LiU-ITN-TEK-A-13/049-SE

Geo-based Mobility Control for Mobile Traffic Simulators

Sankar Saravanan Subramanians

2013-09-27

Linköpings universitet
TEKNISKA HÖGSKOLAN

Department of Science and Technology
Linköping University
SE-601 74 Norrköping, Sweden

Institutionen för teknik och naturvetenskap
Linköpings universitet
601 74 Norrköping

LiU-ITN-TEK-A-13/049-SE

Geo-based Mobility Control for Mobile Traffic Simulators

Examensarbete utfört i Datavetenskap
vid Tekniska högskolan vid
Linköpings universitet

Sankar Saravanan Subramanians

Handledare George Baravdish
Examinator Scott Fowler

Norrköping 2013-09-27

Institutionen för teknik och naturvetenskap
Department of Science and Technology

Examensarbete

Geo-Based Mobility Control for Mobile Traffic Simulators

Examensarbete utfört i Electrical Engineering
vid Tekniska högskolan vid Linköpings universitet
av

Patrik Dahlström
&
Sankar Saravanan Subramanian

LiTH-ITN-EX--YY/NNNN--SE

Norrköping 2013

Linköpings universitet
TEKNISKA HÖGSKOLAN

Department of Science and Technology
Linköpings universitet
SE-601 74 Norrköping, Sweden

Linköpings tekniska högskola
Linköpings universitet, Campus Norrköping
601 74 Norrköping

Geo-Based Mobility Control for Mobile Traffic Simulators

Examensarbete utfört i Electrical Engineering
vid Tekniska högskolan vid Linköpings universitet
av

Patrik Dahlström
&
Sankar Saravanan Subramanian

LiTH-ITN-EX--YY/NNNN--SE

Handledare: **George Baravdish**
ITN, Linköpings universitet
Lars-Anders Cederberg
Ericsson AB

Examinator: **Scott Fowler**
ITN, Linköpings universitet

Norrköping, 32 juni 2013

Avdelning, Institution
Division, Department

Institutionen för teknik och naturvetenskap
Department of Science and Technology
SE-601 74 Norrköping

Datum
Date

2013-06-32

Språk
Language

☐ Svenska/Swedish
☒ Engelska/English
☐ ________________

Rapporttyp
Report category

☐ Licentiatavhandling
☒ Examensarbete
☐ C-uppsats
☐ D-uppsats
☐ Övrig rapport
☐ ________________

ISBN
—

ISRN
LiTH-ITN-EX--YY/NNNN--SE

Serietitel och serienummer
Title of series, numbering

ISSN
—

URL för elektronisk version

http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-XXXXX

Titel
Title

Svensk titel saknas
Geo-Based Mobility Control for Mobile Traffic Simulators

Författare
Author

Patrik Dahlström & Sankar Saravanan Subramanian

Sammanfattning
Abstract

Most mobile traffic simulators of today depend on the user to supply the mobility behavior of the simulated UEs. This becomes a problem when certain wanted mobility characteristics are to be tested, since the user have to go trough a trial-and-error procedure to come up with the proper mobility behavior. This thesis presents two approaches to mobility control, where the aim is to control UE mobility based on certain mobility characteristics supplied by the end user.

The first approach introduces the concept of assigning tasks to UEs, e.g. “cross cell border” or “move to a certain cell”. Furthermore, concepts from control theory are borrowed to control the task assignment process, making it more dynamic and robust.

The second approach iteratively calculate movement patterns for the UEs in an area until it finds a movement pattern that has a high probability of satisfying the user’s requested mobility characteristics.

In order to properly evaluate these two approaches a prototype simulator was developed, as well as a virtual network controller to be tested. This test environment simulate a simplified tree network topology.

Both approaches was tested to control the total number of handovers per second in a simulated area. They both show high accuracy and acceptable precision. Additionally, the task based approach was used to control the cell utilization in a target cell. However, the cell utilization tests showed a lower accuracy and precision than the handover rate control tests.

Nyckelord
Keywords

problem, lösning

Sammanfattning

Svensk sammanfattning saknas

Abstract

Most mobile traffic simulators of today depend on the user to supply the mobility behavior of the simulated UEs. This becomes a problem when certain wanted mobility characteristics are to be tested, since the user have to go trough a trial-and-error procedure to come up with the proper mobility behavior. This thesis presents two approaches to mobility control, where the aim is to control UE mobility based on certain mobility characteristics supplied by the end user.

The first approach introduces the concept of assigning tasks to UEs, e.g. “cross cell border” or “move to a certain cell”. Furthermore, concepts from control theory are borrowed to control the task assignment process, making it more dynamic and robust.

The second approach iteratively calculate movement patterns for the UEs in an area until it finds a movement pattern that has a high probability of satisfying the user’s requested mobility characteristics.

In order to properly evaluate these two approaches a prototype simulator was developed, as well as a virtual network controller to be tested. This test environment simulate a simplified tree network topology.

Both approaches was tested to control the total number of handovers per second in a simulated area. They both show high accuracy and acceptable precision. Additionally, the task based approach was used to control the cell utilization in a target cell. However, the cell utilization tests showed a lower accuracy and precision than the handover rate control tests.

Acknowledgments

We would like to thank Ericsson for giving us an opportunity to write this thesis. With this thesis each of us, Patrik and Sankar complete our "Master of Science in Electrical Engineering " and " Master of Science in Computer Science and Engineering " respectively.

We would like to thank Lars-Anders Cederberg and Esbjörn Rundberg, our supervisors at Ericsson, for sharing their knowledge and providing us with their guidance and support throughout the period of this thesis.

We would like to thank Scott Fowler and George Baravdish, our Examiner and Supervisors at the University, for their involvement in this thesis by providing us with comments and encouragement.

I, Sankar, would like to thank my parents for their unconditional love and support. I would also like to thank all my friends for supporting me.

I, Patrik, would like to give a big *thank you* to my many great friends and amazing family for inspiring me to follow my dreams. This would not have been possible without your love and support.

Linköping, Januari 2020
Patrik Dahlström & Sankar Saravanan Subramanian

Contents

Notation

NOTATIONS USED IN THIS THESIS

Notation	Description
λ_h	Rate of handovers
ρ_c	Cell UE population density
K_P	Proportional constant
K_I	Integral constant
K_D	Derivative constant
M	Overshoot
P_b	Probability of call block
P_d	Probability of call drop
P_H	Probability of Handover
P_{HF}	Probability of Handover Failure
y_{ss}	Steady-State value

ACRONYMS AND ABBREVIATIONS

Abbreviation	Description
2G	Second generation
3G	Third generation
BS	Base Station
LTE	Long Term Evolution
ME	mobility Engine
MS	Mobile Station (UE)
PID	Proportional-Integral-Derivative
QoS	Quality of Service
RNC	Radio Network Controller
RNS	Radio Network Stub
RSS	Received Signal Strength
UE	User Equipment

1 Introduction

Testing a hardware node in a large mobile network using real user equipments (UEs) and surrounding hardware is impractical. Especially in a network where the number of UEs are in the range of thousands. Therefore, *Traffic simulators* are used to simulate user equipments and hardware. Usually, a single hardware node is tested at a time.

Current traffic simulators used by Ericsson are based on the notion that the UE movement behavior is submitted by the user. To properly test a certain aspect of a radio network, the UE movement has to be configured accordingly. This poses a problem when it is uncertain or hard to determine how the UE should move to test the aspect in question – a different simulation approach is needed.

The purpose of this thesis is to find a solution to how mobile traffic simulators can distribute user equipments over a geographic cellular network plan, and move them around in accordance with specified mobility characteristics. Mobility characteristics in this context are made up from requirements and constraints such as number of cell border crossings per hour, maximum number of simultaneous visitors in different cells, allowed/disallowed cells to visit, etc.

1.1 Objective

The objective of this thesis work is

- to formulate an idea on how a specified mobility characteristics can be guaranteed, with small deviations, in a traffic simulator.
- to formulate an idea on how multiple mobility characteristics can be guaranteed at the same time.

- to prove that the aforementioned ideas actually generate the specified characteristics.
- to focus on the following mobility characteristics
  - Rate of handovers
  - UE population density per cell
  - Allowed/disallowed cells to visit

1.2 Scope of Work

The scope of the thesis work is clearly defined through the following:

In this thesis, a simple Geographic area is considered and is built in terms of pixels. As our thesis was into mobility controls, real time geographic area models were not considered.

The Radio Network Controller used in this thesis is a simplified network which issues the basic commands to UE to perform a handover or not.

In this thesis, all the UEs that are used in the geographic area are in active state. There are no new calls considered in this thesis.

A simplified and a general model of resources is considered in this thesis.

In this thesis, we do not have any mechanisms that takes into account handover failures, or if calls are block or dropped.

1.3 Outline

This thesis work is divided into 10 chapters. Chapter 5 and 6 present two ideas on mobility control, and can be regarded as an extension of the Methodology chapter.

1 Introduction Describes the thesis background and motivation

3 Background Gives the necessary background theory and information.

4 Methodology Contains a description of methods, techniques. and tools employed.

5 Task Based Mobility Control Presents a mobility control concept based on tasks and control theory.

6 Probability Based Mobility Control Presents a mobility control algorithm based on calculated movement paths and handover probabilities.

7 Result Gives a comparison between the mobility control algorithms presented here.

8 Discussion Relates the results to the goal.

10 Future Work Summarizes other ideas and extensions that is not part of this thesis.

2

Related Work

When it comes to mobility, there is a lot of research on how to model it to real life scenarios (see Figure 2.1 and [1, Chapter 1]). The Random Waypoint Model was first proposed by Johnson and Maltz, in which nodes move independently to a randomly chosen destination with a randomly selected velocity[2]. Due to its simplicity and wide availability it has become the 'benchmark' mobility model to evaluate Mobile Ad hoc Network routing protocols. Two variants of the Random Waypoint Model are the Random Walk model and the Random Direction model[3, 4].

Other mobility models focus on constraining the node mobility to physical laws of acceleration, velocity, and rate of change of direction. Hence, the current velocity of a mobile node may depend on its previous velocity. This mobility characteristic can be called the *Temporal Dependency* of velocity. The most widely used models with temporal dependencies are the Gauss-Markov model and the Smooth Random Mobility model[5, 6].

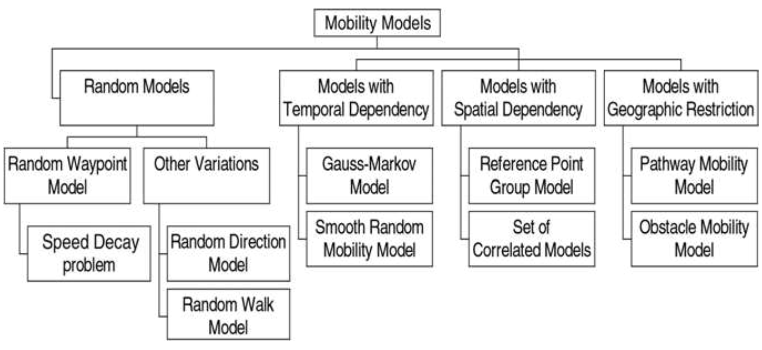


Figure 2.1: *Categories of mobility models*

Additionally, some mobility models discuss *Spatial Dependency* of velocity, where the velocities of different nodes are correlated in space. Examples of such models are the Reference Point Group Mobility model, the Column Mobility model, the Pursue Mobility model, and the Nomadic Mobility model[7, 8].

Furthermore, there are mobility models that take geographic restrictions into account. E.g. the motion of vehicles are bounded to the freeways or local streets, or the movement of pedestrians may be blocked by buildings or other obstacles. Two suchs models are the Pathway Mobility model and the Obstacle Mobility model[9, 10].

However, not much research have been done on how the mobility of nodes can be modeled in order to fulfill the mobility characteristics in section 1.1. Several mobile simulators have been presented in [11, 12, 13, 14, 15, 16, 17], but none of them talk about using the mobility of UEs to fulfill a specification. The UE mobility in these simulations are either random, user-submitted, or not mentioned at all. From discussions with our supervisors at Ericsson, the same can be said about their traffic simulators.

Vlajic, N. and Stevanovic, D. [18] summarize several mobility control algorithms for mobile sinks in wireless sensor networks. Qijun Gu et al. [19] further discuss another mobility control algorithm for mobile sinks. However, the focus of these mobility control algorithms lies in reducing energy consumption or transmission delays and are not applicable in this thesis work.

3

Background

This chapter provides the basic elements of wireless mobile networks and handover processes, such as the classification of handovers, the various steps in the handover process and the various handover schemes in use are described. This chapter also provides a brief introduction to the control theory and probability calculations needed for the methodology in chapter 5 and 6. The path loss and resource management related to mobile networks have also been introduced.

3.1 User Equipment

User Equipment is a device which is used by the end user to communicate while moving around a geographic area. It can be a mobile phone or a laptop computer with a mobile adapter or any other mobile device. It communicates by connecting to a base station of a cellular network provided by a mobile phone operator. It connects with other UEs through a Base Station.

3.2 Base Station

A Base Station is a wireless communications station which communicates with a UE and also communicates with other base stations. There are a number of Base Stations installed at fixed locations within the geographic area. The signals from one or more UEs in an area is received by a particular base station. The Base Station then connects the signal to the UE which is located within the area of a different cell. The Handover are explained in detail under Handover section 3.7.

3.3 Radio Network

The Radio Network is the collection of a number of base stations that the network can hold. The Radio Network also regulates the traffic in the geographic area.

Radio Network Controller (RNC) is the governing element in the mobile telecommunication systems. The RNC is responsible for communication with the Base Station (BS). The BS communicates with the mobile phones directly through radio frequency transmitters and receivers. In such networks the mobile phones can communicate with each other only through the BS. The RNC performs the system information broadcasting, cell resource allocation, radio resource management and mobility management. The RNC also encrypts the data between the sender and receiver. RNC is responsible for handover management and implements mobility functions such as paging and cell update.

3.4 Path Loss Models

The path loss models has been used to estimate the radio wave propagation in different environments. There are various models that has been defined to predict the path loss between the transmitter and receiver. Some of the well known models are the Free Space Path Loss,Okumura-Hata and Walfish-Ikegami models. The Okumura-Hata is used for rural and suburban areas while the Walfish-Ikegami model is used for urban areas. The Free Space path loss model is based on theoretical approach while Okumura-Hata and Walfish-Ikegami model are based on empirical results.[20]

3.4.1 Free Space Path Loss

The Free Space Path Loss is a path loss model in which the transmitter and the receiver have no obstacles to create reflection, diffraction or scattering.

The free space loss, L can be given by,

$$L = 32.4 + 20\ \log_{10}(f) + 20\ \log_{10}(d) \tag{3.1}$$

where f is the frequency in megahertz and d is the distance in kilometers.

3.4.2 Okumura Hata Model

The Okumura-Hata model is a radio frequency path loss model for predicting the behavior of cellular transmissions in a macro cell environment. It is an empirical model which is based on field measurements. The Okumura-Hata Model for path loss prediction can be written as,

$$\begin{aligned} L = A\ +\ B\ \log_{10}(f)\ -\ 13.82\ \log_{10}(H_b)\ -\ a(H_m) \\ +\ [\ 44.9\ -\ 6.55\ \log_{10}(H_b)]\ \log_{10}(d)\ +\ L_{other} \end{aligned} \tag{3.2}$$

where f is the frequency (MHz), H_b is the base station antenna height (m), $a(H_M)$ is the mobile antenna correction factor, d is the distance between the Base Transceiver Station and Mobile Station (km) and L_{other} is an additional correction factor.

The correction factor for a Mobile Station Antenna height for a small or medium sized city is:

$$a(H_M) = [1.1\ \log_{10}(f) -\ 0.7\ ]\ H_M - [1.56\ \log_{10}(f)\ -\ 0.8] \tag{3.3}$$

The correction factor for the large city is constrained to,

$$a(H_m) = \begin{cases} 8.29\ [\ \log_{10}\ (1.54\ H_m)]^2\ -\ 1.1: & \text{f} \ \leq\ 200\ MHz \\ 3.2\ [\ \log_{10}\ (11.75\ H_m)]^2\ -\ 4.97: & \text{f} \ \geq\ 400\ MHz \end{cases} \tag{3.4}$$

where H_m is the Mobile Station antenna height and is given by:

$$1\ \leq\ H_m\ \leq\ 10\ (H_m \text{ in metres}) \tag{3.5}$$

The parameters A and B are dependent on the frequency as follows,

$$A = \begin{cases} 69.55, & f = 150 - 1500\ MHz \\ 46.30, & f = 1500 - 2000\ MHz \end{cases} \tag{3.6}$$

$$B = \begin{cases} 26.16, & f = 150 - 1500\ MHz \\ 33.90, & f = 1500 - 2000\ MHz \end{cases} \tag{3.7}$$

To calculate the path loss in the urban areas, the correction factors are not required, but for rural areas the correction factors are required.

3.4.3 Walfish Ikegami Model

The Walfish Ikegami Model is a path loss model for urban areas. It has been designed for micro cells but it can also be applied to macro cells. The Walfish Ikegami model has two cases: line-of-sight (LOS) and non-line-of-sight situations.

The path loss prediction in the LOS situation is given by,

$$L = 42.6\ +\ 26\ \log(d) + 20\ \log(f) \tag{3.8}$$

where d is the distance (km) and f is the frequency (MHz)

The path loss for the non-line-of-sight condition is as follows:

$$L = \begin{cases} L_0\ +\ L_{rts}\ +\ L_{msd}: & L_{rts}\ +\ L_{msd} > 0 \\ L_0: & L_{rts}\ +\ L_{msd} \leq 0 \end{cases} \tag{3.9}$$

where L_{rts} is the rooftop-street diffraction and scatter loss, L_{msd} is the multiscreen diffraction loss, L_0 is the Free Space Path Loss defined by Equation 3.1.

The rooftop-street diffraction, L_{rts}, can be given as

$$\begin{aligned} L_{rts} = &\ -16.9\ -\ 10\ \log_{10}(w)\ -\ 10\ \log_{10}(f) \\ &\ -20\ \log_{10}(h_{roof}\ -\ h_{RX})\ -\ L_{Ori} \end{aligned} \tag{3.10}$$

where w is the mean value for street widths (meters), h_{roof} is the mean value for the building heights, ϕ is the road orientation angle.

The street orientation loss, L_{Ori}, is given by

$$L_{Ori}(\phi) = \begin{cases} -10 + 0.354\phi : & \text{for } 0 \leq \phi < 35^\circ \\ 2.5 + 0.075(\phi - 35) : & \text{for } 35 \leq \phi < 55^\circ \\ 4.0 - 0.114(\phi - 55) : & \text{for } 35 \leq \phi < 90^\circ \end{cases} \quad (3.11)$$

The multiscreen diffraction loss, L_{msd}, is given by

$$L_{msd} = L_{bsh} + k_a + k_d \log_{10}(d) + k_f \, 10 \, \log_{10}(f) - 9 \, \log(b) \quad (3.12)$$

where b is the mean value for building separation. L_{bsh},k_a,k_d and k_f are given by

$$L_{bsh} = \begin{cases} -18(1 + (h_{BTS} - h_{roof})) : & h_{BTS} > h_{roof} \\ 0 : & h_{BTS} < h_{roof} \end{cases} \quad (3.13)$$

$$k_a = \begin{cases} 54 : & h_{BTS} > h_{roof} \\ 54 - 0.8(h_{BTS} - h_{roof}) : & d \geq 0.5 \text{ km and } h_{BTS} \leq h_{roof} \\ 54 - 0.8(h_{BTS} - h_{roof})\frac{d}{0.5} : & d < 0.5 \text{ km and } h_{BTS} \leq h_{roof} \end{cases} \quad (3.14)$$

$$k_d = -4 \begin{cases} 8 : & h_{BTS} > h_{roof} \\ 18 - 15\frac{h_{BTS}-h_{roof}}{h_{roof}-h_{MS}} : & h_{BTS} < h_{roof} \end{cases} \quad (3.15)$$

$$k_f = \begin{cases} 0.7 \left(\frac{f}{925} - 1\right) : & \text{medium sized city and suburban areas} \\ 1.5 \left(\frac{f}{925} - 1\right) : & \text{urban centers} \end{cases} \quad (3.16)$$

where k_d and k_f controls the dependence between the multi-screen diffraction loss with the distance and the radio frequency and k_a is the increase in path loss for the BS below the rooftop.

3.5 Fading

Fading is the gradual loss of the signal over a propagation media. Fading is an important factor affecting the signal quality in wireless mobile networks. Fading can be divided into two types : slow fading and fast fading.

3.5.1 Slow Fading

The signal fades slowly and hence the name slow fading. The signal fading occurs due to changes in the conditions of atmosphere. The changes in the atmosphere may be due to the changes in temperature, pressure and humidity and the radio-refractivity which changes the k-factor [1].There are two types of refractive conditions: sub-refractive and super-refractive in which both the angle of transmission and angle of reception will change depending upon the atmospheric conditions.

The Slow Fading can also be caused by shadowing. Shadowing takes place when there is

[1] k-factor is defined as the ratio of the effective earth radius to the actual radius of the earth:

a obstruction in the form of buildings between the transmitter and the receiver.

Diffraction Fading

The variations in the atmospheric conditions leads to the variations in k-factor and the signal bends in a way where the earth's surface starts to obstructing the direct path between the transmitter and receiver. The various methods used to calculate diffraction:

Terrain Averaging Model This method is used to calculate the signal loss, if the obstacle is neither sharp nor rounded. The loss can be calculated as,

$$A_d = -20\frac{h}{F_1} + 10 \tag{3.17}$$

where

- h is the difference between path trajectory and the most significant obstacle
- F_1 is the radius of the first frensel zone [2].

Knife-Edge Model When there are more than one obstacle in the first frensel zone, the knife edge model is used. This model is used when there is a sharp object and is obstructing the first frensel zone. The diffraction loss can be calculated as,

$$L = 20\log(l) \tag{3.18}$$

where

- l =1 for v < -0.8
- l = 0.452 - $\sqrt{(v-0.1)^2} + 1 - (v-0.1)$ for -0.8 $\leq v$

and the Fresnel-Kirchhoff diffraction can be calculated as,

$$v = h\sqrt{\frac{2(d_1+d_2)}{\lambda d_1 d_2}} \tag{3.19}$$

where

- λ is the wavelength
- d_1 and d_2 are the distance to the sites from either side of the obstacle
- h is the height of the obstacle

3.5.2 Fast Fading

Under Fast Fading, the signal fades from a fraction of a second to a few minutes and the main cause for this fading is the multipath phenomenon. A signal ideally takes only one path to travel from the transmitting antenna to the receiving antenna. But a signal may also take different paths and the signal which is received by the receiving antenna

[2] first frensel zone is defined as the volume contained in the three dimensional ellipsoid between the transmitting and receiving antennas.

consists of both direct and indirect signals. The indirect signal consists of the signal which are reflected from the surface of the earth and by various atmospheric conditions. When the signals, both the direct and the indirect signals arrive at the receiving antenna with a half wavelength difference, fading will take place.

The probability of fading which exceeds the given fade path will depend upon two factors: amplitude of indirect signals and percentage of time for which fading is present.

3.5.3 Rayleigh Fading

The fading which is experienced in an environment, with a lot of reflections is known as Rayleigh fading. The Rayleigh Fading Model is useful in well built urban environment where there is a lot of reflection from buildings which affects the performance in cellular networks. In this model, there is no single dominating signal path between the transmitter and the receiver and in most cases the signals are scattered between them.

At the receiver, when the signals arrive, the different signals that took different paths are combined to form the original signal. This phase and the strength of the arrived signal is very important. The signals may be in phase or out of phase with the arriving signals.

3.6 Resource Management

The wireless networks has resources like frequency channels, time slots, code channels, transmission power and a number of transceivers. The radio resources should be managed efficiently, which can help in improving the quality of service and the efficiency of wireless networks. Resource management can also help in improving the handover in wireless networks, by reducing the handover failure and handover drop probability and also in maintaining the quality of service during and after the handover process. Admission control and bandwidth reservation are some of the important resource management techniques that are related to the handover process.

The admission control helps the system by preventing it from becoming overloaded. The new calls that are arriving and the ongoing calls can be treated differently. The new calls can be queued and the handover request can be prioritized.The bandwidth is an another important requirement in wireless networks. Handover can be performed when there is enough bandwidth available. Each cell can reserve a fraction of its total capacity and these bandwidth channels should be used only for the handover process and not for the arriving new call requests.

3.7 Handovers

Handover[21] is the process of transferring the connection of the UE from one channel to another. Handover is performed for a UE to make sure that the UE do not loose data when moving from one cell to another. The Handovers can be classified based on the type of network, the type of traffic the network supports, the involved network elements or the number of active connections.Different access technologies have different Handover algorithms implemented and most companies have their own proprietary Handover solutions.

The handovers are classified into Hard and Soft Handovers.

3.7.1 Hard Handover

A Hard Handover is the situation when the UE establishes a connection with a new cell, only after disconnecting from its current cell. In a communication network, where a Hard handover is implemented, the UE breaks off from the initial connection of a Base Station and then connects with the new Base Station. Hence this type of handover is also known as *break-before-make*. This is explained in Figure 3.1.

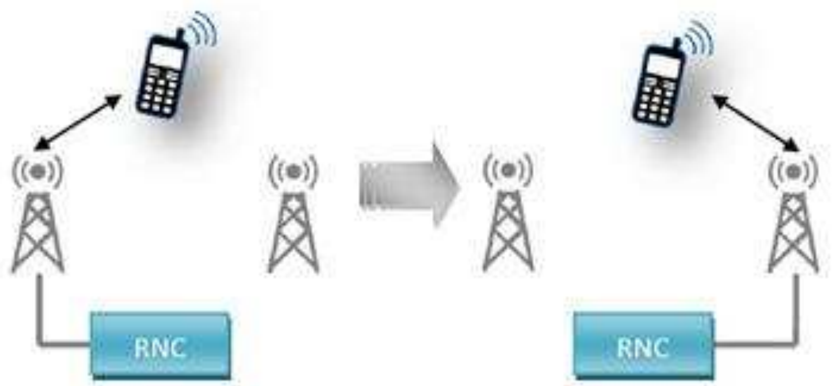


Figure 3.1: *Hard handover*
[22]

3.7.2 Soft Handover

A Soft handover is the situation in which the connection to the source cell is retained in parallel with the connection to the target cell. Using this technique, the connection is established with the target cell before the connection to the source cell is broken. Hence this type of handover is also called *make-before-break* and can be explained through Figure 3.2. Soft Handovers may also involve connection with two or more cells, where the mobile terminal maintains two or three connections leading to *softer handover*.

Figure 3.2: *Soft Handover*

3.7.3 Classification of Handovers

Handovers can also be distinguished into horizontal and vertical handovers. This depends on whether the handover occurs between a single type of network interface or with different types of network interfaces. The horizontal handovers can be further classified into intra-cell and inter-cell handovers. The intra-cell handovers occurs when a user moving within the cell and the radio channels with respect to the user has been changed to minimize the handovers within the same cell. The inter-cell handovers occurs when a UE moves to a nearby cell and all the connections of the UE is be transferred to the new cell.

Vertical Handover is the process of changing the Mobile Terminals connection between different wireless technologies. This can be further divided into *Downward Vertical handover* (DVH) and *Upward Vertical handover* (UVH). When a handover is made to a network of higher bandwidth and limited coverage , it is called as DVH. When the handover takes place with a network of lower bandwidth and higher coverage, it is called as UVH.

Table 3.1: *Types Classification*

Types	Soft	Hard
Horizontal	Intra-cell	Inter-cell
Vertical	Downward	Upward

3.8 Steps in Handover Process

In general, there are four steps involved in performing a handover.

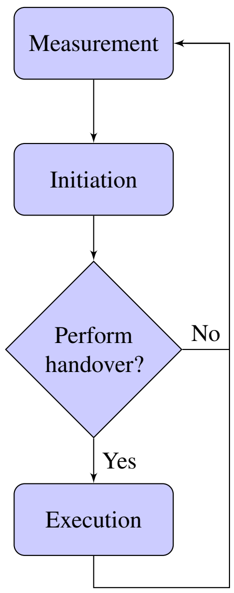


Figure 3.3: *Handover Decision Process*

3.8.1 Handover Measurement

During this phase, measurements of Received Signal Strength (RSS), Signal to Interference Ratio (SIR), distance measure, Bit Error Rate (BER) are measured for the handover process.

3.8.2 Handover Initiation

Handover Initiation[23] is the process of deciding whether a handover process is needed and if so, to initiate the handover process. The handover decision is made, by comparing the Received Signal Strength (RSS)[3] of the current base station and a neighboring base station. The handover initiation also analyses the quality of the currently used channel, the threshold and hysteresis values as parameters during the initiation process. The handover initiation can be explained by Figure 3.4.

In Figure 3.4, we compare the RSS's of two base stations, a current BS (BS1) and a neighboring BS (BS2). When the UE moves away from the current base station (BS1), the RSS1 of the BS1 decreases. But at the same time, as it gets nearer to the neighboring base station (BS2) the RSS2 of BS2 increases as a result of signal propagation. The four main handover techniques can be explained as follows.

Relative Signal Strength

In relative signal strength, the RSS's of both the base stations are measured over time. In Figure 3.4, at point A, the RSS of BS2 exceeds the RSS of BS1 and a new handover is requested by the base station of the current cell. But under certain situations, the handover takes place even though, the RSS of BS1 is sufficient enough to serve the UE. These unnecessary handovers leads to *ping-pong effect*. The increased number of handovers causes the call drop probability to increase. So the unnecessary handovers should be avoided.

Relative Signal Strength with Threshold

In order to avoid the *ping-pong effect*, a threshold value (T1 in Figure 3.4) is introduced in the Relative signal strength. At point B, in Figure 3.4, a handover is initiated when the RSS of BS1 becomes lower than the threshold value and RSS of BS2 is stronger than the RSS of BS1.

Relative Signal Strength with Hysteresis

This type of Relative signal strength uses a hysteresis value h, as noted in Figure 3.4 to initiate the handover process. At point C, when the RSS of BS2 exceeds the RSS of BS2 by a hysteresis value, a handover process is initiated.

Relative Signal Strength with Hysteresis and Threshold

This technique which is a combination of Hysteresis and Threshold formulates the technique which has a minimum number of handovers. When the RSS of BS1 is below a

[3]RSS is a parameter that provides information about total received power including all the interference and noise information.

threshold T1 in Figure 3.4, and the RSS of BS2 is is stronger than the RSS of BS1 by a hysteresis value h.

At point D in Figure 3.4, it is the *receiver threshold* which is the minimum RSS required for call continuation. If the RSS drops below the receiver threshold the call is dropped.

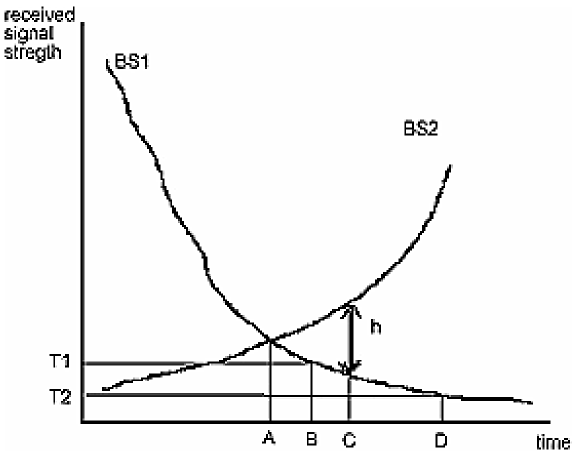


Figure 3.4: *Movement of a UE in a handover zone[4] [24]*

Prediction Techniques

The Prediction techniques[25] are used to predict the future value of the RSS using the information from the previous RSS value by using M-order adaptive filter.

$$Y_{n+1} = \bar{Y}_{n+1} + e_{n+1} \tag{3.20}$$

where e_{n+1} is the prediction error, Y_{n+1} is the current RSS . The next predicted RSS of the next estimate $\bar{Y}_{n+1}$, the predicted RSS of the next estimate and can be expressed as,

$$\bar{Y}_{n+1} = -\sum_{m=1}^{M-1} h_m(n+1)Y_{n-m} \tag{3.21}$$

where M is the order of the filter, and h_{n+1} is the $h_m(n+1)$ is the $(m+1)^{th}$ weight of the predictor at time nT.

This technique is better in reducing the number of unnecessary handovers when compared to the previous techniques of relative signal strength and relative signal strength with hysteresis and threshold.

3.8.3 Handover Decision

This phase decides whether the handover should be performed based on the resource available and the network load. The Handover decisions[26] can be classified into Mobile Controlled Handovers, Mobile-Assisted Handovers and Network-Controlled Handovers.

[4] Image sharpened with the kernel in Figure A.2

Mobile Controlled Handovers (MCHO)

The Mobile Controlled Handovers are used in Digital Enhanced Cordless Telecommunication (DECT). In this MCHO, the Mobile Terminal constantly monitors the surrounding Base Station signals and requests a channel from the target Base Station.

Mobile-Assisted Handovers (MAHO)

In Mobile-Assisted Handovers, the Mobile Terminal measures the signal strength received from the serving base station and the surrounding base stations. The network performs the handover decision based on the measurement reports. The Mobile-Assisted handovers are best suited for micro cells, where handovers are more frequent and the signal quality is good.

Network Controlled Handovers (NCHO)

The mobile telephone switching office (MTSO) is responsible for Network Controlled Handovers (NCHO). In NCHO, the neighboring Base Station signals are measured by the Mobile Terminal. The handover decisions and Relative Signal Strength (RSS) measurements are handled by the network.

3.8.4 Handover Execution

This is the final phase of the handover process and the network allows the Mobile Terminal to communicate with the new base station and transfer its communication to a different cell. Several other process of authentication, database lookup and network configuration are performed in this final step.

3.9 Handover schemes

Handover in a wireless network is very important for the continuation of connections and the Quality of Service perceived by the users. The handover schemes[21] are distinguished into Non-Prioritized Schemes and Prioritized Schemes.

In non-prioritized schemes, both the handover calls and the newly arrived calls are treated equally. When the BS's channel is idle, a first-come first-serve scheme is utilized. Using this scheme there is no difference between new calls and the handover calls. As long there are free channels available, both the calls are served. If there are no free channels the calls will be blocked. There is no priority between the new and the handover calls, and hence there is a increase in the call drop probability (CDP).

The Non-Prioritized schemes uses the policies of Complete Sharing (CS) and Complete Partitioning (CP). The CS provides equal access to the available bandwidth for both the incoming and handover calls. The CP divides the available bandwidth into sub-pools according to the incoming and handover calls.

In Prioritized schemes, the Call Dropping Probability (CDP) and Call Blocking Probability (CBP) is reduced by increasing the priority of the handover calls over the arriving new calls. Since handover calls are prioritized, the call block probability is increased. The handover prioritization schemes lead to increased performance at the expense of the reduction in the total admitted traffic and an increase in the call block probability of new calls. There are several handover prioritization schemes that has been proposed, a few of the most important are described as follows:

3.9.1 Guard Channels

This scheme reserves a fixed number of channels for handover calls only. The rest of the channels are used for both new and the handover calls. As a result of reserving channels for handover calls, there is a decrease in forced termination probability and an increase in the call blocking probability. The number of guard channels are dynamically determined by the neighboring Base Stations.

The number of UEs in the pre-handover zone (PHZ) is determined by the Base Station and informs its neighboring Base Station. The pre-handover zone[24] is a small area which is located near the handover zone and contains the UEs which will enter the handover zone. Whenever the Base station gets the number of UEs in the pre-handover zone, it reserves the amount of guard channels for handover calls.

3.9.2 Queuing Handover Calls

When all the channels in the base station are occupied by calls, the handover calls are queued. When the channel is released, it is assigned with one of the calls in the queued list. If the queue is empty or there is at least one free channel, a new call request may be assigned to the channels.

Queuing handover calls decreases the call drop probability. Queuing handover calls can be used irrespective of the guard channels. There are different types of schemes that uses the queuing handover concept.

In a timer based handover priority scheme a timer is started whenever a channel is released by the base station[27]. If there is a handover request within the time interval, the channel is assigned to it. If the timer expires, the channel is assigned to a new or handover calls according to the order of arrival.

The Measurement Based Prioritization Scheme (MBPS), the handover calls are added to the queue and its priorities changes depending upon their power level. The calls with a power level that is close to the receiver threshold will have higher priority.

The Most Critical First (MCF) will determine the first handover call that will be cut off and assigns the first released channel to that call[24]. The first handover call which will be cut off will have the highest priority. This scheme has a trade off with increase in forced termination probability with a decrease in the call blocking probability.

3.9.3 Channel Transferred Handover Schemes

When there are no available channels to accommodate the handover call request, a channel is transferred from a neighboring cell. After the handover has taken place, the transferred channel may follow up on two decision categories: the Channel Carrying Approach (CCA), that selects its current channel and allows the UE to carry its channel using certain mobility patterns to the new cell.In Channel Borrowing Approach (CBA) where a new channel is selected from the neighboring cell.

3.9.4 Sub Rating Schemes

This scheme degrades the bandwidth of existing calls in order to accept more handover calls. Under these scheme, the ongoing calls are forced to operate under degraded modes in order to accommodate calls into an overloaded system. Under these scheme, certain channels are allowed to divide temporarily into two channels with half the original rate in order to accommodate more calls into the system. Using these scheme, one half of the channel is used to maintain the existing connections while the other half is used to maintain the new handover calls. When a sub-rated channel is released, it is combined with the other sub-rated channel to form the original full-rated channel.This scheme reduces the blocking probability and forced termination probability for handover calls on the contrary with the introduction of degradation in the system.

Table 3.2: *Prioritization Schemes Comparison*

Prioritization Schemes	Advantages	Disadvantages
Channel transferred	Increases system Efficiency	Signaling overhead
Sub rating	Increases system efficiency	QoS degradation
	Increases channel utilization	Delay needed to assign channels

3.9.5 LTE Standard Hard Handover Algorithm

In the Long Term Evolution Standard Hard Handover Algorithm[28], when a mobile starts to move away from its serving cell, its Received Signal Received Power (RSRP) starts deteriorating as the time increases. But at the same time, when it approaches an another cell the RSRP will increase. A handover is triggered when this condition is satisfied for the entire Time to Trigger (TTT) time duration.

$$RSRP_T > RSRP_S + HOM \tag{3.22}$$

where $RSRP_T$ and $RSRP_S$ are the RSRP which is received from the target and the serving cell respectively. HOM is the handover margin (HOM) which represents the threshold for the difference in signal strength between the target and the serving cell. TTT, prevents the UE from making an unnecessary handover. This is illustrated by Figure 3.5

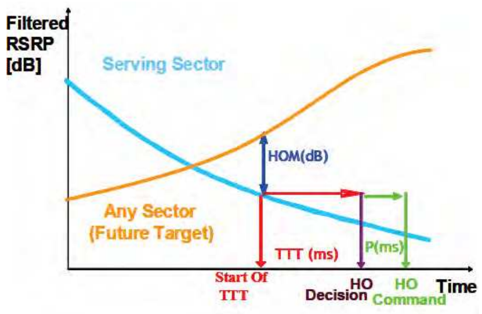


Figure 3.5: *LTE Hard Handover Algorithm*

3.10 Control Theory

Traditionally, control theory is divided in two approaches to control: the classical and the state-space control. Classical is the most straight-forward and generally does not require any inherent knowledge of the system to be controlled. State-space control is able to handle systems with multiple inputs and/or multiple outputs, but assumes knowledge of the system to be controlled.

3.10.1 Classic control

In classic control theory, transfer functions are used to define controllers and systems to be controlled. A transfer function relates the input to the output and is, in classic control theory, often given in the Laplace domain. Block diagrams are also very common to use to visualize a controlled system.

$$h(t) = \frac{y(t)}{u(t)} \Leftrightarrow H(s) = \frac{Y(s)}{U(s)} \tag{3.23}$$

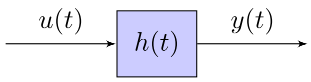


Figure 3.6: *Transfer function in block diagram*

Control Theory - Controllers

A system to be controlled is referred to as a process or plant. If a process is unstable, it may need to be controlled. This can be done by adding a *Controller* to the input signal. This controller is called an open-loop controller. Together, they form an open-loop control system.

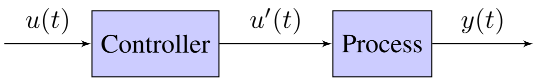


Figure 3.7: *Open-loop control system*

Control Theory - Closed-Loop Systems

Unless the process in a an open-loop system is completely known and predictable, open-loop systems are hard to use. Therefore, closed-loop control systems are used instead. A closed-loop system is where the output of a open-loop system is used as feedback to the controller. The output is compared to a reference input to form the error. This error is used as the input to the controller (see Figure 3.8).

$$e(t) = r(t) - y(t) \tag{3.24}$$

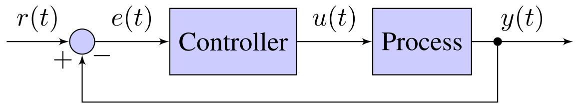


Figure 3.8: *Closed-loop control system*

This can of course also be written as a closed-loop transfer function:

$$h_c(t) = \frac{y(t)}{r(t)}. \tag{3.25}$$

The closed-loop transfer function is further investigated in later sections.

In order to achieve a controlled output in a closed-loop system, a well designed controller is needed. The most simple controller is when

$$u(t) = e(t).^5 \tag{3.26}$$

Control Theory - PID Controller

A common controller is the proportional-integral-derivative (PID) controller:

$$u(t) = \underbrace{K_P e(t)}_{\mathbf{P}} + \underbrace{K_I \int_0^t e(\tau)\, d\tau}_{\mathbf{I}} + \underbrace{K_D \frac{de(t)}{dt}}_{\mathbf{D}} \tag{3.27}$$

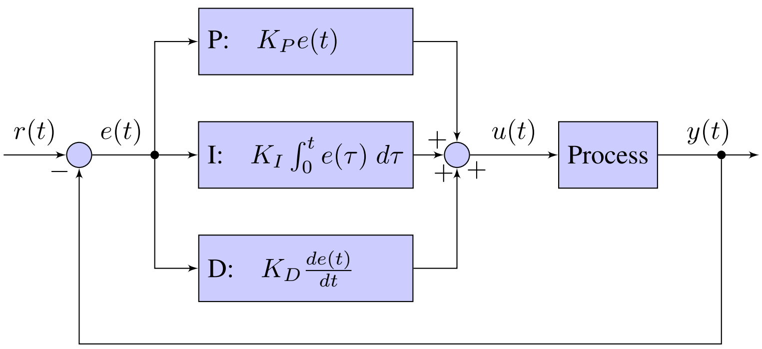


Figure 3.9: *Block diagram of a closed-loop system with a PID controller*

A PID controller can be interpreted in terms of time, where **P** depends on the *present* error, **I** depends on the accumulation of *past* errors, and **D** is a prediction of *future* errors.

A large proportional constant (K_P) magnifies the error signal. Therefore, if K_P is too large, the system can become unstable because it tries to overcompensate the error, overshooting the target.

[5] It may be argued that this does not describe a controller at all, but the lack of a controller.

Likewise, if K_P is small it can make the system slow and unresponsive.

Since the controller operates on a non-zero error, using only **P** will generally generate a steady-state error in the output, sometimes referred to as *droop*. The level of the steady-state error is proportional to the process gain and inversely proportional to K_P. This steady-state error can be corrected by **I** or by adding a bias term to the input.

The **I** term is commonly used to mitigate steady-state error by taking into account previous errors. However, since **I** accumulates previous errors it can overshoot the target level if K_I is too large.

Additionally, if the process is slow to react to changes in the input signal, the accumulated error of **I** can cause the controller to continue to increase its control signal even if the error is decreasing. This is commonly known as *integral windup*. This can also happen if the reference signal is set to a value that the process can never reach, i.e. the process becomes saturated. When the reference signal is later adjusted to a level the process is able to reach, due to the integral windup, the system will take a long time to react.

Since **D** "predicts" future errors, it is used to decrease overshoot and settling time to the system. But as with the other constants, choosing a large K_D may instead make the system unstable. If the steady-state output signal contains a lot of noise, **D** might amplify these errors and make the system more unstable than without the **D** term. Likewise, if the reference signal changes instantaneously, the derivative term might cause the controller to output an unreasonably large control signal.

Control Theory - Discrete PID Controller

Due to the sampled nature of most control system, controllers need to be discretized. The proportional term can be converted directly, but the derivative and integral terms have to be approximated.

Two common approximations are

$$K_I \int_0^t e(\tau)\, d\tau \Rightarrow K_I \sum_{i=0}^{n} e[i]\Delta T \tag{3.28}$$

$$K_D \frac{de(t)}{dt} \Rightarrow K_D \frac{e[n] - e[n-1]}{\Delta T} \tag{3.29}$$

However, to counter integral windup, the following approximation might be used instead

$$K_I \int_0^t e(\tau)\, d\tau \Rightarrow K_I \left(e[n] + e[n-1]\right) \Delta T \tag{3.30}$$

Control Theory - Laplace Transform

When doing calculations for control theory it is common to work in the s-domain to simplify calculations and increase the understanding the system to be controlled. For example, the closed-loop transfer function in s-domain becomes

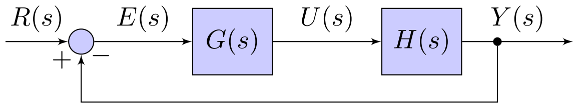


Figure 3.10: *Closed-loop control system in s-domain*

$$\begin{aligned} Y(s) &= H(s)U(s) = \\ H(s)G(s)E(s) &= H(s)G(s)(R(s) - Y(s)) \Leftrightarrow \\ Y(s)(1 + H(s)G(s)) &= H(s)G(s)R(s) \Leftrightarrow \\ \frac{Y(s)}{R(s)} &= H_C(s) = \frac{H(s)G(s)}{1 + H(s)G(s)}, \end{aligned} \tag{3.31}$$

where $H(s)$ and $G(s)$ are the process and controller, respectively. It is clear that the closed-loop system will become unstable if $H(s)G(s) = -1$ for any s.

The s-domain is defined in continuous time, but most control systems are discrete in nature. If the control system has a high enough sample rate this does not pose a problem because the sampled system can safely be approximated as continuous.

Control Theory - Z-transform

In a low sample rate system, continuous-time models and Laplace transforms can no longer be used. Therefore, the Laplace transform is replaced with the Z-transform.

A controller can still be designed in continuous-time, and then transformed to a discrete controller using the Tustin transformation (3.32).

$$s = \frac{2(z-1)}{T(z+1)} \tag{3.32}$$

Control Theory - Smith Predictor

Some processes present significant delays from when a control signal is applied and to when the output is changed. This can cause instability in the system since the controller is acting on outdated information.

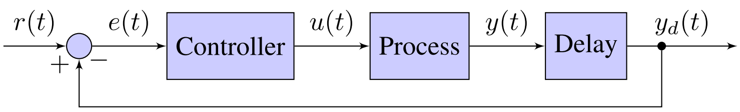


Figure 3.11: *Closed-loop control system with process delay*

One way of combating time delays is to slow down the sample rate of the system so that when the output, $y_d(t)$, is measured, the effect of the last input has already taken place. This however, can lead to a slow system.

Another approach is to use a Smith Predictor [29], which is a predictive feedback controller for the controller itself. Consider two closed-loop systems, with and without time delay of k samples in the z-domain:

$$H(z) = \frac{C(z)G(z)}{1 + C(z)G(z)} \tag{3.33}$$

and

$$H'(z) = \frac{z^{-k}C'(z)G(z)}{1 + z^{-k}C'(z)G(z)}, \tag{3.34}$$

where $G(z)$ represent the process, and $C(z)$ and $C'(z)$ represent controllers designed with no time delay and with time delay, respectively.

A Smith predictor is based on designing $C'(z)$ such that

$$H'(z) = z^{-k}H(z). \tag{3.35}$$

Thus, the time delay is moved out of the control loop.

Substituting (3.33) and (3.34) in (3.35) and solving for $C'(z)$ yield

$$C'(z) = \frac{C(z)}{1 + C(z)G(z)(1 - z^{-k})}. \tag{3.36}$$

This transfer function can be represented by either of the block diagrams in Figure 3.12. $\hat{G}(z)$ is used to reflect that a model of the process is used. The Smith predictor is therefore largely dependent on an accurate process model in order to be effective.

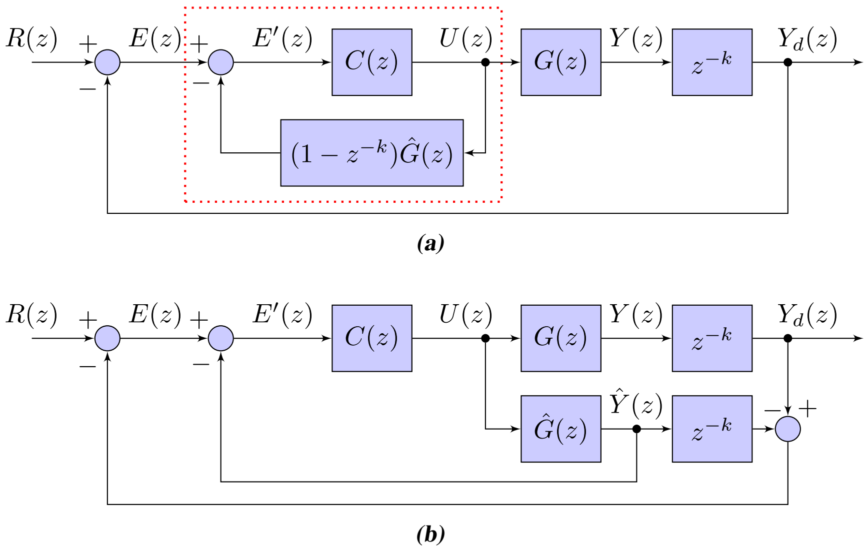


Figure 3.12: *Closed-loop control system with Smith predictor (a) encircled in red and (b) redrawn for clarity*

3.10.2 State space representation and control

Another way of describing a system in control theory is by a set of input, output and state variables related by first-order differential equations. This system representation is called state space representation, where “state space” refers to the space whose axes are the state variables. The state of the system is represented by a vector in that space.

Consider the following system of first order differential equations:

$$\begin{cases} \dot{x}_1(t) - a_1 x_1(t) = b_1 u_1(t) \\ \vdots \\ \dot{x}_n(t) - a_n x_n(t) = b_n u_n(t) \\ y_1(t) = c_1 x_1(t) + d_1 u_1(t) \\ \vdots \\ y_n(t) = c_n x_n(t) + d_n u_n(t) \end{cases} \tag{3.37}$$

System (3.37) can be rewritten in matrix form as

$$\begin{cases} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} a_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & a_n \end{bmatrix} \begin{bmatrix} \dot{x}_1(t) \\ \vdots \\ \dot{x}_n(t) \end{bmatrix} + \begin{bmatrix} b_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & b_n \end{bmatrix} \begin{bmatrix} u_1(t) \\ \vdots \\ u_n(t) \end{bmatrix} \\ \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} c_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & c_n \end{bmatrix} \begin{bmatrix} x_1(t) \\ \vdots \\ x_n(t) \end{bmatrix} + \begin{bmatrix} d_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & d_n \end{bmatrix} \begin{bmatrix} u_1(t) \\ \vdots \\ u_n(t) \end{bmatrix} \end{cases} \tag{3.38}$$

or, more condensed, as

$$\begin{aligned} \vec{\dot{x}}(t) &= \mathbf{A}\vec{x}(t) + \mathbf{B}\vec{u}(t) \\ \vec{y}(t) &= \mathbf{C}\vec{x}(t) + \mathbf{D}\vec{u}(t) \end{aligned} \tag{3.39}$$

The system in (3.39) form a mathematical description of a system without any control added to it. This representation can also be described by the block diagram in Figure 3.13.

In order to control an unstable system a *full state feedback* signal can be added as such

$$\vec{u}(t) = \vec{r}(t) - \mathbf{K}\vec{x}(t). \tag{3.40}$$

This allow the system to be controlled but, because the new input signal ($r(t)$) is compared to the state of the system, this will unfortunately introduce a steady-state error at the output signal. This problem can be resolved by adding a pre-compensation term, $\bar{\mathbf{N}}$, to the input.

$$\vec{u}(t) = \bar{\mathbf{N}}\vec{r}(t) - \mathbf{K}\vec{x}(t) \tag{3.41}$$

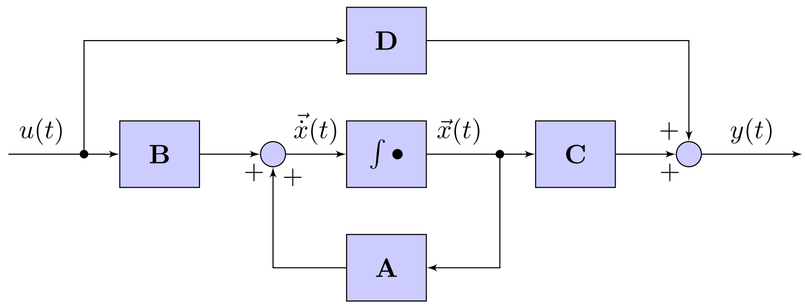


Figure 3.13: *Block diagram of a open-loop state space system*

Finally, (Equation 3.39) can be rewritten using (Equation 3.41)

$$\begin{aligned}\dot{\vec{x}}(t) &= (\mathbf{A} - \mathbf{BK})\,\vec{x}(t) + \mathbf{B}\bar{\mathbf{N}}\vec{r}(t)\\ \vec{y}(t) &= (\mathbf{C} - \mathbf{DK})\,\vec{x}(t) + \mathbf{D}\bar{\mathbf{N}}\vec{r}(t)\end{aligned} \tag{3.42}$$

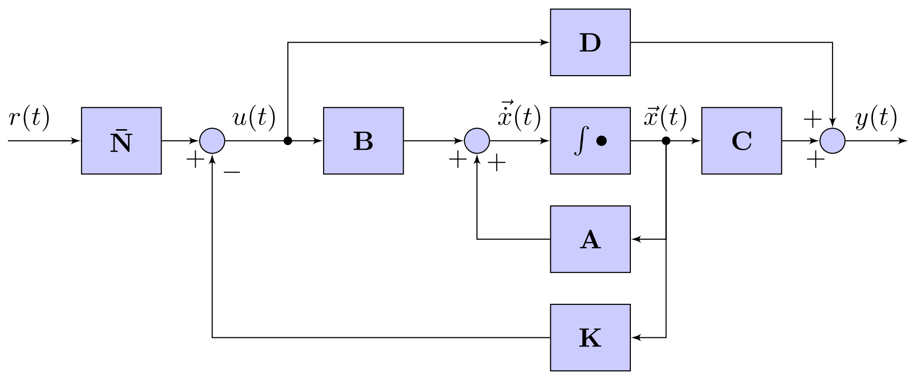


Figure 3.14: *State space system with full state feedback and pre-compensation*

3.10.3 Discrete State Space

Previous sections have given an introduction to state space representation and control in continuous time domain, but most control systems are in fact discrete. If the discrete control system is fast enough, it can safely be approximated as continuous and no further action is needed. However, if this is not the case, a discrete state space representation is needed. One such discrete representation is

$$\begin{aligned}\vec{x}[k+1] &= \mathbf{A_d}\vec{x}[k] + \mathbf{B_d}\vec{u}[k]\\ \vec{y}[k] &= \mathbf{C_d}\vec{x}[k] + \mathbf{D_d}\vec{u}[k],\end{aligned} \tag{3.43}$$

where $\mathbf{A_d}$, $\mathbf{B_d}$, $\mathbf{C_d}$, and $\mathbf{D_d}$ are the discrete versions of $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$. If a continuous state space system is already available, it can be converted into discrete state space by

$$\mathbf{A_d} = \mathrm{e}^{\mathbf{A}T} \tag{3.44}$$

$$\mathbf{B_d} = \int_0^\infty \mathrm{e}^{\mathbf{A}\tau}\, d\tau \mathbf{B} \tag{3.45}$$

$$\mathbf{C_d} = \mathbf{C} \tag{3.46}$$

$$\mathbf{D_d} = \mathbf{D}. \tag{3.47}$$

If $\mathbf{A}$ is a singular matrix, then $\mathbf{B_d}$ can be defined as

$$\mathbf{B_d} = \mathbf{A}^{-1}(\mathbf{A_d} - \mathbf{I})\mathbf{B}, \tag{3.48}$$

where $\mathbf{I}$ is the identity matrix. For derivations of the above equations and more information on state space control, see [30].

3.10.4 Control Theory Performance Metrics

The performance of a controller is usually evaluated by applying the Heaviside step function, with amplitude A, as input to the controlled system and recording the output, also called the *step response*. From a step response, it is possible to evaluate

- Overshoot
- Percentage overshoot
- Steady-state error
- Settling time
- Rise time

To illustrate these metrics, the step response of the second order closed-loop transfer function in Equation 3.49 is shown in Figure 3.15.

$$H_C(s) = \frac{0.5^2}{s^2 + 0.5s + 0.5^2} \tag{3.49}$$

The step input to Equation 3.49 is

$$R(s) = \frac{A}{s} \Leftrightarrow r(t) = Au(t), \tag{3.50}$$

where $A = 1$ and $u(t)$ is the heaviside step function.

Steady-state value is the value of the output signal when the system have stabilized.

$$y_{ss} = y(\infty) = \lim_{t\to\infty} y(t) \tag{3.51}$$

If steady-state oscillation is present, the mean value can be used instead.

Steady-state error is the difference between input and the final output value.

$$e_{ss}(\infty) = \lim_{t\to\infty} |r(t) - y(t)| = |A - y_{ss}|, \tag{3.52}$$

where A is the amplitude of the step input.

Overshoot is how much the output signal misses its steady-state value. Assuming the output signal begins at 0, overshoot can be defined as (M in Figure 3.15)

$$M = \max |y(t)| - |y_{ss}| \tag{3.53}$$

Percentage overshoot is defined as

$$M_p = \frac{M}{|y_{ss}|}. \tag{3.54}$$

Settling time is the time it takes for the output signal to enter and remain in a specified error band (t_s in Figure 3.15).

Rise time is the time it takes for the output signal to go from 10 % to 90 % of its final value (t_r in Figure 3.15).

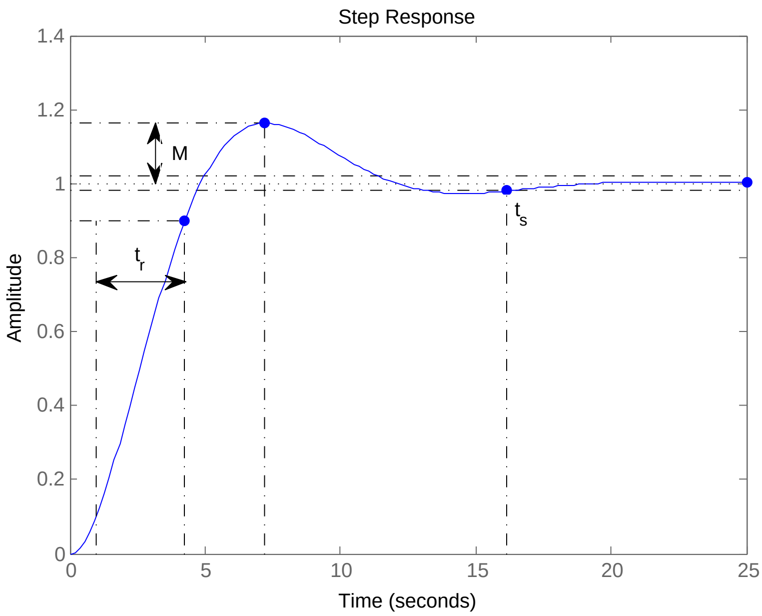


Figure 3.15: *Step response of Equation 3.49.*

3.11 Probabilities

There are several types of handover algorithms that has been defined and implemented in mobile networks. There is a mathematical way of performing the handover process which has been described by Jabbari in [31]. The mathematical analysis includes the calculations in the following sections.

3.11.1 Probability of Handover (PHO)

The probability of a handover is the probability that a handover will occur when it is in the new cell and is given by,

$$P_h = P(T_n > T_h) = \frac{\eta}{\mu + \eta} \tag{3.55}$$

where $\mu = \frac{1}{\tau}$ and τ is the unhindered call duration and its mean is given by $\bar{\tau}$. The cell cross over rate, η, is given by,

$$\eta = V\frac{L}{\pi S} \tag{3.56}$$

where V is the mean velocity and their movement is uniformly distributed over $[0, 2\pi]$. L is the boundary length and S is the surface area of the cell. T_n is the call holding time, exponentially distributed with parameter μ and T_h is the cell dwell time which is exponentially distributed with η.

3.11.2 Probability of Handover Failure (PHF)

The handover failure can occur when the neighboring cells does not have sufficient channels to support the handover. In such cases the particular call is dropped. The probability of a call to be dropped can be calculated by the probability of handover failure and is given by

$$P_{hf}(i = m) = \left(\frac{\lambda_n + \lambda_h}{\mu_c}\right)^{m-g} \frac{1}{m!} \left(\frac{\lambda_h}{\mu_c}\right)^g P_0 \tag{3.57}$$

where λ_n is the average intensity of the new traffic, λ_h is the average rate of handover towards the cell, and m is the the total channels available, g is the number of guard channels and i is the number of channels in use.

3.11.3 Call Drop Probability (CDP)

The forced termination probability, P_d, is defined as a handover call which will be dropped as the UE moves from one cell to another.

$$P_d = \sum_{i=1}^{\infty} P_h^i (1 - P_{hf})^{(i-1)} P_{hf} = \frac{P_h P_{hf}}{1 - P_h(1 - P_{hf})} \tag{3.58}$$

where P_h is the probability of handover, P_{hf} is the probability of handover failure.

3.11.4 Call Block Probability (CBP)

The call block probability in the new cell is the call that will not be accepted by a network due to the lack of available channels in the new cell.

$$P_b(i = m) = \sum_{j=m-g}^{m} P_j = \left(\frac{\lambda_n + \lambda_h}{\mu_c}\right)^{m-g} \sum_{j=m-g}^{m} \frac{1}{j!} \left(\frac{\lambda_h}{\mu_c}\right)^{j-(m-g)} P_0 \tag{3.59}$$

where P_j is given by,

$$P_j = \frac{1}{j!} \left(\frac{\lambda_n + \lambda_h}{\mu_c}\right)^{j} P_0, 1 \leq j \leq m - g \tag{3.60}$$

and P_0 is given by,

$$P_0 = \frac{1}{\sum_{j=0}^{m-g} \frac{\rho^j}{j!} + \rho^{m-g}. \sum_{j=m-g+1}^{m} \frac{\rho_h^{j-(m-g)}}{j!}}, m - g + 1 \leq j \leq m \tag{3.61}$$

where λ_n is the average intensity of the new traffic. λ_h is the average rate of handover towards the cell and is given by,

$$\lambda_h = \frac{P_h(1 - P_b)}{1 - P_h(1 - P_{hf})} \lambda_n \tag{3.62}$$

If the call and block probabilities are negligible,

$$\lambda_h \approx \frac{P_h}{1 - P_h} \lambda_n, P_b, P_{hf} \ll 1 \tag{3.63}$$

m is the the total channels available, g is the number of guard channels and i is the number of channels in use.

4

Methodology

This thesis introduces two approaches to mobility control. These approaches, however, are explained in chapter 5 and 6. This chapter discusses methods and tools that are not directly related to mobility control. A prototype simulator that evaluates the two approaches is presented section 4.1, as well as a few other methods related to UE mobility are discussed in subsection 4.2.1, 4.2.2, and 4.2.3.

4.1 Simulation Setup

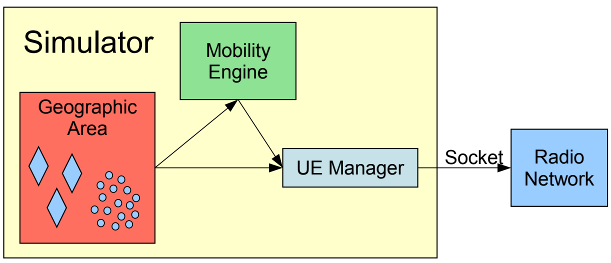


Figure 4.1: *Simulator overview*

The simulation setup consists of two applications:

- a prototype simulator to evaluate the theories described in detail in chapter 5 and 6.
- an emulated radio network to act as the "system under test".

The simulator implements the concepts discussed in chapter 5 and 6, while the emulated radio network is the "system under test". The two applications communicates through a single TCP datagram connection.

While simulating, the simulator sends measurement reports to the radio network and the radio network replies with handover commands. Both the simulator and the radio network applications were developed in C++, with the help of the Qt framework and libQxt.

Sections 4.1.1 through 4.1.5 explain the major parts of the prototype simulator and the radio network, while sections 4.1.6 through 4.1.12 explain certain aspects and details to the prototype simulator.

4.1.1 Prototype Simulator Overview

The theories detailed in chapter 5 and 6 are investigated by implementing them in a prototype simulator. This simulator is divided in 3 parts: the geographic area, the UE manager, and a mobility engine, i.e. 3 main classes (see Figure 4.1). These parts, and the radio network block, are explained in more detail in subsequent sections.

4.1.2 Geographic Area

A geographic area can be regarded as a pixmap where each pixel contain a list of calculated path loss between the pixel and the base stations in its vicinity. Some pixels, of course, contain base stations as well.

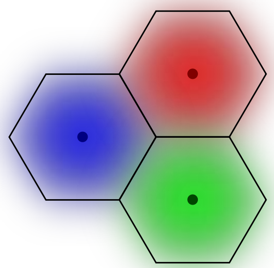


Figure 4.2: *Illustrated signal strength with no fading effects*

To calculate the base station signal strength in a pixel, the calculated path loss and fading effects are subtracted from the output power of the base station. Since the Geographic area only contain path loss, fading effects can be generated at run-time. The signal strength of base station N, measured at pixel (x, y), is calculated by:

$$P_N(x, y) = P_{out,N} - (P_{L,N}(x, y) + P_F), \tag{4.1}$$

where $P_{out,N}$ is the output power of the base station, $P_{L,N}(x, y)$ is the calculated path loss between base station N and pixel (x, y), and P_F is the net sum of all contingent fading effects. The prototype simulator support either Rayleigh fading or no fading at all.

The geographic area is shared between the currently active mobility engine and the UE manager. For more detail, read below.

4.1.3 Mobility Engines

Mobility engines (MEs) are defined in this thesis as the part of code that calculates the UE movement paths. The MEs implement the concepts more thoroughly explained in chapter 5 and 6. Although the two mobility engines present two different approaches, they share a common code base. Therefore, certain functionality is present in both mobility engines.

4.1.4 UE Manager

To assist the Mobility Engines (see subsection 4.1.3) in UE handling, an UE Manager is used. The manager maintain a list of UEs and manages their positions in the geographic area and each UE's serving base station.

It is also the manager that holds the connection to the radio network. Upon request from a Mobility Engine (ME), the UE manager will send measurement reports to the radio network, update UE positions, and act upon received handover commands.

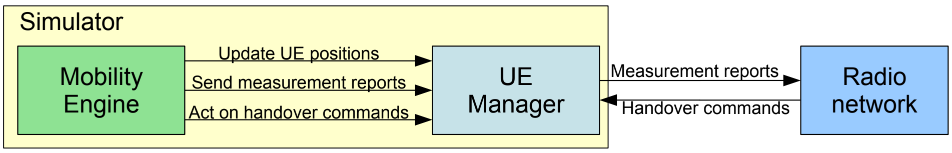


Figure 4.3: *How the UE Manager relates to other parts of the simulator*

4.1.5 Radio Network

Instead of building a simulator for an specific existing network technology, which could introduce considerable coding overhead, a simplified network topology is used. This topology contains three elements

- UEs
- Base stations
- Radio network

In this simplified network, UEs are connected to a serving base station. The base stations routes data traffic from the UEs to the radio network, or to other UEs connected to the same base station. The radio network decides when UEs should change their serving base station, i.e. make handovers.

This network topology is implemented in a separate application called Radio Network Stub[1] (RNS). Through a network socket connection to the simulator, the RNS receives measurement reports from UEs and, based on these reports, issues handover commands. To determine which UE should be handed over and when, the RNS utilizes the LTE Standard Hard Handover algorithm.

[1]Test stubs are programs which simulate the behaviors of software or hardware components that are depended-upon modules of the module being tested[32]. Here, this refers to the RNS being a test stub to replace a real hardware node in testing the prototype simulator.

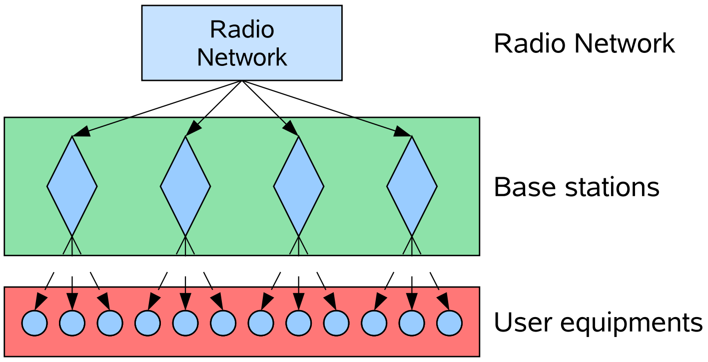


Figure 4.4: *Simplified Network Topology*

The RNS use a slightly modified variant of the LTE standard handover algorithm. When an UE satisfies Equation 3.22, a Time-To-Trigger (TTT) timer is started but, unlike the LTE handover algorithm, a handover is not triggered automatically when the timer expires. A handover is only triggered when the RNS recieves a measurement report from the UE and the UE's TTT timer has expired.

However, with a high enough simulator update rate, the RNS will recieve measurement reports more often than the required LTE TTT timer resolution, effectively making the RNS handover algorithm equivalent to the standard LTE hard handover algorithm. The default simulator update rate is 20 ms.

This is the "system under test".

Subsequent sections contain details on some important aspects of the prototype simulator.

4.1.6 Calculating Handover Rate

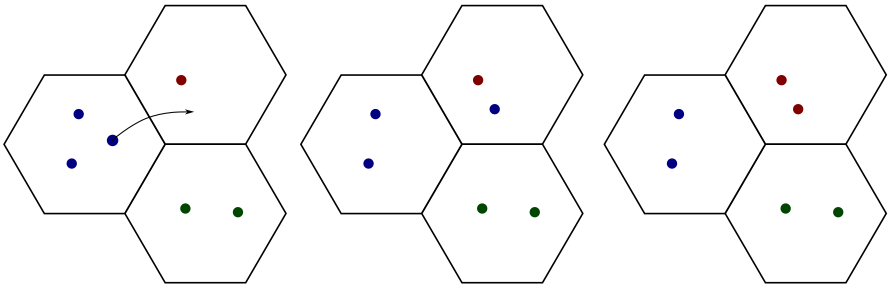
Whenever a UE is moved, a measurement report is sent to the RNS. Acting upon these measurement reports, the RNS issues handover commands. These handover commands arrives asynchronously and are saved in a list in the UE manager. Upon the request from a mobility engine, the manager goes through the list and execute the handovers.

When a mobility engine is running, the manager is requested to execute handovers on regular time intervals (ΔT). The handover process is illustrated in Figure 4.5, where $0 \leq \delta < \Delta T$.

The handover rate can then be calculated as

$$\lambda_h = \frac{N}{\Delta T}, \tag{4.2}$$

where N is the number of handover commands that has arrived during ΔT. This can also be used to determine the rate of handover and departure calls for individual cells by replacing N with the appropriate quantity.



(a) $t = 0$ *(b)* $t = n \cdot \Delta T + \delta$ *(c)* $t = (n+1) \cdot \Delta T$

Figure 4.5: *(a) UE enters another cell. (b) A handover command arrives to the UE manager from the RNS. (c) The manager executes the handover command upon request from a mobility engine.*

4.1.7 Cell Area

The cell area is used in Equation 3.56 for the calculation of the handover probability. The cell area is calculated as the total number of pixels present within the cell area:

$$A = \sum_{x=0}^{width} \sum_{y=0}^{height} \Delta A * p_a(x, y), \tag{4.3}$$

where ΔA is the area of one pixel and

$$p_a(x, y) = \begin{cases} 1 & \text{if } (x, y) \text{ is inside the cell} \\ 0 & \text{if } (x, y) \text{ is outside the cell} \end{cases} \tag{4.4}$$

A pixel is considered to lie within the cell if the highest recorded signal strength in that pixel belongs to the cell.

This can be further illustrated through Figure 4.6

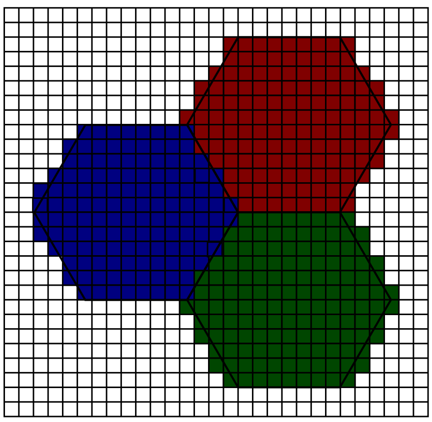


Figure 4.6: *Area Calculation*

4.1.8 Cell Perimeter

Since the geographic area is treated as a pixmap, image processing techniques can be used to calculate the perimeter length of a cell as the sum of edge pixels in the cell area. The perimeter is used in Equation 3.56.

$$L = \sum_{x=0}^{width} \sum_{y=0}^{height} \Delta l \times p_e(x, y), \tag{4.5}$$

where Δl is the average length through a pixel and

$$p_e(x, y) = \begin{cases} 1 & \text{if } (x, y) \text{ is an edge pixel} \\ 0 & \text{if } (x, y) \text{ is not an edge pixel} \end{cases} \tag{4.6}$$

Edge pixels can be detected by first creating a binary image where the pixels which belong to the cell are represented by 1 and all other pixels are 0 (see subsection 4.1.7).

Then, this binary image is convoluted[2] with the kernel K in Equation 4.8. A pixel is regarded as an edge pixel if

$$2 \leq N(x, y) * K \leq 5, \tag{4.7}$$

where $N(x, y)$ is the pixel with a $3x3$ neighborhood at position (x, y) and

$$K = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}. \tag{4.8}$$

The interpretation of Equation 4.7 is that a pixel is regarded as a cell border pixel if it is surrounded by two or more, but five or less pixels that lies outside the cell. If $N(x, y) * K < 0$, it means that the pixel is not part of the cell. Figure 4.7 show Figure 4.6 overlayed with pixels detected as border pixels.

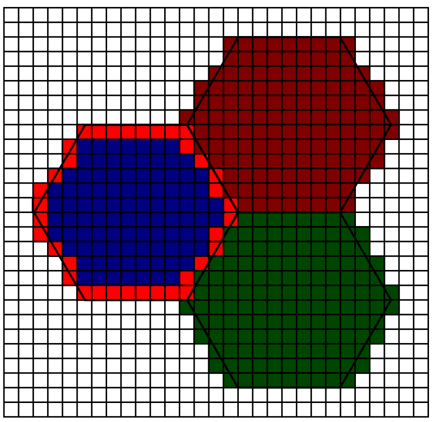


Figure 4.7: *Perimeter Calculation*

[2]See Appendix A for how this image convolution is done

4.1.9 Traffic Model

The simulator only simulates UEs that is in an active call. No calls are dropped and no new calls are made. Additionally, no calls are handed over from other radio network technologies, i.e. only homogenous handovers are simulated.

4.1.10 UE Direction

By default, when deciding in what direction a UE should move, the signal strength of nearby BSs are compared and the UE's direction is set toward the BS with the highest signal strength. See subsection 4.2.1, 4.2.2, and 4.2.3 for extensions to this decision algorihtm, all of which were implemented in the simulator.

4.1.11 Forbidden Cells

BSs in the simulator can be configured so that they are excluded when deciding a UE's direction. This is referred to as marking a BS or cell as forbidden. A UE's direction is not set toward any BSs that are marked as forbidden, but the UE might still enter the forbidden cell if it is moving toward a BS beyond the forbidden cell.

4.1.12 Probability of Handover Failure

For calculating $P_{HO}(x, y)$ in chapter 6, only the probability of handover (Equation 3.55) was used. The handover failure probability, Equation 3.57, tends to be negligible. Since both the handover block and drop probabilities are based on handover failure probability, they too become negligible. Additionally, since the traffic model in the prototype simulator does not consider any new calls, the call block probability is not used.

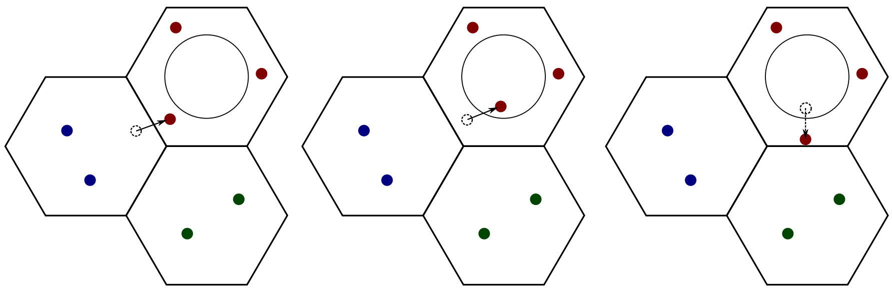
4.2 Other UE Mobility Methodology

The below sections present some theories and methodology that is not directly linked to the simulation setup or any prototype simulator. They are however accessable and configurable in the prototype simulator as optional features.

4.2.1 Velocity Change Radius

Depending on the mobility control concept (see chapter 5 and 6), whenever an UE receives a handover it will change its velocity (speed and direction) to move towards the BS with the highest signal strength that is not their serving BS. If all handovers occur as soon as an UE crosses a cell border, the UEs will only move on the outskirts of a cell. They will simply move rapidly back and forth between two cells.

This behavior is likely to trigger *ping-pong* countermeasures in the hardware node being tested. If such countermeasures are not to be tested or desired, a velocity change radius can be introduced. UEs who have received a handover will not change their velocity until they are within a certain radius of their serving base station. This radius is the velocity change radius (see Figure 4.8, the circle within the top cell is the velocity change radius).

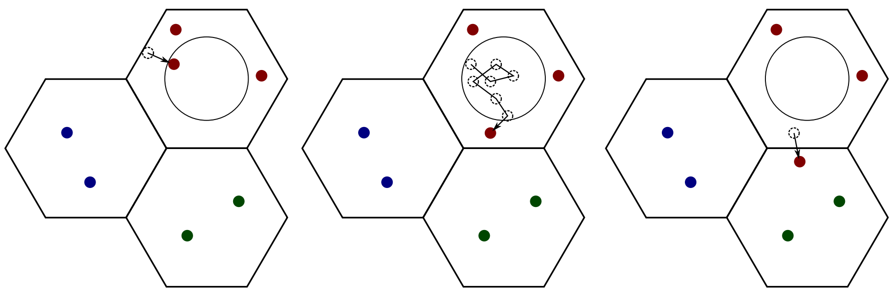


(a) $t = 0$ **(b)** $t = n \cdot \Delta T$ **(c)** $t = (n+1) \cdot \Delta T$

Figure 4.8: *(a) A UE is handed over to a new cell but keeps his current velocity. (b) The UE is moved within the velocity change radius and gets a new velocity. (c) The UE uses the new velocity to move towards the next cell.*

4.2.2 Random Center Movement

In order to make UE movement less deterministic, the UEs can be moved randomly within the velocity change radius. Once a UE exits this area (randomly) it is given a more fixed velocity.



(a) $t = 0$ **(b)** $t = n \cdot \Delta T$ **(c)** $t = (n+1) \cdot \Delta T$

Figure 4.9: *(a) A UE is moved within the velocity change radius. (b) For every ΔT, the UE is given a random velocity until it is moved outside the velocity change radius. (c) The UE is moved normally.*

4.2.3 Base Station History

If the mobility behavior of a UE is to always move towards the base station with the highest signal strength, except its serving base station, it will continuously move back and forth between the same two cells.

To avoid this behavior, the UE's previous serving base stations can be excluded when calculating the UE's next direction. If a new direction cannot be determined under these conditions, the last (oldest) serving base station is used for direction. The number of previous serving base stations can of course be constrained.

5

Task Based Mobility Control

Outlined below is a theory on how UE mobility can be controlled by assigning each UE with a task which describes its mobility behavior. It can be compared to grouping UEs with different mobility behaviors. To make the assignment of tasks dynamic, with as little manual interaction and adjustments as possible, theories and concepts from the area of control theory is used.

5.1 Task Definition

Tasks are a way to tag UEs with different mobility behaviors. A UE might be assigned with the task to

- generate a handover, e.g. move towards an other cell until it receives a handover.
- move to a certain cell, and thus increase the population density in that cell.
- leave its current cell, and thus decrease the population density in its current cell.
- stand still, and wait to be assigned an other task.

UEs will perform their assigned task and then stand still. UEs that are standing still can be regarded as a pool of available UEs that can be assigned with any new task. How UEs are assigned a task is described more thoroughly in the next section.

5.2 Task Assignment using Control Theory

Determining how many and which UEs should be assigned a certain task can be made more dynamic by using theories and principles from control theory. This has the advantage of a strong theoretical and practical background, and many of the principles developed for physical control systems can be applied to determining tasks for UEs.

5.2.1 Handovers per Second

Consider the case where a fixed number of handovers per second is supposed to be generated. This fixed number is the input to the control system. The output is the number of handovers that is actually issued by the radio network. The difference between the input and the output forms the error.

$$r(t) = \text{Requested handovers per second (input)} \tag{5.1}$$
$$y(t) = \text{Measured handovers per second (output)} \tag{5.2}$$
$$e(t) = r(t) - y(t) \tag{5.3}$$

If the error is positive, not enough handovers are generated and more UEs should be assigned the task to generate a handover; if the error is negative, too many handovers are generated and more UEs should be re-assigned with the task to stand still. This single input, single output (SISO) system can be described by the closed-loop system in figure 5.1.

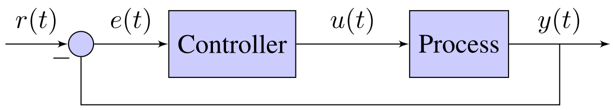


Figure 5.1: *Feedback control system*

Process description

The *Process* block represent assigning tasks and moving UEs; sending measurement reports and acting upon handover commands; and measuring the current rate of handover. This process is controlled by the signal

$$u(t) = \begin{cases} \textit{UEs should be assigned the task to generate a handover,} & u(t) \geq 0 \\ \textit{UEs should be re-assigned with the task to stand still,} & u(t) \leq 0 \end{cases} \tag{5.4}$$

This signal is bounded by

$$-U_{LB}(t) \leq u(t) \leq U_{UP}(t) \tag{5.5}$$

where

$$U_{LB}(t) = \textit{Number of UEs generating a handover}$$
$$U_{UB}(t) = \textit{Number of UEs standing still}$$

That is, the signal's upper bound is the number UEs that are standing still. If there are no UEs currently standing still, no more UEs can be assigned with the task to generate a handover. The reverse is true for the minimum bound. If there are no UEs currently generating a handover, no more UEs can be re-assigned with the task to stand still.

For control theory purposes, the process is mathematically modeled as a pure time delay

$$\frac{Y(z)}{U(z)} = G(z) = z^{-k} \tag{5.6}$$

Controller

For controlling handover task assignment, a discrete PID controller with a Smith predictor was used (see section 3.10). Additionally, a low-pass filter was applied on the input to the controller to reduce the effect of spikes in the error signal. This low-pass filter was implemented as a moving average of 10 samples.

5.2.2 Cell Density

The density of a cell is defined as the ration of used resources and available resources.

$$\rho_c = \textit{Cell density} = \frac{\textit{Used resources}}{\textit{Available resources}} \tag{5.7}$$

Consider the case where the cell density of an individual cell is supposed to be kept at a certain level. This can be controlled by the same type of closed-loop system (Figure 5.1) and reasoning as in the previous section, but with the following changes

$$r(t) = \text{Wanted cell density (input)} \tag{5.8}$$

$$y(t) = \text{Measured cell density (output)} \tag{5.9}$$

$$u(t) = \text{Number of UEs to move towards or out of the cell.} \tag{5.10}$$

$$e(t) = r(t) - y(t) \tag{5.11}$$

If the error is positive, the cell density is too low and more UEs should be assigned the task to move towards the cell; if the error is negative, too many UEs are already in the cell and more UEs should be re-assigned with the task to leave the cell.

Process description

The *process* block represent assigning tasks, moving UEs, and measuring the cell density. This process is controlled by the signal

$$u(t) = \begin{cases} \textit{UEs that should be assigned the task to move towards the cell,} & u(t) \geq 0 \\ \textit{UEs that should be re-assigned the task to move out of the cell,} & u(t) \leq 0 \end{cases} \tag{5.12}$$

Analogous to the process description in subsection 5.2.1, $u(t)$ is bounded by

$$-U_{LB}(t) \leq u(t) \leq U_{UP}(t) \tag{5.13}$$

where

$$U_{LB}(t) = \textit{Number of UEs in the cell}$$
$$U_{UB}(t) = \textit{Number of UEs standing still.}$$

The upper bound is the same: number of UEs that are standing still. The lower bound, however, is how many UEs are in the cell, no matter what previous task they have.

For control theory purposes, the process is mathematically modeled as a pure time delay

$$\frac{Y(z)}{U(z)} = G(z) = z^{-k} \tag{5.14}$$

Controller

For controlling the cell density, a discrete PID controller was used (see section 3.10).

6

Probability Based Mobility Control

The probability based mobility control is an approach to UE mobility control where the mobility of the UE is pre-calculated and evaluated before execution. This approach uses the equations presented in the section 3.11 to calculate the movement path of each and every UE in a geographic area. If the calculated path does not generate the specified mobility characteristics, the parameters are changed and the movement path is re-calculated.

6.1 Overview

The probability based mobility control approach can be explained as a two step process.

The first step involves moving the UEs in a geographic area without any connection to a radio network. The movement path for each and every UE is recorded. At the end of the calculation, the total handover rate is calculated by dividing the noted handovers with the complete runtime of the simulation.

The second step is to check whether the noted handover rate in the first step is within the given specification. If it is within the specification, the simulator connects to a radio network and replay the UE movement recorded in the first step. If it does not satisfy the specifications, then parameters of the UEs are changed and the entire path is re-calculated.

The various steps that are followed in calculating the movement path can be explained through the flow chart in Figure 6.1.

6.1.1 Initialize

The pre-requisites for calculating the movement path is to have a geographic area, complete with BSs and their path loss to different coordinates in the area. Also, UEs are distributed, in some way (e.g. evenly), in that geographic area.

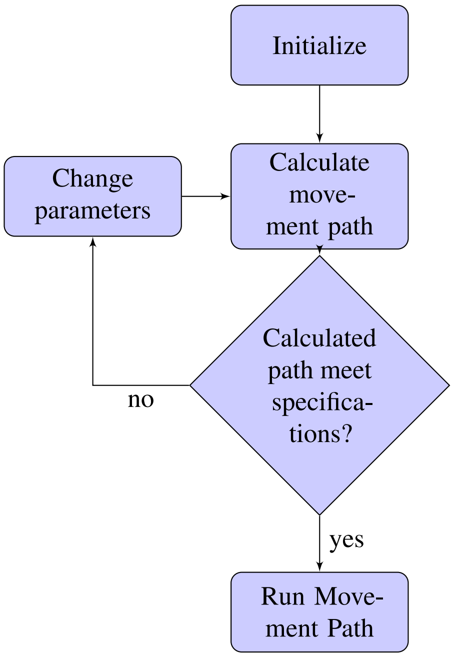


Figure 6.1: *Overview of Mobility Control*

6.1.2 Calculate Movement Path

The movement path is calculated for a specified time. At each time instant, $n \cdot \Delta T$, all UEs are moved one step. If the UE enters a new cell, a number ρ is chosen uniformly and randomly from $[0, 1]$ and compared to the calculated handover probability at the current position of the UE – $P_{HO}(x, y)$[1]. If $\rho \leq P_{HO}(x, y)$, then a handover is noted to have taken place and the UE is assigned the new cell as its serving cell and the direction of the UE is updated. This can be explained through Algorithm 1.

6.1.3 Calculated Path Meets Specifications

The next step is to check whether the calculated path meets the specifications of generating the required handover rate. If the path generates the handovers required, then the simulator connects to the radio network and replay the recorded path. If it does not satisfy the condition, the simulation parameters are changed (see subsection 6.1.4). Algorithm 2 gives an overview of this process.

The generated handover rate could be considered "good enough", if they lie within a small interval about the specified target level, e.g. 950-1050 handovers per second for 1000 HO/s target level.

[1] This calculated probability could be the combination of probability of handover, call block probability, call drop probability, and probability of handover failure (see Equation 3.55, 3.57, 3.58, and 3.59)

Algorithm 1 Movement Path Calculation

```
for t = 0, 1·ΔT, 2·ΔT, ..., T_final do
    for all UEs do
        Move UE
        if UE is in new cell then
            Roll dice
            if P_ho >die result then
                Note handover
                Set UE serving cell
                Update UE direction
            end if
        end if
    end for
end for
```

Algorithm 2 Specification Condition

```
if Noted handover rate is within specification then
    Connect to network
    Play back recorded UE movement path
else
    change parameters and recalculate
end if
```

Startup Spike

When the simulation is run, there is the possibility of a startup spike. This particular spike occurs when all the UEs start to move away from their base station to generate handovers. As a result of this movement, there is a large number of noted handover which is more than the required specifications. To reduce the influence of this startup spike on the final value of the handover rate, handovers are noted only after a certain delay.

6.1.4 Change Parameters

If the specification is not met, the parameters are changed. After the parameters are changed, the movement path is re-calculated. The variable that mainly affects the handover probability is the speed of the UE.

Table 6.1: *Effects of change in parameters*

	Speed	Perimeter to Area Ratio
Decrease	Decreases Handover Probability	Decreases Handover Probability
Increase	Increases Handover Probability	Increases Handover Probability

In this mobility control, the noted handovers are mainly affected by the speed of the UE. The speed of the UE is increased or decreased depending upon the "good enough" handover rate is obtained. When the handover rate is less than the specification, the speed of the UEs are increased to satisfy the required specification. The same can be said for its contradiction.

In the handover probability calculation, the area and the perimeter of the cell in which the base stations are located are constant.

But when there is an increase in the perimeter to area ration, there is an increase in handover probability and the same is true for the reverse.

In addition to that, the strength of the base stations are also constant.

6.1.5 Run Movement Path

Once the specifications are met, the optimum calculated movement path is executed in a radio network controller.

7

Result

This chapter represents the simulation results from the prototype simulator presented in section 4.1. The two mobility control concepts in chapter 5 and 6 were simulated with near identical simulation setups. Both concepts support mobility control for a fixed handover rate, but only the task based mobility control concept support control of cell density.

7.1 Handover Rate

Three different handover rate targets were simulated:

- 500 handovers/second
- 1000 handovers/second
- 2000 handovers/second

These target handover rates reflect the total number of handover commands issued by the RNS (section 4.1) per second, i.e. they do not reflect the handover rate of individual cells.

Each section below begins with a simulation setup, followed by simulation data statistics and details about the first second(s), and ends with the simulation results as a function of time and their probability density functions.

In the simulation settings below, *Delta speed* refers to how much the UE speed is changed when the calculated handover rate doesn't meet the specifications. For more details about the different simulation settings, see section 4.1. To avoid the effects of a startup spike, some results are presented with the first number of samples removed. Additionally, since the handover rate is measured over ΔT, which can be sub-second, the measured handovers/second rate can be larger than the actual number of UEs in the area.

7.1.1 Handover Rate - 500 HO/s

The simulation setups are presented in Table 7.1. Data statistics for full length simulations are found in Table 7.2 and Table 7.3. Control theory statistics for the very first second of simulation can be found in Table 7.4.

Figure 7.1 and Figure 7.2 show the first second and full length simulation results, respectively. In Figure 7.3, the probability density functions of Figure 7.2 are presented.

Table 7.1: *Simulation setup*

	Task Based	**Probability Based**
Area	200x200 m^2	200x200 m^2
Base stations	36	36
ΔT	20 ms	20 ms
Velocity change radius	10 m	10 m
Simulation runtime	100 s	100 s
Number of UEs in area	841	841
Random center movement	no	no
UEs idle after Handover	no	
Low speed (3 m/s) UEs	20 %	
Medium speed (30 m/s) UEs	70 %	
High speed (50 m/s) UEs	10 %	
K_P	0.5	
K_I	5.0	
K_D	0.0	
Smith controller used	no	
Starting speed		10 m/s
Delta speed		1 m/s
Final UE speed		22 m/s

Table 7.2: *Data statistics with startup spike. All entries in handovers/s*

	Task Based	**Probability Based**
Min	0	0
Max	2050	18850
Mean	498	582
Median	500	550
Mode	500	550
Stdev	187.7	358.7
Range	2050	18850

Table 7.3: *Data statistics with first 10 samples removed. All entries in handovers/s*

	Task Based	**Probability Based**
Min	0	50
Max	2050	1300
Mean	499	575
Median	500	550
Mode	500	550
Stdev	186.7	169.2
Range	2050	1250

Table 7.4: *Control theory performance metrics of the first second (task based approach only)*

Rise time	0.37 s
Settling time	0.93 s
Percentage overshoot	60.0 %
Peak	800 HO/s
Peak time	0.64 s

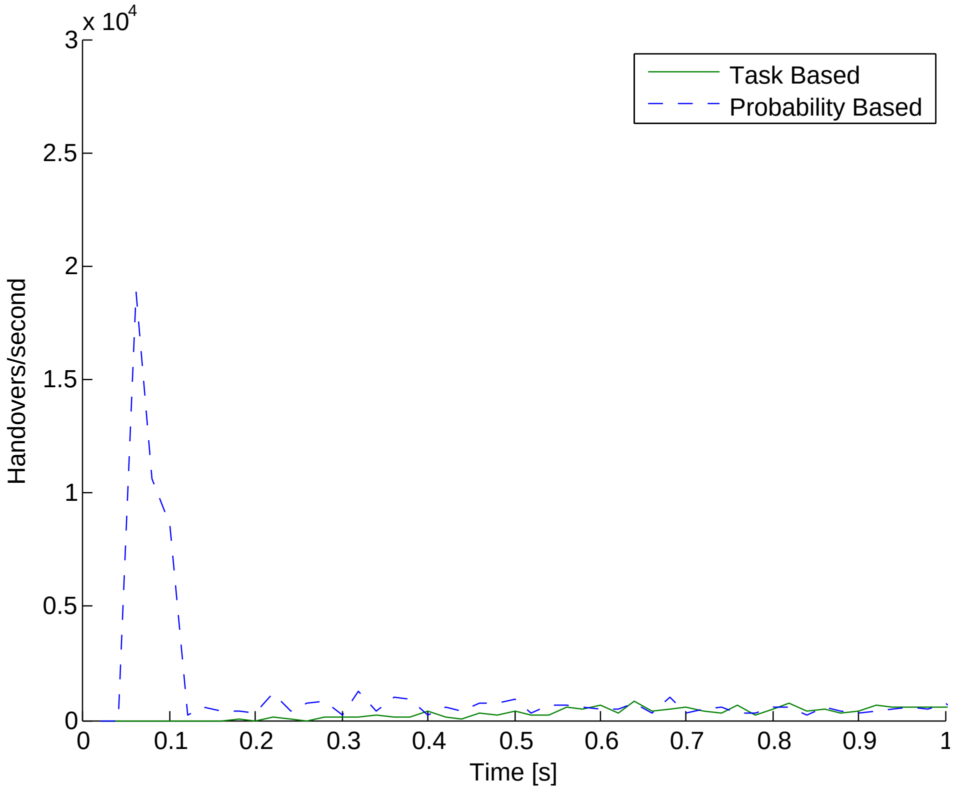
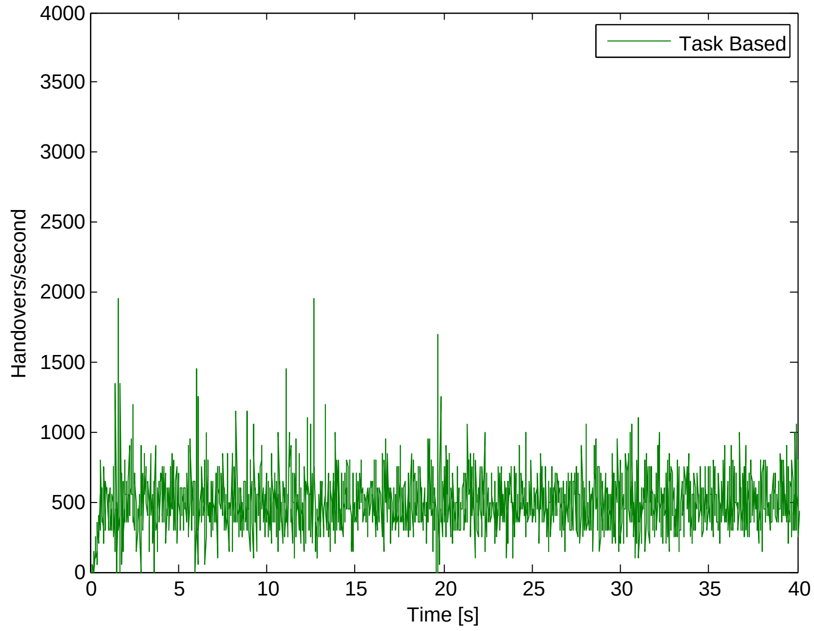
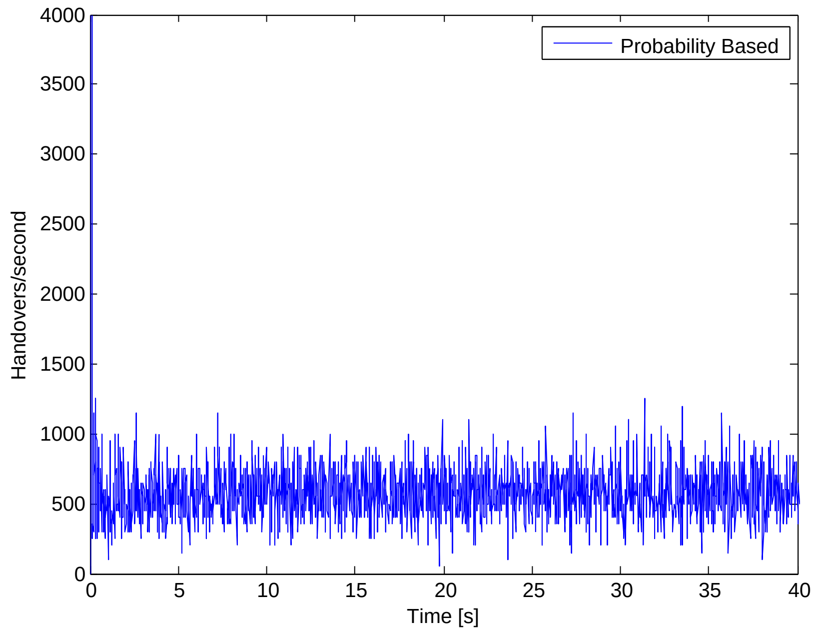


Figure 7.1: *First second with 500 HO/s target*

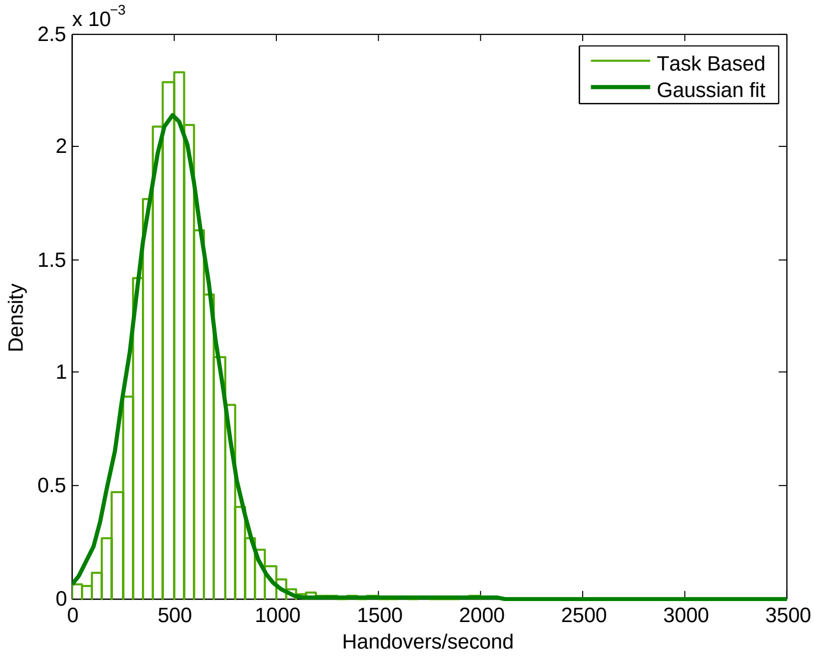


(a)

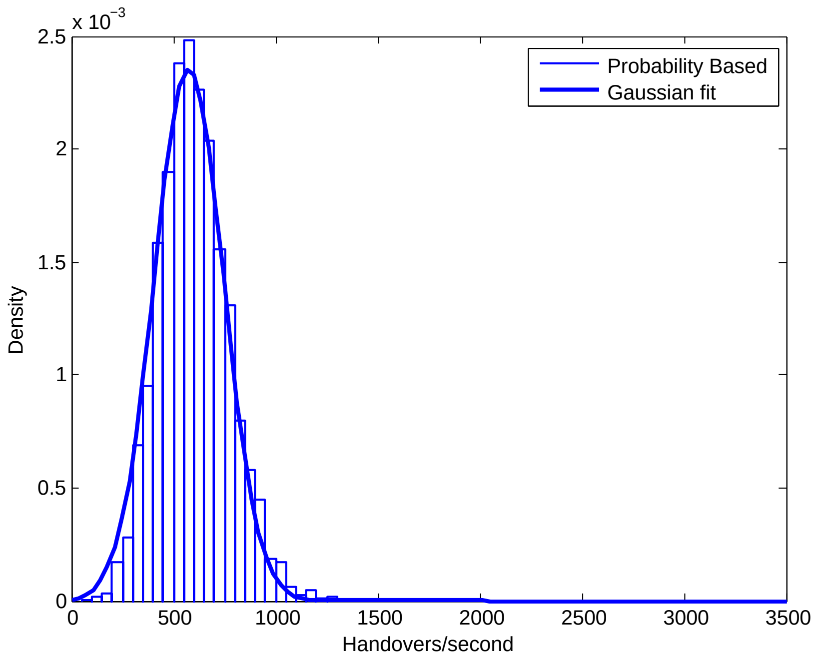


(b)

Figure 7.2: *Handover rate as a function of time, with (a) task based and (b) probability based mobility control, for a 500 HO/s target simulation*



(a)



(b)

Figure 7.3: *Probability density function, with (a) task based and (b) probability based mobility control, for a 500 HO/s target simulation with the first 10 samples removed*

7.1.2 Handover Rate - 1000 HO/s

The simulation setups are presented in Table 7.5. Data statistics for full length simulations are found in Table 7.6 and Table 7.7. Control theory statistics for the very first second of simulation can be found in Table 7.8.

Figure 7.4 and Figure 7.5 show the first second and full length simulation results, respectively. In Figure 7.6, the probability density functions of Figure 7.5 are presented.

Table 7.5: *Simulation setup*

	Task Based	**Probability Based**
Area	200x200 m^2	200x200 m^2
Base stations	36	36
ΔT	20 ms	20 ms
Velocity change radius	10 m	10 m
Simulation runtime	100 s	100 s
Number of UEs in area	841	841
Random center movement	no	no
UEs idle after Handover	no	
Low speed (3 m/s) UEs	0 %	
Medium speed (30 m/s) UEs	67 %	
High speed (50 m/s) UEs	33 %	
K_P	0.5	
K_I	5.0	
K_D	0.0	
Smith controller used	no	
Starting speed		10 m/s
Delta speed		5 m/s
Final UE speed		35 m/s

Table 7.6: *Data statistics with startup spike. All entries in handovers/s*

	Task Based	**Probability Based**
Min	0	0
Max	3150	18950
Mean	998	941
Median	950	900
Mode	900	850
Stdev	297.6	379.6
Range	3150	18950

Table 7.7: *Data statistics with first 10 samples removed. All entries in handovers/s*

	Task Based	**Probability Based**
Min	50	150
Max	3150	1950
Mean	1000	934
Median	950	900
Mode	900	850
Stdev	295.0	215.9
Range	3100	1800

Table 7.8: *Control theory performance metrics of the first second (task based approach only)*

Rise time	0.31 s
Settling time	0.85 s
Percentage overshoot	55.0 %
Peak	1550
Peak time	0.60 s

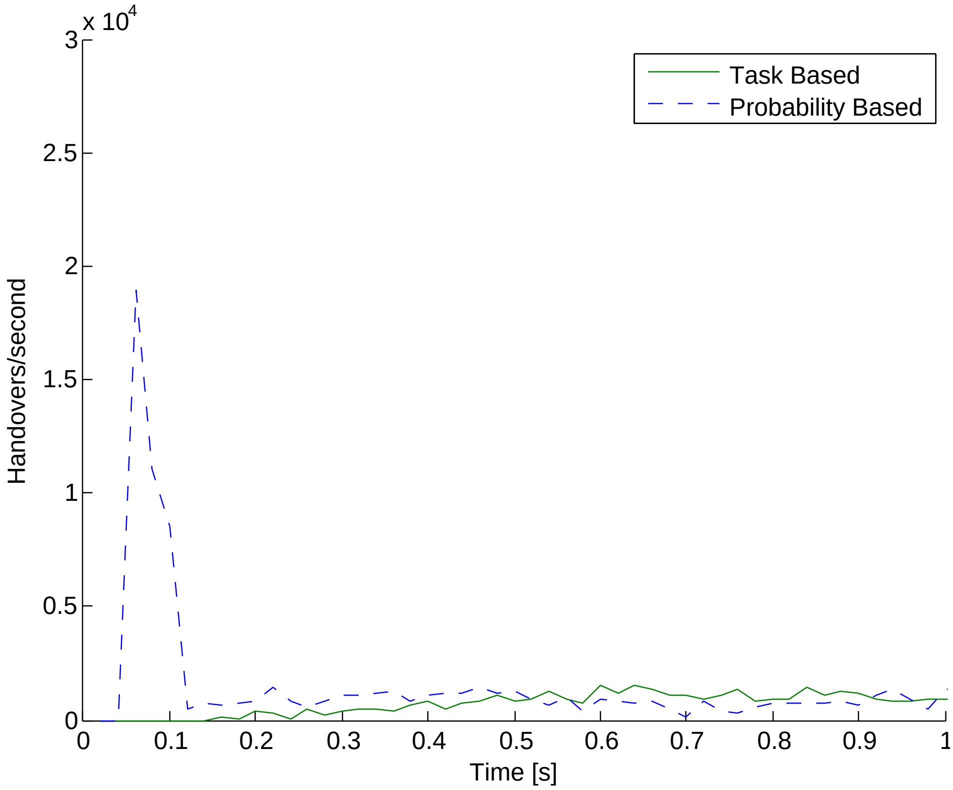


Figure 7.4: *First second with 1000 HO/s target*

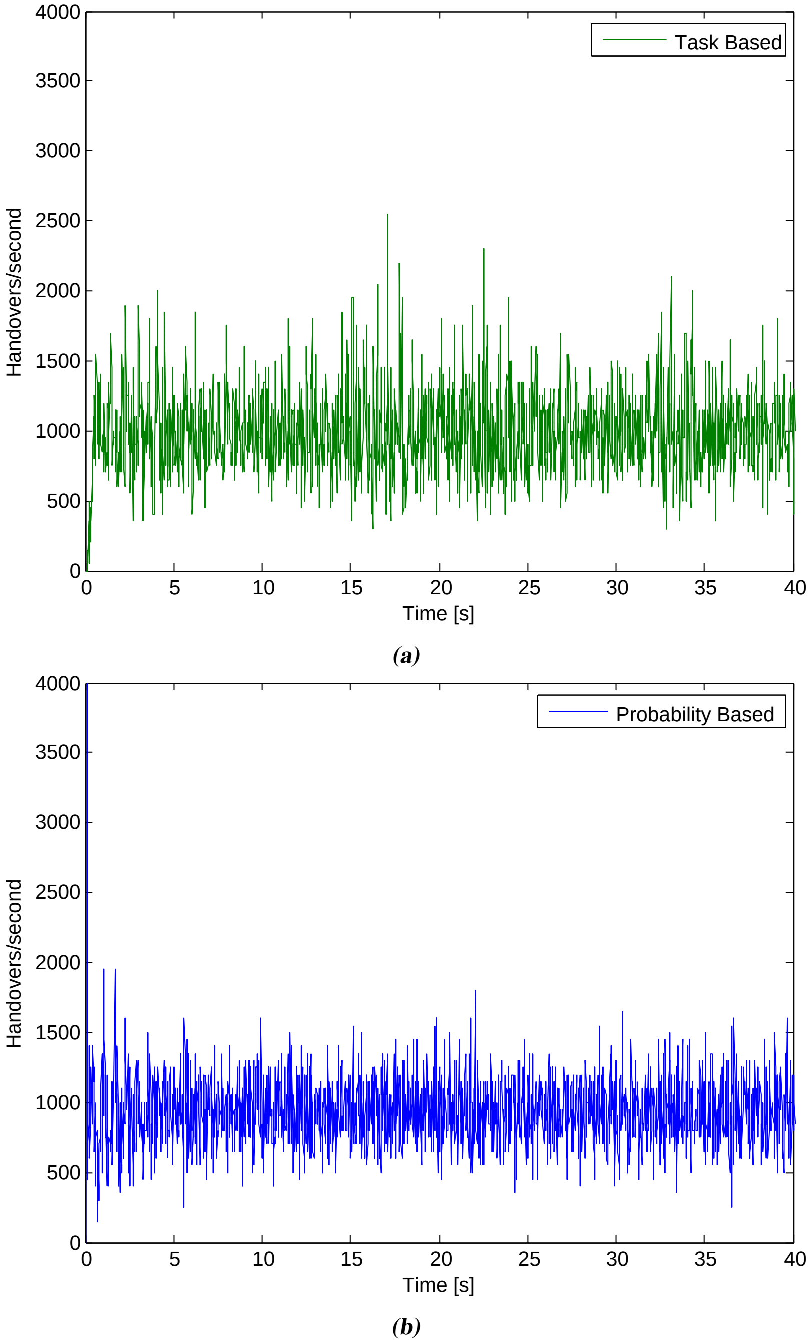
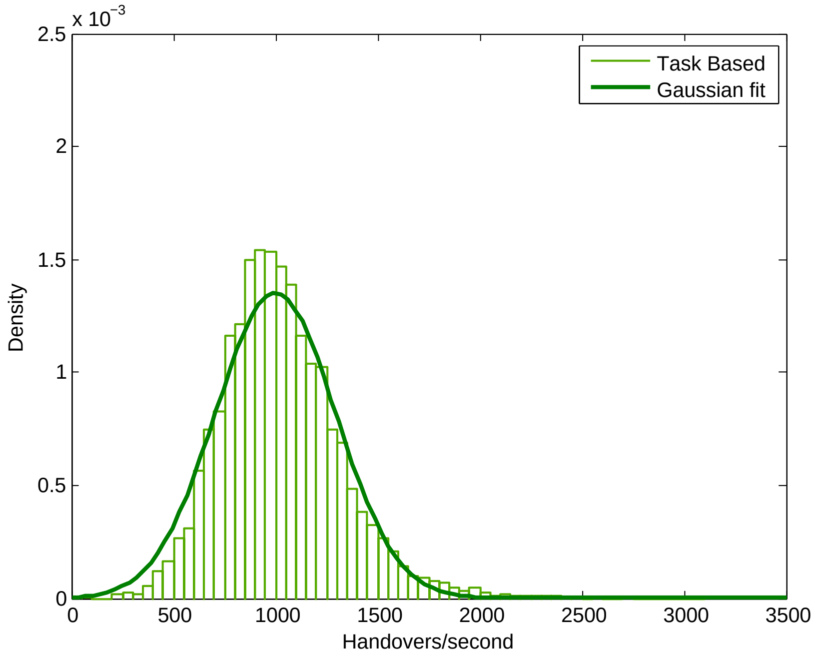
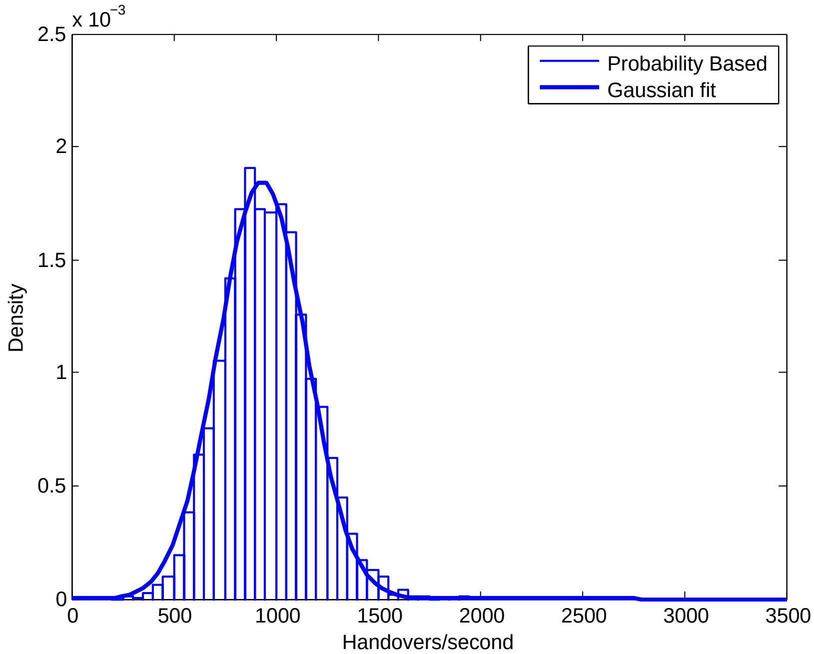


Figure 7.5: *Handover rate as a function of time, with (a) task based and (b) probability based mobility control, for a 1000 HO/s target simulation*



(a)



(b)

Figure 7.6: *Probability density function, with (a) task based and (b) probability based mobility control, for a 1000 HO/s target simulation with the first 10 samples removed*

7.1.3 Handover Rate - 2000 HO/s

The simulation setups are presented in Table 7.9. Data statistics for full length simulations are found in Table 7.10 and Table 7.11. Control theory statistics for the first two seconds of simulation can be found in Table 7.12.

Figure 7.7 and Figure 7.8 show the first two seconds and full length simulation results, respectively. In Figure 7.9, the probability density functions of Figure 7.8 are presented.

Table 7.9: *Simulation setup*

	Task Based	**Probability Based**
Area	200x200 m^2	200x200 m^2
Base stations	36	36
ΔT	20 ms	20 ms
Velocity change radius	10 m	10 m
Simulation runtime	100 s	100 s
Number of UEs in area	841	841
Random center movement	no	no
UEs idle after Handover	no	
Low speed (3 m/s) UEs	0 %	
Medium speed (30 m/s) UEs	0 %	
High speed (50 m/s) UEs	100 %	
K_P	0.5	
K_I	5.0	
K_D	0.0	
Smith controller used	no	
Starting speed		10 m/s
Delta speed		5 m/s
Final UE speed		80 m/s

Table 7.10: *Data statistics with startup spike. All entries in handovers/s*

	Task Based	**Probability Based**
Min	0	0
Max	3650	29450
Mean	1917	1973
Median	1900	1950
Mode	1800	1900
Stdev	401.2	494.2
Range	3650	29450

Table 7.11: *Data statistics with first 10 samples removed. All entries in handovers/s*

	Task Based	Probability Based
Min	750	1000
Max	3650	3050
Mean	1923	1969
Median	1900	1950
Mode	1800	1900
Stdev	390.9	300.7
Range	2900	2050

Table 7.12: *Control theory performance metrics of the first 2 seconds (task based approach only)*

Rise time	0.26 s
Settling time	1.95 s
Percentage overshoot	67.5 %
Peak	3350
Peak time	1.94 s

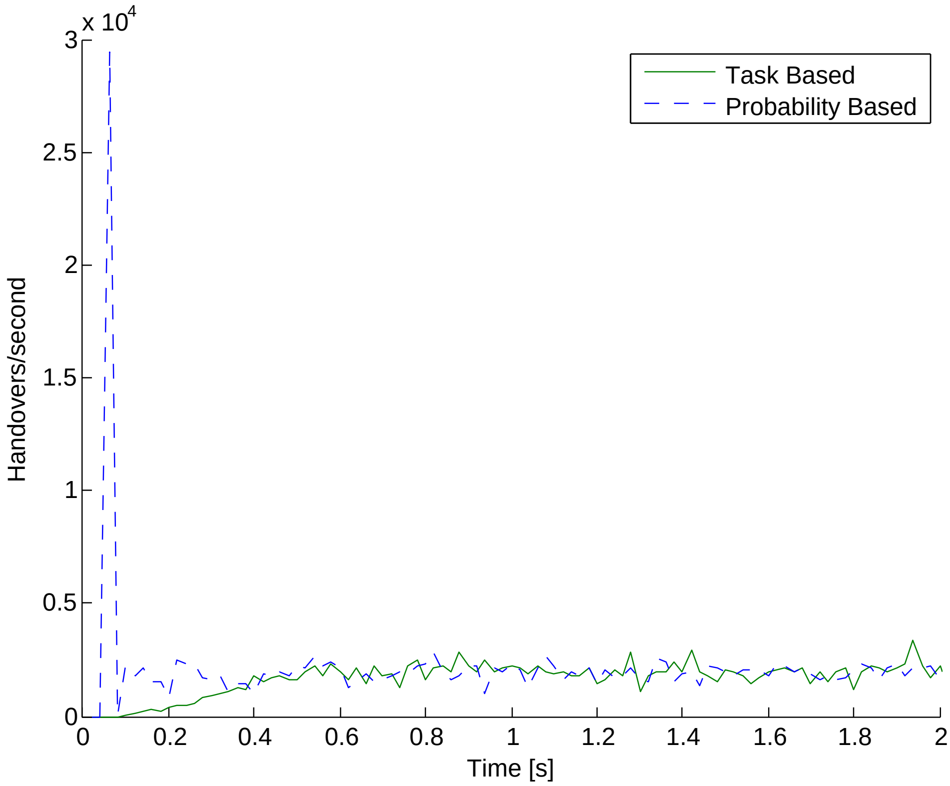
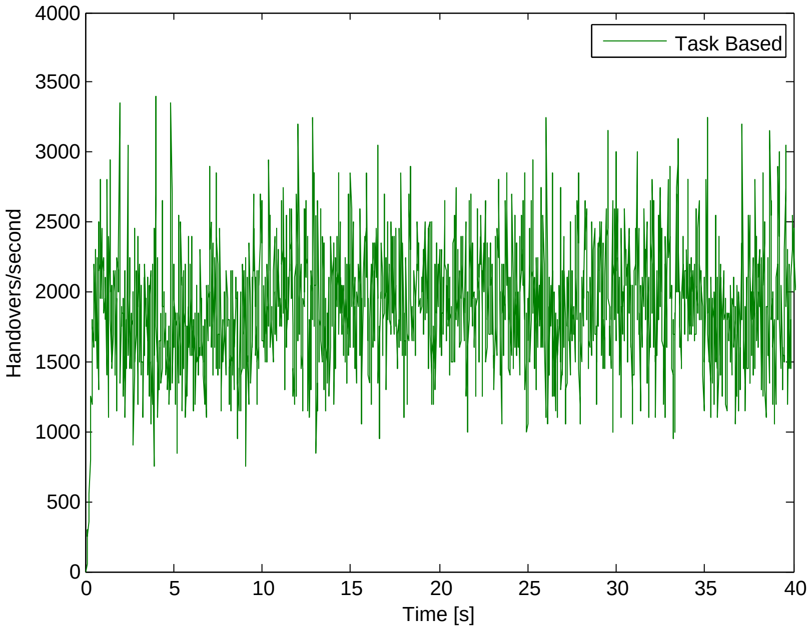
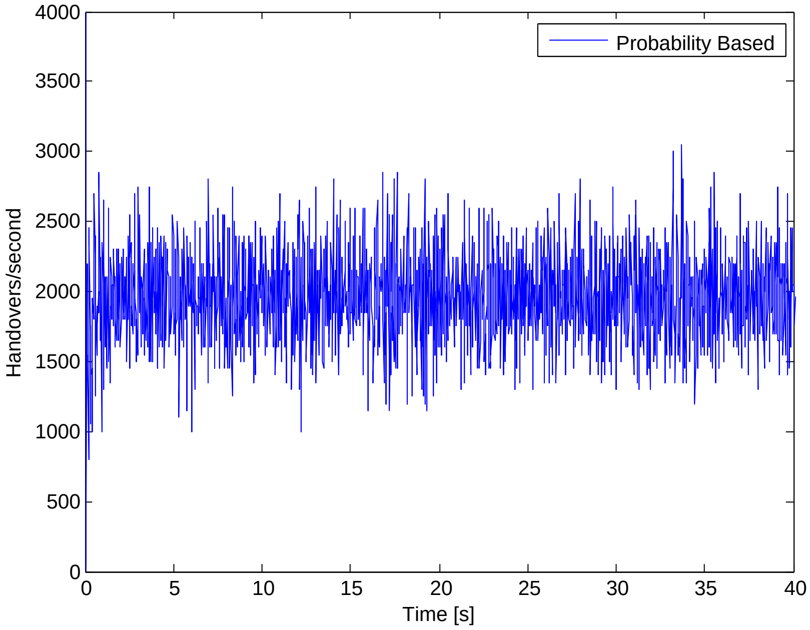


Figure 7.7: *First 2 seconds with 2000 HO/s target*



(a)



(b)

Figure 7.8: *Handover rate as a function of time, with (a) task based and (b) probability based mobility control, for a 2000 HO/s target simulation*

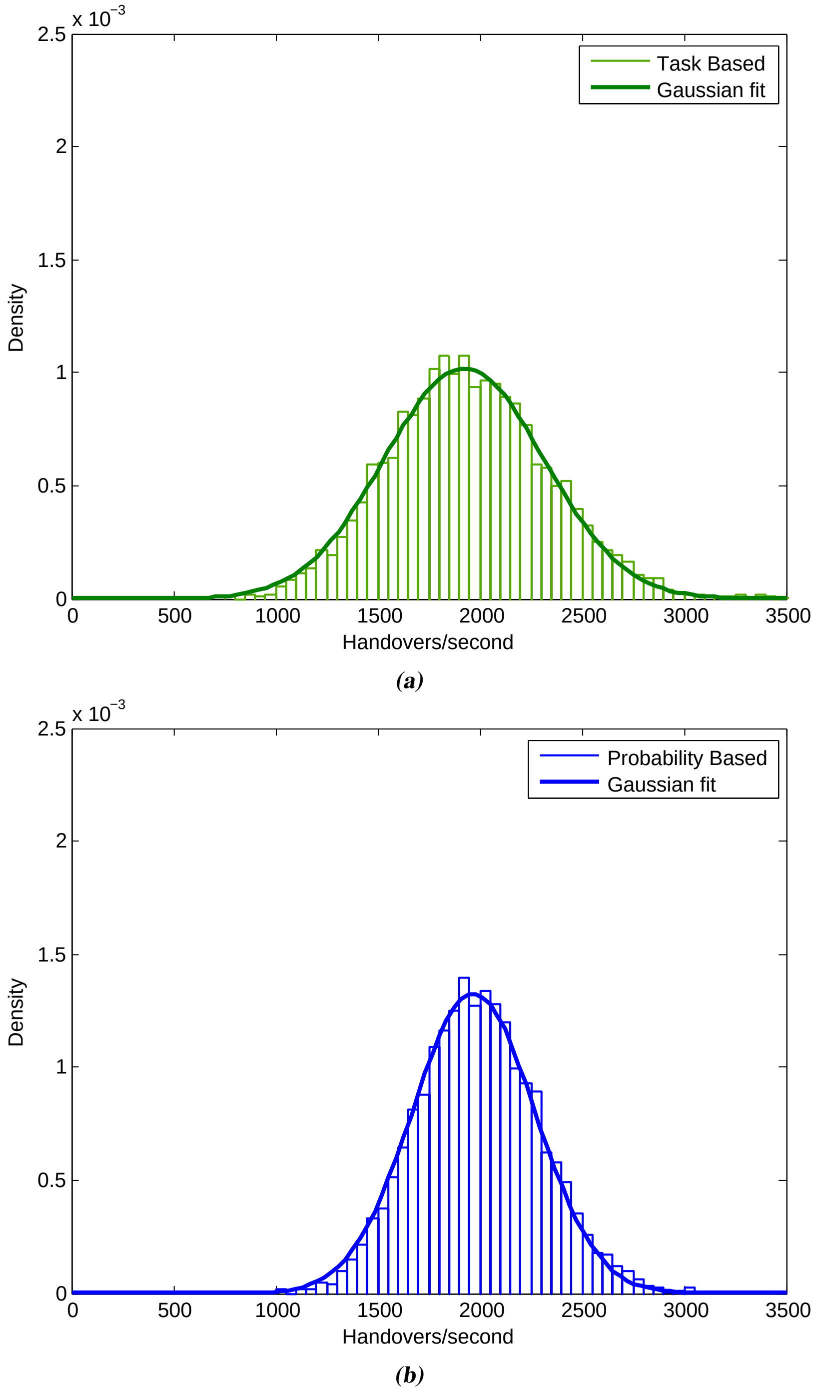


Figure 7.9: *Probability density function, with (a) task based and (b) probability based mobility control, for a 2000 HO/s target simulation with the first 10 samples removed*

7.2 Cell Utilization

Only the task based mobility control concept in chapter 5 define how to control UE mobility in order to achieve a fixed cell utilization (CU). Hence, only the task based approach was evaluated.

The simulation setup that was used is found in Table 7.13. Five different levels of CU (0 %, 25 %, 50 %, 72 %, and 100 %) was simulated in both a centrally located cell and a corner cell. The CU was controlled for only one single cell at a time.

The cell in which the CU was controlled was also marked as forbidden for UEs generating handovers, i.e. UEs that have been assigned the task to generate a handover will never enter that cell. This was done to minimize the influence the handover rate control system have on the CU control system.

The simulation results can be found in Figure 7.10 and 7.11, with their respective probability density function plotted in Figure 7.12 and 7.13.

Table 7.13: *Simulation setup*

Area	200x200
Base stations	36
ΔT	20 ms
Velocity change radius	10 m
Simulation runtime	50 s
Number of UEs in area	841
Random center movement	no
UEs idle after Handover	yes
Low speed (3 m/s) UEs	0 %
Medium speed (30 m/s) UEs	100 %
High speed (50 m/s) UEs	0 %
K_P	0.1
K_I	0.0
K_D	0.0
Smith controller used	no
Target cell marked as forbidden	yes

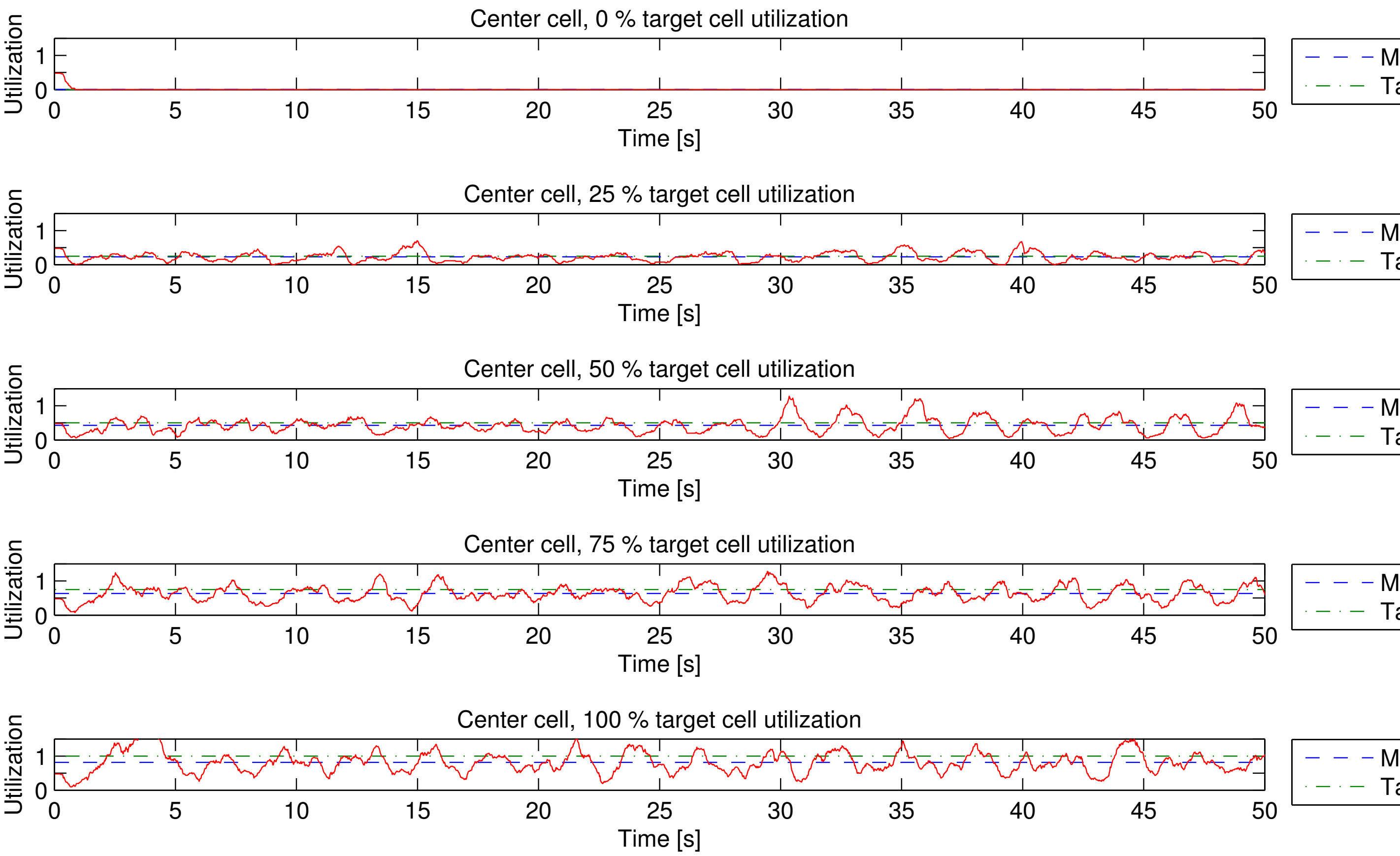


Figure 7.10: *Cell utilization (0 %, 25 %, 50 %, 75 %, and 100 %) as a function of time for a center cell*

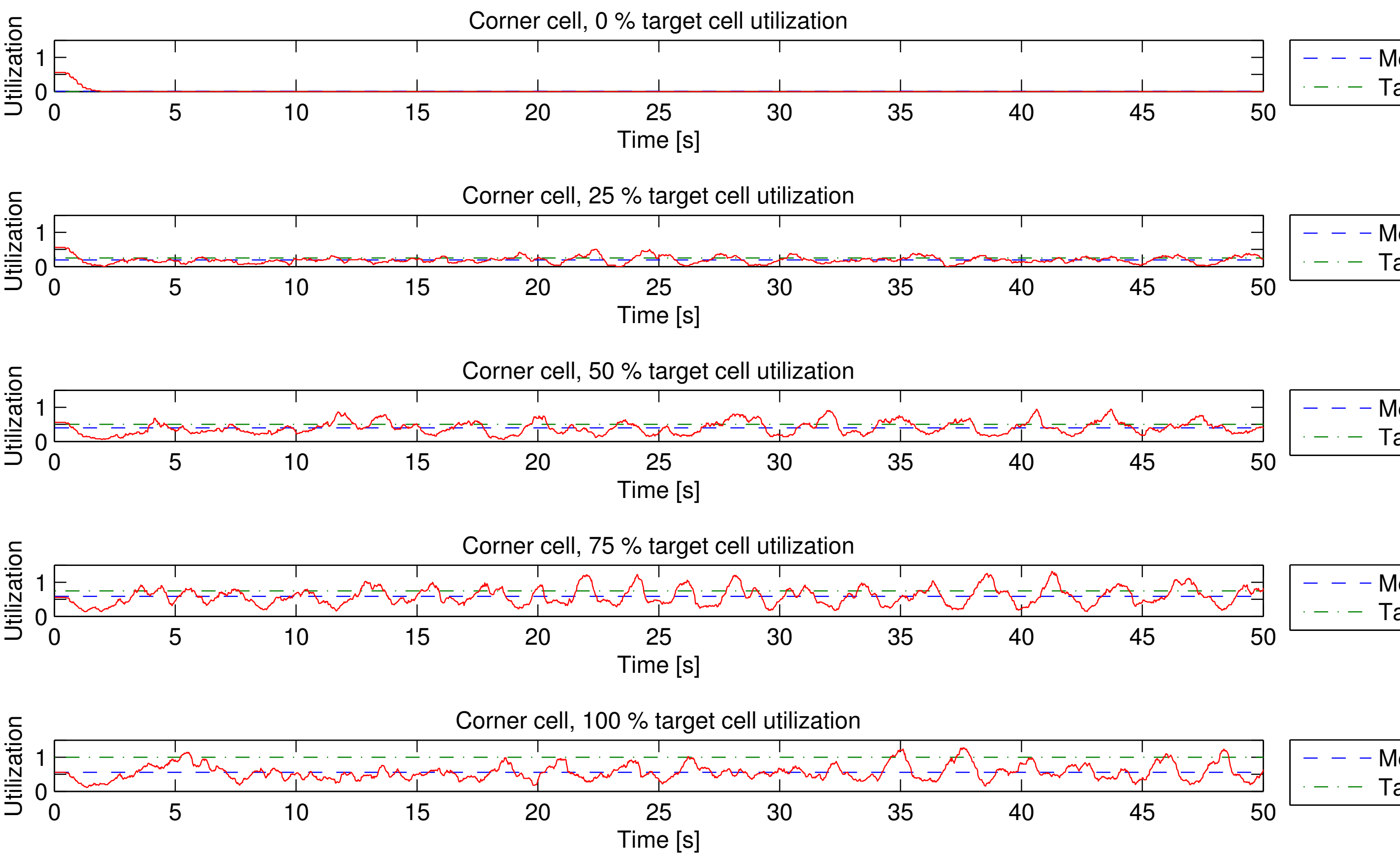
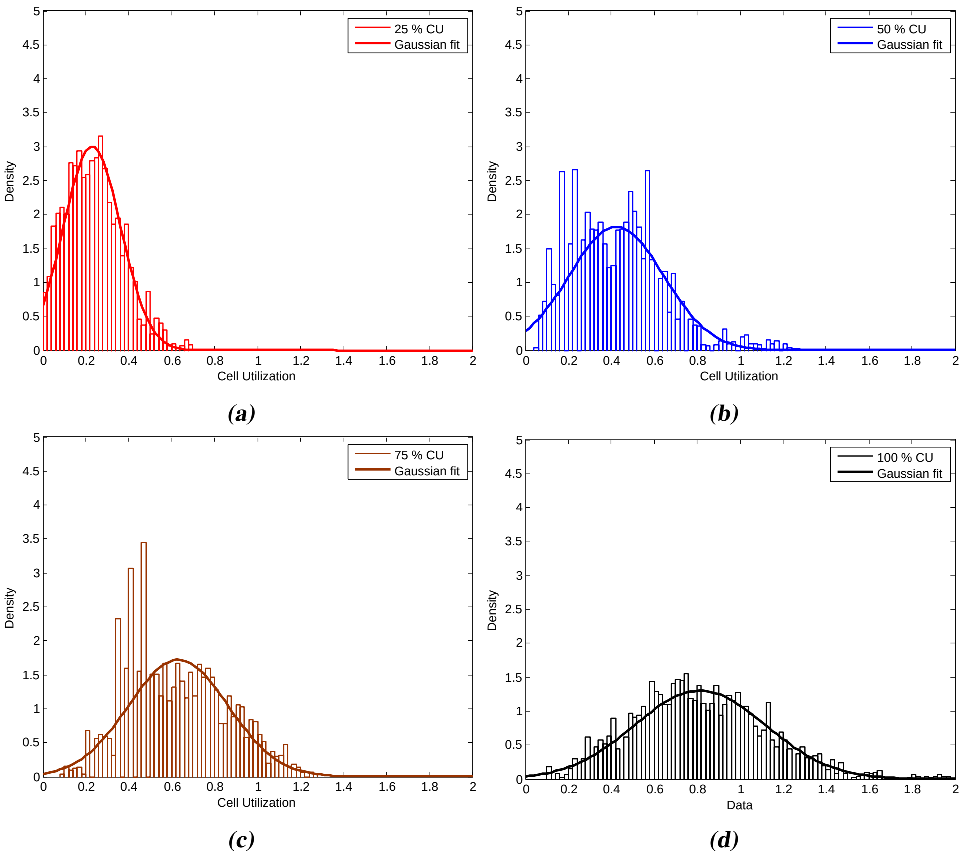


Figure 7.11: *Cell utilization (0 %, 25 %, 50 %, 75 %, and 100 %) as a function of time for a corner cell*



(a) *(b)* *(c)* *(d)*

Figure 7.12: *Probability density functions for simulations with (a) 25 %, (b) 50 %, (c) 75 %, and (d) 100 % target cell utilization in a center cell.*

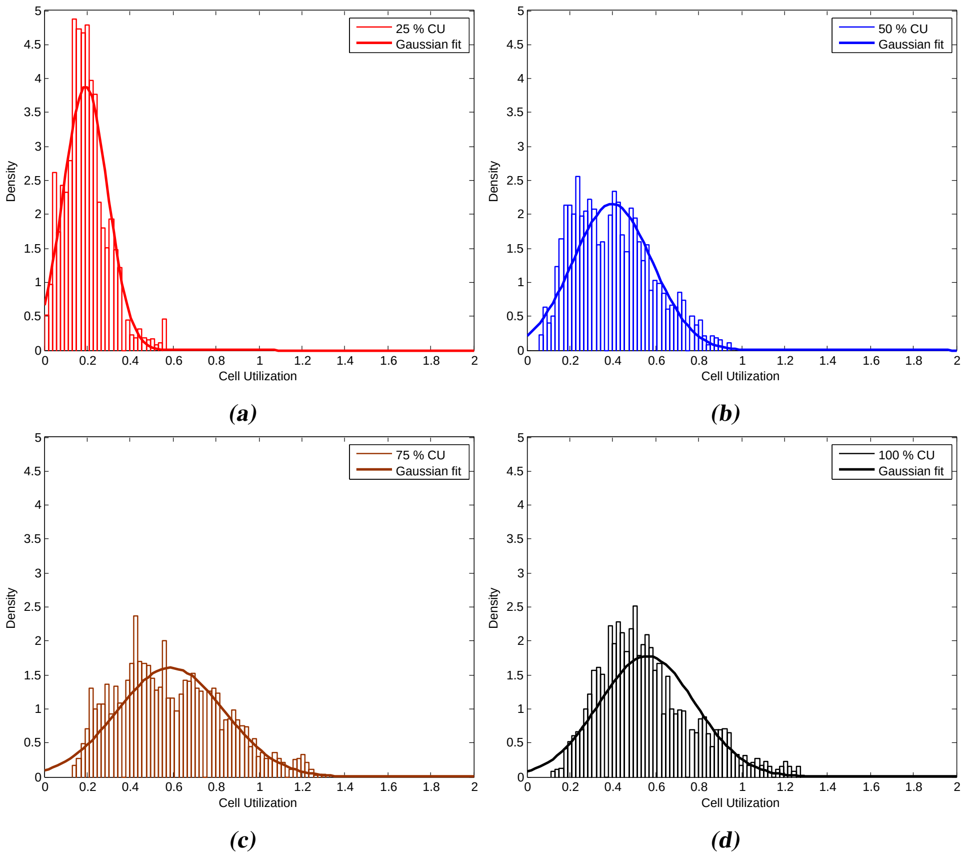
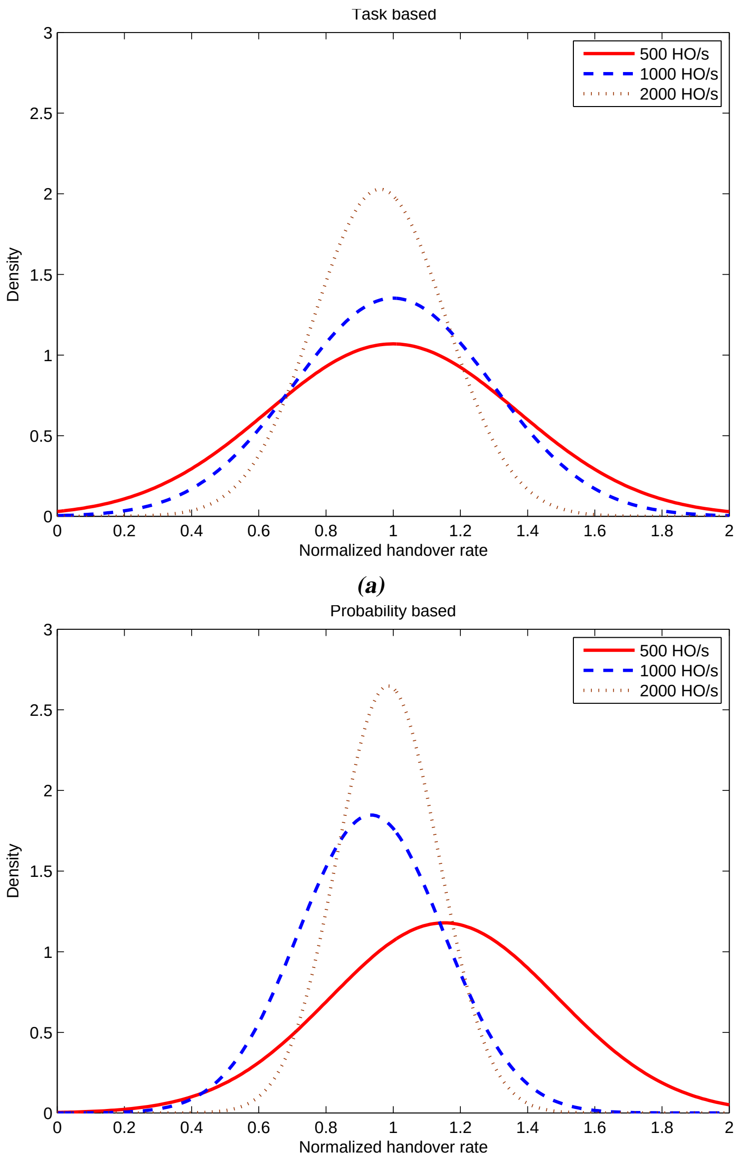


Figure 7.13: *Probability density functions for simulations with (a) 25 %, (b) 50 %, (c) 75 %, and (d) 100 % target cell utilization in a corner cell.*

7.3 Result summary

Below follows a result summary where the probability density functions of each simulation have been normalized to its target handover rate or cell utilization. Figure 7.14 show the result of handover rate control and Figure 7.15 show of cell utilization control.



(b)

Figure 7.14: *Normalized probability density functions, with (a) task based and (b) probability based mobility control, for handover rate simulations.*

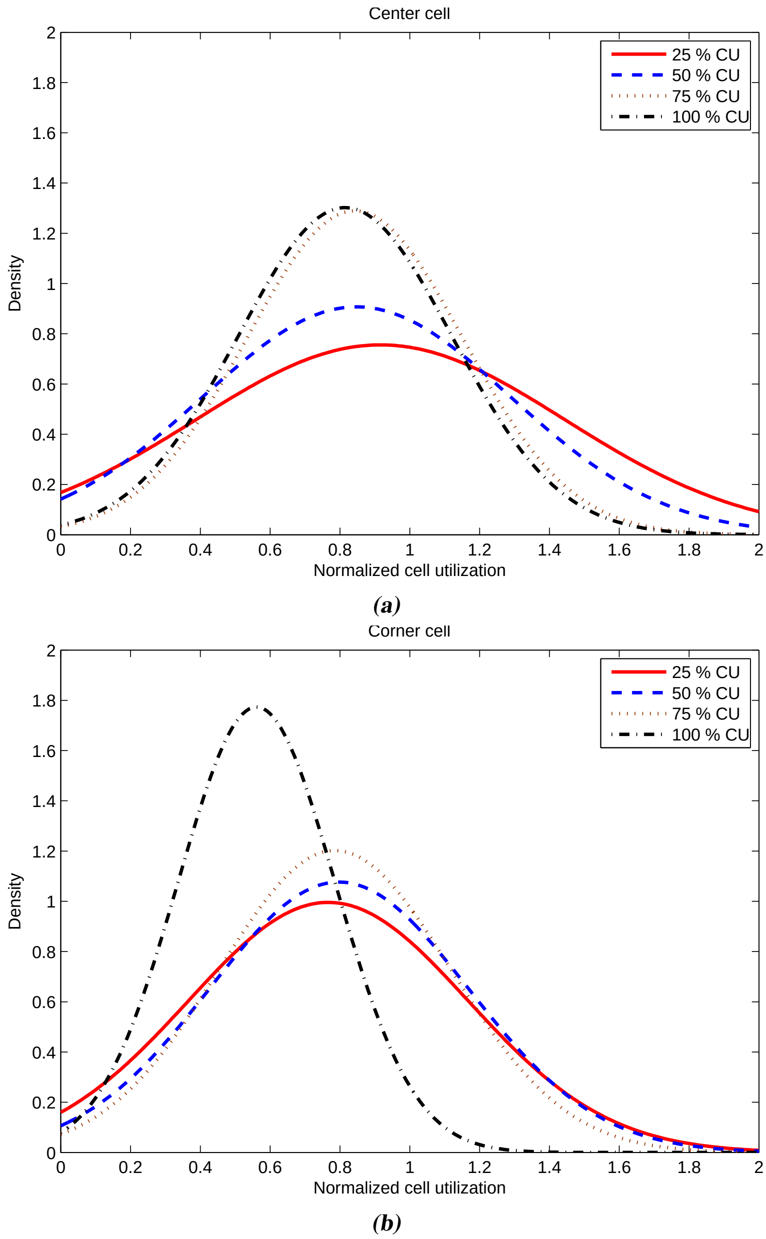


Figure 7.15: *Normalized probability density functions for cell utilization simulations in a (a) center and (b) corner cell.*

8

Discussion

The simulation results from chapter 7 are here discussed. The discussion focus on the handover rate since it is the only characteristic that both mobility control concepts cover. A discussion on cell utilization control by the task based approach can be found in section 8.3.

8.1 Accuracy and Precision

By comparing Figure 7.14 (a) and (b), it can be stated that, in general, the task based mobility control is more accurate but less precise than the probability based mobility control.

The precision of the task based mobility control could be improved by adjusting the number of UEs and UE speed distribution, or further tuning of the PID controller.

The lack of accuracy in the probability based simulation results can be explained by how the final speed of the UEs is determined. If the starting UE speed is too low, the probability based algorithm will continue to increase the speed of all UEs until the calculated handover rate is "good enough". Since "good enough" is actually lower than the target handover rate, the subsequent simulation results will also reflect this. The reverse is true when the starting speed is too high. By restraining what is considered "good enough", the accuracy could increase at the expense of a longer calculation time and the possibility that the calculated value is never "good enough".

8.2 Startup Spike

As can be seen in Figure 7.1, 7.4 and 7.7 and read from Table 7.2, 7.6, and 7.10, the probability based approach suffer from a quite significant startup spike, This stems from the fact that most UEs reaches a new cell almost at the same time during the first ΔT time segments. This could however be mitigated by more clever UE placement or if groups of UEs starts moving with slight time delay.

Another approach is to first perform a initial simulation, where the simulator is not connected to the network, and then use the final position of the UEs in the initial simulation as starting positions for the first real simulation.

The task based approach does not suffer from startup spike since not all UEs are given the task to perform handovers at once but gradually over time. This is due to the low pass filtering of the error signal and the generally low value of the PID parameters. While the error signal continues to be positive (too few handovers generated), the number of UE generating handovers increases. If, however, the PID controller parameters would be drastically increased, startup spike could potentially still be an issue.

8.3 Cell Utilization Results

From the simulation results in Figure 7.10 and 7.11, the conclusion can be made that the system is unstable and oscillating. This oscillation is mainly caused by the big time delays in the system compared to the speed of the control loop. When the error signal is positive, more UEs are assigned the task to move to the target cell. Since these UEs take a significant long time to reach the cell, the control loop continues to assign more and more UEs with the task to move to the target cell.

When the first UEs finally start to arrive to the target cell, almost every UE available have been assigned to reach that cell. The cell utilization quickly rises above its target level and the controller starts assigning UEs with the task to move to a (random) cell *outside* the target cell.

Eventually, more UEs are moving out of the cell than into it and the cell utilization drops below the target level. The controller again starts looking for UEs to assign with the task to move towards the target cell, but most UEs are already busy with either moving into or out of the cell, and the task assignment is delayed.

For controlling the cell utilization, the PID controller have been made deliberately slow in an effort to counter this behavior. However, due to time constraints, no extensive PID tuning has been done, nor has a Smith controller been tested thoroughly.

That being said, the comparison in Figure 7.15 show that a lower CU target level is less precise but slightly more accurate. The highest CU target level is considerably less accurate in the case of a corner cell. This is likely because the UEs have to travel much farther on average than for a center cell. This also has the effect of spreading out the time delays in the system and could be the cause of the higher precision of the corner cell case.

9 Conclusion

In this report, two concepts for mobility control have been presented (see chapter 5 and chapter 6). Both of these concepts are able to control the mobility of UEs in order to reach a set specification. The task based mobility control concept is even able to control multiple mobility characteristics at the same time. Moreover, the mobility concepts can achieve high accuracy and precision when controlling the handover rate (see Figure 7.14). Therefore, the first two objectives of this thesis can be considered to be met (see section 1.1).

To verify the mobility control concepts, a prototype simulator was developed which simulates a fictional radio network (see section 4.1). With the simulator, it was possible to verify that the mobility control concepts were effective in what they were designed to control and thereby achieving the third objective.

The task based mobility control concept is able to control handover rate and UE population density, the probability based mobility control concept is able to control handover rate, and the prototype simulator is able to mark cells as forbidden independently of mobility control concept. Thus, the fourth and final objective can be considered met.

10

Future Work

This section discuss additional thoughts and other ideas which relate to mobility control and the prototype simulator in chapter 4, but have not been implemented due to limited time constraints.

10.1 User Equipment Groups

The mobility control concepts introduced in this thesis handle each and every UE individually. This could cause a traffic simulator to become slow due to the extra resources spent on managing individual UEs. Instead, UEs could be grouped together so that the mobility of a group of UEs is controlled.

10.2 Incoming Calls

The prototype simulator in chapter 4 currently use a traffic model where every UE is always in an active call. No new calls are made, nor is any call dropped. This make it impossible to test such behaviors in a radio network.

Furthermore, since no new calls are generated, the rate of incoming calls is zero. The incoming call rate is used to calculate the call drop and block probability, which could be useful in extending the probability based mobility control concept in chapter 6.

10.2.1 Call Priority

If new calls are introduced by the prototype simulator, they could be prioritized by the RNS as explained in section 3.9. Adding call priority to the RNS would make it simulate a real hardware node more accurately.

10.2.2 Call Queuing

When new calls has been introduced into the mobility control, a queuing system should be introduced. Two Queue list can be introduced in the mobility control, one for the handover calls and the other for the new or originating calls.

10.3 Resources Types

The RNS used to evaluate the prototype simulator currently use a single type of resource that acts as an umbrella resource to cover every possible type of resource. To more accurately evaluate the mobility concepts in chapter 5 and 6 multiple resources could be implemented in both the prototype simulator and the RNS.

10.4 Weight-Based Direction

In this report, every time the direction of a UE is calculated, the received signal strength from surrounding base stations are used exclusively. The only exception to this is when using the task based mobility control and an UE has been assigned the task to move to a certain cell.

One future improvement to this would be to introduce weights for every cell based on signal strength, cell utilization, available resources, rate of incoming calls, etc. The weight of every cell close to a UE is compared and the cell with the highest weight is used to determine the UE's next direction.

10.5 Probability Based Mobility Control

The probability approach in our mobility control uses only the handover probability calculation and the mobility control can be upgraded to use the calculations of handover failure, call block and call drop probabilities.

10.5.1 Handover Failure

In our mobility control, the handover failure value which is calculated is very small. This particular phenomenon is caused because the denominator in Equation 3.57 contains the factorial of available channels. This denominator quickly becomes very large and the handover failure probability goes to zero.

10.5.2 Call Drop and Call Block Probabilities

Once the originating calls has been introduced inside the controller, the probability approach can incorporate the use of Call Drop and Call Block probabilities in calculating the movement path. The call drop and call block probabilities can be used in combination with handover failure and handover probability values in determining the UE movement path.

10.5.3 Continuous Probability Simulation

When using the probability based mobility control concept, the UE movement path is calculated and executed for a user-specified time T. When implementing this concept in a traffic simulator, the simulation can be run continuously by calculating the movement path for the next T seconds while the current movement path is executed.

10.6 Other Handover Algorithms

Both mobility control concepts presented in this report can be further evaluated using other handover algorithms, e.g. other hard or soft handover, as well as vertical or horizontal handover, algorithms.

10.7 Interfacing Real Hardware Nodes

Since the mobility control concepts presented herein are targeted at improving the performance of traffic simulators, it is reasonable to assume that further testing and development would gain immensely from interfacing with real hardware nodes, as opposed to the very simplified RNS in section 4.1.

10.8 Task Based Mobility Control

The task based approach to mobility control succeeds in most of what it is meant to do, but some things can be improved.

10.8.1 State Space

The task based mobility control concept could be extended to use state space control (see subsection 3.10.2). The advantage of this is that state space control theory allow multiple separate systems to influence the control of each other in a way that is not possible in classic control theory.

For example, let

$$\vec{x}[n] = \begin{cases} x_1[n] = \textit{UEs that are generating handovers} \\ x_2[n] = \textit{UEs that are moving to a certain cell} \end{cases}, \tag{10.1}$$

$$\vec{u}[n] = \begin{cases} u_1[n] = \textit{UEs that should generate handovers} \\ u_2[n] = \textit{UEs that should move to a certain cell} \end{cases}, \tag{10.2}$$

and

$$\vec{y}[n] = \begin{cases} y_1[n] = \textit{Generated handovers} \\ y_2[n] = \textit{UEs in a certain cell} \end{cases} \tag{10.3}$$

Since UEs moving to a certain cell will generate handovers while moving to that cell, u_2 will influence x_1 to a certain degree. The prototype simulator can thus be modeled, approximately, as the following state space system

$$\vec{x}[n+1] = \begin{bmatrix} x_1[n] - kx_2[n-b] \\ x_2[n] - x_2[n-b] \end{bmatrix} + \begin{bmatrix} 1 & k \\ 0 & 1 \end{bmatrix} \vec{u}[n] \quad (10.4)$$

$$\vec{y}[n] = \begin{bmatrix} x_1[n-a] \\ \sum_{m=n-b}^{-\infty} x_2[m] \end{bmatrix},$$

where a and b represent the time delays in the system, and k reflects to what degree u_2 influence x_1. Additionally, $x_2[n-b]$ is subtracted from the state of the system to reflect that UEs have reached the specified cell and is no longer moving towards it. Remember that this state space system represent a model of the prototype simulator without any kind of feedback control.

10.8.2 SIMO/MISO/MIMO System

The control systems for controlling the handover rate and cell density outlined in chapter 5 are two separate single input single output (SISO) systems. However, multiple factors can influence the output of a system, i.e. multiple inputs influence a single output (MISO). Likewise, a single input can influence the output of multiple systems (SIMO). The state space system in Equation 10.4 represent an example MIMO system that take into account multiple inputs to produce multiple outputs.

Using MIMO system control might increase the accuracy of the mobility control.

A

Image convolution

In image processing, an image can be transformed by means of convolution with a kernel (or convolution matrix). Depending on what kernel is used, an image can be blurred, sharpened, embossed, etc. A kernel can also be used for edge detection.[33]

Image convolution is done by convolving every pixel and its neighbourhood with the kernel. The pixel neighbourhood is simply the pixel and its surrounding pixels. If a pixel lies on the edge of the image, its neighbourhood will extend outside the image. There are a number of ways to deal with this

- The nearest border pixels are extended as far as necessary to provide values for the convolution (Figure A.1b).
- The image is wrapped and pixels are taken from the opposite border or corner of the image (Figure A.1c).
- Border pixels are removed (*cropped*, Figure A.1d).

(a) ***(b)*** ***(c)*** ***(d)***

Figure A.1: *Example images, where (a) is the original, (b) have its borders extended, (c) is wrapped, and (d) is cropped.[34]*

The convolution per pixel is performed by multiplying each pixel value in the pixel neighbourhood with its corresponding kernel value, and then sum up all multiplications. That is, for a a-*by*-a kernel K, each and every pixel of the resulting image is determined by

$$p_c(x,y) = \sum_{n=y-b}^{y+b} \sum_{m=x-b}^{x+b} K(m,n)\, p(m,n), \quad b = \frac{a-1}{2}, \tag{A.1}$$

where $p(x,y)$ and $p_c(x,y)$ represent the pixel values at (x,y) of the original and the convolved image, respectively. Figure A.2 show how the first pixel in an example image is convolved with a 3-*by*-3 sharpening kernel.

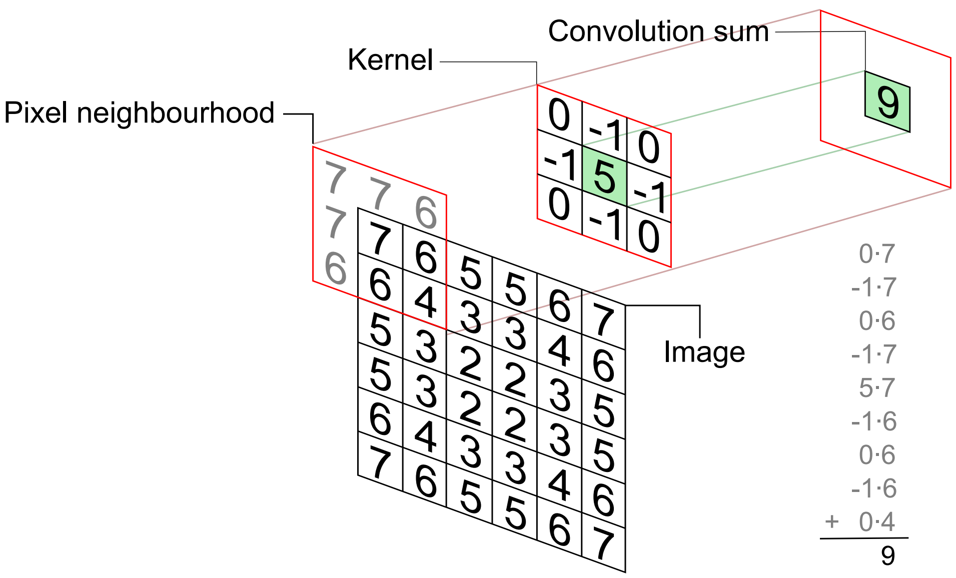


Figure A.2: *Example image convolution with a sharpening kernel*[1]

[1] Image based on original by Michael Plotke: `http://en.wikipedia.org/wiki/File:3D_Convolution_Animation.gif`

Bibliography

[1] Fan Bai and Ahmed Helmy. *Wireless Ad-Hoc Networks*. Kluwer Academic Publishers, USA., 2006.

[2] J. Broch, D. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. In *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom98)*, October 1998.

[3] T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. In *Wireless Communication and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends, and Applications*, 2002.

[4] E.M. Royer, P.M. Melliar-Smith, and L.E. Moser. An analysis of the optimum node density for ad hoc mobile networks. In *IEEE International Conference on Communications*, 2001.

[5] B. Liang and Z.J. Haas. Predictive distance-based mobility management for pcs networks. In *Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies.*, 1999.

[6] Christian Bettstetter. Smooth is better than sharp: A random mobility model for simulation of wireless networks. In *Workshop on Modeling, Analysis, and Simulation of Wireless and Mobile Systems (MSWiM).Proc. ACM Inter.*, 2001.

[7] Xiaoyan Hong, Mario Gerla, Guangyu Pei, and Ching-Chuan Chiang. A group mobility model for ad hoc wireless networks. In *Proceedings of the 2nd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, 1999.

[8] Miguel Sánchez López and Pietro Manzoni. Anejos: a java based simulator for ad hoc networks. *Future Generation Comp. Syst.*, 2001.

[9] J. Tian, J. Hähner, C. Becker, I. Stepanov, and K. Rothermel. Graph-based mobility model for mobile ad hoc network simulation. In *Proceedings of the 35th Annual Simulation Symposium*, 2002.

[10] Per Johansson, Tony Larsson, Nicklas Hedman, Bartosz Mielczarek, and Mikael Degermark. Scenario-based performance analysis of routing protocols for mobile ad-hoc networks. In *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, 1999.

[11] T. van der Vorm and H. Wilberg. A performance evaluation tool for cellular mobile networks. In *Proceedings of Fifth Nordic Seminar on Digital Mobile Radio Communications DMR V*, 1992.

[12] G. Drozdy, J. Niemella, J. Valimaki, J. Rosenberg, and V. Hara. Study of gsm system performance by a gsm network computer simulator. In *IEEE International Conference on Selected Topics in Wireless Communications,Conference Proceedings.*, 1992.

[13] G. Edbom and P. Stjernholm. Simulation of cellular networks. In *IEEE 44th Vehicular Technology Conference*, 1994.

[14] I. Martin-Escalona, F. Barcelo, and J. Casademont. Teletraffic simulation of cellular networks: modeling the handoff arrivals and the handoff delay. In *The 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, volume 5, pages 2209–2213 vol.5, 2002.

[15] J.A. Khoja, M.A. Al-Shalash, and V.K. Prabhu. Dynamic system simulator for the modeling of cdma systems. In *International Mobility and Wireless Access Workshop*, 2002.

[16] E.L. Andrade N, A.A. Shinoda, M.E. Pellenz, and M.D. Yacoub. A comprehensive 3g link level simulator. In *35th annual Simulation Symposium*, 2002.

[17] K. Meier, D. Wehrle, K. Rechert, and D. von Suchodoletz. Testbed for mobile telephony networks. In *Sixth International Conference on Availability, Reliability and Security (ARES)*, 2011.

[18] N. Vlajic and D. Stevanovic. Sink mobility in wireless sensor networks: When theory meets reality. In *IEEE Sarnoff Symposium*, 2009.

[19] Qijun Gu, Xiao Chen, Zhen Jiang, and Jie Wu. Sink-anonymity mobility control in wireless sensor networks. In *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, 2009.

[20] Ajay R. Mishra. *Advanced Cellular Network Planning and Optimisation*, pages 40–44. WILEY, February 2007.

[21] A. Sgora and D.D. Vergados. Handoff prioritization and decision schemes in wireless cellular networks: a survey. *Communications Surveys Tutorials, IEEE*, 2009.

[22] L'Institut d'électronique et d'informatique Gaspard-Monge (IGM). Umts and mobile broadband. URL `http://www-igm.univ-mlv.fr/~dr/XPOSE2006/eric_meurisse/umts.php`. Accessed: 2013-09-8.

[23] G.P. Pollini. Trends in handover design. *Communications Magazine, IEEE*, 1996.

[24] Nasıf Ekiz Tara Salih Sibel Küçüköner and Kemal Fidanboylu. An overview of handoff techniques in cellular networks. *WorldAcademyofScience*, 2007.

[25] V. Kapor, G. Edwards, and R. Sankar. Handoff criteria for personal communication networks. In *IEEE International Conference on Communications*, 1994.

[26] K. Sanon and S. Joshi. Preemptive mobile assisted and guard channel based handoff queuing scheme. In *Annual IEEE India Conference (INDICON)*, 2011.

[27] P. Marichamy, S. Chakrabarti, and S.L. Maskara. Overview of handoff schemes in cellular mobile networks and their comparative performance evaluation. In *Vehicular Technology Conference.*, 1999.

[28] Cheng-Chung Lin, K. Sandrasegaran, H. A M Ramli, R. Basukala, R. Patachaianand, Lu Chen, and T.S. Afrin. Optimization of handover algorithms in 3gpp long term evolution system. In *Modeling, Simulation and Applied Optimization (ICMSAO)*, 2011.

[29] O.J.M. Smith. Closer control of loops with dead time. *Chemical Engineering Progress*, 1957.

[30] D.J. Powell G.F. Franklin and M.L. Workman. *Digital Control of Dynamic Systems(3rd Edition)*. Prentice Hall, 1997.

[31] B. Jabbari. Teletraffic aspects of evolving and next-generation wireless communication networks. *IEEE Personal Communications Magazine*, 1996.

[32] Martin Fowler. Mocks aren't stubs. URL `martinfowler.com/articles/mocksArentStubs.html`. Accessed: 2013-09-08.

[33] Mubarak Shah. *Fundamentals of Computer Vision*. Computer Science Department, University of Central Florida, Orlando, FL32816, USA, 1997. URL `http://www.cs.ucf.edu/courses/cap6411/book.pdf`. Accessed: 2013-07-14.

[34] The GIMP Documentation Team. Gnu image manipulation program user manual. URL `http://docs.gimp.org/en/`. Accessed: 2013-07-14.

Upphovsrätt

Copyright