collect activity data of an ambulant care patient, which is optimally collected during times when the patient is alone at home (to make sure the data originates from the person in question only), and *b)* to activate security measures such as fall detection and automated emergency calls, which are only necessary when the person is alone temporarily.

Researchers at the Tokyo Medical and Dental University have collected data from homes instrumented with binary sensors (motion detectors and contact switches) for one year. Patterns of activity such as absence, use of stove and sleeping times were "clearly identifiable" [23]. Researchers at the University of Virginia used an array of motion detectors and contact switches to detect ADLs [24]. Sensor readings are clustered to groups based on room, duration, and time of day to show that many clusters correspond to ADLs.

Numerous studies show that data collected from sensor networks in living spaces allow for the deduction of information relevant in applications of activity monitoring, care assessments and behavior modeling. Logan et al. showed that ambient motion-based sensors provide the most useful information for detection and classification of daily in-home activities in a study compared to RFID, on-body and on-object sensors. In their study, infrared motion sensors yielded the best results overall, although classification performance on

this data was better on activities that are strongly correlated with locations in the home, such as "watching TV" and "meal preparation" [25].

Data from binary sensors can also be used to calculate average room residence time and frequency: Assessment tests are partly realizable by using recordings from light barriers and reed contacts alone [15]. The authors argue that light barriers alone do not constitute sufficient evidence of a person entering a room, because people may change directions between rooms. It is suggested to combine light barriers with sensors covering larger areas. Room residence times are calculated by manually labeling the sensors constituting a room using a floor plan and knowledge of the sensors' placement. In a similar study, the authors model user behavior of a resident from the probability of location at a certain time of day and the frequency of presence in a location in a defined period of time [26]. Models are created for rooms individually (bathroom, bedroom, living room, kitchen). Based on the number of anomalous behavior detected, the authors conclude that the models' performance varies by room: Presence in the bathroom is best modeled duration-based, while the timeslot-based model yielded better results for the other rooms.

Frenken et al. [10] use ambient sensors in an attempt to automate measurement of mobility and gait velocity, as required in the Timed Up and Go assessment [27]. For this, five apartments are equipped with home automation sensors and one with an additional laser range scanner. It is shown that the data is suitable to compute gait velocity at home. While data from the laser range scanner is proven to be more precise than home automation sensor data, no statistical post-processing or filtering was performed on the latter.

Shin et al. use motion sensors to determine normal behavior and to detect unusual behavior [28]. The behavior is described by "activity level, mobility level and non-response interval (NRI)". Days are divided into timeslots for which the features are computed. The activity level measures a person's motion at a certain timeslot by summing up the number of motion sensor events detected in its coverage area. The mobility level is computed by counting location changes during a certain timeslot as depicted by two different sensors triggering. The non-response interval is the duration between two consecutive movements and is calculated by summarizing these no-motion durations for each timeslot. Using an approach coined "support vector data description", normal behavior and anomalies are correctly classified with an accuracy of more than 90%.

Skubic et al. use motion and temperature sensors as well as a pneumatic bed sensor to determine alert parameters (bathroom activity, bed restlessness, kitchen activity, living room activity) to model normal behavior [29]. The parameters are computed for three time periods, a 24-hour day, daytime (8am to 8pm) and nighttime (midnight to 6am) to describe the behavior learned over the span of two weeks. Fuzzy pattern tree and support vector machine classification is used to determine whether the feature vector of the current day is abnormal. Both classifiers achieved similar results.

Floek et al. use automation sensors to identify human behavior by means of inactivity profiles [30]. Inactivity is defined as the time between consecutive sensor events, so that inactivity ends with any detected sensor event. Plots of inactivity intervals over a learning period of 31 days are

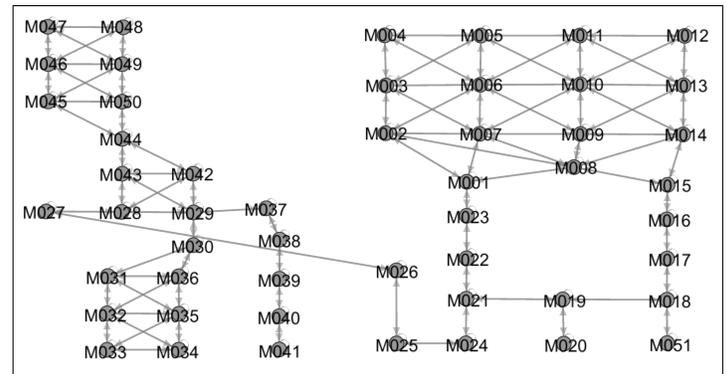


Figure 1. Graph of sensors (with their internal IDs and their spatial relations used in the evaluation (adapted from [31]).

created for reference, whereby inactivities due to absences are not regarded. To classify the current day as normal or abnormal day, the day's inactivity vector is compared with the corresponding timeslot in the reference vector. Dice's coefficient is used to measure the similarity of these vectors: An inactivity value is identical to the corresponding reference value if it is in a predefined tolerance region of the reference value. A modified approach uses the maximum duration of inactivity over several days: A threshold is calculated based on recorded maximum durations plus an additional tolerance of 30 minutes. When abnormal behavior is detected, an alert is generated.

As can be seen, most studies of activity monitoring are designed to be used in single-person households, although the foundations for data association and identification have been laid. Whatever the approach is, it is important that a tracking algorithm works on a multitude of sensor technologies, and without restricting the setup in terms of sensor density or placement. In the following section, such an approach, based on multi-hypothesis tracking, is presented. An evaluation of its performance using different numbers and placement of sensors follows.

### III. APPROACH

To describe the tracking approach used in these studies, the idea of the sensor graph is introduced first. Then, Bayesian filters and how they can be used to help track individuals on this graph will be explained. Finally, the multi-hypothesis tracking approach and how it differs from previous implementations is described.

#### A. Sensor Graph

A graph of sensors  $s_1, \dots, s_N$  is defined as a weighted, directed graph  $G = (V, L)$ , where  $V = \{1, \dots, N\}$  is the set of nodes in the graph representing sensors, and  $L$  is the set of all edges  $(u, v)$  for which there is a direct passage from the sensing region of sensor  $u$  to the sensing region of  $v$  which does not intersect any other sensing regions. Informally, two sensors  $u, v$  are connected if it is possible for a person to traverse from the sensing region of  $u$  to the sensing region of  $v$  without activating any other sensor.

Each resident in the target space is represented by a discrete Bayesian filter on an unweighted, undirected graph

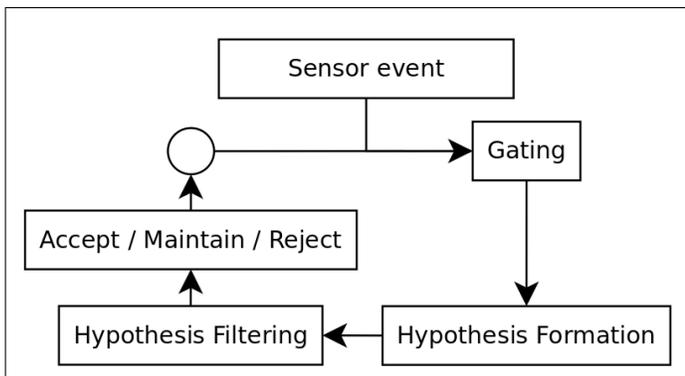


Figure 2. Hypothesis formation overview.

consisting of sensors as nodes and edges representing their spatial adjacency. For the evaluation data, this graph was published by Crandall et al. [31] (Figure 1). If the adjacency relations are not known, they can be approximated by a path planning algorithm using a floor plan, if available [32], or can be generated from prerecorded data [33].

### B. Tracking of individuals

Bayesian filters estimate the state of a dynamic system from noisy data. The location of a person is described as a probability distribution over all nodes in the graph, rather than selecting the most likely sensor as their location, to represent the location of each individual. This is done because it may help to calculate a more precise location later on, especially when sensor regions overlap. More importantly still, it helps the tracker to recover more quickly when a noisy measurement is assigned to the individual's track. Lastly, the aim is to replace the manually constructed, unweighted graph with a weighted graph that is constructed from in-situ recorded data, which reflects individual behavior by using transition probabilities between sensors as edge weights (cf. [22]).

All sensors are subject to noise, and although only binary data is sought, the sensors may still fail to detect activity or detect irrelevant activity, such as from pets or through the window. Due to duty cycle regulations in the range of 868 Mhz, many motion and home automation sensors come with an additional source for noise: measurement delay. Many sensors do not measure or report measurements for a specified amount of time after triggering. This period can last from a few seconds to several minutes due to radio communication regulations. This results in sensors missing the presence or movement, thus breaking the continuity of measurements of a motion track.

At which point and how many filters are created – that is to say, how many individuals are assumed to be present – depends on the performance (*belief*) of the previously existing track: When new measurements cause the current data to be more likely when assigned to more or fewer tracks (= individuals) than before, a new filter is spawned or an existing filter is discarded. The data could be bundled into larger updates within reasonable time frames (cf. [34]), but in this case an update occurs for each new sensor event.

### C. Multi-target tracking

Fundamentally, to distinguish activity of two or more persons in an area monitored by binary sensors, the activity

must be spatially separable on the graph. Ideally, there is an inactive sensor between two persons. This concept has been thoroughly described by Oh and Sastry [21]. However, whether two neighboring sensor events are assigned to one or two targets is determined by the weights of the graph and the evaluation function.

1) *Multi-Hypothesis Tracking*: When new sensor data arrives, hypotheses are created by considering all possible assignments of the data to existing and new tracks ("hypotheses") until the filter's *window size* is reached. This is particularly useful in a low-resolution setting like this, where individuals may occlude each other in sensor readings for any period of time. The window size in multi-hypothesis tracking (MHT) describes the maximum number of events (or time steps) that are considered before choosing a likely hypothesis. Windowing is necessary to limit the number of possible hypotheses and to limit the information loss in case no acceptable hypothesis is found and the data is discarded. The influence of the window size on tracking accuracy has been shown previously [35]. For this evaluation, a window size of 10 events is used. The sensors used in the field trial collected an average of 25.5 events per hour, making the average windows size 23.5 minutes. The sensors used in the home lab have a delay of approximately two seconds, making the temporal extent of the windows much smaller.

The idea of multi-hypothesis tracking dates back to 1979, when Donald B. Reid published "An algorithm for tracking multiple targets" [36]. Reid's algorithm was developed to work on data from a continuous scale sensor (radar). Therefore, Reid speaks of associating measurements to clusters. In the work presented here, the target space is discrete (nodes on a graph), and targets and their locations are stored as a probability distribution over the space using Bayesian filters.

There are two significant differences between Reid's original work and the approach described here. First, in accordance with Reid's *type 2* sensor, the sensor model used in this work expects *positive reports* only, meaning that only sensor data reporting activity is considered. However, tracks are updated per hypothesis, rather than generated and filtered individually (*hypothesis-oriented MHT*). This means that hypotheses are not constructed from *compatible* tracks, but all possible combinations of updates of existing hypotheses. Second, the tracker is updated every time a sensor reports activity. Because of this, and the fact that the given state space is discrete, computational complexity is reduced. For a more detailed description of track- and hypothesis-oriented MHT, see Blackman [34].

For each triggered sensor, a new hypothesis based on all previously existing hypotheses is created, in which the triggered sensor is

- considered noise and discarded,
- used to update one of the existing filters, or
- assigned to a new filter.

2) *Filtering*: For an unspecified number of possible targets the number of possible hypotheses follows Bell's Number  $B_{n+1}$ . Due to the exponential growth ( $> 4.74 \times 10^{13}$  for 20 events), a number of filters to optimize computation efficiency must be employed.

All hypotheses must pass a gating function before they are considered for evaluation (see Figure 2). In this case, this

gating function is a simple comparison of the prior probability of each filter to a threshold value. Afterwards, hypotheses are filtered based on confidence, noise ratio and similarity. This procedure is performed until a single hypothesis remains or the window size is reached. In the former case, the hypothesis is accepted, the underlying Bayesian filters updated, and the window size reset. In the latter case, all hypotheses are evaluated. If no single, dominating hypothesis can be found, all hypotheses are discarded and the underlying filters reset.

The size of the window strongly influences the performance of the algorithm. A larger window size will result in a larger number of correct associations, but also in a larger number of discarded sensor events [35].

Figure 2 depicts the general multi-hypothesis tracking logic. For a more in-depth description of multi-hypothesis tracking, see Blackman [34] or Reid [36].

#### IV. EVALUATION

In the following, the data that was collected for both the home lab tests and field trial in terms of how it was collected and what was evaluated is described. For the home lab tests, several subsets of sensors were evaluated to study the impact of sensor placement on tracking performance. For the field trial, sensors were installed based on these results.

##### A. Experiment I: Home Lab

1) *Data*: The data used for this evaluation was recorded at the Center for Advanced Studies in Adaptive Systems (CASAS) at the University of Washington [38]. It shows activity of two residents of a smart home environment, residing in a 4-room, 2-story apartment for approximately 8 months. For this evaluation, subsets of the data recorded by the 50 motion sensors mounted to the ceiling are used. The smart home is also equipped with contact sensors on doors and cabinets, temperature, water and electricity sensors. For the purpose of this trial, however, motion sensors offer the most precise and least noisy data.

Data for which at least both residents are present and active is used. Among those, time frames

- that last at least 20 minutes or contain at least 300 sensor events,
- in which both residents change rooms at least once, and
- in which neither resident is inactive for more than 20% of the time

are chosen. The result are twenty time frames, with 330 to 910 sensor events with durations between 24 and 530 minutes. After selection, each of the 13321 sensor events was labelled as originating from Resident 1, Resident 2 or a third person using the manually labelled events and the laboratory's floor plan.

2) *Data Association*: The algorithm can track any number of targets. However, the intended area of application – small households – allows one to use an evaluation function that is tailored towards few targets (1-3). For this evaluation, the algorithm was optimized to track two targets by using an evaluation function that favors one- and two-track hypotheses. Equation (1) describes the evaluation function, where  $h$  is the hypothesis in question,  $conf(p_n)$  is the belief of the Bayesian

filter at the most recent event location  $n$ ,  $\|p\|$  is the number of paths (= targets) in  $h$ , and  $m$  is the expected number of targets in the sensor space.

$$eval(h) = \frac{\sum_{i=1}^n conf(p_n)}{\frac{\|p\|^2 + m}{m+1}} \quad (1)$$

3) *Sensor Placement*: To get a better understanding of how the number of sensors affects tracking accuracy, the algorithm is also applied to subsets of the original set of sensors in decreasing size (40, 30 and 20 sensors). Instead of choosing the sensors randomly, characteristics of sensors deemed possibly influential on tracking performance were chosen:

a) *Number of neighboring sensors*: Based on the assumption that sensors in doorways, which usually have few neighboring sensors, are critical in tracking room transitions, those in larger areas with many neighboring sensors are removed. The number of neighboring sensors can be calculated from the sensor graph.

b) *Duration of stay*: Given that tracking stationary targets is much simpler than moving targets, subsets of sensors that cover areas in which the average duration of stay is short are considered. The duration of stay can be calculated from the duration between consecutive sensor events in recorded data.

c) *Activity*: Considering the application of in-home activity monitoring, it is imperative that the placement of sensors for tracking accuracy improvement does not interfere with the necessity of covering those areas in which the majority of activity is taking place. Thus, sensors based on the amount of activity covered are selected and filtered. The amount of activity covered by a sensor is simply calculated by the number of times it is triggered.

These criteria were used to create subsets of data of varying size, selected by increasing, as well as decreasing order of the respective criterium (cf. Figure 6).

4) *Sensor Clustering*: The procedure of selecting subsets of sensors for tracking performance evaluation was also conducted for sets of 10 sensors. However, due to the selection criteria, most of the sets had removed whole rooms, and in one case all data from one individual. Thus, in order to evaluate tracking performance on 10 sensors, the sensors are clustered manually by rooms and spatial adjacency (see Figure 3), and resulting clusters are treated as individual sensors. This results in a more realistic scenario, in which motion sensors often cover different size areas up to whole rooms.

For this evaluation, data from all sensors is used, but the sensor IDs are replaced with IDs for their corresponding cluster. This way, use of all sensor events is made, but their spatial resolution is decreased.

##### B. Experiment II: Field Trial

To test the applicability of the approach in a more realistic setting, it is tested in an evaluation in three households equipped with retrofitted sensors. Since the identity of the residents cannot be reconstructed without video recordings or other identifying data, this study focuses on comparing the actual and the calculated number of people present. Thus, claims about the false associations between residents cannot be made. However, this information helps developing home security and care support services that rely on information on the number of people present.

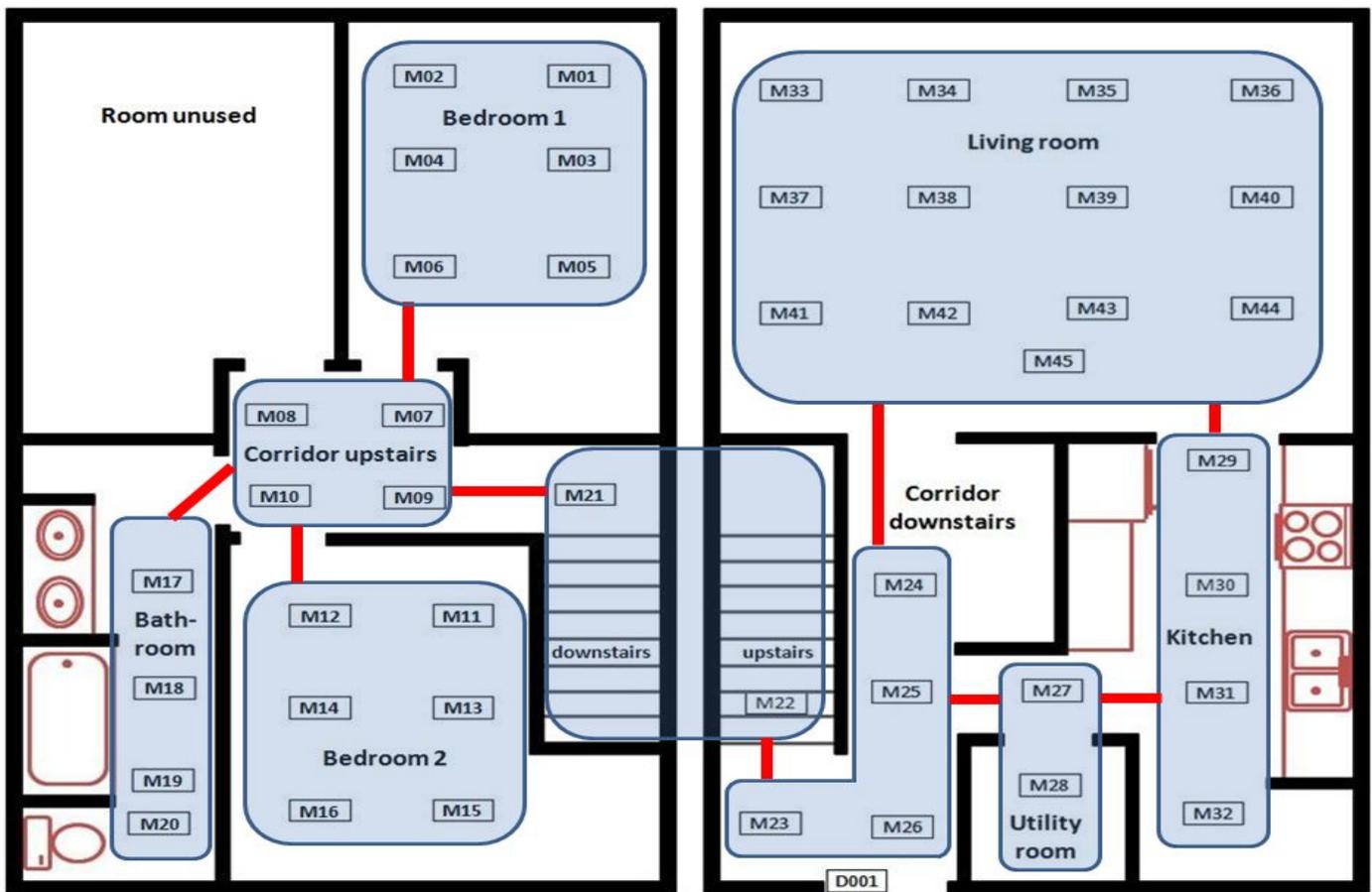


Figure 3. Layout of home lab, with placement of sensors and their corresponding clusters. Red lines between clusters lines indicate spatial adjacency as implemented in the sensor graph. After Crandall & Cook [37].

1) *Setup*: Each apartment was equipped with 13 to 15 PIR motion sensors, depending on the size of the apartment. Five of the sensors were installed to cover as large an area as possible, to enable an "inactivity monitoring". These sensors were also used in a separate study to detect unusual behavior (e.g., falls) in order to automatically invoke an emergency call. The remaining sensors were placed to cover room transitions (e.g., hallways and door frames) and smaller areas in which most activity takes place (such as the kitchen counter and the bathroom sink). The sensors further included two light barriers and two pressure mats near the front entrance. Their primary task was to enable detection of people entering and leaving the apartment. For the sake of this evaluation, however, they only provide further tracking data. A schema of the hardware setup is shown in Figure 4. A detailed description of the care related services implemented in the project has been written by Pauls et al. [39], [40] and Gerdes et al. [41].

The participants were recruited by the Johanniter-Unfall-Hilfe e.V. as part of the research project *Cicely*. Three participant couples were recruited. All lived in a two-room apartment, which was subsequently equipped with the sensors, a "small-form-factor PC", a radio receiver to collect the sensor data and a UMTS router to place emergency calls and to remotely monitor the correct functioning of the setup.

2) *Data*: During the evaluation using home lab data, the tracking results could be validated using video recordings from video cameras installed in the home lab. During the field trial, use of video cameras was considered impractical as well as a burden on the participants' privacy. Instead, they were asked to fill out an "activity log" (see Figure 5), which states sleep times and the number of people present for the whole duration of the trial. The sensors recorded data for a period of up to six weeks. With an average of approximately 610 sensor events per day, 55000 sensor events were recorded.

Not all data could be used for analysis, as participants missed to fill out the activity log for parts of the trial. Thus, this analysis focuses on the time frames in which the number of targets can be precisely reconstructed.

3) *Data Analysis*: During the trial, the algorithm ran as part of a software framework developed for management of Ambient Assisted Living (AAL) installations. Sensors were connected to this framework using two 868 Mhz radio receivers, which were in turn connected to the PC via USB. The tracking algorithm is implemented in Java, and was started as a service. Each time a sensor event was registered, the tracker was updated. At the end of the trial, the raw, recorded data as well as the tracking output was collected from the households and the estimated number of people present was extracted from the tracking results.

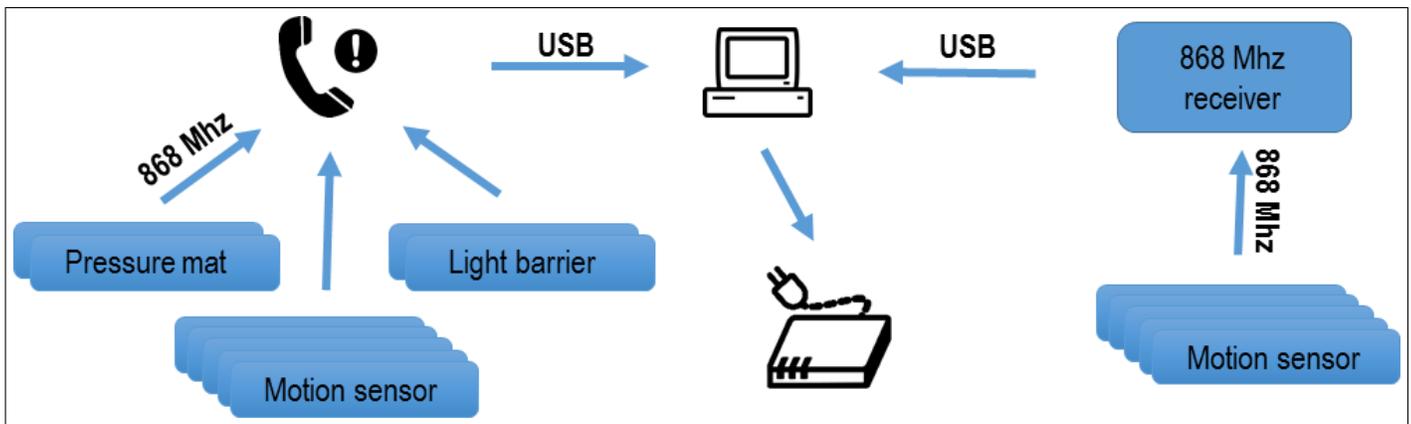


Figure 4. Hardware setup for the field trial including emergency phone, router, radio receiver and sensors.

Apart from the sensor graph, which had to be implemented after installation, the software setup was equal across all households. The number of sensors varied based on the size of the apartment.

## V. RESULTS

### A. Home Lab

Tracking accuracy using all sensors is 90.3%. This is the percentage of all sensor events across all time frames that are correctly associated to any of the targets. Correct association is determined by the labels in the dataset, which have been added manually after reviewing video recordings. The accuracy of individual time frames ranges from 62.1% to 99.5%, with a median of 93.1%. After reviewing the individual time frames, it can be seen that the variance in accuracy between time frames can be largely attributed to the varying complexity of the recorded activities. In some time frames, the residents spend little time in the same room, but move independently across the lab, thus making it easy for the tracking algorithm to separate the data. In other time frames, the residents spend much time close to another.

The median rate of false associations (events that are falsely associated to another target) per time frame is 5.88%, making false associations the largest source of error. It is clear why: While the other errors (wrongly discarding the data as noise (0.44%) and failing to associate the data to any target (0.59%)) commonly affect singular sensor events, false associations usually cause a large number of consecutive events to be associated falsely.

1) *Subset Data*: Figure 6 shows tracking accuracy across sensor subsets. The sets vary by the number of included sensors (x-axis) and their selection criterion. As can be seen, tracking

Datum	Uhrzeit		Betruhe	Abwesenheit	Besuch	Personenzahl
	von	bis				
28.08.15	14:20	23:30		X		1
29.08.15	17:00	22:00		X		2
30.08.15	06:20	15:45		X		2

Figure 5. Example excerpt of an activity log showing date, time (from, to), bed rest, absence, visit and number of people.

accuracy generally decreases with reduced sensor count. This is to be expected as the resolution of the tracking space decreases and situations with overlapping motion increases. Down to 30 sensors, tracking accuracy decreases only slightly for all but one sensor set. For the set of sensors with much overall activity, accuracy even increases slightly. The graph also shows that performance variation increases with a decreasing number of sensors. While tracking accuracy varies between 85.8% and 90.6% with 40 sensors, with 20 sensors accuracy ranges between 59.5% and 88.2%.

2) *Clustered Data*: Tracking accuracy on the clustered data set is 77.2%. This result is considerably lower than results with ten individual sensors. However, using the selection criteria explained above, more than half of the 10-sensor sets either removed whole rooms from the data, or all data from one of the residents. As a result, the evaluation using ten individual sensors is considered invalid. Considering that each of the rooms the clusters represent could be monitored using a single well-placed sensor, the tracking accuracy is surprisingly high relative to the number of sensors used.

In terms of estimation of people present, the algorithm was correct 84.5% of the time. This means that, for 84.5% of sensor events and tracking updates, the number of tracks spawned by the algorithm matches the actual number of people present. As previous studies have shown, this performance is at least as good as with the full, unclustered set of sensors [42].

### B. Field Trial

Across all households, the number of people present was correctly estimated 84.0% of the time. This means that, for 84.0% of sensor events and tracking updates, the number of tracks spawned by the algorithm matches the number of people present according to the activity log. This includes data from three households and all motion sensors. This matches the results of the clustered home lab data. The average time for the algorithm to recover (to restore the correct number of tracks) after a tracking error amounts to 9.7 sensor events, or 101 seconds.

Upon further inspection, it can be seen that the largest share of incorrect estimations lies in the night time. It becomes obvious that, when an individual is inactive, their track profile will slowly fade during updates caused by activity of other

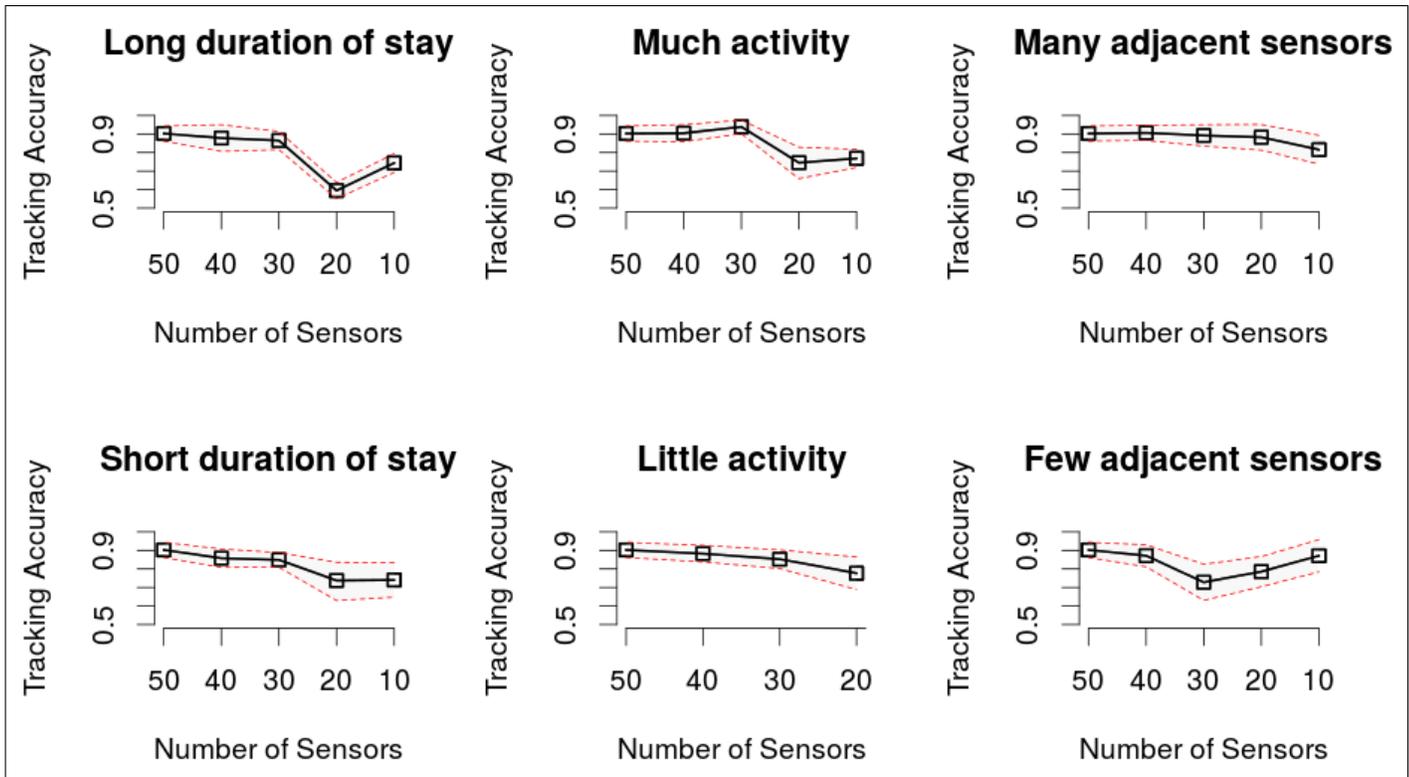


Figure 6. Tracking performance across sensor groups.

individuals. This happens, for example, when one of the residents is restless at night, or when two individuals are monitored by one and the same sensor.

Furthermore, the estimation error in terms of their impact on home security and ambulant care scenarios is considered: The algorithm could be used to activate or deactivate an alarm system when the house is empty, or to place an automated emergency call when a sole individual shows unusual behavior. Thanks to the light barriers and pressure mats, it is known when an individual left the apartment. Thus, the timestamps of the outer pressure mat with an appropriate change in the number of tracks in the output of the algorithm are compared. On average, it took 40 seconds between a person leaving the apartment and the appropriate fading of their track path. Since the number of updates depends on the amount of activity, this amount of time varied up to 174 seconds.

## VI. CONCLUSION

Technical systems to measure mobility parameters and to implement safety concepts in the home are one possibility to help caretakers manage the future challenges imposed by current and future changes in demography and care. In the upcoming decades, there will be continuously more people requiring care and fewer young people to carry out this care.

The article at hand describes an algorithm to enable previously developed concepts of care support and health monitoring in the home for households of more than one person. This is achieved through a multi-target tracking algorithm in a space monitored by binary sensors. The system utilizes ambient sensor technologies, which is considered less obtrusive

than video cameras and microphones, and more suitable for older people. It enables the separation of sensor data generated by multiple persons in smart home environments without the need for identifying sensors. The approach makes it possible to install services of home security and ambulant care support in multi-person households, which previously have only been possible to use in single-person households.

The algorithm makes use of a graph consisting of sensors as nodes and their spatial relations as vertices. Compared to other related works, the algorithm works particularly well in low-resolution settings (i.e., with few binary sensors). It was shown that tracking accuracy can be improved by placing sensors based on activity characteristics. For example, sensors with many neighboring sensors provide a consistently higher accuracy than those with few, and sensors in places where the duration of stay is long on average prove to be less beneficial than those where duration of stay is short.

It was shown that the decrease of tracking accuracy resulting from smaller sets of sensors (i.e., decreased target space resolution) can be largely absorbed by selective placement of sensors. It was shown that tracking two targets in a network of 20 sensors or more can be achieved for over 90% of the time.

It must be noted that differences in tracking performance may not only be due to advantageous sensor placement, but also due to favorable data: While tracking in space with many adjacent sensors works well, it neglects in part space where tracking might be particularly difficult but useful, such as in narrow hallways. The share of total events covered by the different subsets of sensors range from 11 to 98%.

The algorithm was also successfully used in a field trial and it was shown that it is possible to correctly estimate the number of people present over long periods of time. The algorithm was also found to be useful for applications of home security and technical care support, as it swiftly determined when a person has left or entered the monitored area. A conclusion of whether the error rates are sufficiently small for use in practical applications cannot be given, because this depends largely on the application itself. For example, in order to collect mobility data of an individual during times they are alone at home, only a small percentage of the overall time is required to collect this data. However, an accuracy of more than 90% may still not be sufficient for the data to be used in medical decision making. It was shown that the implementation of the algorithm works reliably with different sensors and sensing technologies to separate sensor data of two residents in a shared apartment. The number of people present was correctly calculated 84.0% of the time. This is less than expected, but can largely be explained by difficulties differentiating between absence and inactivity, such as when people are sleeping.

The results of the field trial show that the tracking algorithm is sensitive to differences in the amount of activity between individuals. This is due to the fact that the number of updates is based on the number of sensor events received, and every track that is not assigned any event during an update fades, i.e., a smoothing function to this track's probability distribution is applied. When the peak probability of a track falls below a certain threshold, it is removed from the hypothesis space. This is done to enable fading of tracks when individuals leave the house without the need to rely on special sensors monitoring the entrance and exit doors. However, this is cause for tracking errors when one individual is inactive, such as during night. Neither the installation of additional sensors nor the omission of the smoothing function seem viable options at this point, and the problem remains unsolved.

## VII. FUTURE WORK

The home lab experiment gives insight into the importance of sensor placement for multi-target tracking using binary sensors. The next step must be to find the ideal sensor setup for the data used in this evaluation, which may be a mixture of the sensor subsets and criteria examined here, and may contain criteria which were not yet considered.

Furthermore, the algorithm's performance with more than two targets must be evaluated. The tracking of two to three individuals is sufficient for most applications this approach is targeted towards. However, the approach is not theoretically limited in its applicability and studies with three or more individuals may provide better insight into the limitations of the algorithm.

It is further planned to include identifying information in the algorithm so as to not only associate the data to tracks, but to identify the target. This could be easily achieved using the labelled data from the home lab (c.f. Nillius et al. [43]), but ultimately it may be possible to assign tracks to individuals using unsupervised clustering techniques, using the meta information of sensor placement and expected number of people present. This way, the sensor graph could be replaced by individual motion models, further improving tracking accuracy.

## ACKNOWLEDGMENT

The authors would like to thank the team of the Center for Advanced Studies in Adaptive Systems (CASAS) at the Washington State University for making their data publicly available. The authors would further like to thank the *Cicely* consortium for making the field trial possible.

This work was funded by the German Ministry for Education and Research (BMBF) within the research project *Cicely* (grant 16SV5896).

## REFERENCES

- [1] S. M. Müller and A. Hein, "Multi-Target Data Association in Binary Sensor Networks," in AMBIENT 2015, The Fifth International Conference on Ambient Computing, Applications, Services and Technologies, July 2015, pp. 79–84, [Last access: 2016-06-15]. [Online]. Available: [http://www.thinkmind.org/index.php?view=article&articleid=ambient\\_2015\\_3\\_50\\_40044](http://www.thinkmind.org/index.php?view=article&articleid=ambient_2015_3_50_40044)
- [2] United Nations, "World mortality report," Tech. Rep., 2013. [Online]. Available: [http://www.un.org/en/development/desa/population/publications/pdf/mortality/WMR2013/World\\_Mortality\\_2013\\_Report.pdf](http://www.un.org/en/development/desa/population/publications/pdf/mortality/WMR2013/World_Mortality_2013_Report.pdf)
- [3] J. Vaupel and H. Lundstrom, "The future of mortality at older ages in developed countries." 1994.
- [4] G. E. Stockmayer, "The demographic foundations of change in us households in the twentieth century," Ph.D. dissertation, University of California, Berkeley, 2004.
- [5] M. Eisenmenger, O. Pötzsch and B. Sommer, "Germany's population by 2050. Results of the 11th coordinated population projection," Federal Statistical Office, Tech. Rep., 2006.
- [6] S. Ruggles, M. Sobek, T. Alexander, C. A. Fitch, R. Goeken, P. K. Hall, M. King, and C. Ronnander, "Integrated public use microdata series: Version 3.0 [machine-readable database]." 2004.
- [7] H. Rothgang, R. Müller, and R. Unger, Themenreport "Pflege 2030": was ist zu erwarten - was ist zu tun? Bertelsmann Stiftung, 2012.
- [8] W. H. Organization et al., "The european health report 2009: Health and health systems," 2009.
- [9] A. H. Association et al., The AHA annual survey database. American Hospital Association, 2005.
- [10] T. Frenken, B. Vester, M. Brell, and A. Hein, "atug: Fully-automated timed up and go assessment using ambient sensor technologies," in Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2011 5th International Conference on. IEEE, 2011, pp. 55–62.
- [11] A. Fleury, M. Vacher, and N. Noury, "Svm-based multimodal classification of activities of daily living in health smart homes: sensors, algorithms, and first experimental results," Information Technology in Biomedicine, IEEE Transactions on, vol. 14, no. 2, 2010, pp. 274–283.
- [12] A. Sixsmith, N. Johnson, and R. Whatmore, "Pyroelectric ir sensor arrays for fall detection in the older population," in Journal de Physique IV (Proceedings), vol. 128. EDP sciences, 2005, pp. 153–160.
- [13] S. Katz, A. B. Ford, R. W. Moskowitz, B. A. Jackson, and M. W. Jaffe, "Studies of illness in the aged: the index of adl: a standardized measure of biological and psychosocial function," *Jama*, vol. 185, no. 12, 1963, pp. 914–919.
- [14] A. Mihailidis and G. R. Fernie, "Context-aware assistive devices for older adults with dementia," *Gerontechnology*, vol. 2, no. 2, 2002, pp. 173–188.
- [15] E.-E. Steen, T. Frenken, M. Frenken, and A. Hein, "Functional assessment in elderlies homes: Early results from a field trial," in *Ambient Assisted Living*. Springer, 2014, pp. 3–17.
- [16] N. C. Krishnan and D. J. Cook, "Activity recognition on streaming sensor data," *Pervasive and mobile computing*, vol. 10, 2014, pp. 138–154.
- [17] D. Wilson and C. Atkeson, "Simultaneous tracking and activity recognition (star) using many anonymous, binary sensors," in *Pervasive computing*. Springer, 2005, pp. 62–79.
- [18] M. Addlesee, R. Curwen, S. Hodges, J. Newman, P. Steggle, A. Ward, and A. Hopper, "Implementing a sentient computing system," *Computer*, vol. 34, no. 8, 2001, pp. 50–56.

- [19] T. Kanade, R. Collins, A. Lipton, P. Burt, and L. Wixson, "Advances in cooperative multi-sensor video surveillance," in Proceedings of DARPA Image Understanding Workshop, vol. 1, no. 3-24. Citeseer, 1998, p. 2.
- [20] F. Krüger, M. Kasparick, T. Mundt, and T. Kirste, "Where are my colleagues and why? tracking multiple persons in indoor environments," in Intelligent Environments, International Conference on. IEEE, 2014, pp. 190–197.
- [21] S. Oh and S. Sastry, "Tracking on a graph," in Proceedings of the 4th international symposium on Information processing in sensor networks. IEEE Press, 2005, p. 26.
- [22] D. Marinakis, G. Dudek, and D. Fleet, "Learning sensor network topology through monte carlo expectation maximization," in Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on. IEEE, 2005, pp. 4581–4587.
- [23] M. Ogawa, R. Suzuki, S. Otake, T. Izutsu, T. Iwaya, and T. Togawa, "Long-term remote behavioral monitoring of the elderly using sensors installed in domestic houses," in Engineering in Medicine and Biology, 2002. 24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society EMBS/BMES Conference, 2002. Proceedings of the Second Joint, vol. 3. IEEE, 2002, pp. 1853–1854.
- [24] T. Barger, D. Brown, and M. Alwan, "Health-status monitoring through analysis of behavioral patterns," Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, vol. 35, no. 1, 2005, pp. 22–27.
- [25] B. Logan, J. Healey, M. Philipose, E. Tapia, and S. Intille, "A long-term evaluation of sensing modalities for activity recognition," in UbiComp 2007: Ubiquitous Computing. Springer, 2007, pp. 483–500.
- [26] E.-E. Steen, T. Frenken, M. Eichelberg, M. Frenken, and A. Hein, "Modeling individual healthy behavior using home automation sensor data: Results from a field trial," Journal of Ambient Intelligence and Smart Environments, vol. 5, no. 5, 2013, pp. 503–523.
- [27] D. Podsiadlo and S. Richardson, "The timed" up & go": a test of basic functional mobility for frail elderly persons." Journal of the American geriatrics Society, vol. 39, no. 2, 1991, pp. 142–148.
- [28] J. H. Shin, B. Lee, and K. S. Park, "Detection of abnormal living patterns for elderly living alone using support vector data description," Information Technology in Biomedicine, IEEE Transactions on, vol. 15, no. 3, 2011, pp. 438–448.
- [29] M. Skubic, R. D. Guevara, and M. Rantz, "Testing classifiers for embedded health assessment," in Impact Analysis of Solutions for Chronic Disease Prevention and Management. Springer, 2012, pp. 198–205.
- [30] M. Floeck and L. Litz, "Activity-and inactivity-based approaches to analyze an assisted living environment," in Emerging Security Information, Systems and Technologies, 2008. SECURWARE'08. Second International Conference on. IEEE, 2008, pp. 311–316.
- [31] A. Crandall and D. Cook, Human Behavior Recognition Technologies, H. W. Guesgen and S. Marsland, Eds. IGI Global, Mar. 2013.
- [32] T. Frenken, E.-E. Steen, M. Brell, W. Nebel, and A. Hein, "Motion pattern generation and recognition for mobility assessments in domestic environments." in Proceedings of the 1st Int. Living Usability Lab Workshop on AAL Latest Solutions, 2011, pp. 3–12.
- [33] S. Müller, A. Helmer, E.-E. Steen, M. Frenken, and A. Hein, "Automated clustering of home sensor networks to functional regions for the deduction of presence information for medical applications," Wohnen–Pflege–Teilhabe–Besser leben durch Technik , 2014.
- [34] S. Blackman, "Multiple hypothesis tracking for multiple target tracking," Aerospace and Electronic Systems Magazine, IEEE, vol. 19, no. 1, 2004, pp. 5–18.
- [35] S. Müller and A. Hein, "Multi-target tracking in home sensor networks," in Proc. of the Workshop on AI Problems and Approaches for Intelligent Environments. Prague, Czech Republic: CEUR-WS.org, Aug. 2014, to appear.
- [36] D. Reid, "An algorithm for tracking multiple targets," Automatic Control, IEEE Transactions on, vol. 24, no. 6, 1979, pp. 843–854.
- [37] A. Crandall and D. Cook, "Tracking systems for multiple smart home residents," Human Behavior Recognition Technologies, 2010.
- [38] D. Cook and M. Schmitter-Edgecombe, "Assessing the quality of activities in a smart environment," Methods of information in medicine, vol. 48, no. 5, 2009, p. 480.
- [39] A. Pauls, S. Simon, M. Schwarz-Eywill, V. Gerdes, and C. Scheve, "Intelligentes monitoring von subjektiven und objektiven parametern in der spezialisierten ambulanten palliativversorgung (sapv) – das bmbf-projekt cicely," Zeitschrift für Palliativmedizin, vol. 15, no. 03, 2014, p. PC207.
- [40] —, "Kurze informationswege in der spezialisierten ambulanten palliativversorgung (sapv) – das bmbf-projekt cicely," Zeitschrift für Palliativmedizin, vol. 15, no. 03, 2014, p. V60.
- [41] V. Gerdes, S. Simon, A. Pauls, H. Rolfes, and C. Scheve, "Unterstützung der selbstbestimmung am lebensende: Empowerment für patienten in der spezialisierten ambulanten palliativversorgung (sapv) – das bmbf-projekt cicely," Zeitschrift für Palliativmedizin, vol. 15, no. 03, 2014, p. V135.
- [42] S. M. Müller, E.-E. Steen, and A. Hein, "Inferring multi-person presence in home sensor networks," in Ambient Assisted Living. Springer, 2016, pp. 47–56.
- [43] P. Nililius, J. Sullivan, and S. Carlsson, "Multi-target tracking-linking identities using bayesian network inference," in Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, vol. 2. IEEE, 2006, pp. 2187–2194.

# Application Server Availability; Measurement and Analysis of the NorNet Core

Sune Jakobsson

Department of Telematics

NTNU

Trondheim, Norway

e-mail: [sune.jakobsson@telenor.com](mailto:sune.jakobsson@telenor.com)

**Abstract**— This article investigates the availability of applications servers running on the NorNet Core test-bed. NorNet Core is the world's first, open, large-scale Internet test-bed for multi-homed systems and applications. Particularly, it is currently used for research on topics like multi-path transport and resilience. The NorNet Core test-bed provides access to worldwide distributed nodes, connected with multiple interfaces over a set of ISPs (Internet Service Providers), providing independent transport paths between them. Each node has a set of programmable nodes that can be used for network experiments. The objective of this paper is to describe a practical approach to assess how suitable this test-bed is for distributed computing, and application servers. In conclusion, the approach taken in this paper shows that by using a simple client to check all communication links, and then create plots of the data, one can observe changes over time, due to network failures, or changes to the network between the sites.

**Keywords**- *Test-bed; Java virtual machines; application servers; availability; tunnelling.*

## I. INTRODUCTION

This paper is an extended version of a conference paper by the author [1]. This paper addresses the behaviour and availability of distributed computing resources in a “virtual” network built on top of academic and commercial networks, like the NorNet test-bed [2]. The main goal is to use simple tools and mechanism already available, to assess the properties of the NorNet test-bed. It is often necessary either do this assessment, before the test-bed is taken into use, or run the assessment in parallel with the trials. The NorNet test-bed provides a multi-homed IP infrastructure with nodes distributed over the world. The conference paper [1], where it is demonstrated that simple monitoring, using HTTP requests and monitoring memory usage is sufficient to access the availability of a server. Using this technique over months of time, one gets a reasonable view of the availability for the measured system, including long-term memory leakages. A previous paper discusses the availability of web servers in commercial settings using providers (e.g., Amazon, Google and other providers) hosting the computing resources, that use the Internet as the transport network [3]. In such setting, you as a customer, have little or no control over the computing resources. In that case it is hard to assess to what extent the computing resources are shared or virtualized, but one can assume that the Internet itself is a reasonably stable platform for transport. However, as a customer of the cloud computing resources, one has little or no control of the

instance of deployment, or how the instance is configured, and how well it is monitored. This is due to the fact that commercial providers do not disclose any or very little information regarding their internal infrastructure, keeping this as company secrets. In the NorNet Core setting, one has near complete control over and information about the computing resources, but limited control over the point-to-point tunnels running between the sites. For additional publications regarding the NorNet Core, please visit [4].

The objective of this paper is to assess the behaviour and availability of servers serving Web pages and running in the NorNet Core network and the transport of IP packets between the sites. It describes a series of simple experiments at the application level, i.e., invocation of Web servers and how to capture their continuous operation and long term behaviour.

In Section II, we describe the NorNet test-bed in detail and how the experiments were carried out. The goal of these measurements was to detect network changes and service quality in the test-bed over periods of days, weeks and months, due to issues that can be traced back to the inter-site communication links or the software (SW) running at the sites and how these issues affect application servers running on the test-bed. The NorNet Core test-bed was continually updated and upgraded and the packet route the tunnels use was entirely up to the ISP, so there was a number of factors that impacted the availability.

Section III addresses the details of the monitoring, and what tools were used to collect the data, and validate the measurements.

Section IV provides some highlights of selected subsets of the results, and discusses the visualisation used in the plots.

Finally, Section V discusses the shortcomings of a best effort worldwide distributed test-bed and provides recommendations for test-beds in general and areas for improvement of the NorNet Core.

## II. THE NORNET CORE TEST-BED

### A. Test-bed structure

As of writing May 2016, the NorNet Core test-bed [2] is deployed on 20 sites physically distributed across the world and interconnected with tunnels over 15 different ISP's networks. The majority of the sites are at the major universities in Norway, at Simula Research Laboratory in

Oslo and the rest are at universities in Sweden, Germany, China, Korea Australia and USA. The 16 ISP's provide connectivity across the sites, so that roughly half of the sites are connected using tunnels with more than one ISP involved. As an example, the University in Trondheim, is connected to Simula Research Laboratory by Uninett and PowerTech, and hence provides two independent communication tunnels with their respective interfaces, hence providing a true multi-homing configuration.

A screen shot of the testbed console is shown in Figure 1. The colour of the links between the sites, indicate their status, and a similarly the pins on the sites indicate the site status. Green indicates that the site or link is available. The site names and their ISPs are shown in Table I. We do not know how the ISPs route the packets between the sites, but we can observe when the underlying packet routing changes over time.

Each site has a set of research systems running virtual machines for experiments, a control-box for management functions, and a tunnel-box that terminates the tunnel end-points between the sites. The tunnel-box terminates all the tunnels at a site, and also provides the assignment of the local IP addresses, which allows an experimenter his own IP range. The NorNet Core runs its own domain name service (DNS), and the tunnels provide both IPv4 and IPv6 connectivity between the sites. The tunnels provide site to site connectivity over academic and commercial IP networks. The overall structure of a node in the NorNet Core is illustrated in Figure 2.

The thin red line from the control box is the connection to the central management system in Oslo. The primary ISP is used for the configuration from the central Simula site.

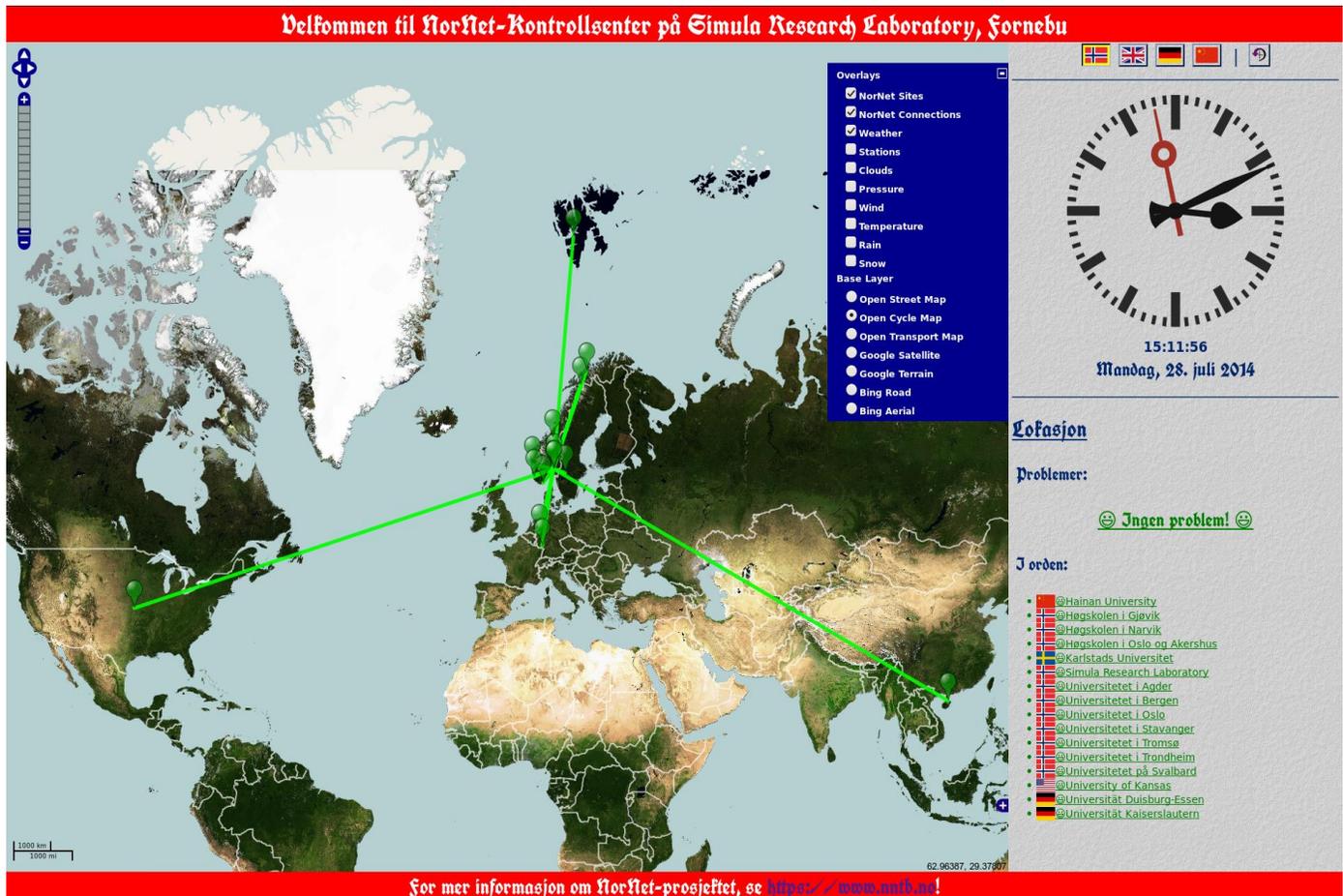


Figure 1. Screen shot of Control center for NorNet Core servers.

Table 1. Locations of NorNet Core server and their respective ISP's

Index	Site	ISP1	ISP2	ISP3	ISP4
1	Simula Research Laboratory	Uninett	Kvantel	Telenor	PowerTech
2	Universitetet i Oslo	Uninett	Broadnet	PowerTech	-
3	Høgskolen i Gjøvik	Uninett	PowerTech	-	-
4	Universitetet i Tromsø	Uninett	Telenor	PowerTech	-
5	Universitetet i Stavanger	Uninett	Altibox	PowerTech	-
6	Universitetet i Bergen	Uninett	BKK	-	-
7	Universitetet i Agder	Uninett	PowerTech	-	-
8	Universitetet på Svalbard	Uninett	Telenor	-	-
9	Universitetet i Trondheim	Uninett	PowerTech	-	-
10	Høgskolen i Narvik	Uninett	Broadnet	PowerTech	-
11	Høgskolen i Oslo og Akershus	Uninett	-	-	-
30	Karlstad Universitet	SUNET	-	-	-
40	Universität Kaiserslautern	DFN	-	-	-
41	Hochschule Hamburg	DFN	-	-	-
42	Universität Duisburg-Essen	DFN	Versatel	-	-
43	Universität Darmstadt	DFN	-	-	-
88	Hainan University	CERNET	CnUnicom	-	-
100	Univercidade Federal de São Carlos	RNP	-	-	-
160	Korea University	KREONET	-	-	-
200	National ICT Australia	AARNet	-	-	-

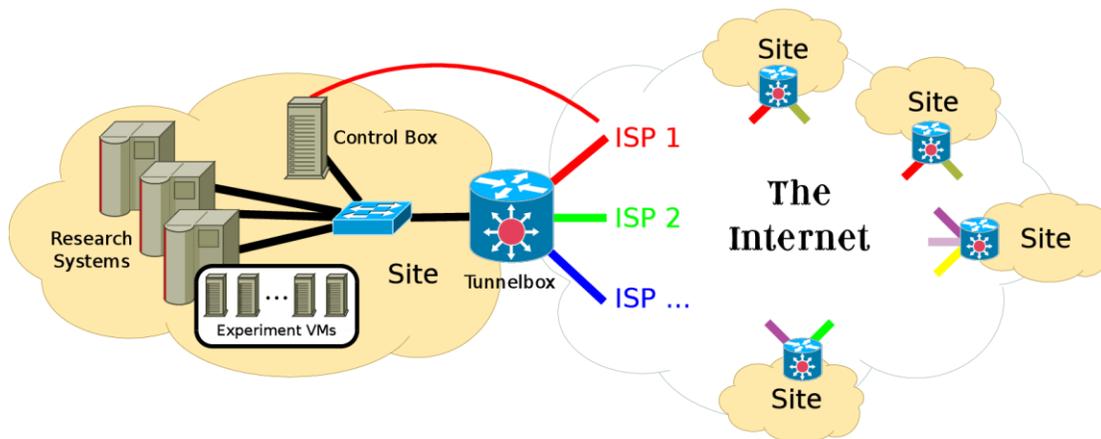


Figure 2. Overall structure of a single NorNet Core site.

Each site contains a set of physical servers that host individual virtual machines running instances of Planet lab software [5] for managing the sites. The virtual machines (VM) run the Fedora version 18 operating system [6], and connect to all available VPN tunnels at the site through the tunnel-box, and researchers can use them for multi-homed experiments as needed. Each site contains a number of VM instances, and they are all connected to all ISP's at the site. The experimenters are free to install SW on the VM's as needed. These VM instances are referred as slivers in the NorNet terminology. Please note that the term sliver in this paper refers to a running VM at a site. The term is also used

by Fedora, but with a different meaning in their setting. The test-bed is configured so that the users get global access to all nodes, and they are able to do experiments on each node as needed, by accessing each virtual machine on an instance by instance basis. This allows individual users to get assigned VM's with private IP addresses, and do not need to consider sharing network interfaces with other users. There are some restrictions on what access rights a user is assigned to the operating systems on each site, and access to all operating system instances is done through the central site at Simula Research Laboratory in Oslo, Norway. These restrictions include tunnel configuration, and the underlying

management of the research systems at a site. The entire NorNet Core infrastructure is managed from Simula Research Laboratory and their technical staff in Oslo, Norway.

### B. The measurement setup

The NorNet Core test-bed at Simula Research Laboratory has kindly provided an additional central node in Oslo for measurement purposes, which has direct access to the network interface at the site, and is able to do packet capture on the wire, so that the behaviour of the network can be captured and studied in retrospect. As shown in Figure 3, when site A invokes the Web server at site B, using all three networks in between, this traffic is captured by the measurement node at Simula Research Laboratory. This means that each server has a dedicated interface for each of the tunnels, and is able to send and receive IP traffic independently on each ISP. This implies that each site has a separate IP address for each tunnel, and one can therefore use IP addresses to select which tunnel to use.

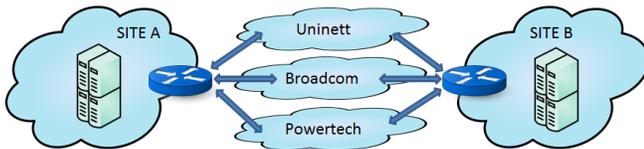


Figure 3. Tunnels between two sites

The measurement server at the Simula site is able to record the entire communication between the Simula site and any arbitrary chosen site, allowing investigation at the packet level of IP. The detailed measurements are done on HTTP calls issued from the Simula site in Oslo, where the calls are issued at fixed intervals to a selected set of sites, making calls over all tunnels, and thereby using the infrastructure provided by all involved ISP's. Since this is also the node that manages all other tunnels and nodes, and has other usages within Simula Research Laboratory, it is fair to assume that any operational issues are observed and rectified within reasonable time. This central measurement node runs the Ubuntu operating system and is directly connected to four ISP's. The measurement node is hosted in an Experiment VM, as shown in Figure 2, with access to the site traffic. The HTTP calls from the measurement node are issued by a shell script using the *curl* command [7] and the *crontab* [8] mechanism to schedule the commands every minute, and the results are captured in a log file, by the Web server (Jetty) as shown in Figure 4. The results from two different days without down-time and invocation errors on a particular ISP (Broadnet) and site (University in Oslo) are shown in Figure 6 and Figure 7, where the status (200 OK) is shown in a horizontal pink line and the black lines are the  $T_{\text{dns}}$  (DNS lookup times), the blue lines are the  $T_c$  (connection setup times), and the green lines are the  $T_t$  (total invocation time). Note that in most invocations the black DNS lookup time is so small that it is nearly invisible, and hidden at the bottom. However, some of the invocations might exceed the acceptable time constraints, but this depends on the actual application, and its requirements. The

amount of free memory is shown in bytes, and highlights the staircase pattern, that is typical for reoccurring garbage collection. The regular pattern is due to the fixed interval of the invocations, which allocate the same amount of buffers for each call.

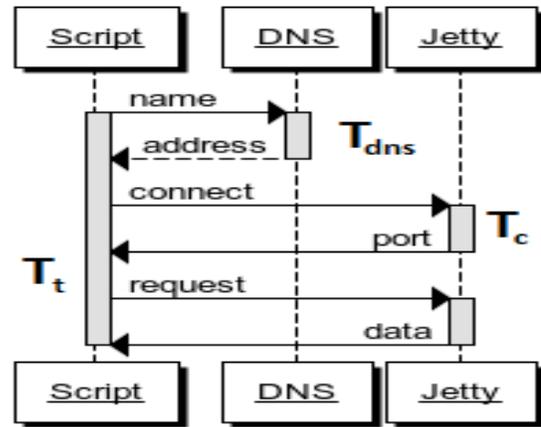


Figure 4. Invocation sequence diagram for the measurement script.

### C. Experiment VM at sites

For the availability experiment, an instance of an Embedded Jetty Server [9] runs as a Web Server and listens to HTTP requests on all interfaces, connected to the individual ISP tunnels at all participating sites. The HTTP requests are issued with the *curl* command, and scheduled with *crontab*. The invocations are scheduled at one minute intervals, and each ISP tunnel is used in 1440 measurements per day. The number "1440" comes from the number of minutes in a day, and is also the shortest interval one can define in "crontab". The measurement period is over several months, so using a granularity of one minute invocations, is sufficient to detect irregularities in the long run. The measurements are also considered to be independent events. Since the network is virtualized and each sliver has its own unique IP addresses on each of the ISP tunnels, the HTTP requests are tagged with time-stamps and also logged locally on each Web server. The Web Server on each site logs locally the incoming request and their unique invocation tags, and responds with a short response containing the amount of available memory on its Java instance. The triplet of IP source and destination addresses and time-stamp provides a unique identifier in the local logs. This also eases the identification of the packets captured on the wire between servers and clients, and makes it possible to observe the network behaviour at the packet level, with the packet sniffing tools.

## III. MONITORING OF THE TEST-BED

There are multiple issues that one wants to observe in order to determine the stability of a test-bed. One is the nodes themselves, their ability to communicate, and what

changes over time are observable. Given that the test-bed should be able to run Internet scale experiments, observing them from an application point of view will give a real life view of its abilities. The physical layout of the NorNet Core needs a set of experiments carried out in parallel in order to pinpoint possible performance or availability issues that can occur, whether they occur on a single sliver, on an entire site, or on the NorNet Core as a whole.

The measuring node runs two distinct sets of measurements in parallel:

1: The first set of measurements runs towards physically distributed nodes, to detect communication issues between the sites, the tunnel SW and configuration issues towards the other sites.

2: The other set of measurements runs towards the slivers residing on one physical location, to detect local issues and if the slivers are communication, and in good health. The reasoning behind this is to be able to detect internal issues on a node, i.e., if there are issues that can be traced back to the tunnel-box and the installation at that site vs. general operation issues in the test-bed as a whole. Since all requests are issued on all ISP's available for transport at that particular site, one can determine if an issue is related to a site or to transport.

Silver: furuset, ISP: broadnet, site: UIO  
23-2-2015

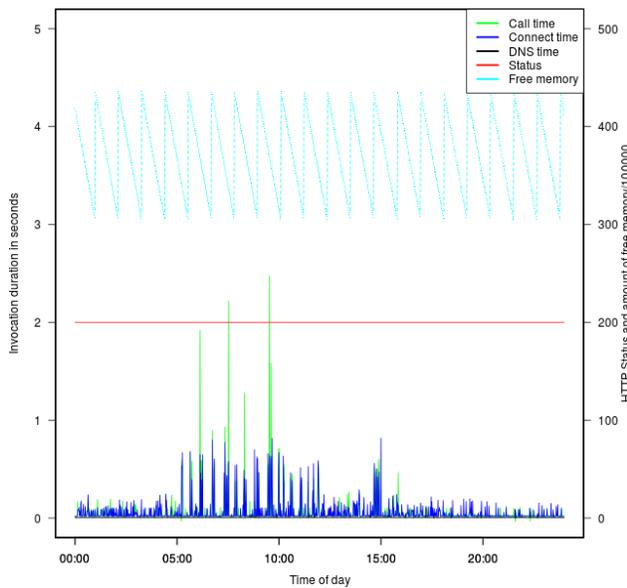


Figure 5. Invocation times and status for 24 hours

Each tunnel on each ISP can then be plotted every day as a graph shown in Figure 5 or Figure 6, to give a visual overview of the behaviour from day to day.

Silver: furuset, ISP: broadnet, site: UIO  
24-2-2015

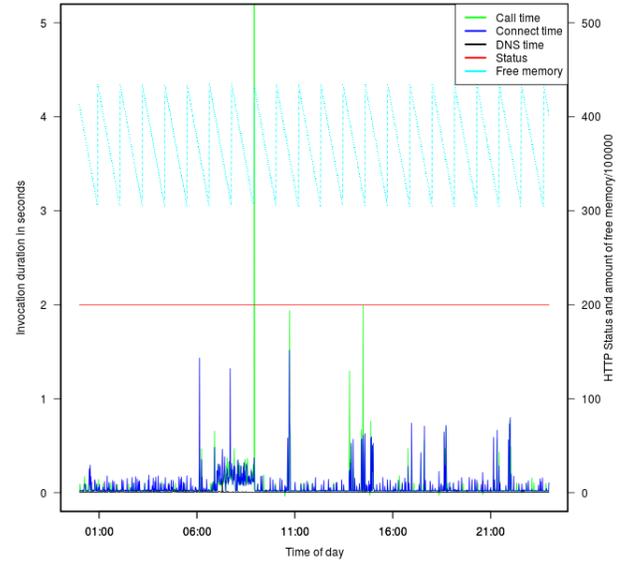


Figure 6. Invocation times and status for 24 hour

The “curl” command gives the connection time, DNS lookup time and connection time for each invocation. In addition each invocation is tagged with a time-stamp so that it is possible to explore the network behaviour at the packet level, using tools like “Wireshark” [10] and to do post-mortem inspection of unusual or odd behaviour in the HTTP communication between the sites. Wireshark, a network analysis tool formerly known as Ethereal, captures packets in real time and display them in human-readable format. For the users of the testbed the packet capture is only available at the Simula site. In addition the local logs are available at each node that is invoked.

With the redundant transport between the nodes it is easy to determine the overall condition of an individual tunnel and an individual sliver between the measuring node and the sliver. By automating the plot generation, daily plots are easily generated like the ones shown in Figure 5 and Figure 6. This, however, results in great numbers of plots and checking them all for abnormalities can be a daunting task. By enforcing a limit on the invocation time  $T_t$  on tunnels or sites, when this limit is exceeded, issues are easily identified and can then be inspected further. By visually comparing plots between different physical sites, it is straight forward to identify global issues or particular issues only manifesting themselves at one site or on one single tunnel (ISP) providing connectivity to that site.

It is also desirable to assess the network characteristics of the tunnels on a daily basis by a statistical analysis. Since the

tunnels are tunnelled over Internet or some local transport, their characteristics varies over time. The connection times  $T_c$  for a particular tunnel (Broadnet) and site pair (University in Oslo) for two days, are shown as a density plot in Figure 7 or as a visual plot of an empirical cumulative distribution as shown in Figure 8. Given the shape of the distributions and the number of samples per day, the Kolmogorov-Smirnov test [11] is chosen to be the most suitable test to compare the daily connection time data.

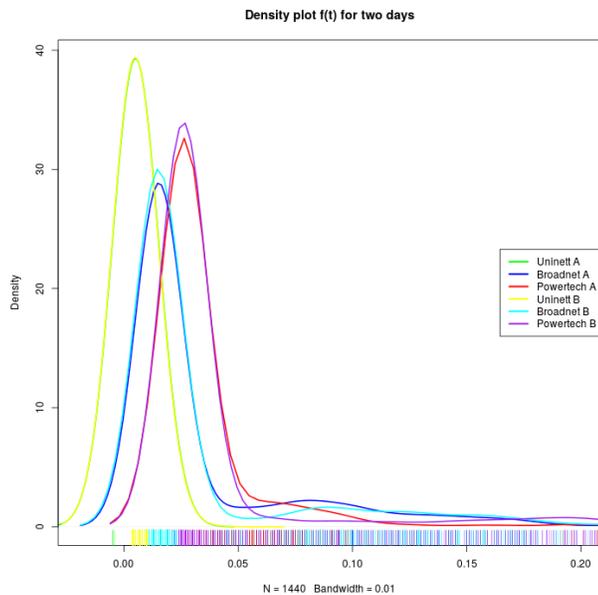


Figure 7. Density plot of connection times.

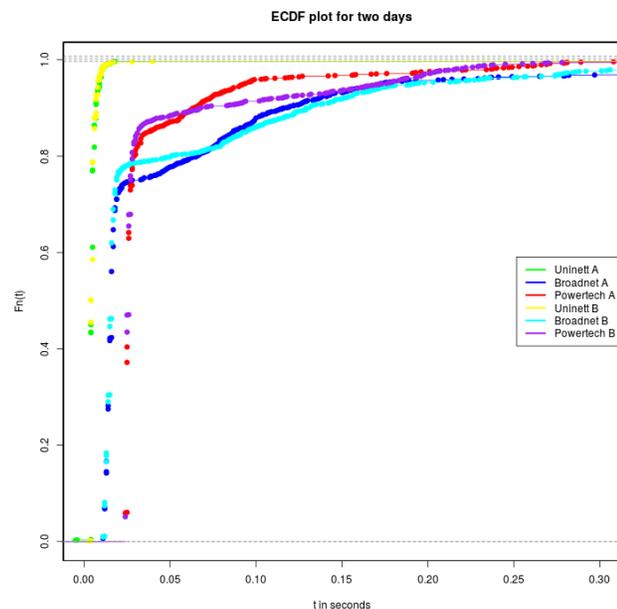


Figure 8. Empirical Cumulative Distribution Function.

The daily connection time distribution can be determined for each tunnel and sliver pair and the result gives an indication if there are changes in the communication between the measuring node and a particular sliver. Uninett is the ISP for the universities in Norway, whereas Telenor, PowerTech, Kvintel, Altibox, BKK and Broadnet are commercial ISP providers in Norway.

#### IV. RESULTS

By assessing daily measurements first at HTTP invocation level, and by defining an acceptable maximum invocation time, depending on the application and the usage, and comparing connection-time distributions and slivers memory usage patterns, enables detection of changes in all involved parts of the test-bed. For a typical Web site used by a human one would expect a maximum tolerable page loading time of 5 second. After that a user would either abandon the site or try to reload the Web page in order to get a response, and hence reissue the original request. We therefore consider invocations that take more than 5 seconds as lost or missing, even if the request would succeed at a later point. By overlying the daily HTTP invocation and status plots one can identify “global” issues affecting the entire test-bed, and as well as “local” issues affecting one site, or one ISP tunnel between the measuring node and the site or sliver. By “global” issues we mean events that impact the entire NorNet Core network, like the DNS or the management functions, whereas “local” issues are issues that affect only one site. As shown in the sub-plots for each ISP in Figure 9. They show the maximum connection time for each ISP, spanning over several weeks, and also the average connection time as a black line. We have a period where the entire test-bed was down, shown as lack of data, and also show the long-term average difference between the individual tunnels provided by the individual ISPs, indicating a “global” issue. The long term connection time also changes significantly from the time before the entire test-bed was upgraded to afterwards, indicating that the new test-bed SW impacts the connection time in a negative fashion. There are also some observable similarities between two of the ISPs (PowerTech and Broadcom) that could indicate that their respective tunnels are indeed sharing some IP transport links between the physical sites. Whereas in Figure 7 between time 7:00 and 9:00, there is a two hour period, where the packets on that particular ISP used on average an extra 100 milliseconds in transport. This indicates a rerouting issue for that particular ISP. Even though there are variations the connection-time the distribution is stable, unless there is a change in the IP packet route or a change in the tunnel-box SW. By viewing the daily density plots of two consecutive days (Figure 7) or daily empirical cumulative distribution functions (ECDF) (Figure 8) on tunnel and sliver pairs, the tunnels repeat the same plots, unless there is some underlying issue. Day one in the plot is referred to as A, and day two as B. In addition the Kolmogorov-Smirnov tests have been run to show that the days without changes or downtime give the same distribution. It turns out that the visual presentation of two consecutive days is better in the

ECDP plot, than in the density plot, when there are substantial differences between two days.

In addition the memory usage on the slivers has also been monitored, and the slivers do not appear to be disturbed by other processes on each sliver. They all show a regular pattern in the amount of free available memory for the virtual

machine running the Web server, and are hence not disturbed or affected by external factors, as shown in Figure 5 and Figure 6.

The Web Server SW and the scripts used to run the experiments are all available at github [12].

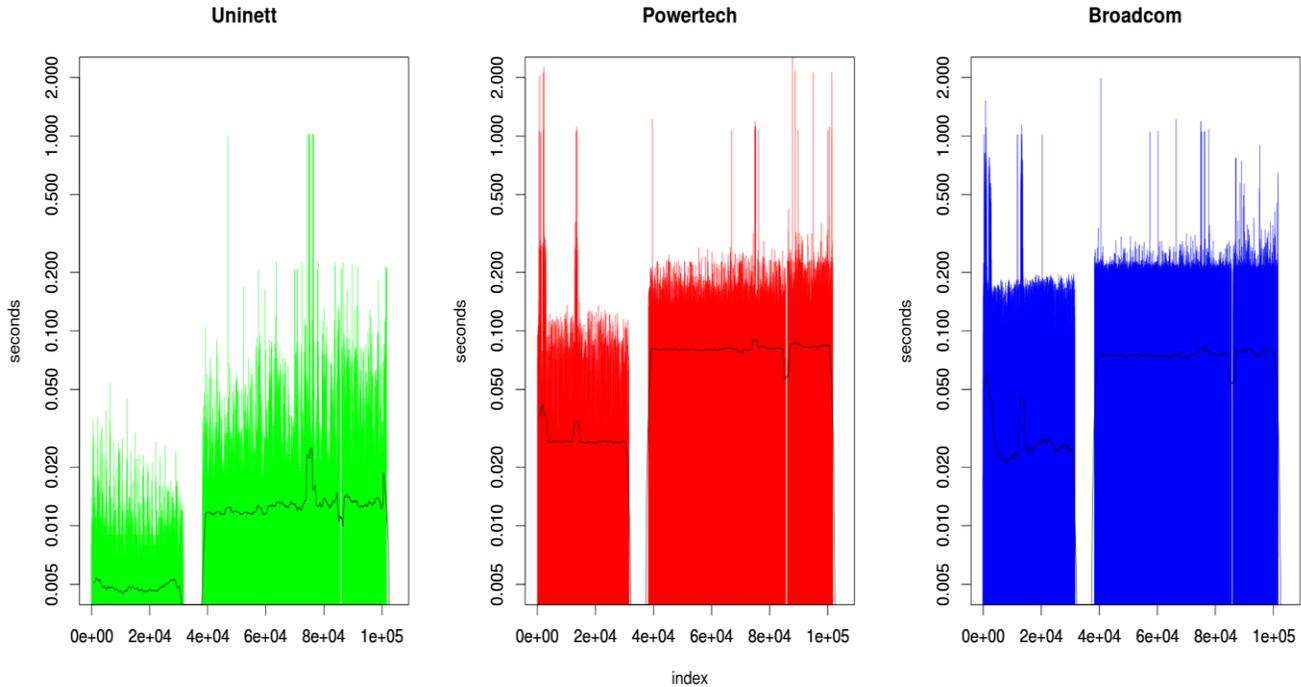


Figure 9. Longterm connection time performance (seconds) over several weeks.

## V. RECOMMENDATION

The NorNet test-bed provides a multi-homed environment for large scale Internet experiments, but it has unfortunately focused the technical aspects of such a test-bed, and primarily at the transport level between the slivers. Most of the experiments published address multi-home transport and their protocols, and are not addressing Internet style client-server usage [13].

The physical distribution of sites adds some transport time between them, and occasionally the routing changes between sites add a constant to the transport time.

The NorNet Core test-bed provides monitoring tools with graphical interfaces. The monitoring provides a colour coded overview of the sites and the links, but does not address the individual slivers at the site. However, this does not give a detailed picture of the communication between the sites nor the status or quality of the tunnels between the sites. When a site goes off-line there is limited support for bringing the site back on-line other than contacting the personnel at the site. This has some grave implications on availability if parts of the test-bed run into issues or go down outside office hours or vacation times. Being a research network NorNet Core

does not provide a service level agreement (SLA) for their users, so you do not get your money back when there are failures [14].

The lack of 24/7 operation management, it is hard to plan and carry out long term experiments. It is necessary with more than only best effort guarantees on a test-bed, to provide repeatable experiments, without adding the operational uncertainties.

The NorNet Core should add some rudimental monitoring SW for each node and each tunnel, and provide this information on the NorNet Core web page. This information could also be used internally at Simula Research Laboratory to alert the personnel in charge to quicker respond to failures or errors that are bound to happen at some point. A SLA for the users of NorNet Core could be beneficial for all parties involved.

## VI. CONCLUSIONS

The approach described in the paper shows that simple monitoring at the application level is sufficient to pinpoint issues connected to tunnels, site SW and operation of the test-bed as a whole, by only collecting the invocation times, and amount of memory on the servers at the sites. This has been accomplished by invoking simple HTTP services

between the involved sites, and collecting data about call setup, response times and memory usage. Using graphics enables one to quickly get an overview, and to pinpoint irregularities. One can then zoom down to interesting abnormalities, to post analyse the root cause of the issue. However, to fully automate the process and to use it to predict future behaviour is still work in progress. The lack of 24/7 operational management limits the possibility to run stable long terms experiments, and should be addressed for the success of the NorNet Core test-bed.

#### ACKNOWLEDGMENT

I would like to thank Professor Rolv Bræk and Professor Bjarne Helvik at Department of Telecommunication at NTNU, for their advice and guidance in my research work, and my wife for proof reading my papers.

#### REFERENCES

- [1] Sune Jakobsson, “Estimation of Performance and Availability of Cloud Application Servers through External Clients”, in *DEPEND 2013, 6th International Conference on Dependability*, pp. 1-5, ISBN: 978-1-61208-301-8
- [2] NorNet Core, <https://www.nntb.no/nornet-core>, Last seen May 2016
- [3] Sune Jakobsson, “Measuring Application Server Availability on the NorNet Core” in *DEPEND 2015, 8th International Conference on Dependability*, pp. 1-4, ISBN: 978-1-61208-429-9
- [4] NorNet publications, <https://www.nntb.no/publications>, Last seen January 2016
- [5] Planet lab, <https://www.planet-lab.org>, Last seen January 2016
- [6] Fedora 18 (Spherical Cow), [https://docs.fedoraproject.org/en-US/Fedora/18/html/Release\\_Notes/index.html](https://docs.fedoraproject.org/en-US/Fedora/18/html/Release_Notes/index.html), Last seen January 2016
- [7] Curl, tool, <http://curl.haxx.se/docs/manual.html>, Last seen January 2016
- [8] Crontab, tool, <http://www.unix.com/manpage/linux/5/crontab>, Last seen January 2016
- [9] Jetty, application server, <http://eclipse.org/jetty>, Last seen January 2016
- [10] Wireshark, tool, <https://www.wireshark.org>, Last seen January 2016
- [11] Vito Ricci, “Fitting distributions with R”, <http://cran.r-project.org/doc/contrib/Ricci-distributions-en.pdf>, Last seen January 2016
- [12] GitHub, code base, <https://github.com/sunejak/EmbeddedJetty>, Last seen January 2016
- [13] Thomas Dreibholz, Jarle Børgeengen, and Jonas Werme: “Monitoring and Maintaining the Infrastructure of the NorNet Testbed for Multi-Homed Systems”, in *5th International Workshop on Protocols and Applications with Multi-Homing Support (PAMS) 2015*, pp. 611–616, ISBN 978-1-4799-1775-4, DOI 10.1109/WAINA.2015.76
- [14] Brian Harry, “How do you measure quality of a service?”, <http://blogs.msdn.com/b/bharry/archive/2013/10/14/how-do-you-measure-quality-of-a-service.aspx>, Last seen January 2016



[www.iariajournals.org](http://www.iariajournals.org)

**International Journal On Advances in Intelligent Systems**

🔗 issn: 1942-2679

**International Journal On Advances in Internet Technology**

🔗 issn: 1942-2652

**International Journal On Advances in Life Sciences**

🔗 issn: 1942-2660

**International Journal On Advances in Networks and Services**

🔗 issn: 1942-2644

**International Journal On Advances in Security**

🔗 issn: 1942-2636

**International Journal On Advances in Software**

🔗 issn: 1942-2628

**International Journal On Advances in Systems and Measurements**

🔗 issn: 1942-261x

**International Journal On Advances in Telecommunications**

🔗 issn: 1942-2601