Quality of Context Enhancements and Cost Effective Radio over Fiber Network Planning

Shawky, Ahmed Sherif Mahmoud

DOI (link to publication from Publisher):
10.5278/vbn.phd.engsci.00089

Publication date:
2016

Document Version
Publisher's PDF, also known as Version of record

Link to publication from Aalborg University

Citation for published version (APA):
Shawky, A. S. M. (2016). *Quality of Context Enhancements and Cost Effective Radio over Fiber Network Planning*. Aalborg Universitetsforlag. https://doi.org/10.5278/vbn.phd.engsci.00089

QUALITY OF CONTEXT ENHANCEMENTS AND COST EFFECTIVE RADIO OVER FIBER NETWORK PLANNING

BY
AHMED SHAWKY

DISSERTATION SUBMITTED 2016

AALBORG UNIVERSITY
DENMARK

Quality of Context Enhancements and Cost Effective Radio over Fiber Network Planning

Ahmed Shawky

Department of Electronic Systems

Aalborg University

A thesis submitted for the degree of

Doctor of Philosophy

2016

Dissertation submitted: February 2016

PhD supervisor: Associate Professor Jens Myrup Pedersen
Aalborg University, Denmark

Assistant PhD supervisor: Associate Professor Rasmus Løvenstein Olsen
Aalborg University, Denmark

Professor Hans-Peter Schwefel
Aalborg University, Denmark

PhD committee: Associate Professor Jens Dalsgaard Nielsen (chairman)
Aalborg University, Denmark

Professor Emeritus Ignatius Niemegeers
Delft, The Netherlands

Professor Ahmed Patel
Universiti Kebangsaan, Malaysia

PhD Series: Faculty of Engineering and Science, Aalborg University

ISSN (online): 2246-1248
ISBN (online): 978-87-7112-512-2

Published by:
Aalborg University Press
Skjernvej 4A, 2nd floor
DK – 9220 Aalborg Ø
Phone: +45 99407140
aauf@forlag.aau.dk
forlag.aau.dk

Printed in Denmark by Rosendahls, 2016

This work is dedicated to my loving family in Denmark and Egypt Iman, Jes, Anders, Adam, Ragaa, Sherif, Rasha, Omar and Farah

Quality of Context Enhancements and Cost Effective Radio over Fiber Network Planning

PhD Student

Ahmed Sherif Mahmoud Shawky

Supervisor & Co-Supervisors

Associate Professor Jens Myrup Pedersen (Supervisor)

Associate Professor Rasmus Løvenstein Olsen (Co-Supervisor)

Professor Hans-Peter Schwefel (Co-Supervisor)

Published Papers & Journals

Shawky, A.S.; Bergheim, H.R.; Ragnarsson, O.P.; Riaz, T.M.; Pedersen, J.M., "An automated planning model for RoF heterogeneous wireless networks," Advanced Communication Technology (ICACT), 2010 The 12th International Conference on , vol.1, no., pp.633,638, 7-10 Feb. 2010

Shawky, A.; Olsen, R.; Pedersen, J.; Rasmussen, J.G., "Service Degradation in Context Management Frameworks," Computer Communications and Networks (ICCCN), 2011 Proceedings of 20th International Conference on , vol., no., pp.1,6, July 31 2011-Aug. 4 2011

Shawky, Ahmed; Olsen, Rasmus Løvenstein; Pedersen, Jens Myrup; Schwefel, Hans-Peter / Optimizing the Quality of Dynamic Context Subscriptions for Scarce Network Resources. Proceedings of the 1st European Workshop on AppRoaches to MObiquiTous Resilience, ARMOR '12 . Association for Computing Machinery, s. Article No. 6. 2012.

Shawky, Ahmed; Olsen, Rasmus Løvenstein; Pedersen, Jens Myrup; Schwefel, Hans-Peter / Class-Based Context Quality Optimization For Context

Management Frameworks. Computer Communications and Networks (ICCCN), 2012 21st International Conference on. IEEE Press, 2012.

Shawky, Ahmed; Olsen, Rasmus Løvenstein; Pedersen, Jens Myrup; Schwefel, Hans-Peter / Network aware dynamic context subscription management. I: Computer Networks, Vol. 58, 2014, s. 239–253.

This thesis has been submitted for assessment in partial fulfillment of the PhD degree. The thesis is based on the submitted and published scientific papers which are listed above. Parts of the papers are used directly or indirectly in the thesis. As part of the assessment, co-author statements have been made available to the assessment committee and are also available at the Faculty. The thesis is not in its present form acceptable for open publication but only in limited and closed circulation as copyright may not be ensured

Acknowledgements

I would like to thank my supervisors Jens Pedersen, Rasmus Olsen and Hans-Peter Schwefel for the fruitful meetings, discussions, and valuable comments as well as for the constant care and support they have provided during this journey. I hope that I could be as lively, enthusiastic, and energetic as Jens who truly made a difference not only during the PhD journey but also during my Master degree. I would also like to thank my colleagues and staff at the Networks and Security (NetSec) section at Aalborg University for their inspiration, support and valuable brainstorms. A special thanks goes out to Kasper Revsbech and Kim Højgaard-Hansen my colleagues, office mates and valuable friends whom I have learned so much from. I also would like to thank our great and cheerful department secretary Dorthe Sparre; whenever I was lost she always managed to give the right directions. I would also like to thank my family in Denmark, My mother Iman who always pushed me to do better and pursue higher education, my stepfather Jes for always being positive and concerned and my brothers Anders and Adam for believing in their older brother. I would also like to thank my family in Egypt, my grandmother Ragaa who have endured much during my high-school and bachelor years in Egypt, my father Sherif for teaching me how to be independent, my stepmother Rasha for being caring and loving and my brother and sister Omar and Farah who also believed in me. Finally, I would like to thank my friends Mazen and Amin for reminding me, how much fun one can have during the PhD journey.

Preface

This thesis is a result of my Ph.D. study at the department of Electronic Systems at Aalborg university, Denmark. The work has been done under the supervision of Associate professor Rasmus Olsen, Associate professor Jens Pedersen and Professor Hans-Peter Schwefel.

The thesis describes methods for enhancement of quality of context and cost effective RoF network planning. The thesis is divided into four parts:

1. Introduction and Background
2. Quality of Context Enhancements
3. Cost Effective RoF Network Planning
4. Conclusions and Outlook

Part 1 gives an overview for the motivation, state-of-the-art and current work in the field. Part 2 is divided into two chapters and is based on three conference papers and one accepted manuscript at Elsevier's Computer Networks journal. All papers have been reviewed, accepted, presented and published in conference proceedings.

Part 3 includes one chapter which is based on the master thesis, and a conference paper which also has been reviewed, accepted, presented and published in conference proceedings.

Following is a full list of publications:

1. Service Degradation in Context Management Frameworks
2. Optimizing the Quality of Dynamic Context Subscriptions for Scarce Network Resources
3. Class-Based Context Quality Optimization For Context Management Frameworks
4. Network aware dynamic context subscription management
5. An Automated Planning Model for RoF Heterogeneous Wireless Networks

Abstract

In Communication networks today there is an endless quest for increased capacity and improved quality. With wireless systems being now popular worldwide for allowing users and devices to communicate and share information with each other irrespective of their location, the development of sustainable and reliable mobile applications is becoming a rising issue for next generation networks. Manufacturers and service providers are not only looking to increase capacity, bandwidth and performance properties, but are also looking into the network's ability to support new applications, features and improved services. The increased number of mobile users puts a demand on today's networks in terms of application sensitivity, mobility and reliability.

Mobile operators are offering various context-aware services and applications to the user. Today, mobile users are demanding access to dynamic context information at any time, thus due to the sensitivity of such context applications, users can easily get frustrated in case the context information received is no longer valid. Mobile operators are in search for methods to ensure reliability of dynamic context information.

Another main concern for both mobile operators and Internet Service Providers (ISPs) is providing their customers with higher bandwidth and better network services. Mobile applications today running on various devices such as smartphones, tablets and other connected devices are consuming large amounts of network bandwidth. Mobile operators are now looking into heterogeneous networks to improve existing mobile broadband in order to accommodate such mobile applications and devices.

This PhD thesis provides mobile service providers and operators with a set of tools that aim to improve network services provided to the user. The PhD study looks into dynamic context information reliability by developing models and online algorithms that ensure increased reliability for context information exchange. Eventually, the PhD looks into improving network planning by implementing an automated network planning model that takes advantage of both Radio over Fiber (RoF) and heterogeneous wireless networks to meet the increasing demands for higher bandwidth.

Synopsis

I nutidens kommunikationsnetværk er der en endeløs søgen efter øget kapacitet og forbedret kvalitet. Trådløse kommunikationsnetværk er blevet populære verden over, fordi de tillader brugere og enheder at kommunikere og dele informationer med hinanden, uanset hvor de er. Et voksende emne for næste generation af de trådløse kommunikationsnetværk er udviklingen af bæredygtige og pålidelige mobile applikationer. Producenter og tjenesteudbydere arbejder ikke blot med at øge kapacitet, båndbredde og funktionsegenskaber, men ser også på netværkets evne til at understøtte nye applikationer, funktioner og forbedrede tjenester. Det voksende antal af mobilbrugere stiller krav til det eksisterende netværk, hvad angår applikationsfølsomhed, mobilitet og pålidelighed.

Mobiloperatører tilbyder brugeren forskellige services og applikationer, der kender til konteksten, de eksekveres i. Nutidens mobilbrugere kræver adgang til dymamisk kontekst information til enhver tid, og risikerer at blive frustrerede, hvis denne information er ikke længere er gyldig ved modtagelsen. Mobiloperatører leder efter metoder, der kan sikre pålidelighed for dynamisk informtion om kontekst.

En anden vigtig udfordring for både mobiloperatører og internet udbydere er at forsyne deres kunder med større båndbredde og bedre netværk services. Mobile applikationer, der bruges fra forskellige apparater såsom smart-phones, tablets og andre tilsluttede enheder, konsumerer store mængder netværksbåndbredde. Mobiloperatører er derfor begyndt at se på heterogene netværk for at forbedre de eksisterende mobilnetværk, så sådanne mobile applikationer og enheder kan rummes.

Denne afhandling forsyner mobil service udbydere og operatører med et værktøjssæt, som har til hensigt at forbedre de netværksservices, som leveres til brugeren. Phd-afhandlingen ser på påideligheden af dynamisk kontekst information ved at udvikle modeller og online algoritmer, som sikrer øget pålidelighed ved udveksling af kontekst information. Endelig ser afhandlingen på at forbedre netværksplanlægning ved at implementere en automatiseret 'network planning' model, som drager fordel af både 'Radio over Fibre' (RoF) og heterogene trådløse netværk for at møde de voksende krav om mere båndbredde.

Contents

List of Figures

Part I

Introduction and Background

Chapter 1

Introduction

1.1 Motivation

In today's online world there is an evolving global market for high quality services in the mobile industry. This comes as no surprise since the mobile industry is continuously evolving day by day and this evolution can be traced back since the customizable mobile ring tones and logos, and from there it has become an attractive billion dollars industry that opened room for different players such as mobile vendors and Internet Service Providers (ISPs) to continuously develop and offer high-end services to the daily mobile user.

According to Cisco [21] more than 85% of the world's population today are mobile owners in 105 countries around the world and according to [4] there are over one billion smart-phones in use today. This became a main driver for creative applications to emerge starting from bandwidth consuming applications such as Netflix which enables users to stream HD videos on their mobile devices to context sensitive applications which can be migration based applications, for example a Skype session migration from one network device to another based on context information exchange. It is a clear indicator that user entertainment has shifted from stationary devices and personal computers to our smaller hand-held smart-phones and tablets. According to [68] almost half of all mobile users are experiencing HD videos, online music, digital books, and online games on their mobile devices on a regular basis.

Since there is a large demand on mobile networks, ISPs are in continuous development of their mobile network in order to ensure network quality and to enhance the user experience. Therefore quality for both mobile services and network performance have always been a topic of research. Today mobile applications are heading towards other types of services which are unavailable on desktops and PCs. This is due to the fact that mobile applications are able to deliver context sensitive information. The combination of user mobility and the continuous increase of number of smart-phones on mobile networks is a direct result of context information becoming more dynamic. This is due to the fact that the user environment is constantly changing due to the increase in mobility and network coverage. An example of dynamic context information can be expressed in the form of application migration from a network device to another [44]. Application migration functions in a way where a user migrates a service or application from one network device to another based on dynamic context information. The service migration decision therefore requires reliable context information to insure a successful migration. If the information is not reliable, it can either lead to an incomplete migration process or a migration can be triggered to the wrong device.

The ISPs are also looking into cheaper planning methods and tools, due to the fact that the increase of mobility, network devices and bandwidth demands leads to the constant expansion of the network which has a significant cost impact. ISPs are looking into different methods to reduce both capital expenditures and operational expenditures of the network. Today, wireless base stations is a significant part of the ISPs budget, and looking into different planning technologies can be a way for ISPs to save money on the planning process.

This thesis is divided in two main sections, both sections investigate methods to improve quality and reliability of future mobile networks. The first sections of the thesis researches the area of dynamic context information exchange over mobile networks. The thesis investigates different quality assurance algorithms, and the effect they have on context sensitive applications. The second section of the thesis investigates future network requirements for efficient and cost effective network planning. Since ISPs are yearly upgrading their network infrastructure to meet market demands, it is always a challenge to plan for next generation mobile networks, while fully utilizing the current infrastructure. The thesis investigates implementation of Radio over Fiber (RoF) technology for wireless networks. RoF is an integration of wireless and fiber optic networks. This technology is supposed to have a large economical benefit when deployed as last mile solution, since it combines the good benefits of fiber networks such as large bandwidth characteristics and low cable and equipment cost. The thesis investigates an optimized planning method that enables reliable network planning of future mobile networks. The planning method should be done automatically and implements RoF technology for heterogeneous wireless networks.

1.2 Problem Formulation

The thesis is divided into three parts. Part I holds the introduction and background used in this thesis. Part II discusses context quality optimization, while Part III discusses network planning optimization. Accordingly, the problem formulation discusses both context sensitivity and planning of next generation networks.

1.2.1 Quality of Context Optimization

The ability of applications to adapt to the users' environment is often referred to as context awareness, [18], and is becoming a key factor in today's mobile networks, since users need to be able to efficiently interact with applications and platforms in a highly dynamic world. Context awareness is achieved by accessing dynamic context information, provided by context agents in a context management system and is a highly desirable feature for future mobile applications. However, the access to dynamic context information distributed in the environment has to be carefully designed to achieve scalability and context reliability. European projects like MAGNET Beyond [57], SPICE, [20] or E-SENSE[3], and others outside Europe, have been researching and developing concepts for context management for some time, whereas reliability and accuracy indicators for context information just recently caught the attention e.g. in the project SENSEI, [19].

An example on why context information reliability is an important issue can be addressed by looking into the case of service migration from one device to another. Service migration is context sensitive in the manner that information from devices is gathered and an informed decision is made by a context triggre manager in regards to migration of the service running on one device to another device on the network. If the context information gathered is unreliable, this may result in an incomplete migration process for this session running on the device. An example of such migration process can be a video call or online HD video running on a smart phone, the user arrives home and the device connects to the home network and service migration is triggered to his smart TV where a better picture resolution for the video is available. A cause of frustration to the user could be entire loss of the session, and thus video call or online video must be restarted [45; 46] .

Therefore, the reliability of the accessed information is the key to the success of any context aware application, since application adaptation should respond to current events and not to earlier ones.

Context information reliability relies on different parameters. The dynamics of the information, the type of access strategy used to obtain the remote dynamic context information and the network properties such as delay and packet loss. Attempting to optimize one parameter has an effect on another parameter, thus the context information reliability task becomes a rather complex task. An illustration of the conceptual relations/impact between the different elements that affects the reliability of the context information can be seen in figure 1.1. By attempting to change the type of access strategy it will have an effect on the network properties which can result in higher or lower delay and packet loss, which will in turn affect the reliability of context information. A stronger notion that affects context reliability in this sense would be the so-called mismatch probability (mmPr). The mmPr is defined as the probability that the received information at the time of execution is outdated. Therefore, if we cosndider a parameter such as information, via it's size it has a direct impact on network properties, however, via its dynamics it influences the reliability and the mmPr. This parameter dependency makes it quite challenging to maximize context reliability. Therefore, in this thesis the problem addressed is how to increase context reliability taking into account the different parameters affecting context reliability and their dependencies. The thesis does not analyze specific context elements, but rather stays generic in respect to the type of context element.

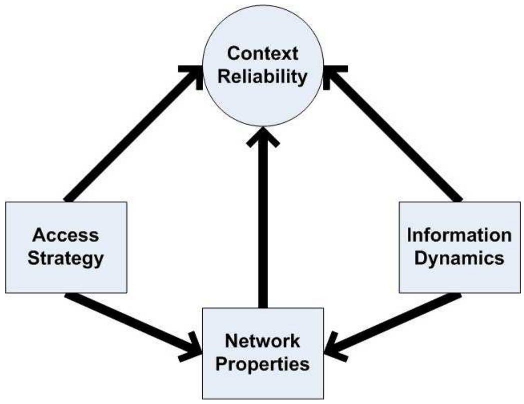

Figure 1.1: Parameters affecting context reliability

1.2.2 Planning of next generation networks

The expansion of wireless communication services has been catching service provider's attentions for quite some time. With increased demands for mobility by the user, wireless services are required to provide users with larger bandwidth and more reliable connections, also future mobile services and application are classified as bandwidth-hungry and sophisticated. While wireless systems are required to cope with the user's increasing bandwidth demands, it should also maintain the network planning and management costs at feasible economical levels.

The deployment of a cost-effective Base Station (BSs) for wireless mobile networks is the main key for service providers to remain competitive and successful in the market. Radio-over-fiber (RoF) technology reduces base station prices to a new level and therefore is a promising solution for building wireless mobile networks [48; 54]. RoF functions by modulating radio frequency onto an optical fiber. Unlike coax and copper, optical fiber is attractive for RoF systems since it is immune to electromagnetic radiations, high bandwidth, low latency and longer distances.

The main issue with 4G networks is that in order to accommodate for higher bit rates and more users, the operator has to reduce the cell sizes, this is the so called micro-cells or pico-cells concept. Another solution can be to operate on higher frequencies such as the 40 to 90 GHz region. However, both solutions are not cost-effective because if the cell size is reduced, it will lead to a higher number of BSs in order to cover the service area. If the frequency on the other hand is increased, it leads to more equipment being installed and maintained. Both solutions will contribute to higher capital and operational expenditures for the service provider.

One of the main advantages of RoF is that the BS only requires a simple optical/electrical conversion. This significantly reduces the cost of the BS RoF because the control station is shared with many BSs. This control station is responsible for functions such as signal routing, processing and resource management.

By implementing a Rof solution the ISP is able to meet both the user's demands while maintaining a cost-effective network. The thesis provides an automated planning tool for planning RoF heterogeneous wireless networks which is aimed to reduce the efforts of the network planner. It also compares the deployment price of regular BSs and RoF BSs. The thesis takes into consideration the reuse of the ISPs infrastructure if it is present, in this way ISPs can either build over the existing network or build a new network from scratch. This means that the planning tool should be able to make use of legacy systems such as 3G and WiMAX. Since the architecture is vastly different from the ones that are deployed today, it requires a different planning process [22; 48; 54].

The main challenges for implementing such an automated planning tool is that the tool itself should combine both fiber planning as well as wireless and radio planning. The tool would need to be able to introduce new algorithms that are able to facilitate the planning of wireless networks and also should be aware of the infrastructure already available for the ISP, in order to take advantage of the centralized processing offered by RoF heterogeneous wireless network architecture. The tool should also be able to make optimal fiber and radio planning scenarios i.e. it should choose the minimum number of antennas, centralized units and fiber, so that the planning process is reliable, flexible and cost-effective.

The thesis investigates how to build RoF heterogeneous wireless networks while taking into account the ISP's currently available legacy systems. The thesis provides algorithms and results for building a RoF network at the Danish municipality of Aalborg using GIS data provided by TDC. The thesis examines different algorithms and developes a planning tool that aims at decreasing the deployment cost. Finally the thesis reviews a price comparison for deploying regular mobile networks versus RoF wireless networks as proposed by the FUTON project.

1.3 Contributions

The thesis is divided into two different areas. This is due to working on different projects and topics during the PhD. The first contribution of the thesis is the study and optimization of context quality. The contributions were done during the work with Open

Pervasive Environments for migratory iNteractive services [46]. Although [1], [25] and [38] have considered the age of information or up-to-dateness, but as mentioned in section 1.2 age of information alone is not a sufficient measure of quality, since there are dependencies between the access strategy, the information dynamics and the network parameters. Chapter 3 makes an effort to explore directions to improve QoC by exploring the dependent metrics mentioned in a practical lab setting. The chapter also suggests how QoC can be improved. Chapter 4 focuses on the so-called mismatch probability quality metric. The chapter provides a set of optimization algorithms that run real-time in-order to improve context management framework's communications. There are three different set of algorithms. The first two algorithms are used to improve context information reliability and quality in a best-effort networking scenario. The last algorithm improves the context information quality under a differentiated network scenario. The second contribution was done during collaborations with FUTON project both during masters and start of PhD [54]. The thesis considers an implementation of a new automated planning model for RoF heterogeneous wireless networks. The thesis introduces planning algorithms, Fiber planning and Radio planning. The outcome of this is a tool that is able to automatically plan RoF heterogeneous wireless networks with a set of GIS data as input.

1.4 Thesis Outline

The remaining of the thesis is structured as follows:

Chapter 2 - Background and Existing Work In this chapter a brief background is presented about QoS, context information frameworks and future network planning.

Chapter 3 - Service Degradation in Context Management Frameworks The chapter introduces results of the dependent degradation metrics measured during a live test-bed setting. The chapter also proposes methods to improve context reliability.

Chapter 4 - Network Aware Dynamic Context Subscription Management This chapter is aimed to improve QoC. This chapter presents an extension to the OPEN Context control framework, the chapter introduces an extra QoC control section to the framework that optimizes context information exchange. The framework is represented and the proposed algorithms are presented as well. The chapter also shows simulation results using the 3 different algorithms and the results are compared. The chapter's contribution is a set of algorithms that can be used in a context management framework to improve Quality of Context.

Chapter 5 - An Automated Planning Model System for Radio over Fiber Heterogeneous Wireless Networks This chapter is based on the master thesis [62]. The text has been rewritten and includes in many cases new reflections from the situation today in comparison to how the situation was at the time of writing. The chapter introduces a new automated planning model system for planning RoF heterogeneous wireless networks. It presents the new model, algorithms and results for planning a wireless RoF heterogeneous wireless network in the Danish municipality of Aalborg. The chapter also presents a new deployment cost comparison between the proposed FUTON architecture and traditional wireless planning.

1.4.1 Roadmap

Chapter 3 is based on the "Service degradation in context management frameworks" conference paper. Here as the lead author of the paper I was responsible for setting up the lab experiments, collection and analysis of results along with other co-authors. The chapter's contributions is addressing the Context reliability problem, by presenting live experiment results showing the affect network degradation parameters has on context quality. The chapter also suggests methods to reduce context information degradation. Chapter 4 is based on the journal paper "Network aware dynamic context subscription management". As lead author of the paper I generated algorithms, ran simulations and collected and analyzed data together with the co-authors. The chapter continues addressing context reliability and QoC by focusing on the mismatch probability quality metric, it investigates proper subscription and class of service management. The chapter is based on the journal paper "Network aware dynamic context subscription management", and presents a set of algorithms that address proper subscription methods as well as proper class of service assignation which directly results in higher context reliability. Finally, Chapter 5 which is based on the conference paper "An Automated Planning Model for RoF Heterogeneous Wireless Networks". The paper is based on the master thesis [62] and discusses network planning issues. The motivation to continue and submit a journal paper was to find potential cost savings, which also was a motivation after starting work at TDC, which is the largest Mobile and internet service provider in Denmark. As lead author I generated algorithms, programmed algorithms, ran simulations, collected and analyzed data and collected information regarding pricing. The chapter addresses the 4G demand for higher bandwidth and expensive equipment cost, by looking into RoF network planning. The chapter introduces different automatic planning algorithms and provides cost comparison between regular mobile planning and Rof architecture.

Chapter 2

Background and Existing Work

This chapter provides a general background and state-of-the-art research for the previously mentioned research goals. The chapter is divided into two main sections, the first section introduces the Quality of Context (QoC) concept which relates to context sensitive mobile applications. The second section provides current research over RoF planning and state-of-the-art RoF architectures.

2.1 Quality of Context

Since the vast increase in smart-phone users globally, ISPs are keen on offering better Context-Aware Services (CAS). The concept of context awareness in the mobile networks world means that applications must be able to process context information explicitly in order for it to be able to adapt its behavior accordingly. An example on such CASs is as mentioned earlier service migration from one network device to another. Service migration requires access to dynamic context information, since the devices need to register information about their capabilities and applications running on them, at the same time the user location is required by the context manager in order to know when to initiate a service migration based on the devices closest to the user's position. The same applies for mobile apps such as dating and friend finder apps. Since the users are constantly mobile, location information is being constantly registered at the context manager and distributed amongst nearby users. This means that those apps are dependent on the availability of context information. Not only the availability is important, but also the up-to-dateness of the information i.e. the context information being provided at the exact right time. QoC is a set of parameters that measure the quality of Context information such as precision, probability of correctness and up-to-dateness [9; 33; 34].

According to [50] location based mobile application, such as context based city guides, friend finder applications, news and weather services tailored to the user's current location, these applications are context sensitive and suffer from the location estimation measurement process [65]. If the user is moving, the user is introduced to processing and communication delay which in turn causes additional errors on real-time mobile location applications. Since the user is constantly moving, this adds an increase to the (static) positioning accuracy which is often coming at the costs of increased delay of the positioning system.

Since [50] proven that mobility results in degradation that can be translated to delay, the thesis investigates the impact on QoC while running context information exchange at Best Effort (BE) priority, versus running context information exchange at Expedited Forwarding (EF) priority [37]. IP QoS plays a key role in prevention of real-time services application degradation [56]. It is of great importance for ISPs to set IP QoS in bottle neck areas, such as access networks, since capacity becomes congested at that segment of the network. Therefore, it is of note to see, how more reliable information exchange would be with minimized delay and see, if it truly belongs in BE class of service, or if it is too sensitive to delay, loss and jitter as VoIP traffic is.

Chapter 3 of the thesis introduces a live lab test bed experiment where this scenario has been tested.

This section provides background and current research for the following topics:

Context Management Frameworks

Quality of Services

2.1.1 Context Management Frameworks

As mentioned before the ability of applications to adapt to the user's environment is often referred to as context awareness, [57], and is becoming a key factor in today's mobile networks, since users need to be able to efficiently interact with applications and platforms in a highly dynamic world. The main example of context awareness considered in the thesis is service migration. However, another example for need of context awareness and context management could be an E-commerce application. The scenario could be as follows: If a client of an E-commerce merchant makes purchases of all kinds, the client's profile may be used to offer a customized service. Let's say the client wants for example to buy a bus ticket to visit a city near the beach. Based on the client's choice of destination, the client can then receive offers to buy pullovers because of the "temperature" measured by the weather station near the mobile cell located in the client's final destination [66].

Context awareness is achieved by accessing dynamic context information provided by context agents in a context management system and is a highly desirable feature for future mobile applications. However, the access to dynamic context information distributed in the environment has to be carefully designed to achieve scalability and context reliability. European projects like MAGNET Beyond [57], SPICE [20], or E-SENSE [3], and others outside Europe, have been researching and developing concepts for context management for some time, whereas reliability and accuracy indicators for context information just recently caught the attention e.g. in the project SENSEI [19].

In those projects as well as within the research field, see e.g. [9], [67], reliability of context information has been acknowledged as an important meta data to context information, as the end user's satisfaction obtained from context-aware systems is directly depending on the reliability of the context information. In the respect of reliability, some work focuses on information reliability in terms of the uncertainty of the information source, e.g. using fuzzy logic approaches, [64] or [58], in which reliability is related to inexactness and uncertainty of obtained information and not the timely aspects of the information. Other work focuses on reliability of the information by considering the age of the information, see e.g. [38], [1] or [25]. However, the information age needs to be related to the temporal dynamics of the information in order to provide a useful notion of reliability.

The reliability of context information is challenged by network delay, information dynamics, the context access strategy used and its parameter settings. The choice of the context access strategy on the other hand also influences network performance through the context access traffic.

For applications to be able to adapt to their environment, access to dynamic context information [57], which describes the current situation, is required. Context Management systems offer flexible access often via dedicated query languages, e.g. [17; 59], which allow applications easy access to distributed information.

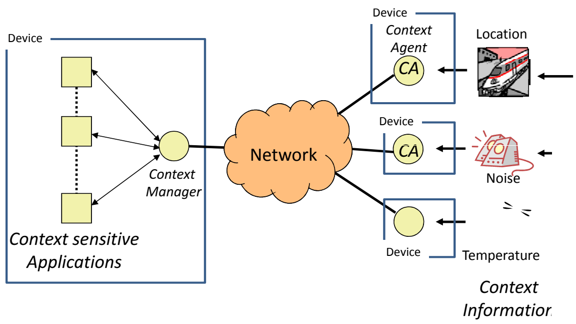

Figure 2.1: Example of Context Management system - The Context Manager resides on the Context Management Node.

A generic context management framework is illustrated in figure 2.1, which contains a set of entities called Context Agents (CA). Each CA is responsible for collecting data from its own local environment, as shown to the right in the figure: one CA exists on each device collecting information on, for instance, location, noise and temperature, respectively. A server in the network will act as Context Manager, which is responsible for maintaining the overlay network of context agents, i.e. it performs agent discovery, maintenance of agent information, information (de)registration and later on also access optimization for access to context information. In our considered architecture, context sensitive applications access context information through the context manager. The node that operates the context manager, is called the Context Management Node (CMN).

As seen to the right of figure 2.1, context information at the source changes according to external events, while the context-sensitive application would like to perform actions matching the true state of the context values at the source. When that information is dynamically changing, communication and processing delays can lead to deviations of the known context value at the application from the true value at the source. Mismatch probability (mmPr), [6] is defined as *the probability that at the time instant of using a certain information for processing in the context-sensitive application, this information does not match the value at the (physical) source.* The reliability of context information has previously been acknowledged and considered an important part of the quality of context, [9], [67], [1] or [38], but not used in any effective way. The thesis uses this quantitative context reliability metric for more accurate choices of access strategies to improve reliability for all context sensitive applications requiring access to various dynamic distributed information elements.

The mismatch probability depends not only on the two stochastic processes a) network delay and b) information change process (See Chapter 4 for description of network delay and information change), but also on the strategy by which the information is accessed. The following three strategies will be used as principle mechanisms to access remote dynamic context information:

Reactive strategy: whenever the application intends to process a certain context value, the context manager sends a request to the context providing agent, and gets a response with the information value in return. Figure 2.2 illustrates this access strategy.

Proactive, event driven: the context manager has set up a subscription to the context providing agent, and each time information changes value, the context agent sends an update to the context manager. For continuous information types, such change events can be defined via discretization intervals. The proactive event-driven strategy is illustrated in figure 2.3.

Proactive, periodic update: the context manager has set up a subscription to the context providing agent, which after recurring time interval, sends the current value to the context manager. Figure 2.4 illustrates this periodic strategy.

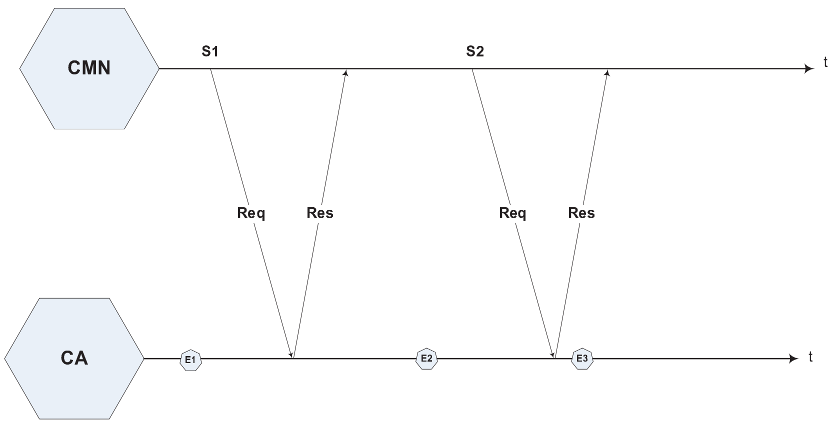

Figure 2.2: Reactive Strategy - The CMN sends request to the CA. The first request in the shown example leads to a matching information element, whereas the second ones leads to a mismatching information element. The end-to-end delay for sending the response impacts the mismatch probability.

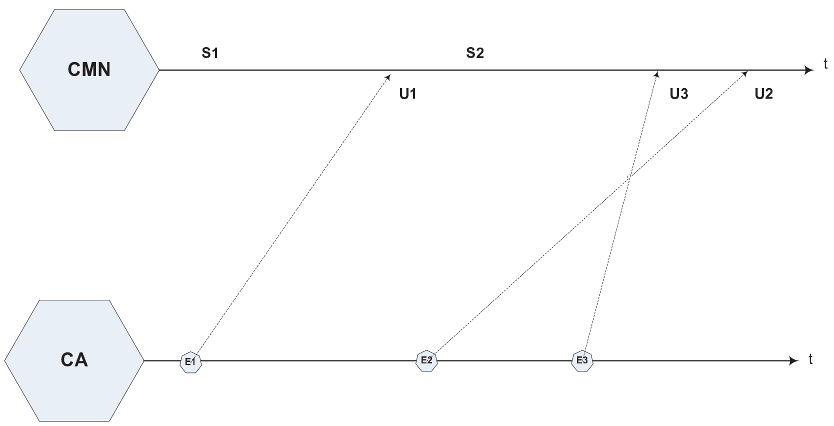

Figure 2.3: Proactive Event Driven Strategy - in the shown example, reception of the update from Event 1 leads to a correct value at the CMN until Event 2 happens at the CA. Update 2 will be filtered out at the CMN by the use of sequence numbers.

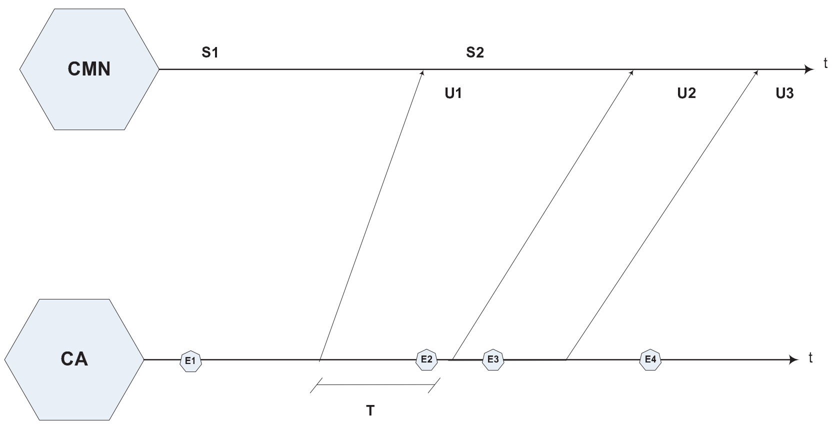

Figure 2.4: Proactive Periodic Update - with a time interval T, an update of the most current value at the context agent is sent to the context manager. Reordered outdated updates are filtered out via sequence numbers.

2.1.2 Quality of Services

With today's networks being shared by converged services such as voice, video and data, end-to-end quality of services are required in order to met the diverse performance requirements from the network. QoS mechanisms play a key role in how packets traverse the network, and how they are being handled at transit on routers. At a router traffic is handled through two key factors, first comes queue management which handles the packet input at the queue, and second the queue scheduler which handles the packet departure from the queue. Using QoS mechanisms at bottleneck locations for example at the aggregation layer is quite critical. The rule of thumb from queuing theory indicates that delay increases when buffer size increases, i.e. the higher the utilization (arrival-rate/service-rate) level is, the higher the delay will be. At access networks utilization is normally quite high, specially in urban areas, with regular utilization reaching 80%, and due to lead time of ISPs, they can reach over 95% in some cases. This means that at some point the queuing buffers are fully loaded and drops and delays will take place. Real-time services can suffer a great loss in quality, since congested queues result in packets arriving out of contract i.e either dropped or arrive too late to be of value. ISPs today configure QoS with the intention of congestion prevention and many of today's real-time services are still being treated as Best Effort (BE) traffic including context sensitive applications. This can be unfair to context sensitive applications, since reliability of such applications depends on low delay, jitter and loss [50]. The fact that such applications reside in the BE class, makes the queue fully loaded at busy hour traffic giving no guarantees for delay, jitter or loss for such applications.

Differentiated Services

Diffserv [37] is a highly scalable QoS solution that runs almost on every router today, including private home routers/access points. Diffserv makes use of the 8-bits Type of Service (ToS) field that is found in the IPv4 header. The first 6 bits in the ToS header are used to define the Differentiated Services Code Point (DSCP) field.

The 6-bit DSCP field indicates how the packets are to be forwarded within the Diff-Serv domain of different network providers. This behavior is called the Per Hop Behavior (PHB) and it is predefined by the network provider. The DiffServ QoS classifies and aggregates the network traffic into different traffic classes, these traffic classes are distinguished from each other through the DSCP field.

The marking and assigning of packets is performed at the network's edge routers based on the network provider's predefined policy. Within the Diffserv domain the network's core routers manage packets according to the value found in the DSCP field. The class selector PHB that can be marked in the DSCP field offers three types of forwarding priorities:

1. Expedited Forwarding (EF): This type of forwarding is characterized by a minimum configurable service rate, independent of the other aggregates within the router, and oriented to low delay and low loss services.

2. Assured Forwarding (AF): This type of forwarding is defined and recommended in for 4 independent classes (AF1, AF2, AF3, AF4), within each of the classes traffic is differentiated into 3 drop precedence categories (low, medium, high). The packets marked with the highest drop precedence are dropped with lower probability than those that are marked with lower drop precedence.

3. Best Effort (BE): This type of forwarding does not provide any performance guarantee and does not define any QoS level.

2.2 Radio over Fiber wireless Networks

Today's mobile 4G networks are required to fulfill several tasks, such as provisioning of true broadband wireless access, and enhanced system capacity in comparison to the older 3G networks. From an ISP's point of view generalized deployment is constrained by Operation and Capital expenditures (OPex and CAPex). 4G networks should be flexible and allow for flexibility and upgradeability in order to enable ISPs to reuse currently deployed legacy networks. Since mobile users demand more and more bandwidth and sophisticated services, ISPs are in search of the perfect balance between three major factors, which can be identified as: the demands for higher bit rates, network capacity and deployment/infrastructure costs [22; 54].

RoF System is the concept of transmitting analog radio signals such as mobile radio signals over an optical fiber link. Thus, a single antenna can therefore receive any and all radio signals carried over a single-fiber cable to a central location where equipment then converts the signals. This is opposed to the traditional way where each protocol type (3G, LTE, WiFi and WiMAX) requires separate equipment at the location of the antenna. There are many forms and architectures associated with RoF wireless transmissions. [22] proposes an architecture introduced by the fiber-optics networks for distributed, extendible heterogeneous radio architecture and service provisioning (FUTON) project. The aim of the FUTON architecture is to use simplified base-stations Remote Antenna Units (RAUs). Those RAUs are transparent multi-frequency transceivers, that receives/-transmits radio signals from the users. From the RAU, the radio signals are forwarded to the Centralized Unit (CU) that is set to serve an area where joint processing of the forwarded radio signals take place. This proposed infrastructure will allow for use of heterogeneous networks, which means that 3G, LTE, WiMAX, WLAN, LTE Advanced and other technologies can all co-exist at the same RAU-location.

RoF physical components are:

1. **Mobile Terminals** (MTs) refer to hand held devices such as mobile phones, smart phones, laptops. It is in general any device that supports one or more of the known wireless technologies i.e. they are wireless devices running 3G,WiMAX,WLAN,..., etc.

2. **The Hybrid Optical-Radio Structure** consists of the RAUs connected to the CU through optical links on an optical fiber. RAUs are the solution to increasingly more complex and expensive Base Stations (BSs). The BSs are simplified to transparent wireless transceivers with optical-electronic converters that sends/transmits wireless signals to and from the CU. The RAUs will be connected to the CUs via fiber, using a tree or other topology, as seen in figure 2.5. This structure allows for lower installation costs than today's wireless systems so the CAPex will be lower. Since there is no expensive processing electronics in the BSs there is also a lower OPex.

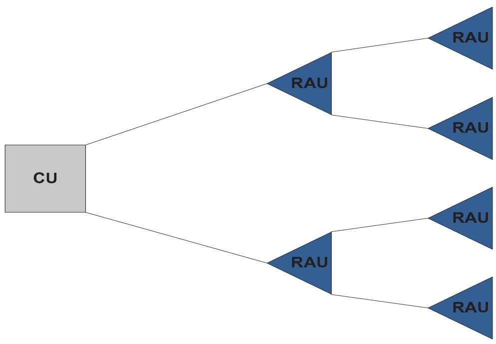

Figure 2.5: CU to RAU tree connection

3. **The Central Units** have the task of processing the radio signals to and from the RAUs. Several fibers can be attached to each CU. Each CU consists of one or more Joint Processing Unit (JPU). Each JPU is responsible for the process of radio signals and the routing of IP packages to and from a limited number of RAUs. Additionally, several CUs can be interconnected, to create even bigger serving areas. A CU and its attached RAUs will provide a geographical area of wireless coverage. This area will be called service area. Depending on the size of the system implemented, several CU's may be connected. The CUs are also able to cooperate for instance with soft handover.

2.2.1 Benefits of RoF Technology

There is a huge benefit of transmitting radio signals over fiber, since optical fiber offers high bandwidth which is up to 100 Terahertz in single-mode fiber. The high optical bandwidth enables high speed signal processing to be implemented. According to [27] optical fiber is a good method for distribution of wireless data transmissions because it has low loss transmission ranging between 0.3 dB/km at 1550 nm and 0.5dB/km at 1310 nm wavelengths. Since fiber optics are the medium of transmission this allows RoF to make use of Dense Wave Division Multiplexing (DWDM). This makes use of multiple wavelengths which enhances the use of fiber topology and bandwidth.

Part II

Context Management Optimization

Chapter 3

Service Degradation in Context Management Frameworks

3.1 Introduction

As mentioned before in chapter 2, an important feature in today's applications and services would be its ability to sense and react to environment changes such as location changes, weather changes or even network changes. An example on such environment-aware i.e. context-aware service would be application migration from one network device to another. But context information is not only utilized by applications so they can adapt to their environments, for example in [44] triggers for service migration are based on the contextual situation of the user and the involved devices. Via the migration framework running applications may shift between devices, e.g. switch a video-stream from the mobile phone to the large TV screen. A Context Management Framework (CMF) as described in [59] offers the service of providing the information needed to perform context aware actions. This study was done during work on the OPEN project therefore the CMF in use is the one described in [53; 59]. Service migration represents a new and interesting use-case to investigate QoC. A CMF's main goal is to manage information distributed in the network such as collecting, storing, processing and delivering relevant information from different sources of context to functions in need of information, which can be either directly measurable or inferred/processed information. The performance of such framework is therefore closely linked to the performance of context aware applications. For example, to carry out timely correct triggers to migrate services, the context information is required to be timely correct as well. The dynamics of context information in combination with any delays leads to potential use of mismatching information, giving grounds to incorrect execution of context aware applications and triggers.

In this thesis, it is assumed that the delay of information occurs due to communication between a context consuming node and a context providing node. In order to obtain the missmatch probability (mmPr) [6] the delay distribution should be obtained.

European projects like MAGNET Beyond [57], SPICE, [20] or E-SENSE, [3], and others outside Europe, have been researching and developing concepts for context management for some time, whereas reliability indicators of context information just recently caught the attention e.g. in the project SENSEI, [19]. Using time as an indicator for the reliability of the information provided to the application is often used, e.g. in [1; 25; 38], where the notion up-to-dateness or freshness is used, which may be useful in time-synchronized networks, however without carrying much of the dynamics of the information or the access method, which has been shown to be a key problem for reliable context information in e.g. [6]. Therefore, the performance investigations focuse on the reliability of context information. However, in that investigation there is a need to understand the realistic properties of context management, which is why the experiments in this chapter are carried out.

Figure 3.1 gives an overview of the OPEN CMF framework, where two or more Context Agents (CAs) that are capable of exchanging information regarding what context information is available on which node. One of the nodes acts as an anchor point, the Context Management Node (CMN), which contains an overview of all other CAs.

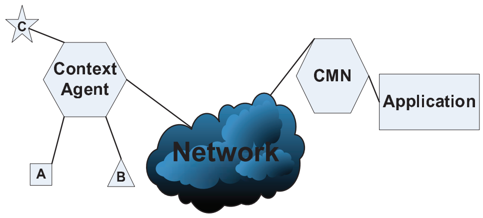

Figure 3.1: CMF Framework

In figure 3.1 an application is gaining access to the remote, dynamic context elements A, B and C over the network, in order for it to perform a context aware reaction upon a change in the contextual situation. Each context agent which has access to context information, has a so-called retriever installed, which ensures the local data abstraction and interfacing between the local source of information and the context management system. the communication between CMN and CA is based on xml-rpc, which is TCP based protocol.

3.2 Degradation Parameters of Context Management Systems

Context information is collected via a reactive access strategy as show in figure 3.2. The figure gives an abstracted view of the communication between the two Context entities in order for the application to obtain the required context information. The defined performance parameters will in the following relate to this figure.

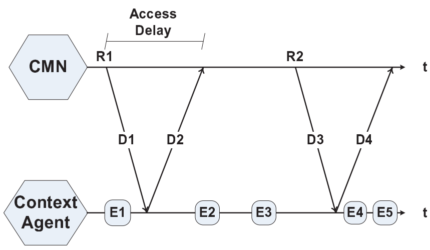

Figure 3.2: Access Delay

1) Traffic Volume: Traffic volume is here defined as the amount of traffic being transmitted in the course of context management as the payload of IP packets. Traffic volume therefore affects the service degradation since the amount of bytes/second generated depends on the request process, the size of the request and response messages. The focus is therefore to measure the sizes of the request and response messages only, whereas the actual network overhead can easily be calculated.

2) Access Delay: Figure 3.2 shows that the CMN sends a request (D1) to the CA to collect context info. The CA then sends response (D2) to the CMN with the requested context information. The access delay is then defined as the time at which a request is sent till a response is received, and is expressed as follows:

$$t_{access} = D_1 + D_2$$

Since, later on mainly the response time is of interest, we assume symmetrical delays.

3) Mismatch probability: The probability of using mismatching information is called mismatch probability (mmPr), [6]. From [6] the mmPr equation for reactive strategy is expressed as follows:

$$mmPr_{full} = \frac{1}{\mathbb{E}(E)} \int_0^\infty F_D F_E dt \tag{3.1}$$

with E as the event process, F_D the forward access delay cdf of D2, F_E the forward event time cdf (the over-line indicates $1 - F$). For the context management framework it is reasonable to assume that meta information regarding the dynamics of the context information is available as this may be a part of the context description itself. Hence, it is assumed that every response received from the context provider, contains not only the value of the information, but also complete meta information about the distribution, F_E. The delay distribution relates mainly to the network, congestion level and QoS will be a central part of the evaluation and influencing factor on the mismatch probability.

3.2.1 Test Bed Setup

To evaluate the CMF degradation effect a test scenario is shown in figure 3.3

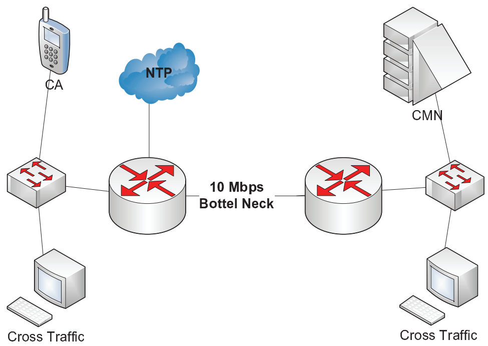

Figure 3.3: Test Bed Setup

The CA is seen on the left side of the network and collects some context info such as location, noise or temperature. The CMN is located on the right side of the network. The CA and CMN are connected to two Cisco 3950 switches each switch is then connected to a Cisco 3620 Router finally the two routers are connected to each other with a 10 Mbps bottle neck link. On each side of the network there are two sources that generate cross traffic. The traffic generated is constant bit rate traffic in order to keep the network constantly busy, the traffic was generated by Packeth traffic generator on a linux machine [52]. A series of tests with varying network utilization can now take place. One set of tests with BE traffic from the CMN and CA side, another is a DiffServ QoS based test where packets from the CMN and CA are marked with a higher EF priority. For each test 50 requests from the CMN are carried out i.e. 50 requests/responses. 50 samples were considered enough since it is a sample size that allows us to generate confidence intervals and generate CDFs. The test scenario will introduce different link utilization between 80% to 97% utilization.

3.3 Result Evaluation Metrics

The following parameters are used as methods for evaluating the results generated from the test scenario.

3.3.1 Network Traffic

Using Wireshark [51] statistics of the data exchange are evaluated. The TCP streams i.e flows [15] with the same source and destination IP, port numbers and transmission protocol, between the Context Agent and the CMN are filtered out for each direction of communication, and statistics about data exchange and the overhead information, including TCP overhead are extracted. The statistics are normalized to the 50 collected samples for each test, so the results are mean values for a single request or response, while in fact according to network utilization sample retransmission and duplicate segments may occur. To evaluate the impact of context state sizes, a reconfigurable retriever is used which allows setting the state of the accessed information. The impact of state size on the traffic is evaluated for the range 10B - 10KB state information i.e. each bandwidth utilization test will be carried out on state sizes varying between 10B - 10KB.

3.3.2 Delay

For retrieving context information, time is essential and thus the delay is examined to see the affect on the access times. For each of the samples time stamps are logged and used to examine the delay. The system time is logged just before making a request call to the CA, and when this returns with a value, the time is logged again. The time difference is used as an indication of the access delay time for the reactive strategy. The impact of the state size on the delay is evaluated for the range 10B - 10KB of state size for both scenarios, BE and QoS.

3.3.3 mmPr

For calculating the mmPr equation 3.1 for the reactive strategy has been used. The equation makes use of the CDF of the empirical delay distribution which has been measured from the test bed. The equation also assumes a Poisson process for event change. The reason for using a Poisson process is that the thesis does not analyze specific context elements, but rather stays generic in respect to the type of context element, therefore it is possible to reduce computational effort of the algorithms and the parameter space stays small. The results are calculated both for BE and QoS.

3.4 Results

The following results are based on the test bed described above.

3.4.1 Network Traffic

The results from the analyzed network traffic for each test are presented in the following tables. Table 3.1 shows the analyzed network traffic between the CMN and CA, where the CMN and CA packets are marked with BE i.e. there is no prioritization. Table 3.2 shows the analyzed network traffic between the CMN and CA, where the CMN and CA packets are marked with EF QoS i.e. the CA and CMN packets are prioritized over other packets and a bandwidth reservation of 10% will be allocated in case of congestion. The notation (1) and (2) are used such as (1) represents traffic in the direction from CA to CMN and (2) represents traffic in the direction from the CMN to CA. Data is the actual payload data and Over-head is TCP headers, TCP retransmission, etc.

Table 3.1: Network Traffic in Bytes for Best Effort Test at 97% utilization

State Size	Data (1)	Data (2)	Over-head (1)	Over-head (2)	Total
10B	1009	1238	304	442	2993
1000B	2066	1238	297	487	4088
2000B	3132	1238	315	530	5215
5000B	6264	1238	247	602	8351
10000B	11590	1238	257	844	13929
Avg.	4812	1238	284	581	6915.2

Table 3.2: Network Traffic in Bytes for EF QoS Test 97% utilization

State Size	Data (1)	Data (2)	Over-head (1)	Over-head (2)	Total
10B	1009	1238	337	292	2876
1000B	2066	1238	337	351	3992
2000B	3132	1238	338	411	5119
5000B	6264	1238	272	470	8244
10000B	11590	1238	272	734	13773
Avg.	4812	1238	284	581	6800.8

As it can be seen from the data in the tables, over head for smaller data-state sizes is quite large. Further analysis of the traffic shows that the cause of increased traffic overhead as a function of increased state size is due to the three main factors of TCP retransmission, lost TCP segments and duplicate TCP acks, this is due to the nature of the TCP protocol. Figure 3.4 shows the different factors of increased overhead for five different state sizes varying from 10 - 10000 Bytes.

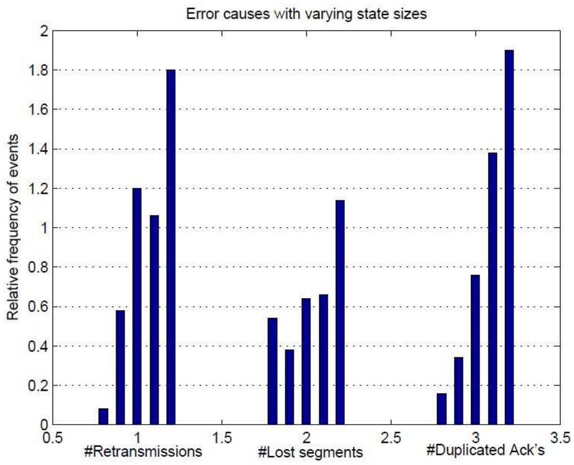

Figure 3.4: Factors of Increased Overhead

3.4.2 Mean access delays due to Utilization

Figure 3.5 and figure 3.6 shows the mean access delay time for the best effort traffic and EF traffic respectively. The Rbar indicates the mean time between requests is set to 5 seconds and three different state sizes 10 Bytes, 2KB and 10KB.

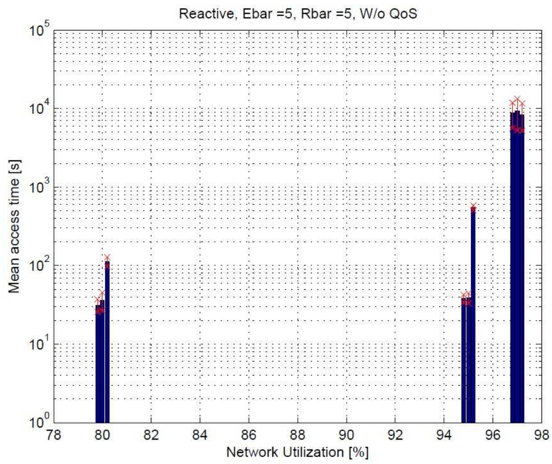

Figure 3.5: Best Effort Mean Access Times with varying Utilization for 10 Bytes, 2KB and 10KB state sizes, respectively

As it can be seen from both figures, the BE traffic access delay is higher for utilization over 95%. It can also bee seen that the sate size has a direct impact over the access delay. BE traffic however gives a degree of control over access delay even under utilization over 95%.

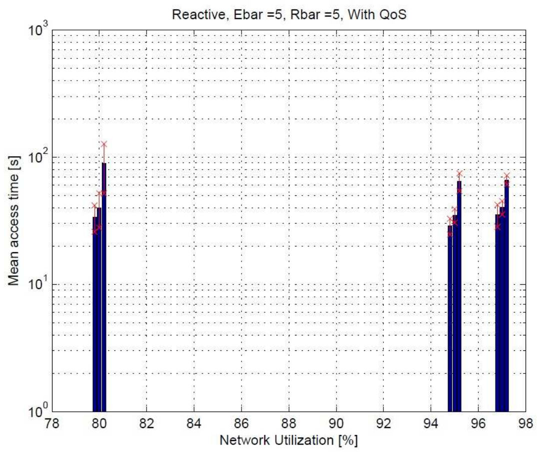

Figure 3.6: EF Mean Access Times with varying Utilization for 10 Bytes, 2KB and 10KB state sizes, respectively

3.4.3 Access Time distributions

The Cumulative Distribution Function (CDF) for access times with best effort under different network utilization is shown in figure 3.7 and figure 3.8.

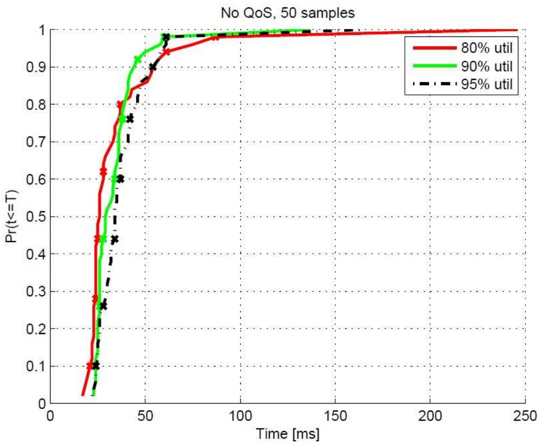

Figure 3.7: Best Effort Access Times CDF, 80%, 90% - 95% util

Figure 3.9 shows the CDF for access times with EF under different network utilization.

As it can be seen from the figures there is very little difference in the access times considering the network utilization interval 80%-90% utilization. This is valid as long as the active QoS class is not saturated i.e. if the CMN should need to exchange large amount of information in this case exceeding the 10% reserved bandwidth, it may need to classify the CMN and CA traffic with more than one class in order to distribute the load.

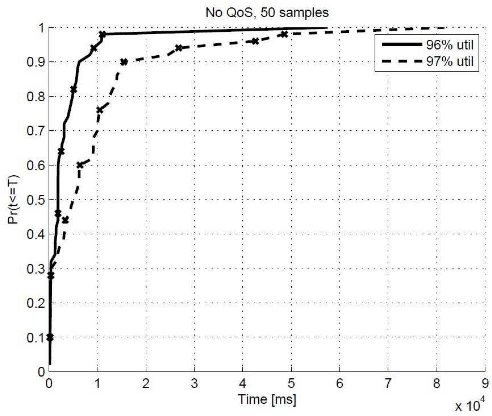

Figure 3.8: Best Effort Access Times CDF, 96% - 97% util

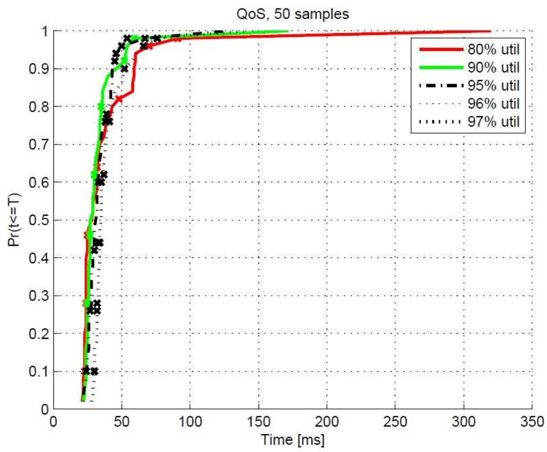

Figure 3.9: Expedited Forwarding Access Times CDF, 80%, 90% - 95% util

3.4.4 Mismatch probability

Recall that the mismatch probability depends on the CDF's of both the event process and delay process as shown in the reactive mmPr equation. For the event process, this may be given as a prior knowledge from context providers, but for the network this is not possible, due to time variation in the network load. From the measurement campaign, we can now plot the resulting mismatch probability with confidence bounds. Figure 3.10 shows the result for two different state sizes, 2KB and 10KB, respectively, how the information dynamics influence the mismatch probability given the delay distributions shown in figure 3.5 and figure 3.6. Thus, the two utilization factors representing the cases using normal and prioritized traffic via QoS marking.

From figure 3.10 it is quite obvious that prioritizing the context management traffic higher than other traffic positively impacts the reliability of the information being accessed. Although not easy to see in figure 3.10, the impact of state size is most significant in low congested scenarios. The increase in mismatch probability for the case, where the event rate is 2 events per second is from around 0.03 to 0.1 for an increase of 8KB in state

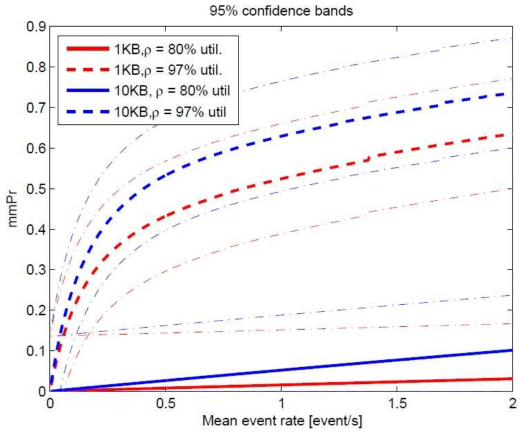

Figure 3.10: mmPr plots of two dynamic information types with varying state size under different network conditions. MmPr's are plotted with 80% and 97% util, the solid lines represent EF traffic and the dashed line represent BE traffic

size, where in the highly congested scenario the increase is from 0.63 to 0.74 for the same information dynamics.

3.5 Context Management Service Improvement

Here an illustration is given on how to use such an investigation to improve QoC. Assume that it is possible to reduce the state size in a response message, either by a query language that efficiently reduces the size, or by any compressing techniques, then figure 3.10 implies that mismatch probability may be reduced simply by focusing on reducing transported state size. Figure 3.11 shows resulting mismatch probability for the 80% utilized network scenario with 1KB, 5KB and 10KB state size, respectively. Usually, query processing or compression techniques also takes additional time, which on top of the network delay poses a challenge to the reliability as we showed in figure 3.10. We now add an additional processing component in the time delay, such that the access delay equation which we for simplicity assume deterministic becomes:

$$t_{access} = t_{D1} + t_{processing} + t_{D2}$$

Figure 3.11 now shows the mismatch probability as a function of this added processing time. If the compressing algorithm or query processing, manages to reduce a 10KB data element into a 5KB data element, then as long it takes less than 30 msec, it will have a positive impact on the reliability. If it takes roughly 30msec, reliability will not be impacted, but if it takes longer, then there will be a loss in reliability at the gain of less traffic being generated. If the compression algorithm is able to reduce data size even further, to e.g. 1KB, then in the case shown in figure 3.11 it will have an additional 10msec to do so in order to gain the benefits of improved reliability and decreased network overhead as described. If there are strict requirements to the reliability, e.g. 0.4, we can see that an upper cap of processing time can be found in similar way. In the case of 12 for a 10KB state, it will give the context provider 200 msec to process the data, before it impacts the reliability. It is important to be aware that figure 3.11 is restrictive in that the processing time is deterministic, access delay as mentioned is based upon measurements with the before-mentioned confidence bands, and an information type which has exponentially distributed time interval between events. For the case where the network is congested, i.e. here 97%, the mismatch probability is much higher than the previous case. Notice, the 5KB and 10KB state size practically leads to the same mismatch probability, hence there is no need to do much additional processing. The reliability gain is simply lost in the higher network delay. However, spending time effort to reduce the data element to a 1KB size may be worth pursuing. Adding less than 270 msec of processing delay to reduce the state size from 10KB to 1KB will reduce the mismatch probability, hence more time may be spend.

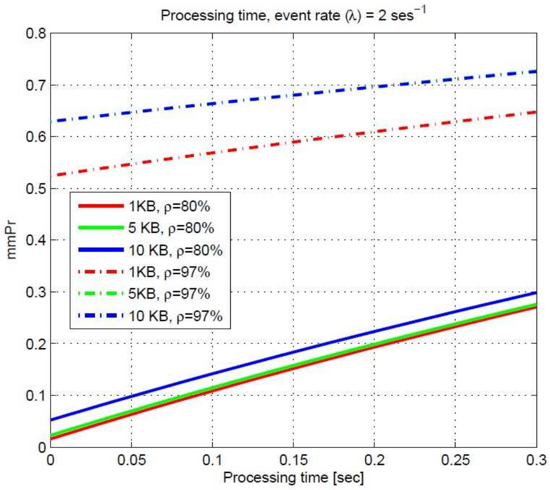

Figure 3.11: Impact of processing time on the reliability of the accessed information

3.6 Summary

This chapter investigated improvement of the reliability of context quality by examining the degradation parameters which are traffic volume, access delay and mmPr. The result for traffic volume showed that overhead in small data sizes was rather large and that its main cause was the number of TCP retransmission, lost TCP segments and duplicate TCP acks. It has also shown the impact large traffic volume has and that taking time for compressing context information data can result in higher reliability for context information. The chapter also showed the impact of access delay and that by applying QoS in higher utilization a degree of control is achieved which minimizes the access delay for context information. Calculations for mmPr have been made using empirical delay distribution measured from the test bed and ws plotted for different state sizes which also showed that the mmPr improved by the combination of reduced state sizes and addition of QoS. Chapter 4 investigates further on how to improve the reliability of context information by minimizing mmPr.

Chapter 4

Network Aware Dynamic Context Subscription Management

4.1 Introduction

This chapter of the thesis introduces a new context management communication framework. The goal of the developed framework and algorithms is that the context manager is able to select and configure the access strategy by which it interacts with the various context agents to provide the information in such way that the reliability of the information is effectively increased.

The decision on which strategy to take is not trivial as it involves several stochastic processes and parameters that may be adjusted, and as each different strategy puts different loads to the network, and any decision effectively feeds back via an increased network delay to the mismatch probability and hereby may render decisions not optimal.

The **objective of the algorithm is therefore to decide upon one of the three access mechanisms that should be used, as well as parameter settings, while considering that the context access traffic also affects the network**; so impact on network delay and loss caused by the higher load is taken into account.

4.1.1 Interactions in the Context Management Framework

The interactions between the different entities in the assumed context management framework are illustrated in figure 4.1. The different entities in the system are as follows:

Context Agents: Collect local context information, and provide remote access to context information. Context Agents register at the CMN at start-up.

Context Management Node: maintains a repository of a) available context agents, b) available context information in the network, and has possibility to control the interactions for context access between the Context Agents.

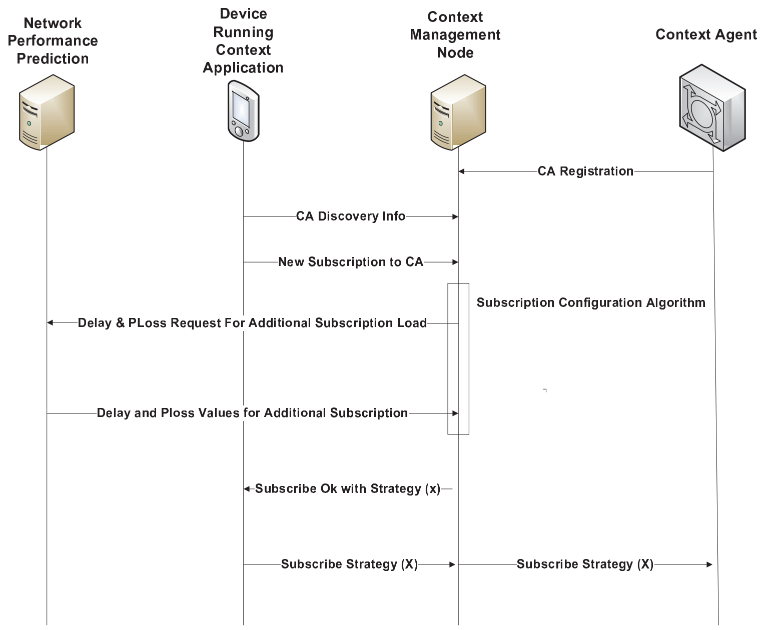

Figure 4.1: Interactions with the Context Management Node during establishment of a context subscription.

Network Performance Prediction: a component that allows to obtain information of the network performance and provide information on how potential additional traffic changes network performance.

Context-dependent application: This application uses dynamic context information for real-time decisions. The correctness of these decisions affects the resulting application quality.

Figure 4.1 shows the basic message flow that occurs during the registration and subscription phases. The basic principles are:

All CAs register at the CMN. They thereby provide information about the type of context elements they can offer and the relevant parameters. Particularly, they provide information on the temporal dynamics of the context element, which in our simplified setting are exponentially distributed inter-event times with a rate λ_i of the context-element i. Furthermore, also information on the size of update messages of the context element is provided.

When an application or middleware function on a node is interested in a certain context element, it sends a subscription to the CMN. Along with other query related specifications, the application also sends its expected request rate and potentially also bounds on mismatch probability of the context element or on access delays.

The CMN then evaluates which context access procedure with what parameter setting is best for this new subscription. In order to do so, it will interact with the Network Performance Prediction function in the network. Having determined the most suitable configuration, the CMN informs the subscriber and the CA about this configuration and starts receiving context updates (in the proactive strategies) respectively relaying requests for context elements in the reactive strategies. All interaction between subscriber and CA is performed via the CMN in order to allow abstraction and efficient processing.

The CMN keeps track of all active subscriptions. Hence, as a variant, when a new subscription comes in, the CMN may not only determine an optimal access configuration for this newly arriving subscription, but it may in addition identify that it is beneficial to change some of the configurations of the already active subscriptions. Both cases will be specified and evaluated further below.

4.2 Analytic Model Representation

4.2.1 Model Overview

In order to calculate the impact of different access strategies on mismatch probability, different model parts need to be combined as illustrated in figure 4.2: Based on the parameters of the context elements accessed (change event rate λ and update size U) in the upper left and based on the request rate, μ, provided within the subscription request, the additionally created network traffic can be calculated in the upper left block. These are straightforward calculations, outlined further below for the individual strategies. The additional traffic is input to the network model, which is used to calculate network performance parameters for the paths between subscriber and context provider. In the example illustrated these parameters are mean delays and packet loss probabilities, but more advanced network models (beyond mean value calculations) are possible.

The delay and loss values are then used together with the context and subscription parameters to calculate the resulting mismatch probability per context element for different access configurations (lower left block). Finally, the calculated individual mismatch values are combined in a single metric called GmmPr as introduced further below and the model part outputs the minimum GmmPr and the corresponding strategy.

4.2.2 Utilization Increase Calculations

When a new subscription is executed by the CMN, additional network traffic is generated. The required additional throughput, η, is calculated in the following for the three different access strategies. U is thereby the size of the different messages (requests or updates), μ is the request rate, λ is the rate of the change events of the context information, and τ is

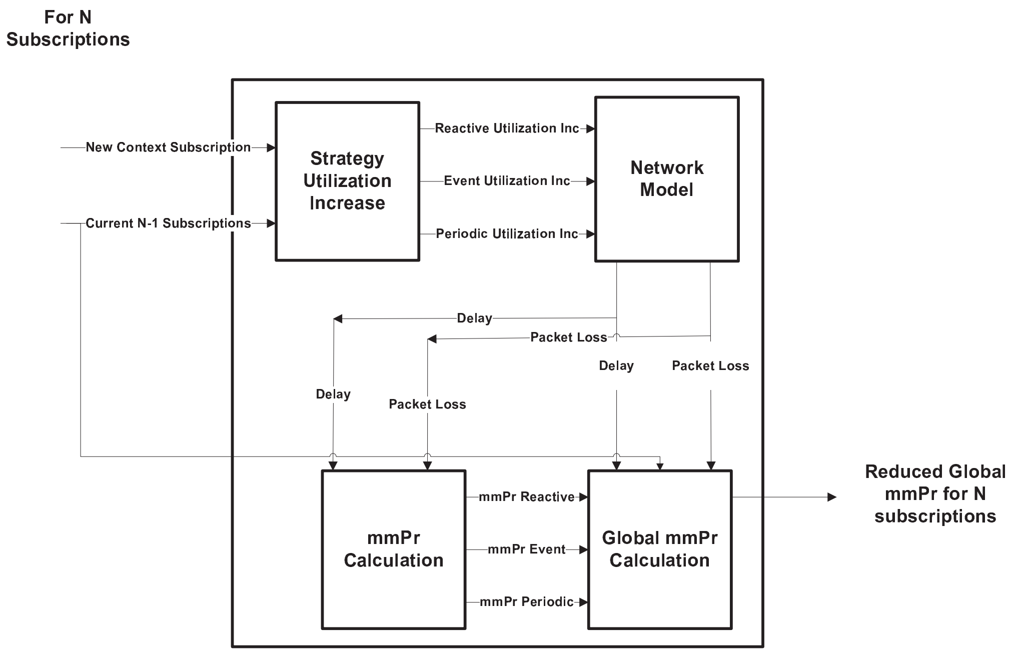

Figure 4.2: Model overview of internal parts of the algorithm that is being executed at the Context Management Node.

the rate of the periodic updates. Messages that are dropped in the network are included in the throughput calculations.

Reactive Strategy The context access traffic generated by the reactive strategy is depending on the request size, the response size and the rate of requests (assuming request arrivals with rate μ). The generated traffic is:

$$\eta_{rea} = (\mathbb{E}(U)_{rqs} + \mathbb{E}(U)_{rsp}) \cdot \mu.$$

Proactive Event Driven Update For the proactive event driven strategy the traffic generated depends entirely on the change rate of the information and the size of the updates:

$$\eta_{evn} = \mathbb{E}(U_{update}) \cdot \lambda.$$

Proactive Periodic Update For proactive periodic update strategy the context access traffic generated is directly depending on the update rate and the size of the updates:

$$\eta_{per} = \mathbb{E}(U_{update}) \cdot \tau$$

4.2.3 Network Model

The network model is modeling the bottle neck behavior of the network and, summarizes the delays and processing time in the network from End-to-End. The model is used to translate utilization into End-to-End delay and packet loss probability. There are two different network models used in this thesis: (1) a M/M/1/K model, which has the advantage of a small parameter space, while still allowing to analyze trade-offs between delay and loss when varying the buffer-size K. (2) A more complex network model consisting of the convolution of a transmission delay with a queuing delay obtained from a bursty MMPP/M/1/K model. The MMPP model has been widely used to describe wireless network behavior [69] [26].

The mean delay and packet drop probabilities at utilization ρ of the M/M/1/K queue are found in any standard queuing theory book [2]:

$$PacketLoss = \frac{((1-\rho)\rho^K)}{(1-\rho^{K+1})}$$

$$QueueLength = \frac{\rho}{(1-\rho)} - \frac{(K+1)(\rho^{K+1})}{(1-\rho^{K+1})}$$

$$Delay = \frac{QueueLength}{\Lambda(1-P_{loss})}$$

Λ is hereby the overall offered load to the queue (in packets per time unit), K the size of the buffer at the bottleneck (measured in packets).

The analysis of the algorithms for context subscription handling also uses a more complex network model which is formed from the convolution of an end-to-end forwarding delay and a queuing delay. The forwarding delay is modeled by an exponential delay. The queuing delay is assuming bursty cross-traffic with exponential ON-OFF pattern superimposed to Poisson context access traffic. The cross-traffic represents a base-load, while the Poisson rate of the context traffic is a function of the number of active subscriptions, their context parameters and the selected access strategies, see Section 4.2.2. This traffic model can be represented by a 2-state Markov-modulated Poisson Process (MMPP) and the performance metrics, mean delay and packet loss probability, of the corresponding MMPP/M/1/K queue can be calculated with matrix-analytic methods [42]. In order to keep the complexity of the mmPr calculations low, the queuing delay is then assumed to be exponentially distributed with a mean that is obtained from the numerical solution of the MMPP/M/1/K queue. That way, the mismatch probability calculations can be performed based on Erlangian-2 type distributions for network delay. Such distribution is a specific case of a general phase-type or matrix-exponential distribution, represented in the notation introduced by Lipsky [30] by a vector/matrix pair $< \mathbf{p}, B >$, such that its cumulative distribution function and density function can be described as

$$F(t) = 1 - \mathbf{p}exp(-t\mathbf{B})\varepsilon', \qquad f(t) = \mathbf{p}\mathbf{B}exp(-t\mathbf{B})\varepsilon'.$$

Here $\mathbf{p}$ is an entry row-vector to the delay model, $\mathbf{B}$ is the generator matrix for the delay model and ε' is a column vector of ones. In the simplest case of an exponentially distribution, $\mathbf{p} = 1$ and $\mathbf{B} = \nu$ (the rate of the process), which later simplifies the used mismatch probability equations. The used Erlangian-type model for the network delay has a Matrix exponential representation:

$$\mathbf{p} = [1 \quad 0], \qquad \mathbf{B} = \begin{bmatrix} \nu_t & -\nu_t \\ 0 & \nu_q \end{bmatrix}$$

where ν_t is the transmission delay rate, and ν_q the queuing delay rate calculated from the queuing model as described above. With this, it is fairly simple to independently adjust the two types of delay occurring in the system being investigated. More complicated delay models can easily be represented by more general matrix-exponential distributions.

The assumption of a single bottle neck is considered to be one of the limitations of the network models, however, it is a simple way to introduce information about network parameters such as delay and packet loss. The exponential assumption is also considered a limitation of the network model, however, under congestion which is the desired study case, an exponential assumption is valid due to retransmissions and drops.

4.2.4 Mismatch Probability Calculations

The equations of [6] that describe the mismatch probabilities for the different access strategies are extended. The main extensions are to include packet loss, which are relevant when UDP transmissions are used for the context access. In case of context access traffic via TCP, packet losses lead to retransmissions, which can be mimicked by an adjusted message delay distribution. We, however, assume UDP context traffic for which losses are relevant to be included in the model. The inter-arrival event process is considered to be a phase-type distribution, represented for different strategies using the Kronecker-product and Kronecker-sum, which are considered to be a good representation according to [47].

Reactive Strategy: In order to receive a correct context information message, the reactive strategy requires successful transmission of two messages; otherwise the CMN does not receive a response which we assume to lead to a mismatch. Under the condition of successful two-way communication, i.e. the request and response are not lost, it is only the delay of the response that may lead to mismatches. Extending the approach of [6] while assuming both the inter-event time, $< \mathbf{p}_E, \mathbf{B}_E >$, and the network delays, $< \mathbf{p}_D, \mathbf{B}_D >$, are Matrix-exponential distributed renewal processes, we can express the mismatch probability as follows:

$$mmPr_{rea,ME} = 1-(1-p_{loss})^2\frac{(\mathbf{p}_E\mathbf{B}_E^{-1})\otimes(\mathbf{p}_D\mathbf{B}_D)}{\mathbf{p}_E\mathbf{B}_E^{-1}\varepsilon'_E}\mathbf{p}_E\mathbf{B}_E^{-1}\varepsilon'_E\left[\mathbf{B}_E\oplus\mathbf{B}_D\right]^{-1}\varepsilon'_{dim(B_E)\cdot dim(B_D)}$$

The symbols $\otimes$ and $\oplus$ represent the Kronecker-product and Kronecker-sum, see [8]. Note also that for reactive strategy the extension includes

$$(1-p_{loss})^2$$

since it is the probability both messages are received correctly i.e probability that request message and respond message are received correctly.

When both the network delay and the time between context change events are exponentially distributed with rates ν and λ, this equation simplifies to:

$$mmPr_{rea,exp} = 1-(1-p_{loss})^2\frac{\nu}{\lambda+\nu}.$$

Note that the mmPr increases with increasing context change rate λ and with increasing network delays (which correspond to decreasing rate, ν, of the delay distribution). For recurrent context information types, the equations from [6] can be analogously extended. In addition to the reactive strategy, it would also be possible to cache data that has been received to be used as response for subsequent request within some finite time interval. This mechanism was evaluated in [60], however, to keep the configuration space low, it was chosen not to allow caching techniques to be applied.

Proactive Event Driven Update: The mismatch probability without packet loss for this approach turns out to be exactly the same as for the reactive strategy, see [6]; in case of potential packet loss, a information match now requires that the last update message has not been lost. We assume here that information updates are complete, i.e. each update completely replaces old information stored at the context manager. An alternative update approach does exist, where updates are incremental, but with a non-zero packet loss probability, this type of update is not useful since one dropped message leads to subsequent mismatches. For the case of Matrix-exponential inter-event times and network delays, the results from [6] are hence extended to

$$mmPr_{rea,ME} = 1-(1-p_{loss})\frac{(\mathbf{p}_E\mathbf{B}_E^{-1})\otimes(\mathbf{p}_D\mathbf{B}_D)}{\mathbf{p}_E\mathbf{B}_E^{-1}\varepsilon'_E}\left[\mathbf{B}_E\oplus\mathbf{B}_D\right]^{-1}\varepsilon'_{dim(B_E)\cdot dim(B_D)},$$

Note also that for proactive event driven update strategy the extension includes

$$(1-p_{loss})$$

since it is the probability the update message is received correctly.

which reduces in case of exponential distributions to

$$mmPr_{per,exp} = 1 - (1 - p_{loss})\frac{\nu}{\lambda + \nu}\,.$$

Proactive Periodic Update: The mmPr model of the periodic update case assumes an exponentially distributed update interval with rate τ in order to simplify the mathematical calculations. If context providing processes are lowly prioritized by the operating system, stochastic fluctuations in the update period are not rare, even if timers are used. The basic approach and the resulting equation to calculate the mismatch probability is the same as in [6], and for Matrix Exponential distribution can be expressed as

$$mmPr_{per,ME} = \frac{e^{\tau\bar{D}}}{\bar{E}} \int_0^\infty e^{-\tau t} exp\left[-\tau \mathbf{p}_D \mathbf{B}_D^{-1} exp(-\mathbf{B}_D t)\varepsilon'_D\right] \cdot \left[\mathbf{p}_E exp(-\mathbf{B}_E t)\varepsilon'_E\right] dt,$$

which can be simplified in case of exponentially distributed processes

$$mmPr_{per,exp} = \phi e^{\psi} \frac{\Gamma(\phi+\psi)}{\psi^{\phi+\psi}} F_{\Gamma(\phi+\psi,\psi)}(1).$$

The latter equation has the advantage that it is integral free, but is limited to only the exponentially distributed case. For the two-stage delays, numerical integration is required for solving the mmPr.

Including packet loss is done by considering the update process as a thinned Poisson process with thinning probability $(1 - p_{loss})$, so that the parameter ψ now becomes $\psi = \tau(1 - p_{loss})/\nu$ (where τ is the update rate, and ν is the delay rate). The other parameter remains as $\phi = \lambda/\nu$ (ratio of event and delay rates). $F_{\Gamma(a,b)}$ is the cdf of a gamma distribution with parameters a and b.

Rejection of Context Requests: As context subscription traffic in any of the above three configurations also influences network delays and loss probabilities, it will increase the mmPr for other subscriptions. Depending on the optimization target, it can hence be beneficial to reject context subscriptions to avoid this negative impact on other ongoing subscriptions. In case of rejection of a subscription, no additional network traffic is generated, but the mismatch probability for the context accesses of this rejected subscriptions are $mmPr = 1$.

4.2.5 Global Mismatch Probability

When there are N context subscriptions, treating all subscriptions as a set of individual subscriptions is not very efficient, since it does not provide a good estimation of the overall CMF performance and thus a more practical measure of the QoC is required. A new metric is therefore required in order to give a measure of quality and performance of the CMF, that is why we define a single metric referred to as the global mmPr (GmmPr). For simplicity in this work we use the average mmPr of the N sources, but for example a priority weighted average could also be applied, and therefore

$$GmmPr = \sum_{i=1}^{N} \frac{mmPr_i}{N}.$$

It is assumed here that all subscriptions are treated equal, i.e subscriptions are not prioritized.

4.3 Optimization of Dynamic Context Subscriptions

The modelling framework of the previous section is now utilized for configuration selection of context subscriptions within the context management framework introduced in section 2.1.1.

4.3.1 Algorithm descriptions

In the following, we present two algorithms that run at the CMN. The first algorithm provides the base case, when a subscription is initiated by a context-dependent application or device middleware, the CMN determines the optimal configuration for this newly starting subscription, while taking into account the network conditions and also the impact of the new subscription traffic on the already ongoing context subscriptions. The second algorithm includes a reconfiguration of already active context subscriptions and as such is an extension of the first one. Although we do not consider this later in the evaluation, the second algorithm could also be triggered by changes in the network resources, e.g. by increase or decrease of the other traffic in the network, or by reduction of available bandwidth/change of delays in wireless communication settings.

Algorithm 1: Configuration of new subscriptions The first algorithm is executed by the CMN upon the reception of a new subscription. As our intention here is to evaluate the benefit from an intelligent configuration choice, the algorithm is not optimized for efficiency, but rather implements a brute-force search over the possible configuration set – taking parameterized strategies (in our example the periodic strategy with update rate τ) via discretized search spaces into account.

Algorithm 2: Reconfigure all subscriptions While the previous algorithm only considers the newly incoming subscription, it may also be worth to adapt the way already

Algorithm 1 New Subscription Configuration Algorithm

```
 function determine_configuration_of_new_subscription
 Input: Current network load L and available network (bottleneck) resources R
               in the path context agent - CMN - subscriber.
        Request Rate mu_i of new subscription.

Available Information (persistently maintained at CMN)
           From registration: Context-value change rate lambda_i, and update size u_i of
           context element of new subscription.
           From previous subscriptions: Parameters and configurations of
                                                                  currently active subscriptions.

Output: access strategy for new subscription
            parameterizations of the access strategy
            (here: rate of proactive periodic strategy).
            resulting new GmmPr best_mmPr.

Begin

best_mmPr=1.0;
best_strategy=not_defined;
For all access strategies in (Reject, reactive, proactive event-driven,
                  proactive periodic_rate1, ... proactive periodic_rateK);
      calculate additional network load L_inc created by NEW subscription
                                 when using this access strategy;
      Call network model to calculate network performance metrics (D, p_l)
                                                          at load L+L_inc for resources R;
      Calculate GmmPr for these network performance metrics and
                                               existing plus new subscriptions;
      If calculated GmmPr < best_mmPr;
               Set currently considered access strategy and parameters as best one;
               Update best_mmPr;
     end-if
end for-loop;

end function;
```

existing subscriptions are executed. As the CMN keeps track of all subscriptions, their access strategies and their parameters, it can use this information to optimize the target metric further. As we are only interested in a potential benefit of such reconfiguration, we also in this case do not implement very sophisticated efficient search approaches, but rather an algorithm that first determines the optimal configuration of the new subscription, and then iterates over all existing subscriptions to check if an improvement can be obtained by changing the configuration. This obviously does not guarantee convergence to the true optimum, as it locally optimizes each individual subscription. Nevertheless, the algorithm is useful in the evaluation part to investigate benefits from reconfigurations.

4.3.2 Evaluation Methodology

To evaluate the algorithms' performance a series of simulations were carried out. These simulations will compare the intelligent mmPr-based configuration optimization algorithms against strategies that assign a constant default access strategy to new subscriptions. The simulation considers the arrival of N context access requests at the CMN; the request parameters as well as the context parameters are thereby stochastically varying (using uniform distributions). All parameters and their values in the simulation are listed in Table 4.1. The simulation evaluation assumes that the parameters of the specifically addressed context elements (change rate, update size) and the resulting individual request rate are known by the context agent respectively the subscribing application and hence

Algorithm 2 Subscription Reconfiguration Algorithm

```
function iterate_over_existing_Subscription
 Input: Current network load L and available network (bottleneck)
          Number of repetitions, rep, for iterative search.

Available Information (persistently maintained at CMN)
          From previous subscriptions: Parameters and configurations of
                                                 currently active subscriptions.

Output: optimized access strategies for existing subscriptions
           parameterizations of the access strategies
           (here: rate of proactive periodic strategy).
           resulting new GmmPr best_mmPr.

Begin

best_mmPr=1.0;
best_strategy=not_defined;

Repeat rep times,

      for all previously existing subscriptions
                     subtract current load L_old caused by this subscription in
                                 current access mode from total load L.
                     Take this subscription out of the currently active set and
                                                       consider it as 'new';
                      determine_configuration_of_new_subscription at load L-L_old
                                   considering the just taken out old subscription;
                     Add subscription to the currently active set with potentially changed access strategy;
                     L=L-L_old+(load created by reconfigured subscription)

                     Call network model to calculate network performance metrics (D, p_l) at load L+L_inc for resources R;

                                        Calculate GmmPr for these network performance metrics and existing plus new subscriptions;

                                        If calculated GmmPr < best_mmPr;

                                              Set currently considered access strategy and parameters as best one;
                                              Update best_mmPr;
                                        end-if
          end for-previously-existing;
 end repeat-rep-times;
```

are correctly specified. The simulations here assume context elements, where the physical information change can be expressed by a Poisson process. The reason for using a Poisson process is that the thesis does not analyze specific context elements, but rather stays generic in respect to the type of context element. Therefore it is possible to reduce computational effort of the algorithms and the parameter space stays small.

For the network, we assume that the context access traffic of all N subscriptions share the same network bottleneck, we consider a scenario where a CMN is running on a wireless connected device, hence the CMN receives and manages these N subscriptions over the same wireless access interface. We use two different models: (1) a network model based solely on the queuing delay of an M/M/1/K model; (2) a network model which uses an exponential forwarding delay in convolution with a queuing delay whose mean is calculated from a bursty MMPP/M/1/K model. The service rate used for both models is motivated by a WLAN access scenario with a link capacity of 10Mb/s. The buffer-size for the used bottleneck link is set to $K = 100$ messages. It is furthermore assumed that the node is not exclusively running the CMN functionality, but other (non-context) traffic is utilizing the same wireless interface, creating a 'base load' taking two different values, bursty traffic with 75% utilization for the MMPP/M/1/K model, and 90% for the M/M/1/K model.

The mmPr models are then used in the simulations to calculate the resulting GmmPr of the context access after each incoming subscription, comparing the selections from the algorithms from the previous section with three default constant access configurations. The comparison cases are: (1) always reactive access;(2) always event-driven access; (3) periodic with a fixed update rate of $\tau = 0.2$ updates/s. Obviously, other heuristic methods of selecting access could be considered, but are not included here due to space limitations.

The algorithm that minimizes the GmmPr based only on the configuration of the newly incoming subscription (Alg 1) is compared with the approach that iterates over all active subscriptions (Alg 2), where the number of iterations is set to 2. Table 4.1 summarizes the context information parameters and the M/M/1/K and MMPP/M/1/K parameters used for all six scenarios. The context parameters are thereby uniformly distributed in the range specified by the given intervals in the table.

Table 4.1: Context Information Parameters

Context Parameter	Value
Context Request Rate	[0.2 - 1] Requests/Sec
Context Event Rate	[0.2 - 1] Events/Sec
Context Update rate range	[1-10] Updates/sec.
Context Request Size	200 Bytes
Context Update/Response Size	[800-1200] Bytes/Update
Packet Size	1000 Bytes

Table 4.2: Network model parameters, M/M/1/K

Service rate (μ)	10 MBit/s (1250 pck/s)
Buffer Size (K)	100 Packets
Cross traffic	9 MBit/s

Table 4.3: Network model parameters, MMPP/M/1/K

Service rate (μ)	10 MBit/s (1250 pck/s)
Buffer Size (K)	100 Packets
Cross traffic rate during OFF	0 MBit/s
Cross traffic rate during ON	10 MBit/s (1250 pck/s)
Mean ON time	0.15 s
Mean OFF time	0.05 s
Exponential Transmission Delay, mean	500 ms

4.3.3 Simulation Results

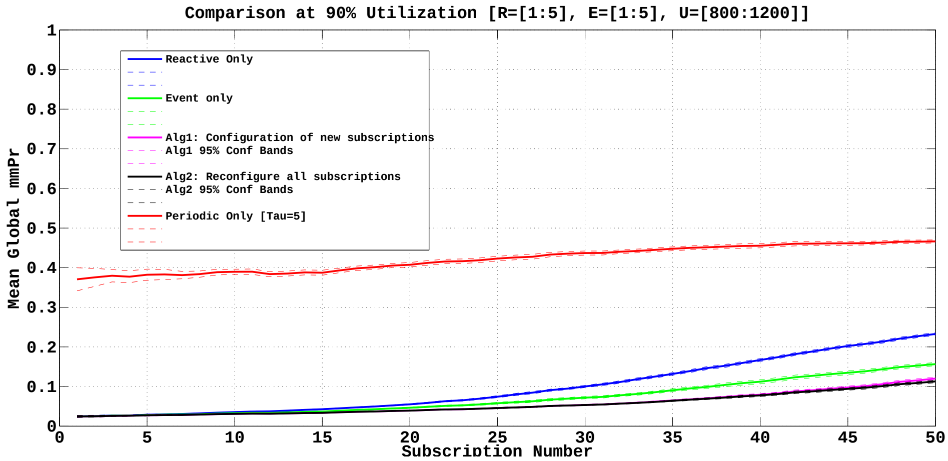

Figure 4.3: Average global context mismatch probabilities when using an M/M/1/K model with 90% cross-traffic utilization, using an update rate of 1/5 updates per second as comparison.

M/M/1/K Network model: Figure 4.3 shows the mean GmmPr of 20 simulations each having 50 subscription requests. The corresponding dashed lines show the 95% confidence intervals of this mean estimate from the 20 simulations. The graph represents a comparison of the fixed access strategies, periodic, reactive, and event-driven, with Algorithm 1 (optimized configuration of newly incoming subscription only) and the iterative Algorithm 2, which optimizes the configuration of the newly incoming subscription and subsequently makes two iterations over the already established subscriptions to reconfigure in case of better configuration options. Rejections of access requests (leading to mismatch equals 1 for this subscription) do not occur. After 50 subscriptions at the right end of the curve, averaged over the 20 runs of Algorithm 1, the configurations that result are: 36.4 event driven, 13.6 reactive, 0 periodic. The reconfiguration of already previously accepted subscriptions (Algorithm 2, black curve) shows a slightly lower average GmmPr compared to only selecting a configuration for the new subscription (Algorithm 1, pink curve). Algorithm 2 configuration results are: 31.45 event driven, 18.55 reactive and 0 periodic. The results show that the model-based optimizations lead to a substantially lower GmmPr than the fixed strategies. The iteration with potential reconfigurations of already existing subscriptions thereby shows a slight further gain, which is statistically relevant judging from the confidence intervals; however, the additional computational effort and also communication effort for reconfiguration of subscriptions appears not justified in the considered scenario.

Network model based on convolution of transmission time with queuing delay from bursty model: Figure 4.4 shows the mean GmmPr of 20 simulations each having 50 subscription requests when using the more complex network model, consisting of an

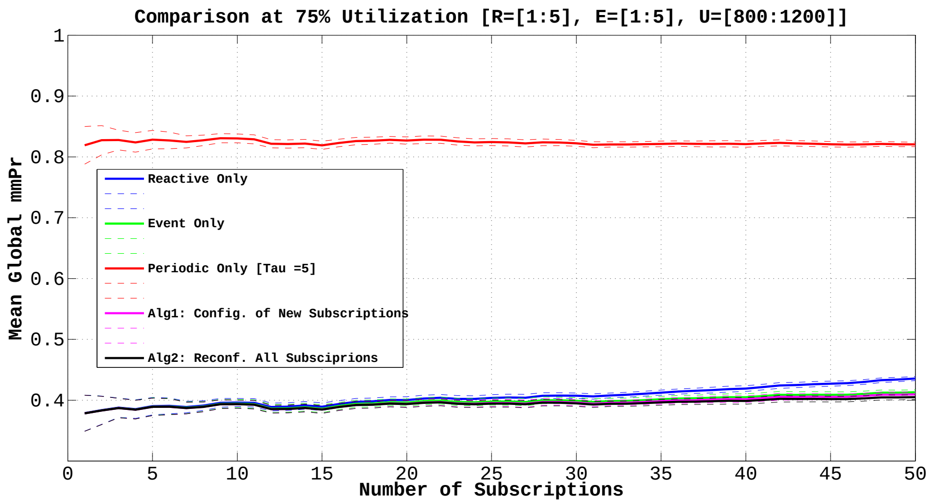

Figure 4.4: Average global context mismatch probabilities when using the more complex network model with 75% Utilization from bursty cross-traffic.

exponentially distributed transmission delay convoluted by a queuing delay whose mean is determined from a MMPP/M/1/k model with increasing utilization upon accepted subscriptions. The Service rate is 10 Mb/s and the Cross traffic is 7.5 MB/s. The corresponding dashed lines show the 95% confidence intervals of this mean estimate from the 20 simulations. The graph again shows a comparison of the fixed access strategies with Algorithm 1 (Optimization of newly incoming subscription) and the iterative Algorithm 2 with two optimization iterations (including potential reconfigurations of previously accepted subscriptions). Rejections of access requests do not occur. After 50 subscriptions at the right end of the curve, averaged over the 20 runs of Algorithm 1, the configurations that result are: 44.85 event driven, 5.15 reactive, 0 periodic. The reconfiguration of already previously accepted subscriptions (Algorithm 2, black curve) shows a slightly lower average GmmPr compared to only selecting a configuration for the new subscription (Algorithm 1, pink curve). Algorithm 2 configuration results are: 31.8 event driven, 18.2 reactive and 0 periodic.

It is of interest to notice the big difference in the numerical values of GmmPr in the two cases using the different network models. Adding the exponential transmission delay (which is independent of network traffic) in the second scenario, the impact of increasing number of subscriptions becomes much lower. On the other hand, the absolute values of the GmmPr are much higher than for the M/M/1/K case, despite the substantially lower utilization. The optimization algorithm is still effective in the scenario with complex network model, but in the chosen case, a fixed event-driven strategy may perform almost as well. Note however, that the result is a consequence of the considered scenario, and the algorithm would still be needed to identify the choice of strategies in a general setting. The benefit from reconfigurations of already existing subscriptions is small in both scenarios. That algorithm, however, is still useful as it can also be applied in settings when network behavior changes drastically (e.g. due to path failure and re-routing), since the existing subscriptions may need to be adapted.

4.3.4 Summary

The model-based optimization algorithms are executed on the Context-Management-Node (CMN), which is the node that receives all context subscriptions and manages the interaction with the context providers. We introduced two algorithm versions: 1) An algorithm that makes decisions on what context access strategy and configuration to use when handling a newly incoming subscription, 2) an algorithm that in addition considers reconfiguration of already established context subscriptions. The target in both cases is a metric called Global mismatch Probability (GmmPr) that is the average over all subscriptions of the probability that at the time of processing context information, the physical context source has changed value. The model used in both algorithms addresses the feedback loop of the impact of decisions on network traffic and hence network performance. Simulations were made to test and compare the algorithms. The simulation results showed that selections of context access configurations based on the mmPr model are able to reduce GmmPr, however, the level of improvements depends highly on the scenario considered i.e. while other scenarios can generate higher GmmPr results, the algorithm still improves the results but the significance of improvement varies from one scenario to the other.

4.4 Subscription Configuration for Class-Based Traffic Differentiation

Modern packet-switched network realizations allow for differentiated treatment of different traffic types, e.g. via the DiffServ architecture [11]. That way it is possible to apply different scheduling strategies and to configure different buffer-sizes for queues in bottleneck routers. The latter degree of freedom allows to trade-off delay for packet-loss. In order to extend the context subscription configuration algorithms from the previous section to a traffic differentiation scenario, we first motivate the expected benefit from class-based traffic treatment by an analysis of the trade-off of loss versus delay on the context mismatch probability. We subsequently extend Algorithms 1 and 2 to the class-based scenario and investigate the resulting context subscription performance, measured by global mismatch probability, in simulation experiments.

4.4.1 Delay versus Loss Trade-off

In a class-based QoS differentiation scheme, it is possible to have different subscriptions treated in different classes which potentially use different buffer-sizes. Using short buffers, lower transmission delays result at the cost of increased packet loss. Using the mmPr models and assuming the M/M/1/K bottleneck buffer model, the impact of the buffer size changes on the resulting context quality can be analyzed in the analytic model. Figure 4.5 shows the mmPr of the reactive access strategy, when it operates in a network which is described by a bottleneck M/M/1/K queuing model with varying buffer size K. The different curves show from bottom to top different context change rates of $\lambda = 1, 2, 5$, while other parameters of the context information (update size equals to one single packet) and the context access rate ($\mu = 2$) are constant, and the service rate of the M/M/1/K model maps a 10Mb/s WLAN scenario with packet size of 1000 bytes, which here is assumed to run in overload at $\rho = 1.2$. Note that in order to analyze the delay-loss trade-off, in contrast to the previous algorithms and evaluations, the additional (constant) context access traffic is here not taken into consideration, so the bottleneck utilization is exactly the same across all strategies.

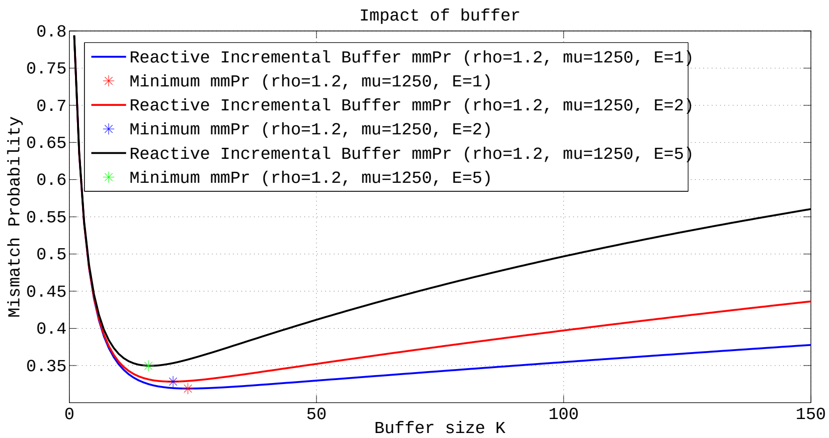

Figure 4.5: Impact of buffer size in the M/M/1/K network model on mismatch probability of different access strategies at high utilization

Figure 4.5 shows that for the chosen example scenario, each curve reaches a minimum mmPr value, marked by the stars on the curves. As we are using analytic formulas, the buffer-size K_{min} at which the minimum mmPr is achieved can be calculated from numerically searching the root of the mmPr derivative for the given parameters. Figure 4.6 shows the resulting K_{min} values for different event rates (x-axis) and for different utilization of the bottleneck queue. For the reactive strategy with the used scenario parameters, scenarios with lower event rates and lower utilization favor delay over loss, so larger buffers are better (K_{min} grows). Note that this section is not intending to formally prove such behavior, but rather acts as motivation that DiffServ-like class differentiation with different buffer sizes can be valuable, and that there exist approaches to determine optimal class parameters.

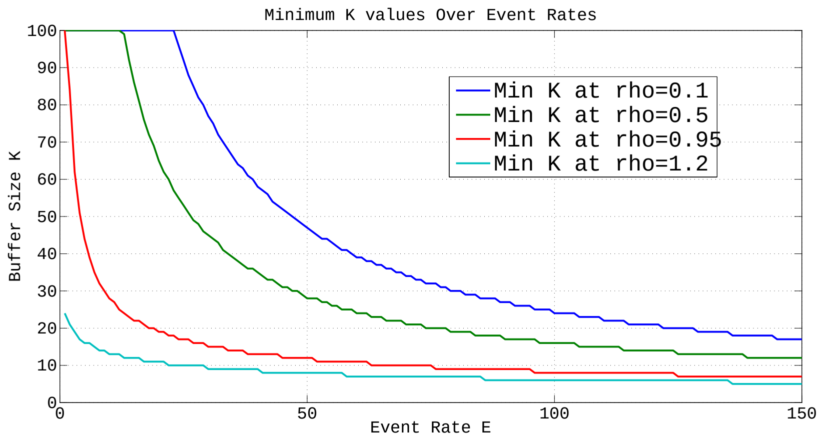

Figure 4.6: Buffer-size K_min that minimizes the mmPr for different change rates of the context information (x-axis) and different utilization (different curves).

4.4.2 Algorithm

Basic approach for a class-based scenario is to extend the algorithms of the previous section by also iterating over the possible C DiffServ classes. The Network model then needs to be able to compute the resulting performance (mean delay and loss probability in our case) given that a certain context traffic is added to a specific class. To keep this part simple in our analysis, we assume a static bandwidth assignment, i.e. the overall link capacity of the bottleneck is distributed to the C classes, so that for each class i, an M/M/1/K_i model can be applied to calculate mean delay and loss within that class. The buffer-size K_i and the service rate μ_i can then be set independently for each class, as long as $\sum_i \mu_i = \mu$.

As before in section 4.3, the extended algorithm is executed at the reception of a new subscription. The algorithm then checks the available QoS classes and calculates the mmPr for brute force over all access strategies for all QoS classes. The algorithm then selects the class and the strategy that results in the lowest GmmPr, taking into account also the impact of the additional context traffic on the already existing subscriptions (only the subscriptions in the same class as the new subscription are affected in the used scheduling strategy). Analogously to Algorithm 2, subsequently already existing subscriptions are reconsidered for potential reconfiguration. The extended pseudo code is found below.

4.4.3 Simulation Model

To evaluate the algorithms, simulations as described in section 4.3.2 were extended to the class-based scenario enabling now to compare Algorithm 3 with the class-less scenario using Algorithm 2. As in section 4.3, the simulation considers the arrival of N context access requests at the CMN. All parameters and their values in the simulation are listed

Algorithm 3 Diffserv Subscription Classification Algorithm

```
 function determine_QoS_class_and_access_configuration_of_new_subscription
 Input: Current load L_j on classes and available network (bottleneck) resources R_j for each class j
                in the path context agent - CMN - subscriber.
        Request Rate mu_i of new subscription.

Available Information (persistently maintained at CMN)
           From registration: Change rate lambda_i, and update size u_i of
           context element of new subscription.
           From previous subscriptions: Parameters and configurations of
                                 currently active subscriptions.

Output: QoS Class for new access strategy
                access strategy for new subscription
        parameterizations of the access strategy
        (here: rate of proactive periodic strategy)
        resulting new GmmPr best_mmPr.

Begin
best_mmPr=1.0;
best_class=NaN;
best_strategy=NaN;
for all classes j=1,ï¿ ,N;
   For all access strategies in (Reject, reactive, proactive event-driven,
            proactive periodic_rate1, ... proactive periodic_rate);
      calculate additional class load Lj_inc created by NEW subscription
                           when using this access strategy (see Table I);
      Call network model to calculate class performance metrics (Dj, Pj)
                                 at class load Lj+Lj_inc for resources R;

      Find Minimum N Class GmmPr


       If calculated GmmPr < best_mmPr;
               Set currently considered access strategy and parameters
               as best one;
               best_Class=j;
               Update best_mmPr;
     end-if
   end for-loop-access strategies;

   Calculate Link Global mmPr (GmmPr) for all classes;
     Select class, strategy and parameter that minimizes Link GmmPr;
     If Subscription_number > 1 for selected class
     Run Reconfigure subscriptions and recalculate GmmPr;
     end for-loop-classes;
end function;
```

in Table 4.4. Same assumptions as for the previous simulations hold: (1) The CMN is assumed to have perfect knowledge of the average request rates for the individual subscriptions. (2) For the network, we assume that the context access traffic of all N subscriptions share the same network bottleneck. (3) a WLAN scenario is used to motivate the network parameters. The corresponding same packet rate is in the class-based scheme mapped to the larger Class 1, leading to class utilizations from background traffic of $\rho_1 = 1$ (while Class 2 has no background load, $\rho_2 = 0$) for the 90% Utilization scenario. For the 100% Utilization scenario the resulting background utilization on Class 1 is $\rho_1 = 1.11$. Mapping all background traffic into Class 1 corresponds to a scenario in which Class 2 is exclusively reserved for context traffic, while the subscription configuration algorithm is also allowed to use Class 1.

Table 4.4: Context Information Parameters

Context Parameter	Value
Context Request Rate (μ)	[0.2-5] Requests/Sec
Context Event Rate (λ)	[0.2-5] Events/Sec
Context Request Size (U_{rqs})	200 Bytes
Context Update Rate (τ)	[1-20] updates/sec
Context Update/Response Size (U_{rsp})	[800-1200] Bytes/Update
Classless Buffer Size (K)	100 Packest
Class 1 Buffer Size (K_1)	100 Packets
Calss 2 Buffer Sive (K_2)	30 Packets
Classless Maximum Link Capacity (μ_{link})	10 Mbit/s
Class 1 Maximum Link Capacity (μ_1)	9 Mbit/s
Class 1 Maximum Link Capacity (μ_2)	1 Mbit/s

NB! Update rate is selected by the algorithm.

4.4.4 Simulation Results

Scenario with 90% utilization from other traffic Figure 4.7 shows the comparison between classless context optimization and class-based context optimization with a base-load of 9 Mbit/s, which in the class-based scheme is put onto Class 1. The results shows that for the class-based scenario the class distribution for 20 simulation runs is: on average 3 out of 50 subscriptions are in Class 1, while Class 2 is used for on average 47 out of 50 subscriptions; no subscriptions are rejected. The subscriptions show on average the following distribution in the class-full scenario: 18.75 using the reactive strategy, on average 31.15 using the proactive event strategy, and 0.1 periodic.

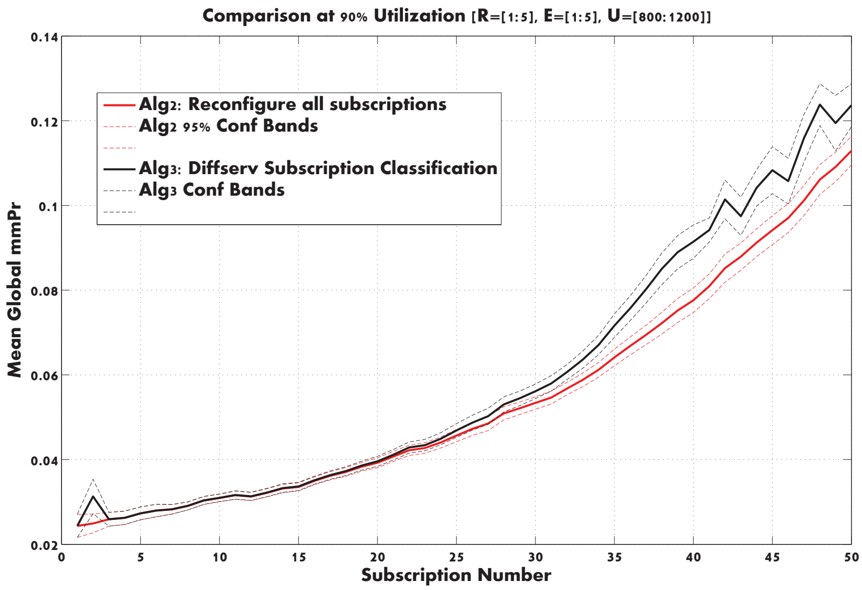

Figure 4.7: Classfull vs Classless Comparison at 90% Background Traffic in the M/M/1/K network model

For the classless scenario the same results as shown in section 4.3.3 apply. As it can be seen from the results, splitting the link into two classes does not benefit the reduction of Global mmPr for active subscriptions in the shown scenario of 90% background traffic utilization.

Scenario with 100% utilization from other traffic figure 4.8 shows the comparison between classless context optimization and class-based context optimization with a base-load of 10 Mbit/s which again is assigned to Class 1 in the class-based scheme. The results shows that for the class-based scenario, the class distribution is on average 0.2 subscriptions for Class 1 and 49.8 subscriptions for Class 2. As for the subscription distribution, for the class-based scenario, it is on average 0.7 rejects, 18.55 reactive, 30.55 proactive event driven and 0.2 for proactive periodic update. The results for the classless scenario were presented in section 4.3.3. In this highly saturated scenario it can be seen that using a split link benefits the Global mmPr for all active subscriptions.

4.4.5 Summary

An investigation over the possibility of increasing context information reliability by using QoS classification has been shown. A new model-based approach was presented which maximizes context reliability by assigning dynamic context subscriptions to an appropriate QoS service class. The concept defines and evaluates an algorithm which is intended to be used as a part of a context QoS control framework. The model's algorithm is executed on the Context-Management-Node (CMN), which is the node that receives all context

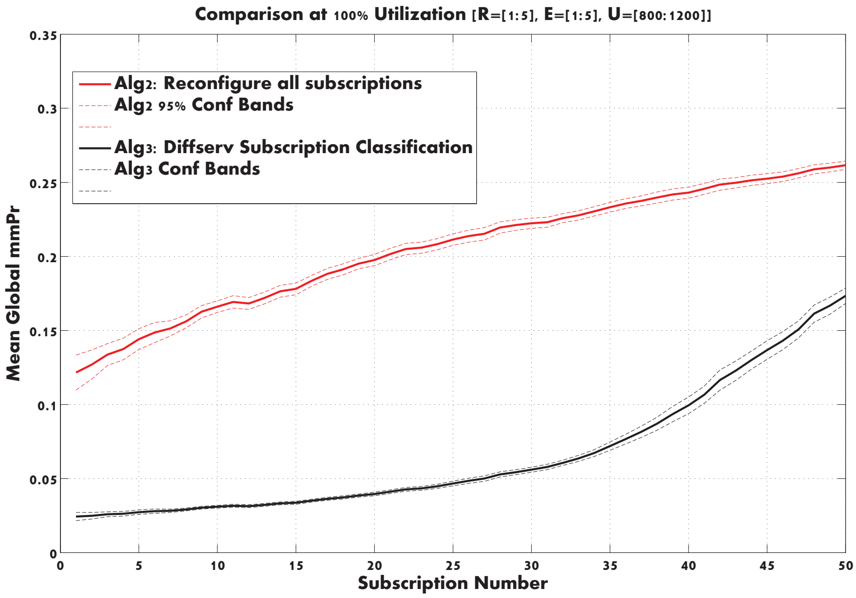

Figure 4.8: Classfull vs Classless Comparison at 100% Background Traffic in the M/M/1/K network model

subscriptions and manages the interaction with the context providers. The algorithm showed that for lower utilization a classless environment would serve best the so-called GmmPr metric, however, for high utilization scenarios the algorithm manages to reduce GmmPr as incoming subscriptions would be assigned to a better serving class with lower GmmPr. It was also shown that subscriptions assigned to class 1 traffic yields very high GmmPr, yet this problem could be solved by introducing mmPr bounds that keep the GmmPr below a certain threshold.

Part III

Network Planning Optimization

Chapter 5

An Automated Planning Model System for Radio over Fiber Heterogeneous Wireless Networks

5.1 Introduction

ISPs today are in search of a cost effective method to cope with increasing bandwidth demands in 4G wireless networks. ISPs are continuously optimizing their planning methods in order to build a more cost effective network, that is also capable of meeting the demands of mobile users today. Higher bit-rates come at the cost of reducing wireless cell sizes, thus coverage by a single base station is reduced and therefore more equipment is required in order to meet the ISPs coverage goals. This directly results in the deployment of more Base-Stations (BSs) and antenna sites to be able to cover large areas. It is therefore in the ISPs' interest to minimize costs for BSs and electronic equipment as much as possible.

By using the RoF architecture it is possible to reduce the deployment costs of such networks by using Remote Antenna Units (RAUs) and Control Units (CUs) **??**. The thesis introduces a new planning model for Radio over Fiber (RoF) networks, the RoF architecture is adopted from the architecture proposed by the FUTON project [39; 54].

The FUTON project addresses the challenges of a 4G network by making it feasible to implement a hybrid radio fiber infrastructure which connects RAUs to CUs where a joint processing can be performed. The FUTON architecture is vastly different than what is normally implemented by Wireless Internet Service Providers (WISPs) today, which means that the normal planning procedures may not be applicable.

The FUTON architecture places all the signal processing at the CU sites that are connected by single fiber to the RAUs.

This chapter of the PhD thesis is based on the master thesis [62]. The text has been rewritten and includes in many cases new reflections from the situation today in comparison to how the situation was at the time of writing. The chapter shows results found for radio coverage algorithms, CU placement algorithms and fiber placing algorithms from the master thesis [62]. The chapter also presents a new price comparison research, which was not included in the master thesis. This research investigates deployment costs between regular mobile planning and the proposed planning method in the Danish municipality of Aalborg.

Below are the main research areas. Each area is an individual section in the chapter:

- Radio planning
- CU location planning
- Fiber planning
- Deployment Cost Comparison

5.2 Radio Planning

This process deals with placing RAUs for different wireless technologies such as 3G, WiMAX and LTE in the Danish municipality of Aalborg. If the planning was done today it would perhaps consider scenarios for 5G networks, however, since ISPs are still running legacy networks it is of interest to compare new and old technologies. Also this new architecture makes it possible for ISPs to make use of the legacy mobile network location. First an estimation providing the total number of potential users and their location is made, this estimation is then used to find the potential locations for the RAUs. Secondly, it is important to calculate the maximum allowable path loss for each wireless technology, this is done by setting up link budgets for all wireless technologies. Finally, the information is used in the COST HATA propagation model [40; 49] in order to find the range of the RAUs for each wireless technology. Today there are several available propagation models. Some of these model are outdoor models i.e. for outdoor coverage such as the ITU Model for Indoor Attenuation [13] and some are indoor models for indoor coverage such as the Okumura model [63], however, at the time of writing there was only available information on the COST HATA model. The goal of the automated planning tool is to provide a coverage solution that covers a defined set of users at the lowest cost.

5.2.1 Distribution of Potential Users

It is important for the planning process to take into account the population distribution of the area. This population represents the potential customers which the ISP wishes to provide services to. The available GIS-data from TDC for Aalborg contains Network

Termination (NT) points of homes and businesses in the municipality. Since this is information about fixed terminations it therefore can't be used for a wireless scenario since mobile users are on the go. Therefore, the data has been approximated, and it has been assumed that there are approximately 2.8 mobile users per NT during peak hour as a worst case scenario. The GIS data included 78818 NTs, and therefore it is assumes that Aalborg holds an average of 201707 mobile users. The distribution of users in the Aalborg area can be seen in figure 5.1.

Figure 5.1: Distribution of potential users in Aalborg Municipality

5.2.2 Distribution of Potential RAUs

The lack of BS location information was overcome by creating a hypothetical grid with possible available BS locations. In the GIS-map of Aalborg municipality every 1000 meters is marked as a potential BS location as shown in figure 5.2. This however is an assumption made due to lack of data, however, in a real deployment scenario potential locations are considered based on factors like local regulations, space requirements, access to power, etc. Since some of the NTs are located outside the municipality boundaries, this leads to some BS locations outside the boundaries. The goal here is to cover 97% of the population. In reality the ISPs have available information on current BS locations.

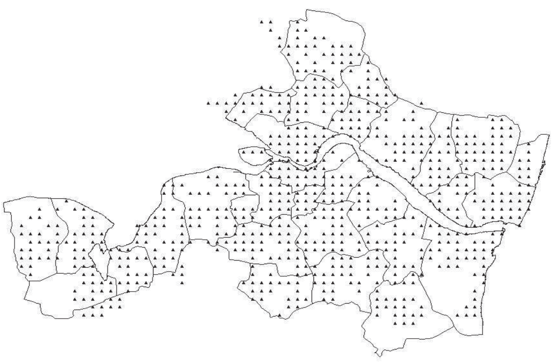

Figure 5.2: Possible BS locations in the Municipality of Aalborg

Due to the higher user density in urban areas, a more fine-grained grid had to be made to deal with the short ranges in those areas. This fine-grained grid is placed over the Aalborg down-town area. Note that the finer grid is placed over the existing grid. Distances between possible RAU locations in the finer grid is 200 meters and can be seen in figure 5.3. Today due to the wide spread of 4G networks and coverage this fine grid might have been done not only for urban areas but for the entire municipality of Aalborg.

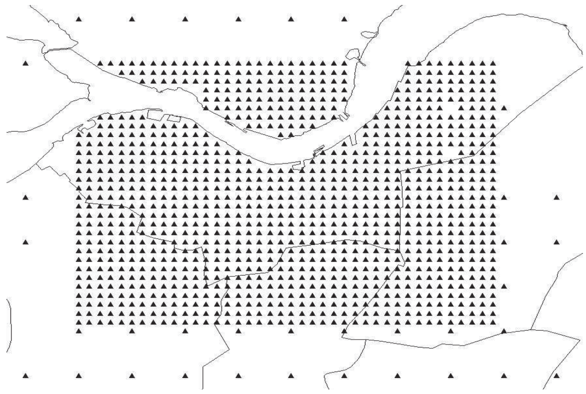

Figure 5.3: Possible BS locations in the Municipality of Aalborg

After finding the positions of possible RAUs, the next step would be finding the coverage range for each RAU in order to know the amount of Mobile Termination's (MT) covered by each RAU. It is therefore required to calculate the radio link budget in order to find out what is the maximum allowed path loss and use this information in the propagation model.

5.2.3 Cell Ranges

By calculating the link budget, and using them in the COST HATA propagation model, it was possible to find the cell ranges for each wireless technology. The only exception was LTE-Advanced. At the time of writing there is no available information about the link budget, therefore a 300 meter range was assumed. The link budget information, however, is available today [24] and research shows that they average between 1 Km to 5 km. All ranges found for all technologies are in fact indoor coverage ranges found by using the lowest modulation for each technology. Ranges for uplink and downlink differ due to the Effective Isotropically Radiated Power (EIRP), which is higher for downlink than it is for uplink, due to BS gain compared to NTs antenna gains. Ranges of all the wireless technologies resulting from link budgets and propagation model can be seen in table 5.1 .

With the current information about cell ranges for all technologies, the next challenge would be to find an automated method to generate optimal network plans with locations of BSs for 97% coverage of all NTs.

Table 5.1: Cell ranges in km for each technology according to environment

	$WiMAX_{down}$	$WiMAX_{up}$	HSDPA	HSUPA	LTE-Advanced
Urban	0,383	0,355	0,682	0,635	0,3
Sub-urban	1,179	1,091	1,852	1,724	-
Rural	4,911	4,543	6,957	6,475	-

5.2.4 Radio Coverage Algorithms

The Link budget determines the range of each RAU. The RAU is assumed to provide adequate services to mobile users in the geographic area. Therefore the cell planning process can be formulated as a set cover problem [23]. The main objective is to find the total number of potential RAUs, that are required to provide the MTs with sufficient signal strength. The set cover problem is formulated mathematically as follows:

$$Minimize_z = \sum_{j \in J} x_j$$

subject to :

$$\sum_{j \in N_i} x_j \geq 1 \forall i \in I, \forall j \in J$$

Where
Z number of BSs
J set of potential BS sites (indexed by j)
I set of MTs (indexed by i)

$$x_j = \begin{cases} 1 & \text{if BS is at j} \\ 0 & \text{otherwise} \end{cases}$$

N_j set of BSs j within range of MT i

by using this set cover problem the number of BSs required is minimized in order to cover all MTs. Since in a real life scenario coverage is not done for 100% of all Mts, as this will be too expensive in CapEx for the ISP, the constraint is changed from "to be valid for all i" to "a fixed percentage of i". In this case 97% of all MTs. It is also worth mentioning that providers today such as TDC offer over 99% coverage.

The set cover problem is considered NP-hard [5], therefore, to solve it heuristic algorithms are used. Several heuristic methods can solve the set covering problem, the thesis implements a greedy approach, and a genetic algorithm approach.

The greedy and genetic algorithms are commonly used to solve the radio planning set cover problem [10; 14; 29]. Both algorithms flows are shown in figures 5.4 & 5.5 respectively. The greedy algorithm is simple to implement, while providing a rather fast solution, however, due to its greedy nature the greedy algorithm may often get caught in a local optima. This would result in a non optimal solution. In order to overcome this, the genetic algorithm was implemented.

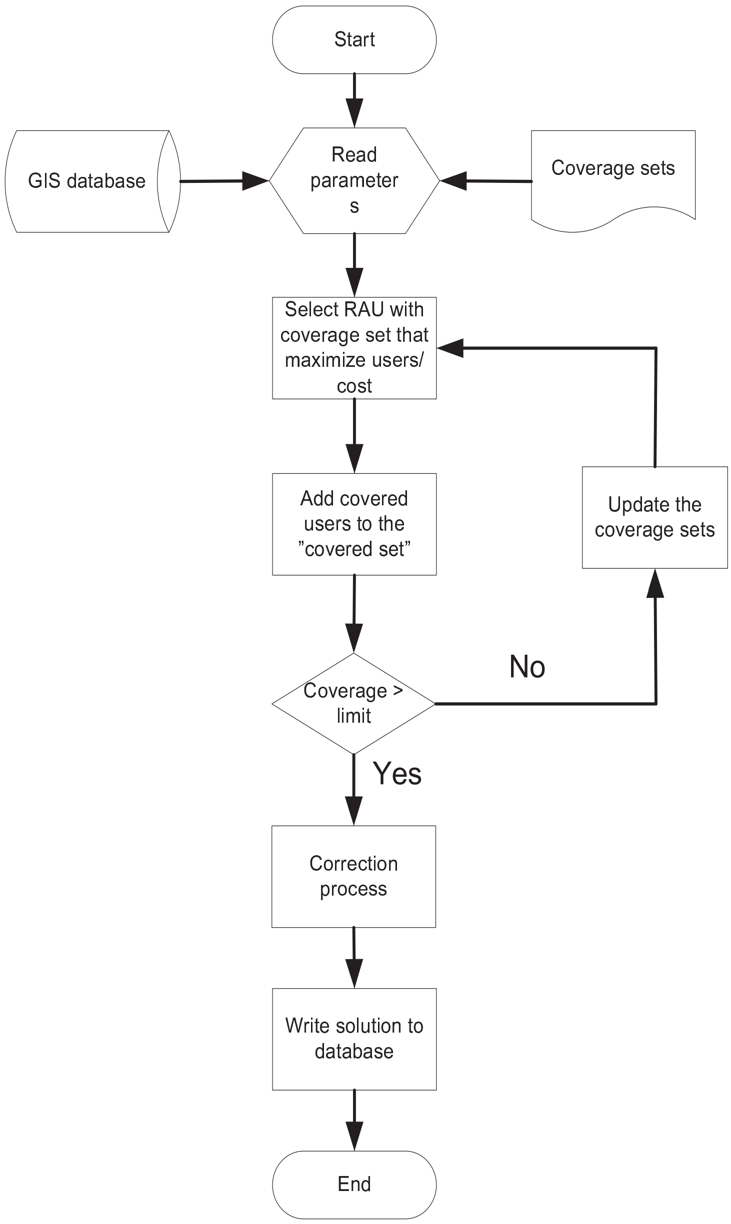

Figure 5.4: The greedy algorithm for solving the set-cover problem

5.2.5 Radio Planning Results

With the current information about RAU/BS locations and RAU/BS ranges it is possible to compute how many RAUs need to be active in order to cover a certain percentage of MTs. Experiments were made with different percentages 95%, 97% and 99% of total

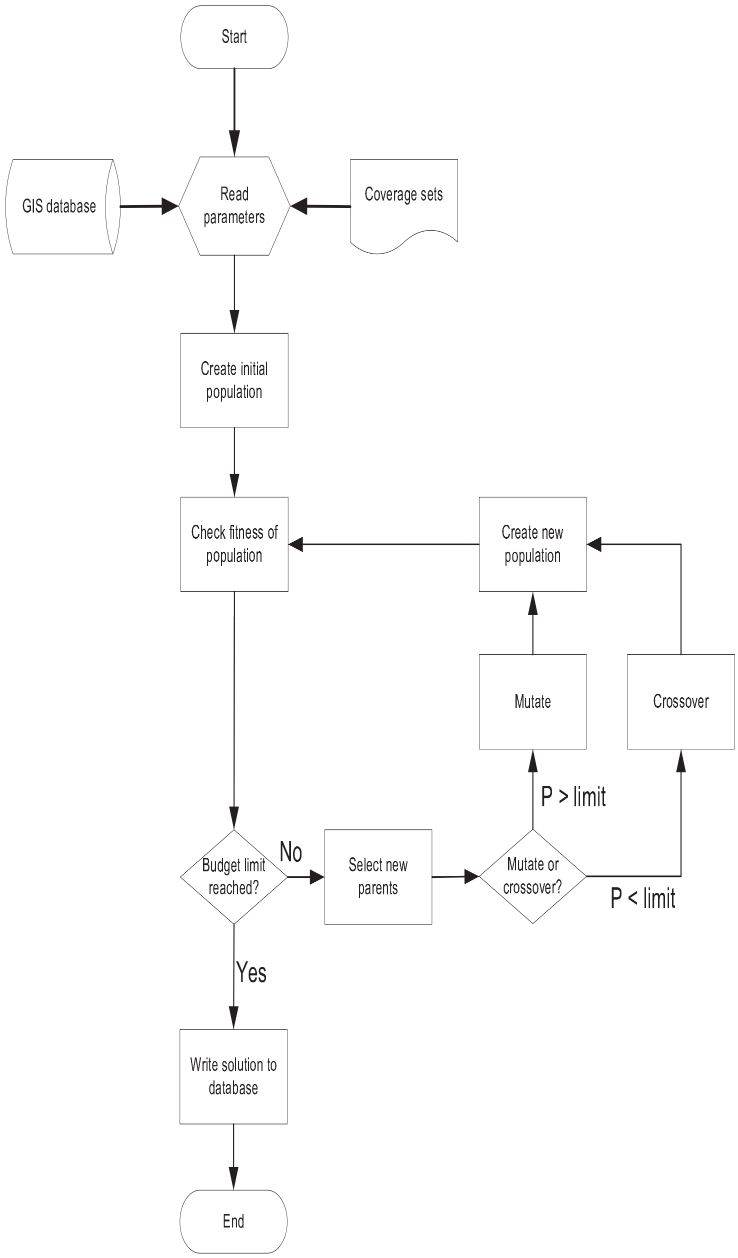

Figure 5.5: Genetic algorithm for solving the set-cover problem

MTs, in order to find a reasonable coverage area. An example can be seen in figure 5.6 which shows the effect of different coverage percentages on the total number of active RAUs for the WiMAX technology.

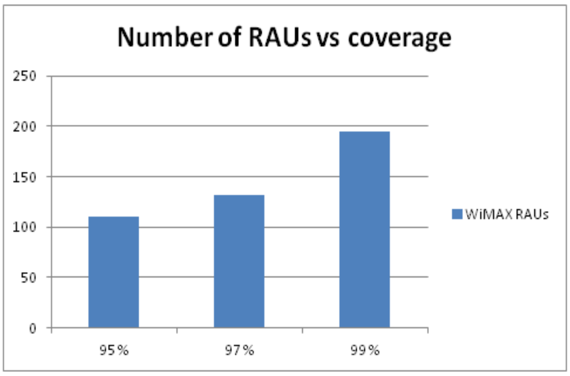

Figure 5.6: Needed RAUs for 95, 97 and 99% indoor coverage

Based on figure 5.6 97% coverage is assumed as the coverage percentage of MTs for each wireless network. Using the greedy algorithm and 20000 generations of the GA, the set covering problem was solved. Using COST 231 Hata propagation model[49], the ranges of RAUs was found and the aim was that each RAU should be able to successfully cover all Mts in its range.

UMTS Results Figure 5.7 shows the 3G indoor coverage. Even though the figure shows that over 97% of MTs were covered, the geographical coverage, however, is not 97% of the area. The figure shows that only the most densely populated areas are covered and at the same time many of the rural areas do not receive indoor coverage. Initially, the greedy algorithm solution resulted in activating 43 RAUs, however, after running 20000 generations of the greedy algorithm on a 2Ghz core processor, the result was reduced by 7% to 40 RAUs. The genetic algorithm complexity is dependent on many factors, the fitness function, the population size and on how many generations. A genetic algorithm is usually said to converge when there is no significant improvement in the values of fitness of the population from one generation to the next.

Figure 5.8 shows the outdoor coverage for 3G. It can be seen from the figure that the entire area is covered for outdoors, where wireless mobility is needed.

WiMAX Results Due to the shorter range of WiMAX compared to that of 3G, the WiMAX solution therefore requires more RAUs to service the same amount of MTs. Figure 5.9 shows the indoor coverage of WiMAX. The greedy algorithm resulted in 132 RAUs initially, however, after running the GA for 20000 generations there were no improvements to the initial result found by the greedy algorithm. The outdoor coverage for WiMAX also covers the entire area.

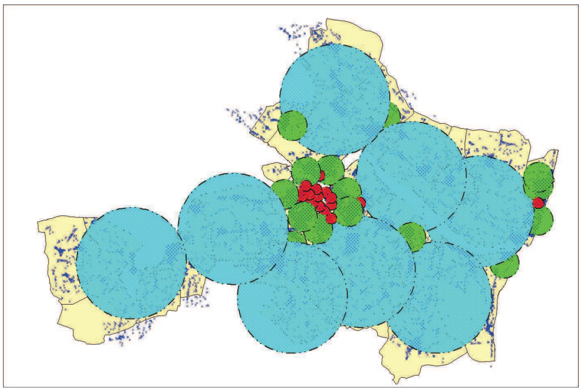

Figure 5.7: Coverage map for 97% indoor coverage for 3G. Blue circles = rural BS, green circle = suburban BS, red circle = urban BS

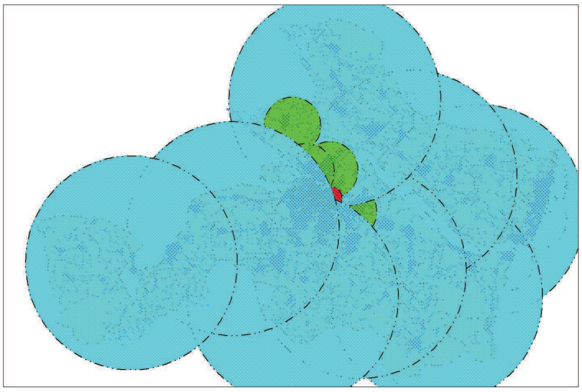

Figure 5.8: Coverage map for 100% outdoor coverage for 3G. Blue circles = rural BS, green circle = suburban BS, red circle = urban BS

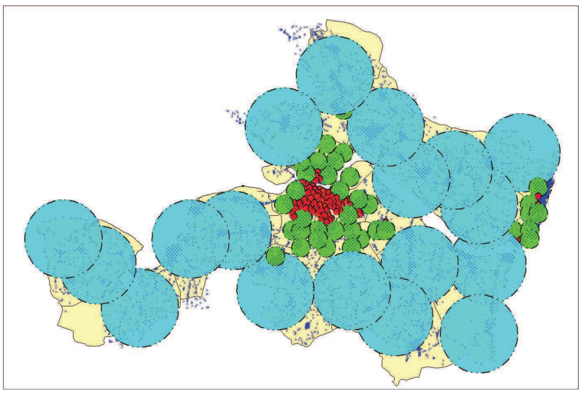

Figure 5.9: Coverage map for 97% indoor coverage for WiMAX. Blue circles = rural BS, green circle = suburban BS, red circle = urban BS

LTE-Advanced Results The aim of the planning algorithms is to reuse sites, where other wireless technologies are already set-up in order to save deployment costs when the LTE-Advanced technology is rolled out. Figure 5.10 shows the indoor coverage of LTE-Advanced technology. Since the cell range is only 300 meters, this will result in a higher number of RAUs to deploy. The greedy algorithm provided an initial result of 162 RAUs, 69 of these RAUs are used only for LTE-Advanced, 3 RAUs are shared between all 3 technologies and the remaining are either a combination of UMTS-LTE or WiMAX-LTE. After running the GA for 20000 generations no improvements could be done to the initial result found by the greedy algorithm.

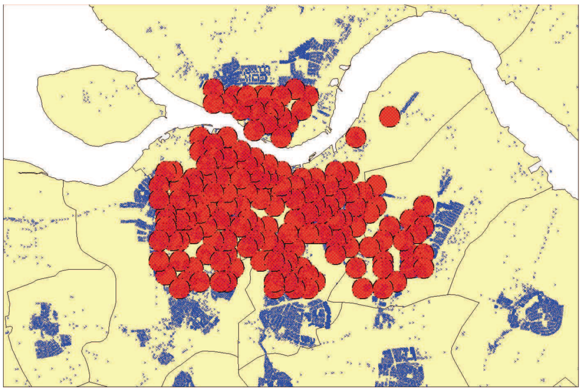

Figure 5.10: LTE-Advanced coverage in the downtown area of Aalborg

For the fiber-planning process, it is intended for all wireless technologies to be connected in a common architecture. All BSs for all wireless technologies 3G, WIMAX and LTE-Advanced will be substituted with transparent multi-frequency RAUs. From the previous results it was found that the total number of RAUs required for all 3 technologies in the Aalborg municipality area was 233. This means that 40% of RAUs have 2 or more co-existing technologies. Figure 5.11 shows the locations of all RAUs. From the figure it can be seen that the RAUs with 2 or more technologies are mostly located around the urban areas.

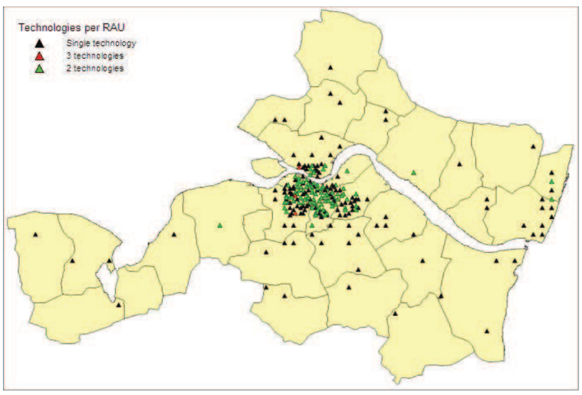

Figure 5.11: RAU locations and technologies per RAU

5.2.6 Summary

Using the link budget and a propagation model it was possible to calculate the range of wireless BSs/RAUs. This information was used to create coverage sets for all the RAUs.

The radio coverage plans were then formulated into a set coverage problem, which automatically generated plan results for the entire area to be covered, by using heuristic methods to find the locations of BSs/RAUs. It was noted that in some cases after running the GA for 20000 generations it was possible to improve the initial results found by the greedy algorithm, this is of high importance since in real scenarios ISPs are looking to reduce deployment costs to minimum. It was also noted that the implementing a pure GA solution was impractical since the algorithm uses too long time to be of any practical use, therefore it was decided to use it in parallel with the greedy algorithm.

Results for coverage plans for all 3 technologies 3G, WiMAX and LTE-Advanced were all made automatically. All plans cover 97% of all MTs in the area, however, it was noticed that due to the short range of WiMAX, it required more RAUs when compared to 3G. LTE-Advanced technology aimed at reusing sites where initially 3G BSs and WiMAX BSs are found. If no available BSs are in place, the algorithm would then activate a location to add an LTE only RAU to meet the coverage limit of 97%. The results also showed that in the case of Aalborg, only 233 RAUs are required and will be connected using the FUTON RoF architecture proposal. The results also showed potential savings of up to 40% less than if all three wireless technologies were deployed separately. All 233 locations will be replaced with multi-frequency RAUs which has a further reduced cost than a regular BS. Radio signal processing, however, will be done at the CUs.

From the results it was also noted that the indoor coverage was 97% of all available MTs, however, the outdoor coverage for all technologies was 100% of the area. This means that outdoor mobile users are covered in the entire geographic area. Even though the planning method itself does not take into account outdoor coverage but only indoor, the 100% outdoor coverage is a result of signals not having to penetrate through buildings and thus the range increases. The planning method is able to quickly drive an automated sub-optimal solution using the greedy algorithm. The result can further be optimized using a GA or any other heuristic optimization algorithm. It also may be utilized as a base for a manual planning solution.

5.3 CU-placement

The next step after completing the radio coverage planning and finding the RAU locations, is to find out where the CUs should be placed. According to [39] it is not specified the maximum number of RAUs that can be connected to a single CU, therefore it is assumed that each CU can hold as many Joint Processing Units (JPUs) as required to service all

RAUs. It is, however, required for redundancy measures to spread out the locations of the CUs geographically. This will protect the network in case of failures. Another requirement is to evenly distribute the number of RAUs amongst the CUs. This is done in order to achieve an even traffic load across the network. Similar to the RAU placement scenario, it is possible in a real implementation scenario that the ISP already has a suitable location to place the CUs. Similarly, the considerations for choosing a CU location is like that of choosing the main distribution locations for a regular IP network. ISPs have available information of the sites: backbone connectivity, power redundancy, accessibility, firewalls and security. All these are of great importance. Therefore, ISPs often choose common locations to be their main distribution sites for the CUs. Due to lack of information of these sites it is therefore assumed that any RAU location can be consider a potential CU location. It is therefore possible to approach this problem in the same manner that was done in the radio planning process. Therefore, the problem is again formulated as a set covering problem. The goal here is to find out the total number of RAU locations that need a CU placement so that all RAUs would have a CU connection. Since fiber is normally laid in trenches, the common rule of thumb of getting from Euclidean distance to wired distance is applied to account for this fact.

$$Wired\,Distance\; = Euclidian\,Distance * \sqrt{2}$$

[55]

It should be noted that in the urban area of Aalborg, many RAUs are within range of only one CU. This is undesirable when planning networks, since the failure of 1 CU will be responsible for the failure of many RAUs. A well planned network would have a more evenly distributed number of RAUs per CU. To achieve this, an upper limit has been set to the 50 closest RAUs in the greedy algorithms coverage sets. The same practice is followed where the greedy algorithm runs first. That assigns an initial CU at the location of the RAU that has the most RAUs within its wired distance. The next step is to run a GA over the produced results, which in turn shifts the CU locations around amongst the available RAUs with the goal of minimizing the variance of the number of RAUs covered by each CU. This ensures that the number of RAUs per CU becomes as evenly distributed as possible. A more evenly distributed number of RAUs ensures an even traffic distribution across the network, avoids congestion and reduces RAU failures in case of a CU failure.

Now that the CUs have been placed, RAUs are assigned to their nearest CU using a path-finding algorithm, in order to provide the most suitable path given the segment information found in the available GIS-data from TDC.

5.3.1 Path finding and the A* Algorithm

In order to optimize the path-finding process and make it as fast as possible, redundant information had to be removed from the GIS-data that was provided by TDC in order to decrease its complexity. The GIS-data contained information about segments i.e. the line that forms the roads, and segment points, which are the points on the map where the segments connect. By quickly scanning the available database it could be seen that many of the segment points had a degree of 2, i.e. only connected 2 segments. All those 2nd degree segment points are presented in this manner in order for the data to look geographically correct when viewed on the map. It is, however, considered redundant information when purely looking at it from a network planning perspective. It was therefore decided to remove those segment points form the data-base, since there will be no loss of information. The segments holding a degree of 2 are removed, while the new segment still holds the length sum value of the 2 old segments. Figure 5.12 shows this procedure.

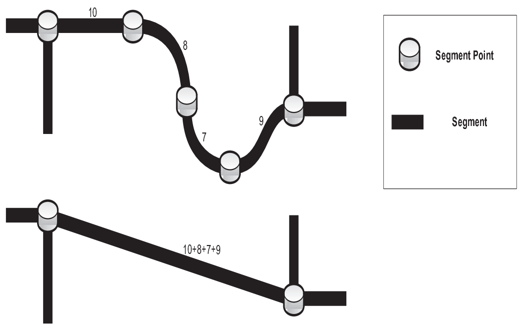

Figure 5.12: Segment table reduction

To find the shortest path 2 algorithms were implemented. The first was Dijkstras shortest path algorithm, which despite the fact of reducing the segment points was very time consuming when calculating the shortest path. The second algorithm implemented was the A* search algorithm [16]. A* algorithm like Dijkstra's algorithm gives optimal results, however, it is considered faster due to its guided search [35; 36]. In order for the A* algorithm to deliver optimal results, there can be no overestimation of the actual cost of reaching a target by the heuristic cost function. After implementing the 2 algorithms it can be seen that there was a huge different in speed between both algorithms. Dijkstra's goes on to search all of the possible paths till the target is found in one of the paths. A*, however, is capable of limiting it's search space. This is done by using an estimated cost to reach the goal vertex, which places a degree of control by guiding the search in the right direction and therefore solves it faster.

The Dijkstra's shortest path algorithm was originally implemented as the path-finding algorithm. Even though the segment-database was reduced, the calculation time for a path was very time-consuming. In order to improve the performance, only the features needed to perform path-finding was imported from the database and read to memory, the segment information was then accessed from there. This improved performance, by reducing the search time for a specific segment approximately 10 times. The overall performance was still too time consuming. Results from comparing both algorithms can be seen in table 5.2.

Table 5.2: Example results of path-finding algorithms

Test	Path-length [m]	Hops	Mean A* [s]	Std-dev A* [s]	Mean Dijkstra [s]	Std-dev Dijkstra [s]
1	3066	2	0.08	0.00	0.18	0.01
2	3100	18	2.0	0.05	18.1	0.3
3	42177	132	62.5	1.0	710.9	3.7
4	44690	188	308.8	3.2	687.0	5.9

Table 5.2 shows that the number of hops has a direct impact over the computation time for both algorithms. As can be seen A* performs faster than Dijkstra's, and longest performance improvements can be seen in similar situations as test 3, in this case, its heuristic cost function is considered a good estimate of the actual path. For test 4 where the path actually goes through the city center, the difference in performance between both algorithms is lower since the A* algorithm also considers many possible paths in the dense city center. Since both algorithms provide optimal solutions, the A* algorithm is therefore the most logical choice to be used as the path-finding algorithm. There are 2 cost functions for the A* algorithm in order to find the shortest path. The first is the cost of getting to the current location, and the second is the estimated cost of getting from the current location to the target. In this thesis, the cost of getting to the location is the length of traveled segments, and the estimated cost to the target is the euclidian distance [16].

5.3.2 CU Placement Results

The final result after running the greedy algorithm then running the GA over the RAU locations was in total 11 CUs for the automatic planning process. The GA attempts to minimize the variance in the number of RAUs covered by a single CU, but it still has a standard divination of 30. Figure 5.13 shows the position of each CU and its assigned RAUs.

From the initial assessment it was clear that the solution had some flaws, such as assigning RAUs on both sides of the Limfjord (water inlet) for the same CU. This can be seen in figure 5.13. When implementing a real network ISPs would not assign RAUs

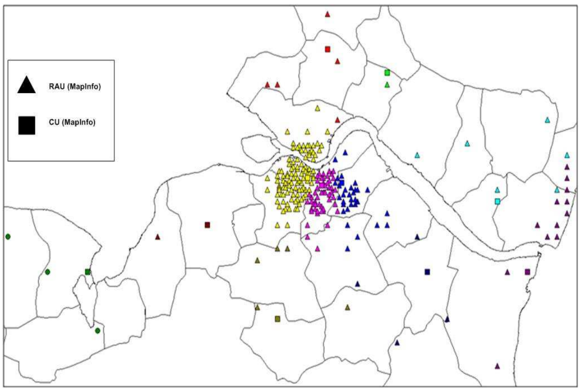

Figure 5.13: Automatic CU Placement

on both sides of the water to a single CU, this however was the case seen in the group of yellow RAUs, which were assigned to the same CU even though they are scattered across the Limfjord. Water cables are very expensive and therefore it would make sense for the ISPs to try to avoid this scenario. It can be also seen that the RAUs could have been distributed better amongst the CUs. Based on this a manual placement scenario was made, where the CUs are manually placed. The manual scenario is presented in order to compare the automatic solution when it comes to fiber planning. The manual solution is basically a visual selection of 11 sites and assigning the RAUs appropriately to them. The idea is to make the RAUs more evenly distributed amongst the CUs. The standard divination of this solution is reduced to 14.5. The manual result can be seen in figure 5.14.

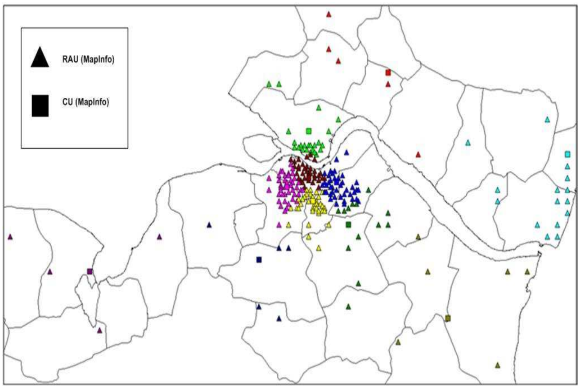

Figure 5.14: Manual CU Placement

5.3.3 Summary

Using a combination of greedy algorithm and GA to solve the set-cover problem provided a solution for the automatic planning. The resulting RAU positions were candidates for CU locations. The distance of CU coverage to an RAU is considered the fiber distance divided by the $\sqrt{2}$. Using path finding algorithms the shortest path from RAU to nearest

CU was found. The thesis started by implementing Dijkstra's shortest-path algorithm, but due to the speed advantage of the A* algorithm, which in a few cases proved to be 11 times faster the choice fell over the A* as the path finding algorithm given that both algorithms provide optimal solutions. Looking at the results of the automatic planning it can be seen that there is still room for improvement as the standard divination of RAUs per CU was found to be high. For this reason a manual solution was implemented which also uses the same number of CUs proposed by the automatic planning. This is done in order for both solutions to be comparable in the upcoming fiber planning process.

5.4 Fiber Planning

There are different ways to setup the optical network between the CU and the RAUs it services. The Futon architecture assumes that an RAU is a simple antenna unit that requires a CU to do all the radio signal processing for it. A CU consists of several JPUs. The function of the JPU is to process radio signals from a group of connected RAUs. The thesis assumes that all JPUs are equipped with the necessary optical interface ports in order to transport radio signaling to and from its corresponding RAUs. ISPs would prefer a passive infrastructure rather than an active infrastructure since this would reduce CapEx and OpEx costs. When deciding on the Optical network there are several connection options. Each option has a different impact on the total network cost. Futon project proposes three different optical network scenarios, all the proposed methods are based on CWDM, since at the time of writing it was more likely that the CWDM architecture will be implemented because of the high cost involved when implementing DWDM.

Table 5.3: Candidate Transmission Architectures

Options	Description
Option A	CWDM mux/demux with two ports for each RAU
Option B	CWDM mux/demux with downstream wavelengths shared by multiple RAUs
Option C	CWDM mux/demux with upstream and downstream in a single CWDM channel

Option A In option A there are 2 dedicated unique wavelengths to service each RAU one for uplink (UL) and one for downlink (DL). There is placed a single mux/demux between the CU/JPUs and the RAUs. Since the thesis considers CWDM there are only 16 wavelengths available, thus since each RAU requires 2 wavelengths for this option a maximum of 8 RAUs can be connected. Figure 5.15 shows option A graphically.

As can be seen from the figure, there is only need for a single fiber cable to connect the JPU to the CWDM mux/demux. For the distribution section between the mux/demux

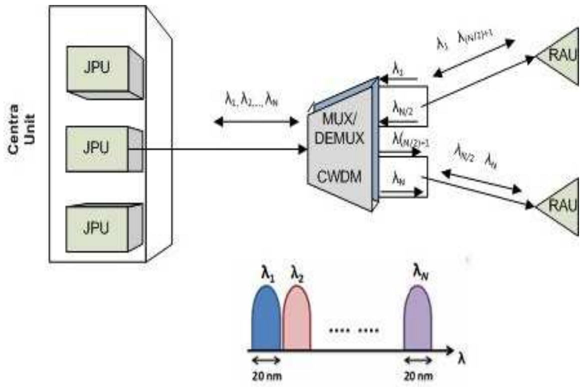

Figure 5.15: Logical overview of CWDM mux/demux with two ports for each RAU.

and RAUs, there are 2 possible implementations . The first implementation is to connect a single fiber to each RAU, but this will require an extra coupler in order to connect UL and DL in the same single fiber. The second implementation requires 2 separate fibers per RAU on for UL and one for DL. while this improves the signal range it has a higher cost since double the fibers are used.

Option B This option is based on option A. Here RAUs get a dedicated UL wavelength, but for DL wavelengths are shared. This can be fulfilled by using Sub-carrier Multiplexing (SCM) in the DL. This option makes it possible to maximize the number of RAUs being serviced by a single JPU. The number of RAUs being supported is dependant on the number of available ports in the CWDM mux/demux. An example of this could be , if we assume that a single wavelength is sufficient to be shared by all RAUs DL, then it is possible to service 15 RAUs with this option [39]. Option B can be seen in figure 5.16.

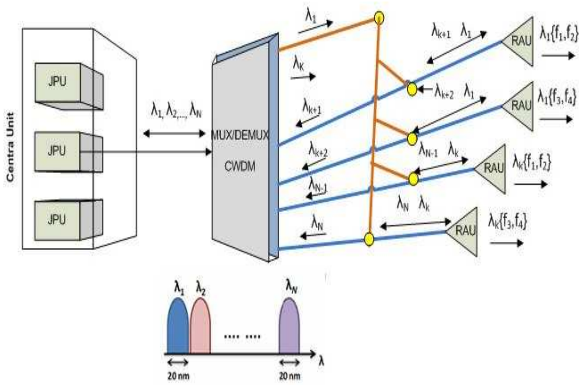

Figure 5.16: Logical overview of CWDM mux/demux with downstream wavelengths shared by multiple RAUs.

Option C This option aims to maximize the number of RAUs per JPU using CDWM technology. This can be achieved by transmitting both DL and UL wavelengths within the 20nm bandwidth for all ports of a mux/demux. Therefore using this option it is possible to connect 16 RAUs to a single JPU (1 RAU to each port of the CWDM mux/demux) [39]. Figure 5.17 shows the physical connectivity of option C.

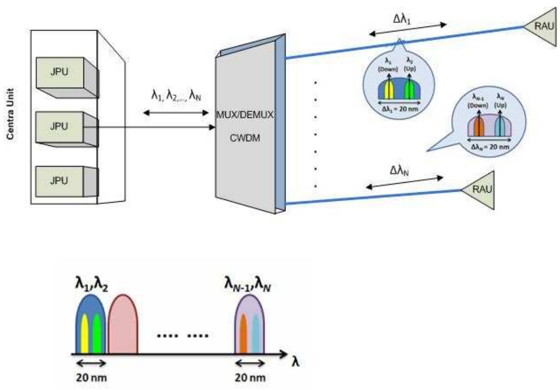

Figure 5.17: Logical overview of CWDM mux/demux with both directions inside the same CWDM channel.

As [39] suggests the length of fiber is restricted to a certain amount of kilometers. It is also specified in [39] that the suitable topology for this kind of networks would be the tree topology, therefore the thesis focuses on connecting each CU to its corresponding RAUs using trees. Although the connection between RAUs and CUs is physically a tree-topology, the logical connection, however, is considered to be point-point, since we are using WDM as can be seen in figure 5.18.

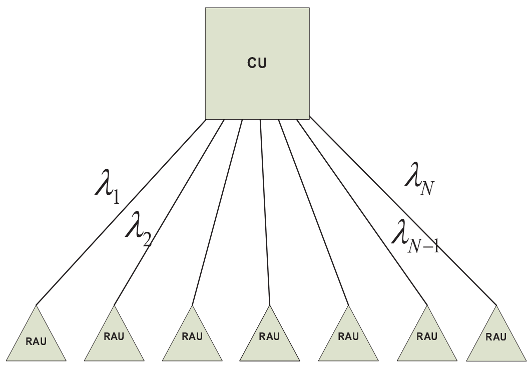

Figure 5.18: CU-RAU Logical Topology

It was decided to use Minimum Spanning Trees (MSTs) when connecting CUs to its corresponding RAUs in order to minimize the number of trenches needed. To accomplish this the thesis implements Prim's MST-algorithm.

Calculating Amount of Fibers Needed For each tree, a calculation of required number of fibers per trench is needed. This is done so that each RAU could have an assigned channel on the fiber. Optimization of fiber assignment is required, in order to minimize the total fiber length needed. A GA is used in order to find the shortest needed fiber length. The GA functions by shuffling the assignment of RAUs to its corresponding CUs, in order to select the solution that results in the shortest fiber length needed.

Implementation Figure 5.19 shows the flow of the RAU to CU connection program. The program starts by loading into memory segment information and locations of the RAUs. The program then goes through all RAUs, in order to locate the RAU locations with CUs. After locating the CUs, the program then starts creating an MST for each CU by adding the closest assigned RAUs to that CU. After the program connects all assigned RAUs, it then calculates the total number of fibers and splitters required by the tree. When the program has investigated all available RAUs it then outputs the segments of the MSTs created, as well as statistical information about fiber and trench lengths and total number of fiber splitters required.

Results The review of the results from the planning of CU to RAU connections are found here. The settings and assumptions that were taken into consideration during the thesis are summarized in the following bullet-points:

No available infrastructure

Optical signals can be amplified where necessary

Optical signals can be split where necessary (for the solution with splitters)

Fiber scenario B (double fiber) with a "range" of 15 km for each CU

CUs placed both automatically and manually

Since the solution with splitters and the solution without splitters are using the same tranches, they therefore are looking visually the same. The difference, however, can be seen in the values from the total number of fibres and the total fiber length used. Table 5.4 shows the results fro both solutions.

Table 5.4: Differences between automatic and manual CU placements.

	Fibers	Splitters	Trench length [m]	Fiber length [m]
Automatic with splitters	26	39	309165	401687*2
Automatic without splitters	79	0	309165	592707*2
Manual with splitters	24	45	319024	364275*2
Manual without splitters	83	0	319024	510095*2

The results show that removing the splitters results in an increase in fiber length equivalent to 47.5% in case of the automatic solution and 40% in case of the manual solution. It can also be noted that there was an increase in the total number of fibers

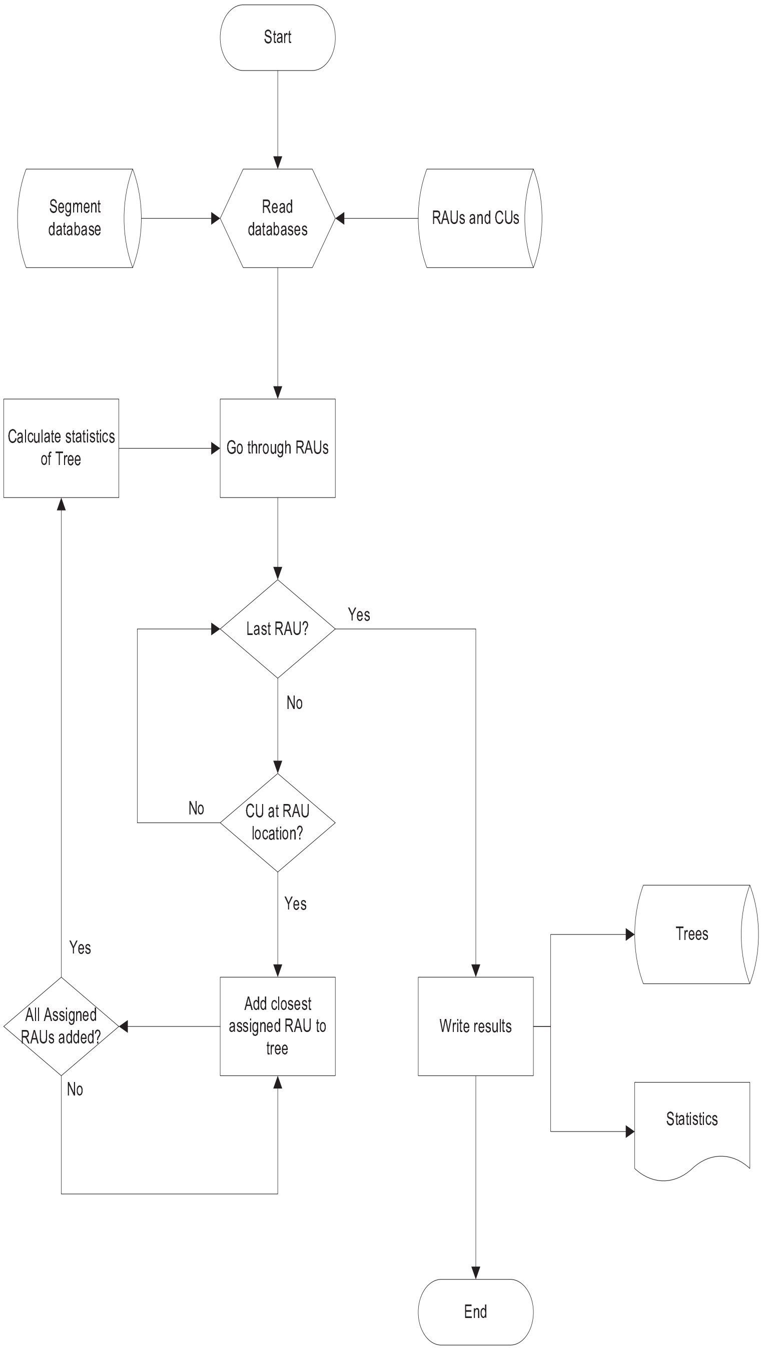

Figure 5.19: Flow of the program planning CU-RAU trees

corresponding to 204% for the automatic solution and 246% for the manual solution. Although the manual solution requires less fibers, it, however, requires slightly longer trenches.

Parts of the MSTs for two of the CUs can be seen in figure 5.20. From the figure it is clear that the program successfully creates MSTs that connects CUs with its assigned RAUs.

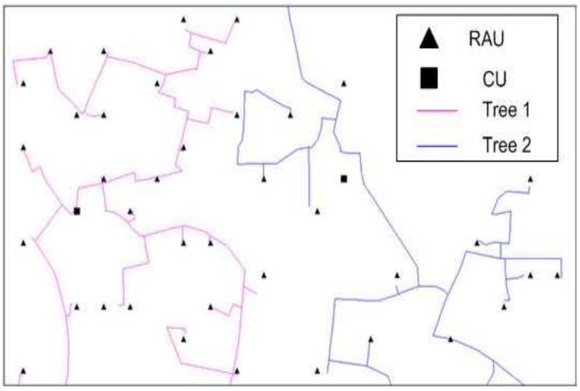

Figure 5.20: Part of the MSTs from the CUs to the RAUs

5.4.1 Connecting CUs

There are many factors to be taken into consideration when deciding on a network topology. According to [31], Structural Quality of Service (SQoS) should be considered for different network with different sizes. Factors such as network size and the different applications and services being provided by the network are a main factor in topology choice. SQoS attempt to improve network performance from a structural point of view, this is achieved by considering factors such as total number of hops, routing schemes, connectivity and network diameter. [32] states that some topologies such as N2R and double ring have better SQoS than single ring topology. In reality, however, single ring topology are till date the most popular and most deployed structure by ISPs, due to its simplicity and cost effectiveness. This is why it was decided to connect the CUs using a ring structure. For the sake of redundancy it is assumed that 2 or more CUs have backbone connectivity, in order to deal with single failures in the network. When it comes to fiber protection method, it was decided to use a 1:1 ring protection i.e. there is an extra unused protection fiber, that becomes active in case of failure. The protection method therefore makes the fiber length twice the trench length.

When connecting the CUs the challenge lies in choosing the ring that will use the least fibers. In graph theory this is known as a Traveling Salesman Problem (TSP) [28]. In this thesis a GA is used to solve this problem. To accomplish this the problem needs to be encoded. In the thesis permutation encoding is used where the CUs are represented as the genes, while the order of visiting the CUs (the tour) is represented as the chromosomes.

This can be seen in figure 5.21. If the start of the CU is 1, the chromosome could look like [1,5,3,2,4] or it could look like [1,4,2,3,5], since it represents the same tour.

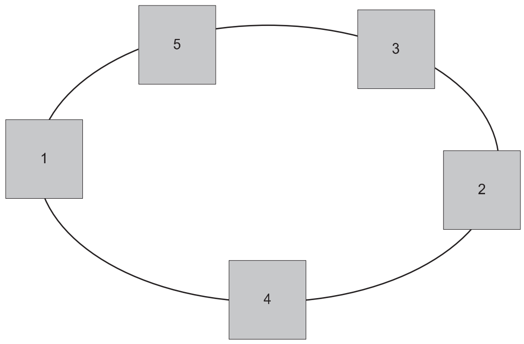

Figure 5.21: CUs interconnected in a ring structure

The flow of the program that connects the CUs can be seen in figure 5.22. First Cu locations and segment information are read into memory, next the paths are calculated from the provided information. The next step is to create a random initial population from the previously created paths, then this population enters into a loop where the GA gets evaluated till the required goal is reached. The program finishes by writing the best possible solution to the database.

Results From table 5.5 it can be seen that both solutions uses almost the same trench length which is about 145 km. The manual solution, however, uses a slightly shorter trench length, which is about 0.9% less than the automatic solution. From the table it can also be seen that the fiber length is twice the trench length, this is due to the extra protection fiber.

Table 5.5: Differences between automatic and manual CU placements.

	Fiber length [m]	Trench length [m]
Automatic	296976	147988
Manual	290492	145246

Figure 5.23 shows the ring that interconnects the automatic solution.

Figure 5.23 shows the ring that interconnects the manual solution.

The program successfully creates a ring that minimizes the fiber length needed to interconnect CUs. The difference in trench length between the manual and automatic solution is almost not noticeable. In real-life, when planning, trenches should be placed on both sides of the road to insure the highest availability.

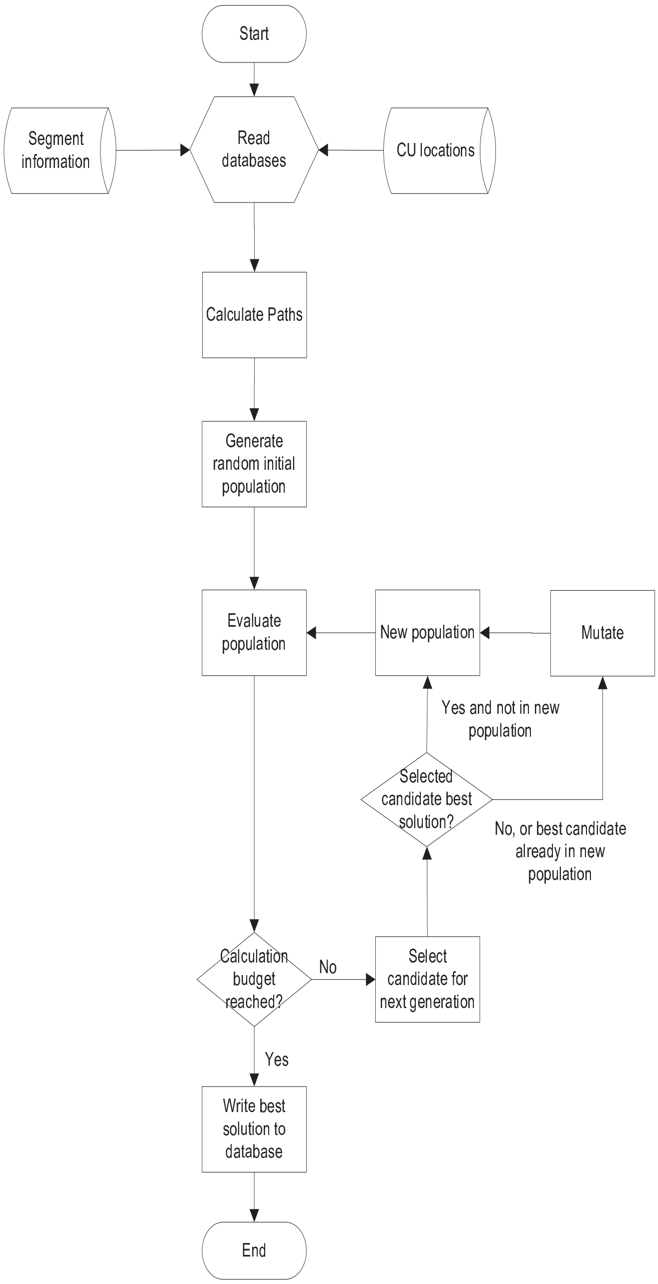

Figure 5.22: Flow of the program planning the CU-CU ring

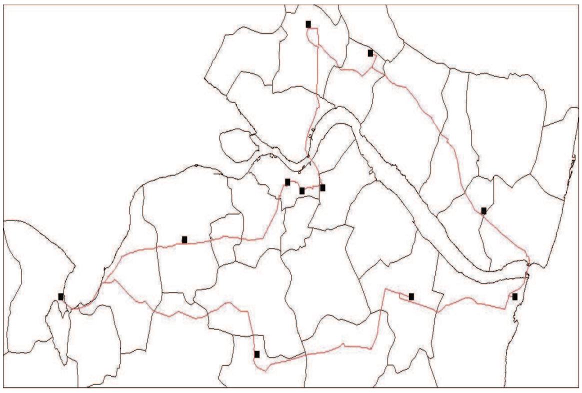

Figure 5.23: CU-ring of automatically placed CUs

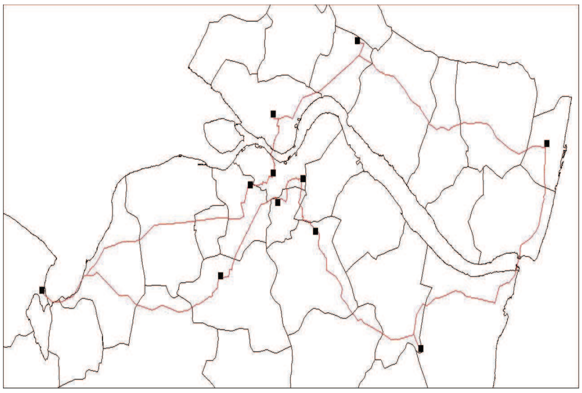

Figure 5.24: CU-ring of manually placed CUs

5.4.2 Summary

The RAUs were connected using a tree topology to the CUs. MST were used in order to minimize the digging needed. A single MST was created for each CU. Two different solutions were presented. One including splitters and one without. It was noted that the solution without splitters resulted in a significantly higher fiber length and total number of fibers. However, this solution might still prove to be cheapest depending on the splitter price and whether or not amplification is required due to splitting. Manually placed Cu solution requires more trench length than that of the automatically placed solution, however it requires less fiber. The cheapest of both solutions therefore, depends on the relationship between cost of trench and cost of fiber. The CUs are then connected using a ring topology. A GA was created and was able to find the optimal solution with the least trenches. Both solutions use almost the same trench-length.

5.5 Deployment Cost Comparison

This section will discuss and introduce the implementation costs for the proposed UMTS network. The focus here is on feasibility and deployment cost. This section will introduce a comparison between typical network planning and RoF network planning. It does not take into account the fact that the ISP may have existing infrastructure deployed, and that it might be cheaper for the ISP to re-locate or reuse available equipment.

The amount of base stations needed for a regular implementation vs the amount of RAU's and CU's needed for the proposed planning architecture are introduced in Table 5.6. For assuming worst case scenario the number of joint processes is set to be as the required RAUs. The minimum number of joint processes is 11 which is the number of CUs with a single joint process.

Table 5.6: Total Number of RAUs/ Base Stations and CUs

	Regular UMTS Implementation	Proposed UMTS implementation
Number of BS/RAU	69	58
Number Of JPs	-	69

According to [12; 43] the average cost of equipments are found in Table 5.7.

Table 5.7: Equipment Costs

Category	Average Price
Regular Base-Station	150,000 USD
Remote Antenna Unit	2,126 USD
Control Unit	3,882 USD

Now that all the prices have been gathered, the following table shows an estimate of the over-all prices for deploying both methods.

Table 5.8: Over All Cost Comparison

	Regular Method	Proposed Method
Base stations	1,035,000 USD	-
Remote Antenna Unit	-	123,308 USD
Control Unit	-	267,858 USD
Total Cost	1,035,000 USD	391,166 USD

The result shows that the proposed method is 60% cheaper to deploy when it comes to wireless equipment assuming worst case scenario, i.e. each RAU requires an individual joint processor at the CU. The average cost of a BS according to [12] is 150,000 USD. In some cases according to [41] base station prices can go up to 400.000 EUR, but by using RoF equipment and by using simple RAUs while applying JPUs at CUs there is a potential cost reduction in CAPEX within this particular scenario.

5.6 Summary

This chapter introduced a new automatic planning model for RoF heterogeneous wireless networks based on the architecture proposed by [54]. The architecture aims at reducing deployment costs by eliminating the expensive costs of wireless base-stations. The chapter proposed several algorithms that are able to solve many planning challenges such as the np-hard problem of RAU locations. The chapter also showed results from the planning of RoF network in the Danish municipality of Aalborg and showed price difference between deploying a regular mobile solution and the proposed RoF architecture.

Part IV

Conclusions and Outlook

Chapter 6

Conclusions and Outlook

This thesis was divided in two main sections, both sections investigate methods to improve quality and reliability of future mobile networks. The first, researched the area of dynamic context information exchange over mobile networks. The thesis investigated different quality assurance algorithms, and the effect they have on context sensitive applications. The second area of optimization was wireless network planning. The second part, investigated how to build RoF heterogeneous wireless networks while taking into account the ISP's currently available legacy systems. The thesis provides algorithms and results for building a RoF network at the Danish municipality of Aalborg using GIS data provided by TDC. The thesis examined different algorithms and developed an automated planning tool that receives GIS data as Input.

For the dynamic context reliability, chapter 3 addressed the Context reliability problem, by presenting live experiment results showing the affect network degradation parameters has on context quality. The chapter also suggested methods to reduce context information degradation. The results of the constructed test bed were investigated and the results for network overhead showed that overhead in small data sizes was rather large and that it's main cause was the number of TCP retransmission, lost TCP segments and duplicate TCP acks. It was also clear the prioritized context traffic resulted in a slightly lower overhead. The mean access delay results stated that for BE traffic a significant increase in access delay can be seen in network utilization between 96%-99%, this is not a surprise however prioritizing CMF traffic gave a degree of normalization and control of access delay in higher utilization. The impact of state-size on mmPr however was more significant in lower congested scenarios. Finally it was shown that taking time to compress the response state has a direct effect on overhead, access delay and mmPr.

The second step was the investigation of the possibility of increasing context information reliability by configuration of access strategies during subscription establishment and by use of QoS classification. Chapter 4 defined and evaluated three different sets of optimization algorithms which are intended to be used as a part of a context QoS control framework. The algorithms use extensions of analytic calculations of mismatch probabilities to scenarios with packet loss, motivated by UDP based context access scenarios. Furthermore, it used a network model to compute the impact of additional context traffic on network performance metrics. The model-based algorithm is executed on the Context-Management-Node (CMN), which is the node that receives all context subscriptions and manages the interaction with the context providers. The algorithm was evaluated for two network models; a simple M/M/1/K bottleneck queuing model and a more complicated network model that uses a convolution of transmission delays and load-dependent queuing delays, in this case for bursty ON/OFF background traffic overlaid to Poisson context traffic. The evaluation results show the effectiveness of the approach. Evaluations of DiffServ like scenarios with two traffic classes allow to quantify the benefit of utilizing dedicated network resources for context traffic.

For cost effective network planning, chapter 5 showed a new automatic planning model for RoF heterogeneous wireless networks based on the architecture proposed by [54]. The architecture aims at reducing deployment costs by eliminating the expensive costs of wireless base-stations. The chapter proposed several algorithms that are able to solve many planning challenges such as the np-hard problem of RAU locations. The chapter also showed results from the planning of RoF network in the Danish municipality of Aalborg, since GIS data was available from TDC. Finally, a cost comparison was made that showed that RoF was 60% cheaper to deploy when it comes to wireless equipment pricing. Below are the different result comparisons generated for the planning tool.

Radio coverage

The radio coverage for all 3 technologies UMTS, WiMAX and LTE-Advanced was made using greedy and genetic algorithms. The idea is to place the RAUs in such a manner so that it would provide maximum coverage to users at the lowest cost. The results for all 3 technologies were made assuming 97% indoor coverage of all MTs. The outdoor coverage, however, is 100%. This is due to the importance of mobility in 4G networks, since users are demanding an always-on service with highest availability. It is worth noting that a more detailed propagation model would benefit the results of the automated planning tool and make them more realistic. Today there are several propagation models available, some are used for indoor planning and some for outdoor. Moreover the Hata model has different variations now not. At the time of writing only information about the Cost 231 HATA model was available, however, today there are various types of city propagation models such as the Young Model, Okumura Model, and Area to Area Lee Model [61].

CU placement

The genetic algorithm was used again to place the CUs. The aim was that each CU would cover the RAUs within its range. The GA was able to provide coverage to all RAUs in the Aalborg municipality, however, the GA does not place the CUs in locations where there are even distributions of the RAUs, therefore a solution with manual CU placement was presented in order to compare it to the automatic solution in terms of fiber planning. For future work, improvements for automatic CU placements should be investigated with the goal of evenly distributing the RAUs amongst the CUs. Linear programming could be considered for future work as a replacement to the greedy algorithm [7].

Fiber planning

The fiber planning process presented results for fiber and trench length and locations. The process is made for both the ring topology that connects the CUs and the MSTs that connect RAUs to CUs. The A* algorithm proved to be faster than Dijkstra's algorithm, given that both provide optimal solutions, therefore, the choice fell on A* as the path finding algorithm. The RAUs were connected to the CUs using MSTs in order to minimize the digging costs. A GA was then used to calculate the required numbers of fibres, splitters, fiber length and trench length.

The result showed that removing the splitters from the CU to RAU trees would have impact on the total fiber length needed, which would lead to a 47.5% increase in fiber length, in case of the automatic solution and a 40% increase for the manual solution. The total number of fibres used would also increase , with values of 204% for the automatic solution and 246% for the manual solution.

The thesis considered a ring topology to interconnect the CUs and RAUs. This is due to the ring topology being widely deployed by ISPs, since it is a simple and cost effective topology with good protection against single failures. The A* algorithm was again used to find the shortest paths between CUs, then a GA is used to find the ring with the least amount of trenches. The results show that for both solutions, the automatic and the manual, the trench length difference is very small.

Finally, a cost comparison has been made which compared wireless equipment costs i.e. a scenario with traditional base-stations being deployed and a scenario with RAUs being deployed. The result showed that the RAU method is 60% cheaper to deploy when it comes to wireless equipment assuming worst case scenario, i.e. each RAU requires an individual joint processor at the CU.

Appendix A: Code For QoC Optimization

Main Function

```

function[nonoptistrat,nonoptigmmpr,nonoptilambda,optistrat,optigmmpr,optilambda]=globalmmprmin(SubNum,lambda,mu,k,tmin,tstep,tmax,rep,Rrate,Erate,Usize,ONbar,OFFbar)
%Input preparation
t=tmin:tstep:tmax;
reqs=200;
strategy=[];
lambda=lambda*1000000/8;
mu=mu*1000000/8;
lamp=1250;
np=lamp*ONbar;
% Code Begins
for i=1:SubNum

    %Utilization Increase Calculation
    [ReqInc EvInc PerInc]=UtilInc(lambda,t,reqs,Rrate(i),Erate(i),Usize(i));

    %for First Subscription
    if isempty(strategy)
```

```
        %Find mmpr vector for all strategies
        [mmpr]=mmprcalc(mu,k,ReqInc,EvInc,PerInc,Rrate(i),Erate(i),t,lamp,np,OFFbar);

        %find strategy with lowst mmpr
        lowest=find(mmpr==min(mmpr));

       [strategy(i) period(i) linc(i) lambda linkglobalmmpr(i)]=minimum(mmpr,lowest,lambda,ReqInc,EvInc,PerInc,t);

        %Comparison variables
        nonoptistrat(i)=strategy(i);
        nonoptiperiod(i)=period(i);
        nonoptigmmpr(i)=linkglobalmmpr(i);
        nonoptilinc(i)=linc(i);
        nonoptilambda(i)=lambda;

        optistrat(i)=strategy(i);
        optiperiod(i)=period(i);
        optigmmpr(i)=linkglobalmmpr(i);
        optilinc(i)=linc(i);
        optilambda(i)=lambda;



    %If Not First Subscription
    else
        %Calculate new Subscription mmPr
        [mmpr]=mmprcalc(mu,k,ReqInc,EvInc,PerInc,Rrate(i),Erate(i),t,lamp,np,OFFbar);
        Inc=[lambda ReqInc EvInc PerInc];
        % Calculate gmmpr vector for all stratgies

        for q=1:length(mmpr)
            comp(q)=linkmmprcalc(i,mmpr(q),Inc(q),mu,k,strategy,period,
```

```
                Rrate,Erate,lamp,np,OFFbar);
        end

        % Find strategy with lowest gmmpr

        lowest=find(comp==min(comp));

        [strategy(i) period(i) linc(i) lambda linkglobalmmpr(i)]=minimum
            (comp,lowest,lambda,ReqInc,EvInc,PerInc,t);

        %saving old values for compariosn
        nonoptistrat(i)=strategy(i);
        nonoptiperiod(i)=period(i);
        nonoptigmmpr(i)=linkglobalmmpr(i);
        nonoptilinc(i)=linc(i);
        nonoptilambda(i)=lambda;

        %Optimization function of current Subscriptions
        [strategy,period,linkglobalmmpr,lambda,linc]=opti(rep,i,linc,
            lambda,mu,k,strategy,t,reqs,period,Rrate,Erate,Usize,lamp,np,
             OFFbar);

        %saving optimized results for comparison
        optistrat(i)=strategy(i);
        optiperiod(i)=period(i);
        optigmmpr(i)=linkglobalmmpr(i);
        optilinc(i)=linc(i);
        optilambda(i)=lambda;



    end
```

```
end
```

Listing 1: Main Function

Overhead Increase Function

```
function[rlinc, elinc, plinc]=UtilInc(lambda,t,reqs,Rrate,Erate,Usize)
%Traffic Increase Calculation

rlinc= lambda + (Rrate *(Usize+reqs));%Increase using  reactive startegy
    .

elinc= lambda + Erate * Usize; %Increase using Event driven strategy.

for j=1:length(t) %Increase using periodic strategy
        plinc(j)= lambda + t(j)*Usize;
end
```

Listing 2: Ovrhead Increase Function

mmPr Calculation

```
function[y]=mmprcalc(mu,k,ReqInc,EvInc,PerInc,Rrate,Erate,t,lamp,np,
    OFFbar)

rejmmpr=1;
x=ReqInc/1000;
mu=mu/1000;
[ReqDelay ReqPloss]=Queue_performance_ON_OFF(k,mu,x,lamp,np, OFFbar);
rmmpr=mmPrConstOffsett(Rrate,1,Erate,1,1,1,1/ReqDelay,200e-3,ReqPloss,0,
    inf,inf); % mmpr calculation for reactive strategy
x=EvInc/1000;
[EvDelay EvPloss]=Queue_performance_ON_OFF(k,mu,x,lamp,np, OFFbar);
emmpr=mmPrConstOffsett(Rrate,1,Erate,1,inf,1,1/EvDelay,200e-3,EvPloss,0,
    inf,inf);


```

```
for j=1:length(t)
    x=PerInc(j)/1000;
    [PerDelay PerPloss]=Queue_performance_ON_OFF(k,mu,x,lamp,np, OFFbar)
        ;
    pmmpr(j)=mmPrConstOffsett(Rrate,1,Erate,1,1/PerDelay,1,1/PerDelay
        ,200e-3,PerPloss,t(j),inf,inf);
end


 y=[rejmmpr rmmpr emmpr  pmmpr];
```

Listing 3: mmPr Calculation

Link GmmPr Function

```
function [linkgmmpr]=linkmmprcalc(n,newmmpr,lambda,mu,k,strategy,period,
    Rrate,Erate,lamp,np, OFFbar)

x=lambda/1000;
mu=mu/1000;
[Delay Ploss]=Queue_performance_ON_OFF(k,mu,x,lamp,np, OFFbar);
z=[];

for i=1:n-1

        if strategy(i)==1
            z(i)=1;
        end

        if strategy(i)==2
            z(i)=mmPrConstOffsett(Rrate(i),1,Erate(i),1,1,1,1/Delay,200e
                -3,Ploss,0,inf,inf);
        end

        if strategy(i)==3
            z(i)=mmPrConstOffsett(Rrate(i),1,Erate(i),1,inf,1,1/Delay
                ,200e-3,Ploss,0,inf,inf);
        end

        if strategy(i)==4

            z(i)=mmPrConstOffsett(Rrate(i),1,Erate(i),1,1/Delay,1,1/
                Delay,200e-3,Ploss,period(i),inf,inf);
        end


end
```

```
linkgmmpr=(sum(z)+newmmpr)/n;
```

Listing 4: GmmPr Function

Strategy Optimization Function

```
function[strategy,period,linkglobalmmpr,lambda,linc]=opti(rep,n,linc,
    lambda,mu,k,strategy,t,reqs,period,Rrate,Erate,Usize,lamp,np, OFFbar)
for q=1:rep
    for i=1:n

        strategy(i)=0;
        lambda=lambda-linc(i); %Remove traffirc of subscription i


        %Calculate Traffic Generated by subscription i

        [ReqInc(1) EvInc(1) PerInc(1,:)]=UtilInc(lambda,t,reqs,Rrate(i),
            Erate(i),Usize(i));

        %Calculate subscription i resulting mmpr

        [mmpr]=mmprcalc(mu,k,ReqInc,EvInc,PerInc,Rrate(i),Erate(i),t,
            lamp,np, OFFbar);

        Inc=[lambda ReqInc EvInc PerInc];



        compute=optilinkmmprcalc(n,mmpr,Inc,mu,k,strategy,period,Rrate,
            Erate,lamp,np, OFFbar);


        lowest=find(compute==min(compute));



```

```
        %calculate new linkglobalmmpr for subscription i


        [strategy(i) period(i) linc(i) lambda linkglobalmmpr(i)]=minimum
            (compute,lowest,lambda,ReqInc,EvInc,PerInc,t);







    end
end
```

Listing 5: Optimization Function

mmPr Function

Constant Offset

```
function[mmPr,nt,ad]=mmPrConstOffsett(R,pe,Be,pu,Bu,pd,Bd,d0,ploss,tau,pc,Bc)
%This function adds an offset to the given input delays. It assumes this
%offset delay is exp. distribution, and adds itself as an addition chain in
%an ME model. It has the following interface:
%
% [mmPr,nt,ad]=mmPrConstOffsett(R,pe,Be,pu,Bu,pd,Bd,d0,ploss,tau,pc,Bc)
%
% R: Request rate, <pe,Be> ME vector/mtx. of event process
% <pu,Bu> and <pd, Bd> ME vector/mtx. of upstream and downstream delays
% - NB! if Bu<inf script assumes reactive strategy
% - NB! For exp. case, set the px=1, and Bx are then rates in [s^-1]
% d0: the offset (here assumed exp. distributed, i.e. the mean delay [s])
% ploss: message loss probability
% tau: update rate [updts/sec]
% - NB!if tau>0 script assumes periodic update
% <pc,Bc>: cache period
% - NB! if not applied, set Bc to inf (if exp., then this is a rate [s^-1])
%
%The script assumes that existing mmPr scripts, mmPrPeriodic, mmPrRea and
%mmPrProEvnFull is present in the same working directory.
%
%The script differentiates the different methods by


dl=length(pd);
```

```
nDelayD(2:(dl+1),2:(dl+1))=Bd;     %Shift the input delay to accommodate an intial delay in the ME description
nDelayD(1,1)=1/d0; nDelayD(1,2)=-1/d0; %Adding the extra delay state in the total delay
pD(1,2:(dl+1))=0;                      %Most entries shall be zero
pD(1)=1;                               %Except the first one


if (tau>0) %Then we assume periodic case
    [mmPr,nt,ad] = mmPrPeriodic(R,pe,Be,pu,Bu,pD,nDelayD,tau,ploss);
elseif (Bu<inf) %If there is an upstream delay given, then we assume reactive strategy
    nDelayU(2:(dl+1),2:(dl+1))=Bd;     %For the reactive, actually, there is also the upstream delay offset
    nDelayU(1,1)=1/d0; nDelay(1,2)=-1/d0;
    pU(1,2:(dl+1))=0;
    pU(1)=1;
    [mmPr,nt,ad]=mmPrRea(R,pe,Be,pU,nDelayU,pD,nDelayD,pc,Bc,0,inf,ploss);
else
    [mmPr,nt,ad]=mmPrProEvnFull(pe,Be,pD,nDelayD,ploss); %Otherwise, assume event driven updates
end
```

Listing 6: Constant Offset

Reactive mmPr

```
function[mmPr,nt,ad]=mmPrRea(mu,pe,Be,pu,Bu,pd,Bd,pc,Bc,d0,T0,ploss)
Ebar=sum(pe/Be);
Ubar=sum(pu/Bu);
Dbar=sum(pd/Bd);
%T0=T0-sum(pu/Bu);

if(Bc(1)<inf)   %If caching is being used
    Cbar=sum(pc/Bc);
    Lbar=1/mu+Ubar+Dbar+Cbar;

    pi1=(1/mu)/Lbar;
    pi2=Ubar/Lbar;
    pi3=Dbar/Lbar;
    pi4=Cbar/Lbar;

    bc_pd=recp(pd,Bd);
    bc_pc=recp(pc,Bc);

    [cv_pd, cv_Bd]=convoluteSpecial(bc_pd,Bd);
    [cv_pc, cv_Bc]=convolute(pd,Bd,bc_pc,Bc);

    mmPr=(pi1+pi2)*baseRea(pe,Be,pd,Bd,pu,Bu,d0,T0,ploss)+pi3*baseRea(pe
        ,Be,cv_pd,cv_Bd,pu,Bu,d0,T0,ploss)+pi4*baseRea(pe,Be,cv_pc,cv_Bc,
        pu,Bu,d0,T0,ploss);

    nt=2/Lbar;
    ad=pi1*(Ubar+Dbar)+pi2*(pu*pinv(Bu)^2*ones(size(Bu,1),1)/Ubar + Dbar
        )+pi3*(pd*pinv(Bd)^2*ones(size(Bd,1),1)/Dbar);

else
    mmPr=baseRea(pe,Be,pd,Bd,pu,Bu,d0,T0,ploss);
```

```
    nt=2*mu;
    ad=Ubar+Dbar;
end


function[mmPr_basis]=baseRea(PE,BE,PD,BD,PU,BU,d0,T0,ploss)

p_prod = kron(PE/BE,PD*BD);
ksum_term=(kron(BE,eye(size(BD))) + kron(eye(size(BE)),BD)) \ ones(size(BE,1)*size(BD,1),1);
mmPr_basis=1-(1-ploss).^2*p_prod*exp(-BE*d0)*ksum_term/sum(PE/BE);


function[p,C]=convolute(pa,A,pb,B)
p=[pa, zeros(1,length(pb))];
C=[A, -sum(A,2)*pb;zeros(length(pb),length(pa)),B];

function[p]=recp(pa,A)
Abar=sum(pa/A);
p=pa*inv(A)/Abar;



function[p,C]=convoluteSpecial(pa,A)
%The downstream period requires a special convolution as the resulting distribution is NOT a phasetype and requires a slight different approach

p=[pa, zeros(1,length(A))];
C=[A, -A;zeros(length(A)), A];
```

Listing 7: Reactive mmPr

Event Driven mmPr

```
function[mmPr,nt,ad]=mmPrProEvnFull(pe,Be,pd,Bd, ploss)

kron_prod=kron(pd,pe);
kron_sum=inv(kron(Be,eye(size(Bd,1)))+kron(eye(size(Be,1)),Bd))*ones(
    size(Be,1)*size(Bd,1),1);


n = pe*inv(Be)*ones(size(Be,1));
t=kron(pe*inv(Be),pd*Bd);

mmPr=1-(1-ploss)*(t/n)*kron_sum;
nt=sum(pe/Be);

ad=0;
```

Listing 8: Event Driven mmPr

Periodic mmPr

```
function[mmPr,nt,ad]=mmPrProEvnFull(pe,Be,pd,Bd, ploss)

kron_prod=kron(pd,pe);
kron_sum=inv(kron(Be,eye(size(Bd,1)))+kron(eye(size(Be,1)),Bd))*ones(
    size(Be,1)*size(Bd,1),1);


n = pe*inv(Be)*ones(size(Be,1));
t=kron(pe*inv(Be),pd*Bd);

mmPr=1-(1-ploss)*(t/n)*kron_sum;
nt=sum(pe/Be);

ad=0;
```

Listing 9: periodic mmPr

Network Models

M/M/1/K

```
function [Ploss,ET] = mm1k(lambda, mu, buffer)

k = buffer;
rho = lambda/mu;


if(rho == 1)
    Ploss = 1/(k+1);
    EN = k/2;                                                    % rho == 1
    Idle = Ploss;
else
    Ploss = ((1-rho)*rho^k)/(1-rho^(k+1))  ;                     %rho != 1
    Idle = (1-rho)/(1-rho^(k+1));
    EN = (rho/(1-rho)) - (k+1)*(rho^(k+1))/(1-rho^(k+1));
end


Util = 1 - Idle;
Throughput=lambda*(1-Ploss);
ET = EN/Throughput;
```

Listing 10: M/M/1/K

MMPP/M/1/K

```
%  [rho,S_bar,snd_S, qbar, snd_q, qa_bar, snd_qa, qc_bar, snd_qc, Ploss] = ...
%  MMPP_M_1_K(K_vect, ql_in, nu, Bcal, Lcal);
%  Analysis of finite MMPP/M/1/m-System for rho<1 using spectral decomposition
%    K_vect(j):  vector of Buffer sizes, m<0 for infinite buffer system;
%    ql_in: vector of queuelengths to calculate probabilities of
%    nu:  service rate nu used
%    Qcal,Lcal: MMPP
%  Output:
%    actual used rho and nu.
%    qbar(j), snd_q(j), qa_bar(j),..., snd_qc(j): first and second moments
%          of q.l.d./q.l.d. at arrival-times/ q.l.d. at arrival times
%          of accepted cells
%    S_bar, snd_S: first two moments of system-times of cells not counting
%             lost cells
%    p_ov(j): Cell-loss probability
%    qbar(j): mean queue-length
%    pi_hat(j,i): scalar queue-length probs. at arbitrary times

function [rho,S_bar, snd_S, qbar, snd_q, qa_bar, snd_qa, qc_bar, snd_qc, Ploss] = ...
                      MMPP_1_1_K(K_vect, nu, Qcal,Lcal);

 y=[]; U=[]; lamR=[]; lamiS=[]; US=[];

Bcal=Lcal-Qcal;

S=size(Bcal,1);
```

```
pi= [zeros(1,S), 1] / [Qcal, ones(S,1)];

E1=1/full(pi*sum(Lcal,2));

rho=1/(nu*E1);

if rho>=1
  error( Queue has to have utilization smaller than 1 ! );
end;

A0=Lcal; A1=-Bcal- nu*speye(S); A2=nu*speye(S);
UR=[];y=[]; US=[]; lamiS=[]; lamR=[];

[UR,lamR US, lamiS]=Rsm_spd(A0,A1,A2);
y=pi/UR;

lamS=1./lamiS;
US(1,:)=pi; % pi is right EVect of S for EV 1
% [lUR,uUR]=lu(UR); [lUS,uUS]=lu(US);

URe=sum(UR,2);
USe=sum(US,2);

URLe=UR*sum(Lcal,2);
USLe=US*sum(Lcal,2);

  WS=inv(US); WR=inv(UR);
  KR1=spdiags(lamR. ,0,S,S) * UR * A2;
  KS1=spdiags(lamS. ,0,S,S) * US * A0;

  H2 = [real(A1+A2+WR*KR1), zeros(S,S), zeros(S,1); ...
          zeros(S,S),          real(WS*KS1+A1+A0), zeros(S,1)];
      %% upper left corner: A1+A2+R*A2, lower right corner: S*A0+A1+A0
      %% rest dependend on m: R^m*(A0-R*A2)  bzw. S^m*(A2-S*A0),
      %% and normalization equation
```

```
  KR2 = UR*A0-KR1;  %% part of R^m(A0-R*A2), for upper right corner of H2
  KS2 = US*A2-KS1;  %% part of S^m(A2-S*A0), for lower left corner of H2
  WSred=WS(:,2:S); USred=US(2:S,:); lamSred=lamS(2:S); % erase eigenvalue 1
  USered=USe(2:S); USLered=USLe(2:S);

j1=prod(size(K_vect));
qbar=zeros(1,j1); snd_q=zeros(1,j1);
qa_bar=zeros(1,j1); snd_qa=zeros(1,j1);
qc_bar=zeros(1,j1); snd_qc=zeros(1,j1);
S_bar=zeros(1,j1); snd_S=zeros(1,j1);
Ploss=zeros(1,j1);
x0WRUe=zeros(S,j1); xmWSUe=zeros(S,j1);


for j=1:j1,
 m=K_vect(j);
%fprintf(1, Finite buffer m=%d CPU-time: %3.1f\n ,m,cputime-t0);
  H2(1:S,S+1:2*S) = real(WR*spdiags((lamR.^m). ,0,S,S)*KR2);
  H2(S+1:2*S,1:S) = real(WS*spdiags((lamS.^m). ,0,S,S)*KS2);

  Rsum_e = real(WR*spdiags(((1-lamR.^(m+1))./(1-lamR)). ,0,S,S)*URe). ;
  Ssum_e = real((m+1)*WS(:,1)+WSred*spdiags(((1-lamSred.^(m+1))./(1-lamSred)). ,0,S-1,S-1)*USered). ;

  H2(:,2*S+1) = [Rsum_e. ;Ssum_e. ];
%% determine x0 and xm of pi_hat(k)=x0*R^k + xm*S^(m-k)
  h= [zeros(1,2*S), 1]/H2; % H2 is not sparse, therefore GMRES of no help !
  x0= h(1:S); xm= h(S+1:2*S);

  x0WR= x0*WR;
  xmWS= xm*WS; xmWS(1)=0;
  xmWSred=xmWS(2:S);
```

```
x0WRUe(:,j)=(x0WR.*URe. ). ;
xmWSUe(:,j)=(xmWS.*USe. ). ;


Ploss(j)= E1*(real(x0WR*spdiags((lamR.^(m)). ,0,S,S)*URLe)+xm*sum(Lcal,2));
qbar(j) = real(x0WR * spdiags( (1./(1-lamR) .* ...
                ((1-lamR.^(m+2))./(1-lamR)-(m+1)*lamR.^(m+1)-1)). ,0,S,S)...
               * URe) ...
        - real(xmWS(2:S) * spdiags((1./(1-lamSred) .*  ...
                ((1-lamSred.^(m+1))./(1-lamSred)-(m+1))). ,0,S-1,S-1) ...
               * USe(2:S));
qa_bar(j) = E1*( real(x0WR * spdiags( (1./(1-lamR) .* ...
                ( (1-lamR.^(m+2))./(1-lamR) - ...
                  (m+1)*lamR.^(m+1) - 1 )). ,0,S,S)...
               * URLe) ...
        - real(xmWS(2:S) * spdiags((1./(1-lamSred) .*  ...
                ((1-lamSred.^(m+1))./(1-lamSred)-(m+1))). ,0,S-1,S-1) ...
               * USLe(2:S)) );

snd_q(j)=real(x0WR * spdiags( ( 1./(1-lamR) .* ...
                ( (1-(m+1)^2*lamR.^(m+1)) + ...
                  2*(1-lamR.^(m+3)) ./ (1-lamR).^2 - ...
                  (3+(2*m+3)*lamR.^(m+2)) ./ (1-lamR) )). ,0,S,S)...
              * URe) ...
        + real(xmWS(2:S) * spdiags((1./(1-lamSred) .*  ...
                ( 2*(1-lamSred.^(m+2)) ./ (1-lamSred).^2 -...
                  (2*m+3+lamSred.^(m+1)) ./ (1-lamSred) +...
                  (m+1)^2 ) ).  ,0,S-1,S-1)...
               * USe(2:S));
snd_qa(j)=E1*( real(x0WR * spdiags( ( 1./(1-lamR) .* ...
```

```
                        ( (1-(m+1)^2*lamR.^(m+1)) + ...
                          2*(1-lamR.^(m+3)) ./ (1-lamR).^2 - ...
                          (3+(2*m+3)*lamR.^(m+2)) ./ (1-lamR) )). ,0,S,S)...
                      * URLe) ...
                + real(xmWS(2:S) * spdiags((1./(1-lamSred) .*  ...
                        ( 2*(1-lamSred.^(m+2)) ./ (1-lamSred).^2 -...
                          (2*m+3+lamSred.^(m+1)) ./ (1-lamSred) +...
                          (m+1)^2 ) ).  ,0,S-1,S-1)...
                       * USLe(2:S)) );
% Performance Parameters
   qc_bar(j)= (qa_bar(j)-m*Ploss(j))/(1-Ploss(j));
   snd_qc(j) = (snd_qa(j)-m^2*Ploss(j))/(1-Ploss(j));
   S_bar(j) = (qc_bar(j)+1)/nu;
   snd_S(j) = (snd_qc(j)+3*qc_bar(j)+2)/nu^2;

end; % j
```

Listing 11: MMPP/M/1/K

Appendix B: Code For Cost Effective RoF Network Planning

Grid Maker

```
#Small script to make a grid of possible RAU-locations

import matplotlib.pylab as plot
import sqlite3 as db
from math import sqrt
from odbc import *
import dbi

MTdb = r D:\MT.db3  #MTs

wrange = 300 #if there are no MTs in this range, it is removed from the grid
distance = 1000 #distance between the potential RAUs
startx = -268500 #x-coordinate of start location
starty = 309500 #y-coordinate of start location
stopx = -210500 #x-coordinate of stop location
stopy = 271500 #y-coordinate of stop location

class MT:
    """Class for the Mobile Terminals"""
    def __init__(self, x, y, id):
```

```
        self.x = x
        self.y = y
        self.id = id

    def findDNsinrange(self,DN_lst):
    """Finds the number of RAUs in range of the MT"""
        for count in range(len(DN_lst)):
            eucledian_distance = sqrt(((self.x-DN_lst[count].x)**2)+((
               self.y-DN_lst[count].y)**2))
            if eucledian_distance < DN_lst[count].range:
                return self.id
        return

#PREPARATIONS --------------------------------------------------
#Establish a connection to the database
conn = db.connect(MTdb)
c = conn.cursor()

c.execute( SELECT * FROM MT )
MT_lst = []
i = 0

for row in c: #Go through table create new instances of the MT class
    id = row[0]
    x = row[1]
    y = row[2]
    MT_lst.append(MT(x, y, id)) #Adds the new instance to a list
    i += 1



#Generate a grid from startx,starty to stopx,stopy
x = []
y = []
newy = starty
```

```
while newy > stopy:
    newx = startx
    while newx < stopx:
        x.append(newx)
        y.append(newy)
        newx += distance
    newy -= distance





#Loops to check if there are MTs around the possible RAU-location. Removes it if none is found.
connected = [0]*len(x)
for i in xrange(len(x)):
    if i%100 == 1:
        print i
    for j in xrange(len(MT_lst)):
        eucledian_distance = sqrt(((x[i]-MT_lst[j].x)**2)+((y[i]-MT_lst[j].y)**2))
        if eucledian_distance < wrange:
            connected[i] += 1
            break
xlist = []
ylist = []
for i in xrange(len(x)):
    if connected[i] > 0:
        xlist.append(x[i])
        ylist.append(y[i])


plot.scatter(xlist,ylist)
plot.show()

```

```

#Write the results to a table
conn = odbc( DN_output )
c = conn.cursor()


try:
    #Create the table
    c.execute(
    CREATE TABLE grid (
    id INTEGER,
    x FLOAT,
    y FLOAT,
    radius FLOAT
    )
        )
except dbi.progError:
    c.execute( DROP TABLE grid )
    #Create the table
    c.execute(
    CREATE TABLE grid (
    id INTEGER,
    x FLOAT,
    y FLOAT,
    radius FLOAT
    )
        )

for i in range(len(xlist)):
    xval = xlist[i]
    yval = ylist[i]
    radius = 0.0
    t = (i,xval,yval,radius)
    c.execute( INSERT INTO grid VALUES (?,?,?,?) , t)
```

Listing 12: Grid Maker

Segment Manipulation

```
from odbc import *
from math import *
import sqlite3 as db
import matplotlib.pylab as plot
import os
from math import sqrt, floor, ceil, pi, log10
from random import uniform, random
from sets import Set
from odbc import *
import dbi
aalborg = r C:\Documents and Settings\Ahmed Shawky\My Documents\python files\Segment_tables.mdb  #the location of the database file
grid= r C:\Documents and Settings\Ahmed Shawky\My Documents\python files\input
conn2= odbc( input )
f=conn2.cursor()
conn = odbc( segment )
c = conn.cursor()
class S: #The segments
    def __init__(self, name , x1, x2, y1, y2, id1, sid, id2, length, id):
        self.name = name
        self.x1 = x1
        self.x2 = x2
        self.y1 = y1
        self.y2 = y2
        self.id1 = id1
        self.sid = sid
        self.id2 = id2
        self.length = length
        self.id = id
```

```
class SP: #The segment points
    def __init__(self, x, y, spid):
        self.x = x
        self.y = y
        self.spid = spid
        self.degree = 1

    def update_degree(self):
        self.degree += 1
def read_databases():


    c.execute( SELECT * FROM S ) #SQL to select all from the NT table
    S_lst = []
    i = 0

    results = c.fetchall()
    for row in results: #Go through table and save the stuff we need
        x1 = row[0]
        x2 = row[1]
        y1 = row[2]
        y2 = row[4]
        id1 = row[5]
        sid = row[6]
        id2 = row[7]
        length = row[9]
        id = row[10]
        S_lst.append(S( s_  + str(i), x1, x2, y1, y2, id1, sid, id2,
            length, id)) #creates a new class-instance for the DN.
        i += 1
    print i,  selected

    c.close()
    return S_lst
def create_SP(S_lst):
```

```
    try:

        c.execute(
        CREATE TABLE SP (
        id INTEGER,
        x FLOAT,
        y FLOAT,
        spid FLOAT,
        degree INTEGER
        )
           )
    except dbi.progError:
        c.execute( DROP TABLE SP )
        c.execute(
        CREATE TABLE SP (
        id INTEGER,
        x FLOAT,
        y FLOAT,
        spid FLOAT,
        degree INTEGER
        )
           )
#
    SP_lst = []
    count = 0
    for i in range(len(S_lst)):
        id1 = S_lst[i].id1
        id2 = S_lst[i].id2
        found1 = 0
        found2 = 0
        for j in range(len(SP_lst)):
            if SP_lst[j].spid == id1:
```

```
                found1 = 1
                SP_lst[j].update_degree()
                t = (SP_lst[j].degree,id1)
                c.execute( UPDATE SP SET degree = (?) WHERE spid = (?) , t)

            if SP_lst[j].spid == id2:
                found2 = 1
                SP_lst[j].update_degree()
                t = (SP_lst[j].degree,id2)
                c.execute( UPDATE SP SET degree = (?) WHERE spid = (?) , t)
            if found1 and found2:
                break

        if not found1:
            SP_lst.append(SP(S_lst[i].x1,S_lst[i].y1,S_lst[i].id1))
            t = (count,S_lst[i].x1,S_lst[i].y1,S_lst[i].id1,1)
            c.execute( INSERT INTO SP (id,x,y,spid,degree) values (%d,%f,%f,%f,%d) %(t[0],t[1],t[2],t[3],t[4]))
            count += 1
        if not found2:
            SP_lst.append(SP(S_lst[i].x2,S_lst[i].y2,S_lst[i].id2))
            t = (count,S_lst[i].x2,S_lst[i].y2,S_lst[i].id2,1)
            c.execute( INSERT INTO SP (id,x,y,spid,degree) values (%d,%f,%f,%f,%d) %(t[0],t[1],t[2],t[3],t[4]))
            count += 1
    conn.commit()
    return SP_lst


f.execute( ALTER TABLE fixed_grid ADD spid int; )     # adds spid column to fixed_grid table
f.execute( ALTER TABLE fixed_grid ADD distance int; ) # adds distance column to fixed_grid table
```

```
c.execute( ALTER TABLE SP ADD flag int )                # adds flag column to SP table

v=0                                                     # counter for fixed_grid table
f.execute( select count(*) from  fixed_grid )
result=f.fetchall()
row= result[0][0]
while v<=row:                                           # goes through each record in fixed_grid
    #print  v ,v
    f.execute( select x,y from fixed_grid where counter=%d %(v))  # selects x and y for current record in fixed_grid
    res=f.fetchall()
    if not res:
        v=v+1                  # if record not found then increment counter
    else:
        x1= res[0][0]
        y1= res[0][1]
        j=0 # second counter for SP table
        c.execute( select count(*) from SP )
        res3=c.fetchall()
        row2= res3[0][0]
        l=100000000 # initial length in meters
        while j<row2: #goes through each record in SP table
            c.execute( select x,y from SP where id=%d %(j)) #selects x and y for current record in SP
            res4=c.fetchall()
            x2=res4[0][0]
            y2=res4[0][1]
            # Euclidean distance calculation
            x=x1-x2
            y=y1-y2
            distance= sqrt(x**2+ y**2)
```

```
            c.execute( select spid from SP where id=%d %(j)) # selects segemnt point id from SP table for current record
            res4=c.fetchall()
            if l > distance: #makes sure to have the shortest distnace kept in l
                l=distance
                spid=res4[0][0]
            j=j+1 #increment counter
        f.execute( update fixed_grid set spid=%d,distance=%d where counter=%d %(spid,l,v))# updates table fixe_grid with closest segment point id and distance
        c.execute( update SP set flag=1 where spid=%d %(spid)) #updates segemnt point table by placing a flag next to the closest segemnt point id
        v=v+1 #increment counter

i=1 # counter for segment  manipulation
c.execute( select count(*) from  s )
result=c.fetchall()
rows= result[0][0]
while i<=rows: # goes through each record in S table
    s=1
    while s:
        s=0
        t=(i,)
        c.execute( select sp_id2, sp_id1 from S where id=(?) , t) # selects the two segemnet points forming the current segement
        res1=c.fetchall()
        if not res1: # if record not found break and increment counter
            break
        else:
            spid2=res1[0][0]
            spid1=res1[0][1]


```

```
            c.execute( select flag from SP where spid=%d %(spid1))# fetch flag for first segment point
            result=c.fetchall()
            flag=result[0][0]
            c.execute( select flag from SP where spid=%d %(spid2))# fetches flag for secon segment point
            result=c.fetchall()
            flag2=result[0][0]
            if flag==1: # if first flag and second flag are found then break and iincrement
                c.execute( select flag from SP where spid=%d %(spid2))
                result2=c.fetchall()
                flag2=result[0][0]
                if flag2==1:
                    break
                else: #if only first flag is found then perform segment manipulation on second segment point
                    a=(spid2,)
                    c.execute( select degree from SP where spid=(?) , a) # selects the degree for second segemtn point
                    res1=c.fetchall()
                    degree=res1[0][0]
                    b=(spid1,)
                    c.execute( select degree from SP where spid=(?) , b) #selects degree for first segemtn points
                    res1=c.fetchall()
                    degree2=res1[0][0]
                    if degree<>2 and degree2<>2:# increment if bothe degrees are not equal to 2
                        s=0
                    if degree==1 and degree2==1: # if both degrees are equal to 1 delete form table
                        c.execute( delete from table S where ID=%d %(i))
                        break
                    if degree==2: # degree of the second segment point
```

```
is equal to 2
s=1 # keep looping
c.execute( select sp_id2 from S where sp_id1=%d %(spid2)) # checks second segment where SP_ID2 is found as SP_ID1
res1=c.fetchall()
if not res1: # if record not found checks second record where SP_ID2 is found as SP_ID2
    c.execute( select sp_id1 from S where sp_id2 =%d and ID<>%i %(spid2,i))
    res2=c.fetchall()
    if not res2:
        break # if both are not found then break loop and increment counter
    newspid=res2[0][0] # holds first segment point in the record where SP_ID2 was found as SP_ID2
    c.execute( select sp_x1,sp_y1 from S where sp_id2=%d and ID<>%i %(spid2,i)) # selects x and y co-ordinates for the new SPID
    res2=c.fetchall()
    spx=res2[0][0] #holds new x co-ordinate
    spy=res2[0][1] #holds new y co-ordinate
else:
    newspid=res1[0][0] #holds second segment point in the record where SP_ID2 was found as SP_ID1
    c.execute( select sp_x2,sp_y2 from S where sp_id1=%d %(spid2)) # selects x and y co-ordinates for the new SPID
    res1=c.fetchall()
    spx=res1[0][0] #holds new x co-ordinate
    spy=res1[0][1] #holds new y co-ordinate

```

```
c.execute( select lenght from S where id=(?) , t ) # selects lenght for current record
res2=c.fetchall()
l1=res2[0][0]
c.execute( select lenght from S where sp_id1=%d and sp_id2=%d %(newspid,spid2)) # select length from next record where SP_ID2 is found as SP_ID2
res1=c.fetchall()
if not res1:
    c.execute( select lenght from S where sp_id2=%d and sp_id1=%d %(newspid,spid2)) # select length from next record where SP_ID2 is found as SP_ID1
    res2=c.fetchall()
    l2=res2[0][0]
    c.execute( delete from S where sp_id2=%d and sp_id1=%d %(newspid,spid2))#deletes record  where SP_ID2 is found as SP_ID1

else:
    l2=res1[0][0]
    c.execute( delete from S where sp_id1=%d and sp_id2=%d %(newspid,spid2))# deletes record where SP_ID2 is found as SP_ID1
lnew=l1+l2 # claculates new length
c.execute( update S set sp_id2=%d where id=%i  %(newspid,i))
c.execute( update S set sp_x2=%d where id=%i  %(spx,i))
c.execute( update S set sp_y2=%d where id=%i  %(spy,i))
c.execute( update S set lenght=%d where id=%i  %(lnew,i)) # up dates table with new degment point ID, Co-ordinates and length
```

```
            elif flag2==1: # same operation as before in reverse
                c.execute( select flag from SP where spid=%d %(spid1))
                result2=c.fetchall()
                flag2=result2[0][0]
                if flag2==1:
                    break
                else:
                    a=(spid2,)
                    c.execute( select degree from SP where spid=(?) , a)
                    res1=c.fetchall()
                    degree=res1[0][0]
                    b=(spid1,)
                    c.execute( select degree from SP where spid=(?) , b)
                    res1=c.fetchall()
                    degree2=res1[0][0]
                    if degree<>2 and degree2<>2:
                        s=0
                    if degree==1 and degree2==1:
                        c.execute( delete from table S where ID=%d %(i))
                        break

                    if degree2==2:
                        s=1
                        c.execute( select sp_id1 from S where sp_id2=%d %(spid1))
                        res1=c.fetchall()
                        if not res1:
                            c.execute( select sp_id2 from S where sp_id1=%d and ID<>%i %(spid1,i))
                            res2=c.fetchall()
```

```
    if not res2:
        break
    newspid=res2[0][0]
    c.execute( select sp_x2,sp_y2 from S where sp_id1=%d and ID<>%i %(spid1,i))
    res2=c.fetchall()
    spx=res2[0][0]
    spy=res2[0][1]
else:
    newspid=res1[0][0]
    c.execute( select sp_x1,sp_y1 from S where sp_id2=%d %(spid1))
    res1=c.fetchall()
    spx=res1[0][0]
    spy=res1[0][1]

c.execute( select lenght from S where id=(?) , t )
res2=c.fetchall()
l1=res2[0][0]
c.execute( select lenght from S where sp_id2=%d and sp_id1=%d %(newspid,spid1))
res1=c.fetchall()
if not res1:
    c.execute( select lenght from S where sp_id1=%d and sp_id2=%d %(newspid,spid1))
    res2=c.fetchall()
    l2=res2[0][0]
    c.execute( delete from S where sp_id1=%d and sp_id2=%d %(newspid,spid1))
else:
    l2=res1[0][0]
    c.execute( delete from S where sp_id2=%d and sp_id1=%d %(newspid,spid1))
lnew=l1+l2
```

```
            c.execute( update S set sp_id1=%d where id=%i  % (newspid,i))
            c.execute( update S set sp_x1=%d where id=%i  % (spx,i))
            c.execute( update S set sp_y1=%d where id=%i  % (spy,i))
            c.execute( update S set lenght=%d where id=%i  % (lnew,i))
else: # combine both operations if no flags are found
    a=(spid2,)
    b=(spid1,)
    c.execute( select degree from SP where spid=(?) , a)
    res1=c.fetchall()
    degree=res1[0][0]
    c.execute( select degree from SP where spid=(?) , b)
    res1=c.fetchall()
    degree2=res1[0][0]
    if degree<>2 and degree2<>2:
        s=0
    if degree==1 and degree2==1:
        c.execute( delete from S where ID=%d %(i))
        break
    if degree==2:
        s=1
        c.execute( select sp_id2 from S where sp_id1=%d %(spid2))
        res1=c.fetchall()
        if not res1:
            c.execute( select sp_id1 from S where sp_id2=%d and ID<>%i %(spid2,i))
            res2=c.fetchall()
            if not res2:
                break
            newspid=res2[0][0]
            c.execute( select sp_x1,sp_y1 from S where
```

```
sp_id2=%d and ID<>%i %(spid2,i))
            res2=c.fetchall()
            spx=res2[0][0]
            spy=res2[0][1]
        else:
            newspid=res1[0][0]
            c.execute( select sp_x2,sp_y2 from S where sp_id1=%d %(spid2))
            res1=c.fetchall()
            spx=res1[0][0]
            spy=res1[0][1]

        c.execute( select lenght from S where id=(?) , t)
        res2=c.fetchall()
        l1=res2[0][0]
        c.execute( select lenght from S where sp_id1=%d and sp_id2=%d %(newspid,spid2))
        res1=c.fetchall()
        if not res1:
            c.execute( select lenght from S where sp_id2=%d and sp_id1=%d %(newspid,spid2))
            res2=c.fetchall()
            l2=res2[0][0]
            c.execute( delete from S where sp_id2=%d and sp_id1=%d %(newspid,spid2))
        else:
            l2=res1[0][0]
            c.execute( delete from S where sp_id1=%d and sp_id2=%d %(newspid,spid2))
        lnew=l1+l2
        c.execute( update S set sp_id2=%d where id=%i  % (newspid,i))
        c.execute( update S set sp_x2=%d where id=%i  % (spx,i))
        c.execute( update S set sp_y2=%d where id=%i  % (spy
```

```
        ,i))
    c.execute( update S set lenght=%d where id=%i  % (lnew,i))


if degree2==2:
    s=1
    c.execute( select sp_id1 from S where sp_id2=%d %(spid1))
    res1=c.fetchall()
    if not res1:
        c.execute( select sp_id2 from S where sp_id1=%d and ID<>%i %(spid1,i))
        res2=c.fetchall()
        if not res2:
            break
        newspid=res2[0][0]
        c.execute( select sp_x2,sp_y2 from S where sp_id1=%d and ID<>%i %(spid1,i))
        res2=c.fetchall()
        spx=res2[0][0]
        spy=res2[0][1]
    else:
        newspid=res1[0][0]
        c.execute( select sp_x1,sp_y1 from S where sp_id2=%d %(spid1))
        res1=c.fetchall()
        spx=res1[0][0]
        spy=res1[0][1]

    c.execute( select lenght from S where id=(?) , t)
    res2=c.fetchall()
    l1=res2[0][0]
    c.execute( select lenght from S where sp_id2=%d and sp_id1=%d %(newspid,spid1))
```

```
                    res1=c.fetchall()
                    if not res1:
                        c.execute( select lenght from S where sp_id1=%d and sp_id2=%d %(newspid,spid1))
                        res2=c.fetchall()
                        l2=res2[0][0]
                        c.execute( delete from S where sp_id1=%d and sp_id2=%d %(newspid,spid1))
                    else:
                        l2=res1[0][0]
                        c.execute( delete from S where sp_id2=%d and sp_id1=%d %(newspid,spid1))
                    lnew=l1+l2
                    c.execute( update S set sp_id1=%d where id=%i  % (newspid,i))
                    c.execute( update S set sp_x1=%d where id=%i  % (spx,i))
                    c.execute( update S set sp_y1=%d where id=%i  % (spy,i))
                    c.execute( update S set lenght=%d where id=%i  % (lnew,i))

    i=i+1     # increment counter




c.execute( SELECT * INTO SPold FROM SP ) # places a copy of the old SP table
c.execute( drop table SP )   # deletes current SP table
S_lst = read_databases()
SP_lst = create_SP(S_lst)     #creates new SP table
```

Listing 13: Segment Manipulation

Cover Set Calculation

```
#Small script to calculate coverage sets for 3G, WiMAX and the DBWS

#Imports
import sqlite3 as db
from odbc import *
import dbi
from math import sqrt, floor, ceil, pi, log10
from random import uniform, random
from sets import *

COVERAGE_FILE = r C:\coverageMIMO.txt  #The location of the output file
TECH =  LTE  #The technology used.  W  = WiMAX,  3G  = 3G,  LTE  = LTE-Advanced
RAU_TABLE =  fixed_grid  #Name of the table containing RAUs
MT_TABLE =  MT  #Name of the table containing MTs
INPUT_DATABASE =  aalborg  #Name of the ODBC-connection to the input database

#Technology parameters that define the range of the RAUs.
if TECH ==  W :
    height, LB, f, hr = (30,133.1,3500,1.5) #RAU height, link budget in decibel, frequency in Mhz, reciever height
if TECH ==  3G :
    height, LB, f, hr = (30, 133.8, 2000, 1.5) #RAU height, link budget in decibel, frequency in Mhz, reciever height
if TECH ==  LTE :
    range1 = 300
    range2 = range1*1.45 #300*1.45 - 45 percent larger range
    #RAU height, link budget in decibel, frequency in Mhz, reciever height


```

```
#Class definitions
class DN:
    """Class for the RAU locations"""
    def __init__(self, name , x, y, id, environment):
        self.name = name
        self.x = x
        self.y = y
        self.id = id
        self.range = 0 #The coordinates are in meters - We will swap this with LB calculations
        self.environment = environment #This will be updated later. 0 is Urban, 1 is Suburban, 2 is Rural
        self.user_density = 0 #population density
        self.maxusers = 0 #Holds the number of NTs in largest radius
        self.covered = 0
        self.coverage0 = []
        self.coverage1 = []


    def reachcalc(self, MT_lst):
        """Function that calculates how many MTs are within range of the RAU"""
        for counter in range(len(MT_lst)):
            if MT_lst[counter].active == 1:
                a = self.x
                b = MT_lst[counter].x
                c = self.y
                d = MT_lst[counter].y
                eucledian_distance = sqrt(((self.x-MT_lst[counter].x)**2)+((self.y-MT_lst[counter].y)**2))
                if  (eucledian_distance < self.range):
                    self.coverage0.append(MT_lst[counter].id)
                    self.maxusers += 1
            else:
                pass
```

```
    def MIMO_reachcalc(self, MT_lst, range2):
        """An extra function for the MIMO coverage sets. Generates a
            second list of MTs covered
        by the extended range
        """
        for counter in range(len(MT_lst)):
            if MT_lst[counter].active == 1:
                eucledian_distance = sqrt(((self.x-MT_lst[counter].x)
                    **2)+((self.y-MT_lst[counter].y)**2))
                if  (eucledian_distance < range2):
                    self.coverage1.append(MT_lst[counter].id)
                    self.maxusers += 1
            else:
                pass
        temp = Set(self.coverage1)
        temp.difference_update(Set(self.coverage0))
        self.coverage1 = list(temp)


class MT:
    """Class for the Mobile Terminals"""
    def __init__(self, x, y, id):
        self.x = x
        self.y = y
        self.id = id
        self.active = 0

    def findDNsinrange(self,DN_lst):
        """Function to find how many RAUs are within range"""
        for count in range(len(DN_lst)):
            eucledian_distance = sqrt(((self.x-DN_lst[count].x)**2)+((
                self.y-DN_lst[count].y)**2))
            if eucledian_distance < DN_lst[count].range:
                return self.id
```

```python
        return

def calculate_ranges(height, LB, f, hr):
    """Function that calculates the range of the RAUs in the three environments, in accordance
    to the Cost 232 HATA model.
    """
    ranges = []
    environments = [0,1,2] #0 = Urban, 1 = Suburban, 2 = Rural
    for i in range(len(environments)):
        environment = environments[i]
        if environment == 0:
            cm = 3 #Urban = 3dB, rural = 0dB
            correction_factor = 0
        elif environment == 1:
            cm = 0
            correction_factor = -2*(log10(f/28.0))*(log10(f/28.0))-5.4
        else:
            cm = 0
            correction_factor = -4.78*log10(f)*log10(f)+18.33*log10(f)-40.94

        hb = height
        ahm = (((1.1*log10(f))-0.7)*hr)-(1.56*log10(f)-0.8)
        A = float(46.3+(33.9*log10(f))-(13.82*log10(hb))-ahm+cm+correction_factor)
        B = float(((44.9-6.55*log10(hb))))
        R = 10**((LB-A)/B)
        R = R*1000.0 #To get ranges in meters, not kilometers
        ranges.append(R)
    return ranges

def read_databases(aalborg,RAUtable, MTtable):
    """function to read from databases"""

```

```
#Initiate a connection to the database
conn = odbc(aalborg)
c = conn.cursor()

c.execute( SELECT * FROM %s  % RAUtable)
dynamic_info = c.description
for i in range(len(dynamic_info)): #Look up what columns in the database to use
    if dynamic_info[i][0] ==  id :
        id_location = i
    if dynamic_info[i][0] ==  x :
        x_location = i
    if dynamic_info[i][0] ==  y :
        y_location = i
    if dynamic_info[i][0] ==  environment :
        environment_location = i

DN_index = {} #Dictionary to look up the index and id of the
i = 0
results = c.fetchall()
DN_lst = [0]*len(results) #List that will hold the id numbers

for row in results: #Go through table create instances of the DN class
    id = row[id_location]
    x = int(row[x_location])
    y = int(row[y_location])
    environment = int(row[environment_location])
    DN_lst[id] = DN( DN_  + str(id), x, y, id, environment) #Adds the instance to the list
    DN_index = i
    i += 1
print i,  selected

#Read all the MTs as well.
```

```
    c.execute( SELECT * FROM %s  % MTtable)
    dynamic_info = c.description
    for i in range(len(dynamic_info)): #Look up what columns in the database to use
        if dynamic_info[i][0] ==  id :
            id_location = i
        if dynamic_info[i][0] ==  x :
            x_location = i
        if dynamic_info[i][0] ==  y :
            y_location = i
    MT_lst = [] #List that will contain the MT objects
    MT_index = {} #Dictionary to look up the index and ids
    i = 0
    results = c.fetchall()
    MT_lst = [0]*len(results) #List that will contain the MT objects
    for row in results: #Go through table and save the stuff we need
        id = row[id_location]
        x = int(row[x_location])
        y = int(row[y_location])
        MT_lst[id] = MT(x, y, id) #Adds the MT instance to the list
        MT_index[id] = i
        i += 1
    print i,  selected
    c.close()
    return DN_lst,MT_lst

if TECH !=  LTE :
    ranges = calculate_ranges(height, LB, f, hr)
else:
    ranges = [range1]*3
    ranges2 = [range2]*3

DN_lst,MT_lst = read_databases(INPUT_DATABASE, RAU_TABLE, MT_TABLE)

all = [] #List containing all MTs, used for calculating coverage
```

```
for i in range(len(MT_lst)):
    if TECH != LTE :
        all.append(MT_lst[i].id)
        MT_lst[i].active = 1
    else:
        if (-240600 < MT_lst[i].x < -232600) and (287600 < MT_lst[i].y < 293600): #If urban, mark it
            all.append(MT_lst[i].id)
            MT_lst[i].active = 1


sss = Set(all) #just a list containing all the MTs. Used for checking total coverage percent when planning
sss = list(all)

#Set the range of each RAU according to what environment it is in
for i in range(len(DN_lst)):
    e = DN_lst[i].environment
    if e == 0:
        DN_lst[i].range = ranges[0]
    elif e == 1:
        DN_lst[i].range = ranges[1]
    else:
        DN_lst[i].range = ranges[2]

#Calculate the coverage set for each RAU
for q in range(len(DN_lst)):
    DN_lst[q].reachcalc(MT_lst)
    if TECH == LTE : #additional loop for calculating the second coverage set of LTE
        DN_lst[q].MIMO_reachcalc(MT_lst,range2)


#-------------------------WRITE TEXTFILE ------------------------------------------
```

```
#Write results to a comma separated textfile
f = open(COVERAGE_FILE, w )
temp = str(sss),  \n
line =    .join(temp)
f.writelines(line)
for i in range(len(DN_lst)):
    temp = str(DN_lst[i].coverage0),  \n
    line =    .join(temp)
    f.writelines(line)
    if TECH ==  LTE :
        temp = str(DN_lst[i].coverage1),  \n
        line =    .join(temp)
        f.writelines(line)

f.close()

print  program successfully executed
```

Listing 14: Cover Set Calculation

References

[1] ZIED ABID, SOPHIE CHABRIDON, AND DENIS CONAN. A framework for quality of context management. In *Quality of Context*, pages 120–131. Springer, 2009.

(Referenced on pages 7, 11, 12, and 21.)

[2] JI AGBINYA. *Planning and Optimization of 3G and 4G Wireless Networks*, **8**. River Publishers, 2010.

(Referenced on page 36.)

[3] Capturing ambient intelligence for mobile communications through wireless sensor networks. `http://www.ist-esense.org/`, 2014. Accessed: 2014-07-23.

(Referenced on pages 4, 10, and 21.)

[4] STRATEGY ANALYTICS. Worldwide smartphone population tops 1 billion in q3 2012. *Retrieved*, **1**:2012, 2012.

(Referenced on page 2.)

[5] STEFAN ARNBORG AND ANDRZEJ PROSKUROWSKI. Linear time algorithms for np-hard problems restricted to partial¡ i¿ k¡/i¿-trees. *Discrete Applied Mathematics*, **23**[1]:11–24, 1989.

(Referenced on page 58.)

[6] MARTIN BØGSTED, RASMUS L OLSEN, AND HANS-PETER SCHWEFEL. Probabilistic models for access strategies to dynamic information elements. *Performance Evaluation*, **67**[1]:43–60, 2010.

(Referenced on pages 12, 21, 22, 37, 38, and 39.)

[7] JOHN ADRIAN BONDY AND UPPALURI SIVA RAMACHANDRA MURTY. *Graph theory with applications*, **290**. Macmillan London, 1976.

(Referenced on page 83.)

[8] JOHN BREWER. Kronecker products and matrix calculus in system theory. *IEEE Transactions on circuits and systems*, **25**[9]:772–781, 1978.

(Referenced on page 38.)

[9] THOMAS BUCHHOLZ, AXEL KÜPPER, AND MICHAEL SCHIFFERS. Quality of context: What it is and why we need it. In *Proceedings of the workshop of the HP OpenView University Association*, **2003**, 2003.

(Referenced on pages 9, 11, and 12.)

[10] PATRICE CALÉGARI, FRÉDÉRIC GUIDEC, PIERRE KUONEN, AND FRANK NIELSEN. Combinatorial optimization algorithms for radio network planning. *Theoretical Computer Science*, **263**[1]:235–245, 2001.

(Referenced on page 58.)

[11] MARK CARLSON, W WEISS, S BLAKE, Z WANG, D BLACK, AND E DAVIES. An architecture for differentiated services. Technical report, RFC 2475, December, 1998.

(Referenced on page 46.)

[12] Avergae cell tower costs. www.statisticbrain.com/cell-phone-tower-statistics/, 2014. Accessed: 2014-07-23.

(Referenced on pages 78 and 79.)

[13] THEOFILOS CHRYSIKOS, GIANNIS GEORGOPOULOS, AND STAVROS KOTSOPOULOS. Site-specific validation of itu indoor path loss model at 2.4 ghz. In *World of Wireless, Mobile and Multimedia Networks & Workshops, 2009. WoWMoM 2009. IEEE International Symposium on a*, pages 1–6. IEEE, 2009.

(Referenced on page 55.)

[14] VASEK CHVATAL. A greedy heuristic for the set-covering problem. *Mathematics of operations research*, **4**[3]:233–235, 1979.

(Referenced on page 58.)

[15] BENOIT CLAISE. *Cisco systems NetFlow services export version 9*, 2004.

(Referenced on page 24.)

[16] RINA DECHTER AND JUDEA PEARL. Generalized best-first search strategies and the optimality of a*. *Journal of the ACM (JACM)*, **32**[3]:505–536, 1985.

(Referenced on pages 66 and 67.)

[17] ALISA DEVLIC AND ERIK KLINTSKOG. Context retrieval and distribution in a mobile distributed environment. In *Third Workshop on Context Awareness for Proactive Systems (CAPS 2007), Guildford, UK*, 2007.

(Referenced on page 11.)

[18] ANIND K DEY. *Providing architectural support for building context-aware applications*. PhD thesis, Georgia Institute of Technology, 2000.

(Referenced on page 4.)

[19] Integrating the physical with the digital world of the network of the futur. `http://www.sensei-project.eu/`, 2014. Accessed: 2014-07-23.
(Referenced on pages 4, 11, and 21.)

[20] European IST-FP6 project spice. `http://www.ist-spice.org/`, 2014. Accessed: 2014-07-23.
(Referenced on pages 4, 10, and 21.)

[21] The future of mobile networks. `http://www.cisco.com`, 2014. Accessed: 2014-07-23.
(Referenced on page 2.)

[22] Nathan J Gomes, Paulo P Monteiro, and Atílio Gameiro. *Next generation wireless communications using radio over fiber*. John Wiley & Sons, 2012.
(Referenced on pages 6 and 16.)

[23] Horst W Hamacher and Zvi Drezner. *Facility location: applications and theory*. Springer, 2002.
(Referenced on page 58.)

[24] Nafiz Imtiaz Bin Hamid, Mohammad T Kawser, and Md Ashraful Hoque. Coverage and capacity analysis of lte radio network planning considering dhaka city. *International Journal of Computer Applications*, **46**[15]:49–56, 2012.
(Referenced on page 57.)

[25] Karen Henricksen, Jadwiga Indulska, and Andry Rakotonirainy. Generating context management infrastructure from high-level context models. In *In 4th International Conference on Mobile Data Management (MDM)-Industrial Track*. Citeseer, 2003.
(Referenced on pages 7, 11, and 21.)

[26] Wanguo Jiao, Min Sheng, King-Shan Lui, and Yan Shi. End-to-end delay distribution analysis for stochastic admission control in multi-hop wireless networks. *Wireless Communications, IEEE Transactions on*, **13**[3]:1308–1320, March 2014.
(Referenced on page 36.)

[27] Ivan Kaminow, Tingye Li, and Alan E Willner. *Optical fiber telecommunications VB: systems and networks*. Elsevier, 2010.
(Referenced on page 17.)

[28] Miguel A Lerma. Notes on discrete mathematics. *Northwestern University*, 2005.
(Referenced on page 74.)

[29] Kai Lieska, Erkki Laitinen, and Jaakko Lähteenmäki. Radio coverage optimization with genetic algorithms. In *Personal, Indoor and Mobile Radio Communications, 1998. The Ninth IEEE International Symposium on*, **1**, pages 318–322. IEEE, 1998.

(Referenced on page 58.)

[30] Lester Lipsky. Queueing theory. *Macmillan, NY.[Liu and Mouftah, 1997] Liu, X. and Mouftah, H.(1997). Queueing performance of copy networks with dynamic cell splitting for multicast ATM switching. IEEE Trans. Selected Areas Commun*, **45**:464–472, 1992.

(Referenced on page 36.)

[31] Ole Brun Madsen, Thomas Phillip Knudsen, and Jens Myrup Pedersen. *SQoS as the base for next generation global infrastructure*. Department of Control Engineering, Aalborg University, 2003.

(Referenced on page 74.)

[32] Ole Brun Madsen, J Dalsgaard Nielsen, and Henrik Schiøler. *Convergence*, 2002.

(Referenced on page 74.)

[33] Atif Manzoor, Hong-Linh Truong, and Schahram Dustdar. On the evaluation of quality of context. In *Smart Sensing and Context*, pages 140–153. Springer, 2008.

(Referenced on page 9.)

[34] Pierrick Marie, Thierry Desprats, Sophie Chabridon, and Michelle Sibilla. Qocim: A meta-model for quality of context. In *Modeling and Using Context*, pages 302–315. Springer, 2013.

(Referenced on page 9.)

[35] Alberto Martelli. On the complexity of admissible search algorithms. *Artificial Intelligence*, **8**[1]:1–13, 1977.

(Referenced on page 66.)

[36] Alberto Martelli and Ugo Montanari. Optimizing decision trees through heuristically guided search. *Communications of the ACM*, **21**[12]:1025–1039, 1978.

(Referenced on page 66.)

[37] Keith McCloghrie. *Differentiated services quality of service policy information base*, 2003.

(Referenced on pages 10 and 15.)

[38] Susan McKeever, Juan Ye, Lorcan Coyle, and Simon Dobson. A context quality model to support transparent reasoning with uncertain context. In *Quality of Context*, pages 65–75. Springer, 2009.

(Referenced on pages 7, 11, 12, and 21.)

[39] Paulo Monteiro, Sílvia Pato, Esther López, David Wake, Nathan J Gomes, and Atílio Gameiro. Fiber optic networks for distributed radio architectures: Futon concept and operation. In *Wireless Communications and Networking Conference Workshops (WCNCW), 2010 IEEE*, pages 1–5. IEEE, 2010.

(Referenced on pages 54, 64, 70, and 71.)

[40] Z Nadir, N Elfadhil, and F Touati. Pathloss determination using okumura-hata model and spline interpolation for missing data for oman. In *Proceedings of the World Congress on Engineering*, **1**, pages 2–4, 2008.

(Referenced on page 55.)

[41] Nokia Siemens Network. Mobile broadband with hspa and lte-capacity and cost aspects. *Nokia Siemens Network, White Paper*, 2010.

(Referenced on page 79.)

[42] Marcel F Neuts. *Structured stochastic matrices of M/G/1 type and their applications*, **701**. Marcel Dekker New York, 1989.

(Referenced on page 36.)

[43] Razali Ngah, Teguh Prakoso, and Tharek Abdul Rahman. Coverage range and cost comparison of remote antenna unit designs for inbuilding radio over fiber technology. *Journal of ICT Research and Applications*, **2**[1]:24–41, 2008.

(Referenced on page 78.)

[44] Anders Nickelsen, Rasmus L Olsen, and Carmen Santoro. *Open Pervasive Environments for migratory iNteractive services (OPEN)*, 2008.

(Referenced on pages 3 and 20.)

[45] Anders Nickelsen, Rasmus Løvenstein Olsen, and Hans-Peter Schwefel. Model-based decision framework for autonomous application migration. *Lecture Notes in Computer Science*, **6751**, 2011.

(Referenced on page 4.)

[46] Anders Nickelsen, Fabio Paternó, Agnese Grasselli, Kay-Uwe Schmidt, Miquel Martin, Björn Schindler, and Francesca Mureddu. Open: Open pervasive environments for migratory interactive services. In *Proceedings of the 12th International Conference on Information Integration and Web-based*

Applications & Services, pages 639–646. ACM, 2010.
(Referenced on pages 4 and 7.)

[47] Bo Friis Nielsen. Lecture notes on phase-type distributions for 02407 stochastic processes, 2012.
(Referenced on page 37.)

[48] Ampalavanapillai Nirmalathas, Prasanna A Gamage, Christina Lim, Dalma Novak, and Rodney Waterhouse. Digitized radio-over-fiber technologies for converged optical wireless access network. *Journal of Lightwave Technology*, **28**[16]:2366–2375, 2010.
(Referenced on pages 5 and 6.)

[49] NS Nkordeh, AA Atayero, FE Idachaba, and OO Oni. *LTE Network Planning using the Hata-Okumura and the COST-231 Hata Pathloss Models*, 2014.
(Referenced on pages 55 and 61.)

[50] Rasmus L Olsen, Joao Figueiras, Jakob Gulddahl Rasmussen, and H Schwefel. How precise should localization be? a quantitative analysis of the impact of delay and mobility on reliability of location information. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–6. IEEE, 2010.
(Referenced on pages 9, 10, and 15.)

[51] Angela Orebaugh, Gilbert Ramirez, and Jay Beale. *Wireshark & Ethereal network protocol analyzer toolkit.* Syngress, 2006.
(Referenced on page 24.)

[52] Packeth traffic genrator. http://packeth.sourceforge.net/packeth/Home.html.
(Referenced on page 23.)

[53] Fabio Paternò. *Migratory interactive applications for ubiquitous environments.* Springer, 2011.
(Referenced on page 20.)

[54] S Pato, Paulo P Monteiro, NJ Gomes, A Gameiro, and T Kawanishi. *Next generation distributed and heterogeneous radio architectures: the FUTON project*, 2009.
(Referenced on pages 5, 6, 7, 16, 54, 79, and 82.)

[55] Jens Pedersen. *Network Course at AAU.* Aalborg University, 2008.
(Referenced on page 65.)

[56] Jonathan M Pitts and John A Schormans. Configuring ip qos mechanisms for graceful degradation of real-time services. In *Military Communications Conference,*

2006. MILCOM 2006. IEEE, pages 1–7. IEEE, 2006.
(Referenced on page 10.)

[57] Ramjee Prasad. *My personal adaptive global NET (MAGNET).* Springer, 2009.
(Referenced on pages 4, 10, 11, and 21.)

[58] Davy Preuveneers and Yolande Berbers. Quality extensions and uncertainty handling for context ontologies. *W. on Context and Ont. Theory, Practice and Appl*, 2006.
(Referenced on page 11.)

[59] Luis Sanchez, Jorge Lanza, Rasmus Olsen, Martin Bauer, and Marc Girod-Genet. A generic context management framework for personal networking environments. In *Mobile and Ubiquitous Systems-Workshops, 2006. 3rd Annual International Conference on*, pages 1–8. IEEE, 2006.
(Referenced on pages 11 and 20.)

[60] H Schwefel, Martin B Hansen, and Rasmus L Olsen. Adaptive caching strategies for context management systems. In *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on*, pages 1–6. IEEE, 2007.
(Referenced on page 38.)

[61] John S Seybold. *Introduction to RF propagation.* John Wiley & Sons, 2005.
(Referenced on page 82.)

[62] Ahmed Shawky, Hans Bergheim, Ólafur Ragnarsson, Tahir Riaz, and Jens Pedersen. Planning of rof heterogeneous wireless networks. *Master Thesis*, 2009.
(Referenced on pages 8, 54, and 55.)

[63] Yuvraj Singh. Comparison of okumura, hata and cost-231 models on the basis of path loss and signal strength. *International Journal of Computer Applications*, **59**[11], 2012.
(Referenced on page 55.)

[64] Giorgos Stoilos, Nikos Simou, Giorgos Stamou, and Stefanos Kollias. Uncertainty and the semantic web. *Intelligent Systems, IEEE*, **21**[5]:84–87, 2006.
(Referenced on page 11.)

[65] Matthias Strobbe, Olivier Van Laere, Femke Ongenae, Samuel Dauwe, Bart Dhoedt, Filip De Turck, Piet Demeester, and Kris Luyten. *Novel applications integrate location and context information*, 2012.
(Referenced on page 9.)

[66] Chantal Taconet and Zakia Kazi-Aoul. Context-awareness and model driven engineering: Illustration by an e-commerce application scenario. In *Digital Information Management, 2008. ICDIM 2008. Third International Conference on*, pages 864–869. IEEE, 2008.

(Referenced on page 10.)

[67] Claudia Villalonga, Daniel Roggen, Clemens Lombriser, Piero Zappi, and Gerhard Tröster. Bringing quality of context into wearable human activity recognition systems. In *Quality of Context*, pages 164–173. Springer, 2009.

(Referenced on pages 11 and 12.)

[68] What do consumers want from wi-fi? `http://www.cisco.com`, 2014. Accessed: 2014-07-23.

(Referenced on page 2.)

[69] Yulei Wu, Geyong Min, and L.T. Yang. Performance analysis of hybrid wireless networks under bursty and correlated traffic. *Vehicular Technology, IEEE Transactions on*, **62**[1]:449–454, Jan 2013.

(Referenced on page 36.)

SUMMARY

In Communication networks today there is an endless quest for increased capacity and improved quality. With wireless systems being now popular worldwide for allowing users and devices to communicate and share information with each other irrespective of their location, the development of sustainable and reliable mobile applications is becoming a rising issue for next generation networks. Manufacturers and service providers are not only looking to increase capacity, bandwidth and performance properties, but are also looking into the network's ability to support new applications, features and improved services. The increased number of mobile users puts a demand on today's networks in terms of application sensitivity, mobility and reliability.

Mobile operators are offering various context-aware services and applications to the user. Today, mobile users are demanding access to dynamic context information at any time, thus due to the sensitivity of such context applications, users can easily get frustrated in case the context information received is no longer valid. Mobile operators are in search for methods to ensure reliability of dynamic context information.

Another main concern for both mobile operators and Internet Service Providers (ISPs) is providing their customers with higher bandwidth and better network services. Mobile applications today running on various devices such as smart-phones, tablets and other connected devices are consuming large amounts of network bandwidth. Mobile operators are now looking into heterogeneous networks to improve existing mobile broadband in order to accommodate such mobile applications and devices.

This PhD thesis provides mobile service providers and operators with a set of tools that aim to improve network services provided to the user. The PhD study looks into dynamic context information reliability by developing models and online algorithms that ensure increased reliability for context information exchange. Eventually, the PhD looks into improving network planning by implementing an automated network planning model that takes advantage of both Radio over Fiber (RoF) and heterogeneous wireless networks to meet the increasing demands for higher bandwidth.

ISSN (online): 2246-1248
ISBN (online): 978-87-7112-512-2

AALBORG **UNIVERSITY PRESS**