Steering Mobile Users in LTE Networks Based on Their Mobility Behaviour

Het sturen van gebruikers in LTE netwerken gebaseerd op hun mobiliteitsgedrag

Proefschrift voorgelegd tot het behalen van de graad van doctor in de Wetenschappen: Informatica aan de Universiteit Antwerpen te verdedigen door Bart SAS

Promotor: Chris Blondia
Antwerpen, 2017
# Contents

**Acronyms** ix

**Publications** xv

## 1 Introduction and Background 1

### 1 Introduction 3

1.1 Context .................................................. 3

1.2 Problem Description .................................. 4

1.3 Contributions ........................................ 4

1.4 Structure of this Thesis .............................. 6

## 2 Long Term Evolution 7

### 2.1 Introduction ...................................... 7

### 2.2 Standardisation ................................... 7

### 2.3 Network Elements ................................. 8

2.3.1 E-UTRAN ..................................... 8

2.3.2 EPC ........................................ 10

### 2.4 Interfaces ......................................... 10
CONTENTS

6.3.2 Antenna Models ............................................. 59
6.4 Users .......................................................... 60
6.5 Simulator ....................................................... 61

7 Trajectory Classification ......................................... 63
  7.1 Introduction .................................................. 63
  7.2 Measurements ................................................ 63
    7.2.1 Measurement Configuration ............................. 64
    7.2.2 Example .................................................. 66
    7.2.3 Inter-RAT Measurements ................................. 70
  7.3 Trajectory Matching ......................................... 70
    7.3.1 Classical Dynamic Time Warping ....................... 71
    7.3.2 Modified Dynamic Time Warping ....................... 77
    7.3.3 Measurement Distance .................................. 80
  7.4 Evaluation .................................................... 82
    7.4.1 Methodology ............................................ 82
    7.4.2 Accuracy Metric ....................................... 84
    7.4.3 Reliability Criteria .................................... 86
    7.4.4 Optimal Cost Parameters and Length Threshold .... 87
    7.4.5 Influence of the Environment ......................... 92
    7.4.6 Signalling Overhead ................................... 93
    7.4.7 Other Environments .................................... 93
  7.5 Conclusion .................................................... 97

8 Trajectory Identification ....................................... 99
CONTENTS

8.1 Introduction ......................................................... 99
8.2 Reference Trace Selection ........................................... 99
8.3 Relevance Score ....................................................... 100
  8.3.1 Adding a New Reference Trace ................................. 102
  8.3.2 Updating the Relevance Score ................................. 102
  8.3.3 Removing Reference Traces .................................... 103
8.4 Example ............................................................. 104
8.5 Evaluation .......................................................... 106
  8.5.1 Reference Set Coverage ....................................... 107
  8.5.2 Dealing With Changes in the Environment .................... 115
8.6 Conclusion .......................................................... 120

9 Traffic Steering ......................................................... 121
  9.1 Introduction ......................................................... 121
  9.2 Event Extrapolation and Projection ............................... 121
  9.3 Radio Condition Estimation ....................................... 122
  9.4 Traffic Steering Decision ......................................... 124
  9.5 Example ............................................................ 126
  9.6 Evaluation .......................................................... 128
    9.6.1 Simulation Setup ............................................. 131
    9.6.2 Results ......................................................... 133
  9.7 Conclusion .......................................................... 136

10 Extending to Multiple RATs ............................................ 139
  10.1 Introduction ......................................................... 139
Acronyms

1G  1st Generation  
2G  2nd Generation  
3G  3rd Generation  
3GPP  3rd Generation Partnership Project  
4G  4th Generation  
5G  5th Generation  
ACK  Acknowledgement  
AC  Admission Control  
AMC  Adaptive Modulation and Coding  
AMPS  Advanced Mobile Phone System  
ANR  Automatic Neighbour Relation  
ARIB  Association of Radio Industries and Businesses  
ARQ  Automatic Repeat Request  
ATIS  Alliance for Telecommunications Industry Solutions  
BCCH  Broadcast Control Channel  
BCH  Broadcast Channel  
BLER  Block Error Rate  
CAPEX  Capital Expenditure  
CCCH  Common Control Channel  
CCSA  China Communications Standards Association  
CFI  Control Frame Indicator  
CIO  Cell Individual Offset
ACRONYMS

CQI Channel Quality Indicator
CR Chip Rate
CRC Cyclic Redundancy Check
C-RS Cell-Specific Reference Signal
CSI Channel State Information
CSI-RS CSI Reference Signal
CSI-RSRP CSI Reference Signal Received Power
CT Core Network & Terminals
D-AMPS Digital Advanced Mobile Phone System (AMPS)
DCCH Dedicated Control Channel
DL-SCH Downlink Shared Channel
DTCH Dedicated Traffic Channel
DTW Dynamic Time Warping
$E_c/I_0$ Energy per Chip over Total Interference Power Density
$E_c/N_0$ Energy per Chip over Total Noise Power Density
EDGE Enhanced Data rates for GSM Evolution
eNB eNodeB
EPC Evolved Packet Core
ETSI European Telecommunications Standards Institute
E-UTRA Evolved Universal Terrestrial Radio Access
E-UTRAN Evolved Universal Terrestrial Radio Access Network
FDD Frequency Division Duplex
FEC Forward Error Correction
FP7 Seventh Framework Programme
GBR Guaranteed Bit rate
GERAN GSM/EDGE Radio Access Network
GP Guard Period
GPS Global Positioning System
GSM Global System for Mobile Communications
GTP  GPRS Tunnelling Protocol
HARQ  Hybrid ARQ
HOF  Handover Failure
HSS  Home Subscription System
ICIC  Inter Cell Interference Coordination
ID  Identifier
IMS  IP Multimedia Subsystem
IP  Internet Protocol
KPI  Key Performance Indicator
LTE  Long Term Evolution
MAC  Medium Access Control
MBMS  Multimedia Broadcast and Multicast Services
MCCH  Multicast Control Channel
MCH  Multicast Channel
MCS  Modulation and Coding Scheme
MDTW  Modified Dynamic Time Warping
MIMO  Multiple-Input and Multiple-Output
MLB  Mobility Load Balancing
MME  Mobility Management Entity
MRO  Mobility Robustness Optimisation
MTCH  Multicast Traffic Channel
NAK  Negative Acknowledgement
NAS  Non Access Stratum
NeNB  Neighbouring eNodeB
NGMN  Next Generation Mobile Networks Alliance
NMT  Nordic Mobile Telephone
OFDM  Orthogonal Frequency-Division Multiplexing
OPEX  Operational Expenditure
PCell  Primary Cell
PCRF  Policy and Charging Resource Function
PDN  Packet Data Network
P-GW  Packet Data Network Gateway
PSCell  Primary Secondary Cell
QAM  Quadrature Amplitude Modulation
QCI  QoS Class Identifier
QoS  Quality of Service
QPSK  Quadrature Phase-Shift Keying
PBCCH  Physical Broadcast Channel
PCCH  Paging Control Channel
PCFICH  Physical Control Format Indicator Channel
PCH  Paging Channel
PDCCH  Physical Downlink Control Channel
PDSCH  Physical Downlink Shared Channel
PHICH  Physical Hybrid ARQ Indicator Channel
PMCH  Physical Multicast Channel
PPHO  Ping Pong Handover
PRACH  Physical Random Access Channel
PRB  Physical Resource Block
PSS  Primary Synchronisation Signal
PUCCH  Physical Uplink Control Channel
PUSCH  Physical Uplink Shared Channel
RAB  Radio Access Bearer
RACH  Random Access Channel
RAN  Radio Access Network
RAT  Radio Access Technology
RE  Resource Element
RLF  Radio Link Failure
RRC  Radio Resource Control
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS</td>
<td>Reference Signal</td>
</tr>
<tr>
<td>RSCP</td>
<td>CPICH (Common Pilot Channel) Received Signal Code Power</td>
</tr>
<tr>
<td>RSRP</td>
<td>Reference Signal Received Power</td>
</tr>
<tr>
<td>RSRQ</td>
<td>Reference Signal Received Quality</td>
</tr>
<tr>
<td>RSSI</td>
<td>Received Signal Strength Indicator</td>
</tr>
<tr>
<td>S1-AP</td>
<td>S1 Application Protocol</td>
</tr>
<tr>
<td>SA</td>
<td>Service &amp; Systems Aspects</td>
</tr>
<tr>
<td>SAW</td>
<td>Stop-and-Wait</td>
</tr>
<tr>
<td>SCell</td>
<td>Secondary Cell</td>
</tr>
<tr>
<td>SCTP</td>
<td>Stream Control Transmission Protocol</td>
</tr>
<tr>
<td>SeNB</td>
<td>Serving eNodeB</td>
</tr>
<tr>
<td>SF</td>
<td>Spreading Factor</td>
</tr>
<tr>
<td>S-GW</td>
<td>Serving Gateway</td>
</tr>
<tr>
<td>SINR</td>
<td>Signal-to-Interference-and-Noise Ratio</td>
</tr>
<tr>
<td>SSS</td>
<td>Secondary Synchronisation Signal</td>
</tr>
<tr>
<td>SUMO</td>
<td>Simulation of Urban MObility</td>
</tr>
<tr>
<td>SON</td>
<td>Self-Organising Network</td>
</tr>
<tr>
<td>TDD</td>
<td>Time Division Duplex</td>
</tr>
<tr>
<td>TR</td>
<td>Technical Report</td>
</tr>
<tr>
<td>TSIDI</td>
<td>Telecommunications Standards Development Society</td>
</tr>
<tr>
<td>TSG</td>
<td>Technical Specification Group</td>
</tr>
<tr>
<td>TS</td>
<td>Technical Specification</td>
</tr>
<tr>
<td>TTA</td>
<td>Telecommunication Technology Association</td>
</tr>
<tr>
<td>TTC</td>
<td>Telecommunication Technology Committee</td>
</tr>
<tr>
<td>TTI</td>
<td>Transmission Time Interval</td>
</tr>
<tr>
<td>TTT</td>
<td>Time-to-Trigger</td>
</tr>
<tr>
<td>TUBS</td>
<td>Technische Universität Braunschweig</td>
</tr>
<tr>
<td>TeNB</td>
<td>Target eNodeB</td>
</tr>
<tr>
<td>UDP</td>
<td>User Datagram Protocol</td>
</tr>
</tbody>
</table>
UE  User Equipment
UL-SCH  Uplink Shared Channel
UMTS  Universal Mobile Telecommunications System
UTRA  Universal Terrestrial Radio Access
UTRAN  Universal Terrestrial Radio Access Network
VoIP  Voice over IP
WG  Working Group
WiMAX  Worldwide Interoperability for Microwave Access
X2-AP  X2 Application Protocol
Publications


Part I

Introduction and Background
Chapter 1

Introduction

1.1 Context

Cellular networks have been used since the 1970s to provide mobile communication to users over a wide geographic area. With cellular networks the covered area is divided in so-called cells. These are small areas with a radius ranging from a couple of tens of meters to several kilometres that are served by a single wireless base station. By reusing the same frequency in multiple cells the capacity of a cellular network can be increased for the same amount of bandwidth relative to a single large transmitter. Distributing the base stations across the covered area also reduces the power consumption, especially of mobile equipment, as they are closer to the transmitters. In order to maintain connectivity users are handed over from one cell to another when they cross the border of a cell.

The first commercial cellular networks like Advanced Mobile Phone System (AMPS) and Nordic Mobile Telephone (NMT) were analogue systems that provided only voice services and could not handle many users. These so called 1st Generation (1G) systems did not provide encryption and did not allow roaming. The 2nd Generation (2G) cellular systems like Global System for Mobile Communications (GSM) and Digital AMPS (D-AMPS) that replaced these 1G systems resolved many of the problems with these systems. Communication was no longer analogue but digital and encryption was provided. Furthermore 2G systems allowed roaming and improved quality, capacity and coverage. 2G systems however mainly focused on voice communication. 3rd Generation (3G) cellular systems like Universal Mobile Telecommunications System (UMTS) and CDMA2000 provided data services next to voice communication. This not only allowed users to make voice calls but also to browse the web and watch online video. The demand for more capacity led to the development of 4th Generation (4G) cellular systems like Long Term Evolution (LTE) and Worldwide Interoperability for Microwave Access (WiMAX). Furthermore the shift towards data communications continued with voice communications becoming
merely a data service through Voice over IP (VoIP).

As mobile services become more popular and the demand for bandwidth increases, much effort is spent on increasing the available capacity. One way to achieve this is by increasing the efficiency of the radio technologies (modulation schemes, Multiple-Input and Multiple-Output (MIMO), ... ) and increasing the allocated bandwidth. Another way to do this is the increase the number of cells and consequently reducing their size such that the same area is covered by more cells.

1.2 Problem Description

When the number of cells that cover a certain area increases and the size of the cells decreases more handovers will occur when users move through the area. Also when user velocities are high, for instance when users are travelling by car on a highway or on a high speed train handovers will occur more frequently. In situations like these, the total handover interruption time can become as high as 2.5% of the total amount of time [21]. A high handover interruption time, and the radio conditions during handover, have a negative impact of the performance of certain applications, especially real-time applications [17].

The high frequency of handovers will furthermore cause an increased signalling overhead in the core network. The handover procedure is resource-consuming and therefore costly to the network operator [6]. First of all performing a handover requires signalling traffic for requesting handovers, transferring status and switching data paths. These messages are propagated through a large part of the core network and consequently consume resources on various links. Secondly, and more importantly, data traffic should be forwarded from the old Serving eNodeB (SeNB) to the Target eNodeB (TeNB) during a handover. This causes additional resources to be used during handover.

The reduction of unnecessary handovers is therefore considered to be a topic for investigation [8], especially in case of HetNet deployments.

1.3 Contributions

In order to avoid the degradation of user experience and network performance, this thesis develops and assesses a Self-Organising Network (SON) function that aims at mitigating the problems that occur in the presence of frequent handovers. The goal of this SON function is to classify users according to their mobility behaviour and, based on this classification, steer them in an appropriate way such that the number of short stays are reduced while the Quality of Service (QoS) is maintained or possibly improved. In a dense deployment of small cells this can be done by steering users that perform frequent handovers to a macro cell where they can
1.3. CONTRIBUTIONS

stay connected to for a longer time while steering users that have low mobility to smaller micro and pico cells. When there are no larger cells to steer users to, but the frequent handovers are caused by the high velocity of users, the frequency of handovers can be limited by for instance not handing them over to cells through which they only pass for a small amount of time, for instance near the edge (see Figure 1.1). Note that steering a user does not mean that the route along which a user travels is influenced. Steering users means determining when and to which cell a user is handed over.

Figure 1.1: The user only passes through the edge of cell C while it is also covered by cells A and B. A handover can be avoided by not handing over the user from cell A to C but instead handing it over directly from A to B.

As mentioned before the developed SON function will classify users according to their mobility behaviour and predict their future mobility based on this classification by assuming that all users in the same mobility class will behave the same in terms of cells they will visit in the future, velocity at which they will travel and so on. The rationale behind this is that users do not just move in a random fashion but follow certain trajectories along roads. For instance, users that travel on a highway will tend to stay on that highway and will thus likely handover to the same cell as other users that travel on the highway while users in a residential area along the highway will not. In a city centre users that come from a certain street will at a crossroad less likely take a street that will lead them back to where they came from. Users can be steered within the same layer (macro, micro, pico cells) and possibly between Radio Access Technologies (RATs) LTE UMTS etc.) but also between different layers and/or RAT depending on their mobility behaviour and the availability of layers and RAT to steer them to.
CHAPTER 1. INTRODUCTION

1.4 Structure of this Thesis

This thesis is divided in two main parts.

Part I [Introduction and Background] describes the problem that is addressed in this thesis and gives an overview of the technology that is built upon by the developed SON function. Chapter 2 [Long Term Evolution] gives an overview of LTE in general. It describes the different nodes of an LTE network and how these are interconnected. Furthermore a description of the LTE radio layer is given. Chapter 3 [Self-Organising Networks] gives an overview of the principles behind SON and explains typical SON functions. Furthermore an overview of literature and research projects that focus on SON is given. The LTE handover procedure is further detailed in Chapter 4 [Handover]. It explains how measurements are performed and how handovers are carried out. This procedure is crucial for understanding the working of the developed SON function in Part II. Readers that are familiar with how LTE works can skip (certain chapters of) this part.

Part II [SON Function and Results] contains the actual contributions of this thesis. It describes the developed SON function and its evaluation. Chapter 5 [SON Function Overview] explains the general principles behind the developed SON function and briefly describes its components. It also gives an overview of related algorithms. Chapter 6 [Simulation Modelling] gives an overview of the simulation scenario that is used to evaluate the developed SON function. Chapter 7 [Trajectory Classification], Chapter 8 [Trajectory Identification] and Chapter 9 [Traffic Steering] describe the different components of the SON function and the algorithms that are used by them in detail. Each chapter explains the general principles of the component and subsequently details the algorithms that are used in the component. Further each chapter explains how the component is evaluated and gives simulation results. Chapter 11 [Conclusion] wraps up this thesis summarising the results and giving future work.
Chapter 2

Long Term Evolution

2.1 Introduction

This chapter gives an overview of LTE. LTE is the evolution of UMTS. LTE improves amongst others the capacity, latency and speed of UMTS by employing a new radio technology and making significant changes to the core network. Work on LTE began in 2004 with the RAN Evolution Workshop where operators, manufacturers and research institutes presented their views and proposals on the evolution of Universal Terrestrial Radio Access Network (UTRAN) [37]. This first release was frozen in 2009 with the publication of Release 8, since then the LTE specification keeps being evolved.

2.2 Standardisation

Standardisation of LTE is done within 3rd Generation Partnership Project (3GPP). 3GPP unites seven telecommunications standard development organisations form around the world including the European Telecommunications Standards Institute (ETSI) in Europe. The work on extending and improving the 3GPP specifications is split over multiple Technical Specification Groups (TSGs). Each of these TSGs focuses on a different aspect of the standard. The four 3GPP TSGs are Radio Access Network (RAN), Service & Systems Aspects (SA), GSM/EDGE Radio Access Network (GERAN) and Core Network & Terminals (CT). The RAN TSG is responsible for the definition of the Universal Terrestrial Radio Access (UTRA) and Evolved Universal Terrestrial Radio Access (E-UTRA) functions, requirements and interfaces [42]. The SA TSG is responsible for the overall architecture and service capabilities of systems based on 3GPP specifications and, as such, has a responsibility for cross TSG co-ordination [43]. The GERAN TSG is responsible for the
specification of the Radio Access part of [GSM](/en.wikipedia.org/wiki/GSM)/Enhanced Data rates for GSM Evolution [EDGE](/en.wikipedia.org/wiki/EDGE). The [CTTSG](/en.wikipedia.org/wiki/3GPP) is responsible for specifying terminal interfaces (logical and physical), terminal capabilities (such as execution environments) and the Core network part of 3GPP systems [40]. The Working Groups (WGs) within the TSGs meet regularly and come together each quarter for a plenary meeting. In these plenary meetings their work is presented for information, discussion and approval [38].

Specifications are published in so-called “releases”. Each release provides a feature freeze. New functionality is added in further releases. Releases have a start date, a freeze date and an end date. From the start date until the freeze date, new features can be added to the release. After the freeze date no new features can be added to the release anymore. The release becomes obsolete after the close date. Release 8 was the first LTE release, the current release at the time of writing this thesis is Release 14. There are two kinds of documents: Technical Reports (TRs) and Technical Specifications (TSs) [39]. TRs are documents containing mainly informative elements approved by a TSG. TSs are documents containing normative provisions approved by a TSG and form the actual standard.

## 2.3 Network Elements

This section discusses the various network elements in an LTE network. An LTE network comprises two main parts: the Evolved Universal Terrestrial Radio Access Network (E-UTRAN) and the Evolved Packet Core (EPC). Figure 2.1 shows the LTE network architecture.

### 2.3.1 E-UTRAN

The E-UTRAN is the LTE radio access network. It consists of two types of nodes: eNodeBs (eNBs) and the User Equipments (UEs) that connect to them.

The UEs is the device, like, for instance, a smartphone, through which the end user communicates. The UE wirelessly connects to an eNB over the LTE radio interface. The eNB to which the UE is connected is called the SeNB. The UE is involved in setting up, tearing down and maintaining connections with eNBs. This includes transmitting and receiving data, performing and reporting measurements to the SeNB and executing commands coming from the network.

The eNB is the gateway between the RAN and the core network and relay information between the UE and the EPC. eNBs are spread over the landscape and are located next to the radio antenna towers. The eNB is in control of of all radio related functions like Admission Control (AC) and scheduling in the area that is covered by the antenna, called a cell. AC is the process that determines whether
Figure 2.1: The LTE network architecture.
CHAPTER 2. LONG TERM EVOLUTION

A UE is allowed to a cell or not. The scheduler determines which radio resources are assigned to which UE. Furthermore the eNB is in control of mobility management which is the focus of this thesis. The eNB decides when and to which cell to handover a UE based on measurements that are performed by the UE.

2.3.2 EPC

The Evolved Packet Core (EPC) is the LTE core network. It consists of a number of network elements including the Mobility Management Entity (MME), Serving Gateway (S-GW), Packet Data Network Gateway (P-GW), Home Subscription System (HSS) and Policy and Charging Resource Function (PCRF). These network elements usually reside at the operator’s premises. Some network nodes are only involved in control plane traffic, some only in user plane traffic and some in both.

The Mobility Management Entity (MME) is the main control node in the EPC. It is only involved in the control plane. The main functions of the MME are authentication and security, mobility management, managing subscription profile and service connectivity.

The Serving Gateway (S-GW) is responsible for routing data packets. It is mainly involved in the user plane and serves as mobility anchor for both inter eNB and inter RAT handovers. When a S-GW receives data for an idle UE it will trigger paging while buffering the data until the UE is ready to receive it.

The Packet Data Network Gateway (P-GW) is the gateway between the EPC and external packet data networks like the Internet. It is the highest mobility anchor in the network and is responsible for allocating the Internet Protocol (IP) address of the UE.

The Home Subscription System (HSS) is a database that contains all permanent user data. It stores the subscriber file that determines what services are applicable to the user. Furthermore it records the location of a user in the network in terms of which network control node it is connected to.

The Policy and Charging Resource Function (PCRF) is responsible for policy and charging control. It determines which QoS parameters to use for which data flow and is responsible for charging.

2.4 Interfaces

The various network components in LTE are connected through so-called interfaces. The communication between the network elements is split in two types: control plane and user plane. Control plane traffic is traffic that is sent between nodes of the


E-UTRAN and EPC to perform the signalling that is required to make the network operate properly like handover requests or billing information. User plane traffic is IP data traffic coming from the Packet Data Network (PDN) that is tunnelled from the P-GW to the eNB and subsequently sent to the UE and vice-versa.

The most important interfaces in the context of this thesis are the Uu, X2 and S1 interfaces. The Uu interface is the radio interface between the UE and the eNB and is discussed in detail in Section 2.5. The X2 and S1 interfaces connect components of the E-UTRAN namely eNB, either with each other or with the EPC.

### 2.4.1 S1 Interface

The S1 interface is at the boundary between the EPC and the E-UTRAN [14]. It connects the eNB with various components of the EPC. The most important functions of the S1 interface are paging; establishing, maintaining and releasing E-UTRAN Radio Access Bearers (RABs); and performing intra-LTE handover and inter-RAT handover. As is the case with most interfaces, the S1 interface is split in a control plane and a user plane. These are called S1-MME and S1-U as is indicated in Figure 2.1.

The S1 control plane interface carries various signalling traffic between the SeNB and the MME. The most important signalling traffic in the context of this thesis is control traffic related to handover. This includes requesting handovers, transferring status from the SeNB to the TeNB and changing the data path. The protocol stack of S1 Application Protocol (S1-AP) is shown in Figure 2.2. As can be seen the data is carried over a regular IP network. The lower layers are usually Ethernet or fiber although in theory other technologies can be used as well.

![S1 control plane protocol stack](image)

Figure 2.2: The S1 control plane protocol stack uses S1-AP on top of SCTP and IP.

The user plane uses GPRS Tunnelling Protocol (GTP) as is shown in Figure 2.3. The S1 user plane carries tunnelled data packets between the SeNB and S-GW.
2.4.2 X2 Interface

The X2 interface allows to interconnect eNBs with each other [15]. The purpose of the X2 interface is mainly to support the same communication facilities between eNBs as those that are offered via the S1 interface. This includes context transfers from one eNB to another, data transfer between eNBs and handover signalling. The presence of the X2 interface between two eNBs is however not required. An X2 interface will however speed up the communications between two eNBs for instance when performing handover or Inter Cell Interference Coordination (ICIC) and will reduce communication on the S1 interface.

The control plane uses X2 Application Protocol (X2-AP) on top of SCTP and IP as is shown in Figure 2.4. X2 interfaces can be set up manually, or by using Automatic Neighbour Relation (ANR). ANR finds new neighbours of the eNB based on UE measurements.

The user plane uses GTP on top of UDP and IP as is shown in Figure 2.5. The X2 user plane is used to forward data packets from the SeNB to the TeNB during handover.

2.4.3 Other Interfaces

Next to the S1 and X2 interfaces there are various other interfaces that connect the components of the EPC. Most of the interfaces that deal with data tunnelling and tunnel management like S5 and S11 use GTP for either the control plane, user plane
2.5. Physical Layer

This section gives an overview of the Physical Layer of the radio interface between the UE and the eNB, which is called LTE-Uu. The physical layer is discussed in a separate section as it is one of the most important changes in LTE relative to its predecessor and because it is of particular importance for this thesis.

LTE features two duplex modes: Frequency Division Duplex (FDD) and Time Division Duplex (TDD). With FDD, uplink and downlink transmission are separated by using different frequency bands called carriers. These frequency bands are paired and separated by a guard band. With TDD, only a single band is used which is alternatively used for downlink and uplink transmission.

The bandwidth of the bands that are used for downlink and uplink transmission (in case of FDD) or the duration of the downlink and uplink periods (in case of TDD) need to be the same. This can avoid wastage of valuable spectrum as traffic becomes more and more asymmetric with regard to downlink and uplink transmission [22].

Figure 2.6 illustrates the difference between the different duplex modes as well as the different TDD configurations. Note that in case of FDD the bandwidths of the bands that are used for downlink and uplink transmission need not to have the same width. There are seven different TDD configurations. Each TDD configuration has a different downlink versus uplink ratio allowing to accommodate downlink and uplink bandwidth requirements. The first and the sixth subframe of a frame is always a downlink subframe. This is because these subframes contain synchronisation signals. Furthermore, when switching from downlink to uplink transmission a Guard Period (GP) is added.
Figure 2.6: The difference between FDD and the different TDD configurations.
2.5. PHYSICAL LAYER

2.5.1 Resource Structure for FDD

This section assumes FDD mode. The resources in each band are divided, in the time dimension, in frames with a length of 10 ms. Frames are the top-level division of the radio resources in LTE meaning that the same pattern repeats every frame. A frame is divided, in the time dimension, into 10 subframes of 1 ms each, this period is called a Transmission Time Interval (TTI). Not all subframes in a frame are the same, subframes can contain slightly different information. The first and sixth subframe for instance contain synchronisation signals in the central six Physical Resource Blocks (PRBs).

Each subframe is in turn split, in the time dimension, in two time slots of 0.5 ms and, in the frequency dimension, into a certain number of parts of 180 kHz depending on the bandwidth. These subdivisions are called PRBs. The two adjacent PRBs in a subframe play an important role in scheduling as this are the resources that are assigned to users by the scheduler. Each subframe the scheduler will assign zero or more of these resources to each of the users. In the downlink direction any combination of resources can be assigned to a UE with the only limitation that the same Modulation and Coding Scheme (MCS) should be used across the assigned resources in the TTI. This means that at most the lowest MCS that can be used in all the PRBs that are assigned to the user can be used. In the uplink direction there is a further restriction that only a single block of consecutive resources can be assigned within each subframe. How these resources are divided among users is up to the scheduling algorithm that is active in each eNB. A scheduler will typically take the QoS requirements of a user, the MCS it can use to serve each user and how much bandwidth has been assigned to the user in the past into account to determine which resources it assigns to which user. A scheduler is not required to serve every user during each subframe, it only has to make sure that the QoS requirements for each user are fulfilled.

PRBs are subsequently subdivided, in the time domain, into seven (or six in case of an extended cyclic prefix) symbols and, in the frequency domain into twelve subcarriers. These subdivisions are called Resource Elements (RE) and are the smallest subdivision as they contain a single Orthogonal Frequency-Division Multiplexing (OFDM) symbol. Each RE starts with a cyclic prefix that is generated by adding the last part of the symbol at its beginning. This cyclic prefix is added in order to eliminate inter-symbol interference. The length of the cyclic prefix is 4.7 μs. This means that the useful symbol length is 66.7 μs. There is also the option to use an extended cyclic prefix. In this case a PRB is divided in six REs in the time domain instead of seven. The extra time is then divided among the cyclic prefixes of the remaining REs resulting in a cyclic prefix length of 16.7 μs, the useful symbol length remains unchanged.

Table 2.1 gives an overview of the different resources in LTE.

Figure 2.7 summarises the structure of the physical layer in the downlink direction.
Total bandwidth
1.4–20 MHz
1 frame
10 ms
12 subcarriers
180 kHz
1 subframe
1 ms
1 subcarrier
15 kHz
1 symbol
71.4 µs
Cyclic prefix
4.7 µs
66.7 µs

Figure 2.7: The structure of the physical layer.
2.5. PHYSICAL LAYER

<table>
<thead>
<tr>
<th>Name</th>
<th>Domain</th>
<th>Dimensions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frame</td>
<td>time</td>
<td>10 ms</td>
</tr>
<tr>
<td>Subframe</td>
<td>time</td>
<td>1 ms</td>
</tr>
<tr>
<td>Subcarrier</td>
<td>frequency</td>
<td>15 kHz</td>
</tr>
<tr>
<td>Slot</td>
<td>time</td>
<td>0.5 ms</td>
</tr>
<tr>
<td>Physical Resource block</td>
<td>time × frequency</td>
<td>0.5 ms × 180 kHz</td>
</tr>
<tr>
<td>Symbol</td>
<td>time</td>
<td>71.4 µs</td>
</tr>
<tr>
<td>Resource Element</td>
<td>time × frequency</td>
<td>71.4 µs × 15 kHz</td>
</tr>
</tbody>
</table>

Table 2.1: A summary of the resources in LTE

2.5.2 Reference Signals

While connected the UE will provide channel information feedback to the SeNB. This channel feedback is a recommendation to the SeNB which it can use for making scheduling decisions. The most important feedback that is provided by the UE is the Channel Quality Indicator (CQI). The CQI is one of 16 values, each corresponding to a MCS as listed in Table 2.2. The reported CQI is the highest possible MCS of which the Block Error Rate (BLER) will not exceed 10% [51, 7].

Table 2.2: The different CQIs and their properties.

<table>
<thead>
<tr>
<th>CQI</th>
<th>Modulation</th>
<th>Bits per symbol</th>
<th>Coding rate</th>
<th>Effective bits per symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>no transmission possible</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>QPSK</td>
<td>2</td>
<td>78/1024</td>
<td>0.1523</td>
</tr>
<tr>
<td>2</td>
<td>QPSK</td>
<td>2</td>
<td>120/1024</td>
<td>0.2344</td>
</tr>
<tr>
<td>3</td>
<td>QPSK</td>
<td>2</td>
<td>193/1024</td>
<td>0.3770</td>
</tr>
<tr>
<td>4</td>
<td>QPSK</td>
<td>2</td>
<td>308/1024</td>
<td>0.6016</td>
</tr>
<tr>
<td>5</td>
<td>QPSK</td>
<td>2</td>
<td>449/1024</td>
<td>1.1758</td>
</tr>
<tr>
<td>6</td>
<td>QPSK</td>
<td>2</td>
<td>602/1024</td>
<td>1.758</td>
</tr>
<tr>
<td>7</td>
<td>16 QAM</td>
<td>4</td>
<td>378/1024</td>
<td>1.4766</td>
</tr>
<tr>
<td>8</td>
<td>16 QAM</td>
<td>4</td>
<td>490/1024</td>
<td>1.9141</td>
</tr>
<tr>
<td>9</td>
<td>16 QAM</td>
<td>4</td>
<td>616/1024</td>
<td>2.4063</td>
</tr>
<tr>
<td>10</td>
<td>64 QAM</td>
<td>6</td>
<td>466/1024</td>
<td>2.7305</td>
</tr>
<tr>
<td>11</td>
<td>64 QAM</td>
<td>6</td>
<td>567/1024</td>
<td>3.3223</td>
</tr>
<tr>
<td>12</td>
<td>64 QAM</td>
<td>6</td>
<td>666/1024</td>
<td>3.9023</td>
</tr>
<tr>
<td>13</td>
<td>64 QAM</td>
<td>6</td>
<td>772/1024</td>
<td>4.5234</td>
</tr>
<tr>
<td>14</td>
<td>64 QAM</td>
<td>6</td>
<td>873/1024</td>
<td>5.1152</td>
</tr>
<tr>
<td>15</td>
<td>64 QAM</td>
<td>6</td>
<td>948/1024</td>
<td>5.5547</td>
</tr>
</tbody>
</table>

Depending on the CQI the SeNB will adapt the MCS that it uses to serve the UE. This process is called Adaptive Modulation and Coding (AMC). The MCS is the combination of the modulation and coding that is applied to the transmitted data. The modulation determines how a group of bits is transmitted over a wireless carrier. Coding determines the amount of Forward Error Correction (FEC) are
added to the transmitted data. In LTE three different modulations are used:

- Quadrature Phase-Shift Keying (QPSK) (illustrated in Figure 2.8(a)) uses 4 different phases to encode bits. QPSK is more robust against interference but only allows \(\log_2 4 = 2\) bit to be sent per symbol.

- 16 Quadrature Amplitude Modulation (QAM) (illustrated in Figure 2.8(b)) uses 16 different combinations of phase and amplitude to encode bits, allowing \(\log_2 16 = 4\) bit to be transmitted per symbol.

- 64 QAM (illustrated in Figure 2.8(c)) uses 64 different combinations of phase and amplitude to encode bits. 64 QAM allows \(\log_2 64 = 6\) bit to be transmitted per symbol but is more sensitive to interference.

![Constellation diagrams](image)

Figure 2.8: The different modulation schemes that are used in LTE.

The higher the number of bits that are transmitted in a single symbol is, the higher the probability that decoding a symbol will fail. Therefore QPSK will be used when the Signal-to-Interference-and-Noise Ratio (SINR) is low and 16 and 64 QAM will be used with higher SINR.
In order to further reduce the impact of transmission errors, FEC is applied. Applying FEC also allows the UE to estimate the BLER. With FEC not all bits are used for sending data, a certain fraction of bits are used for redundancy. This fraction can range from more than 92% to less than 8%. Table 2.2 lists the number of data bits per block of 1024 bits for a given MCS as well as the number of data bits that can be sent per symbol. Again, when the SINR is low more FEC bits will have to be added in order to be able to decode data than when the SINR is high.

By adapting the MCS it is possible to approach the Shannon bound as is shown in Figure 2.9.

Figure 2.9: By adapting the MCS it is possible to approach the Shannon bound.

2.5.3 Channels

There are three layers of channels: physical channels, transport channels and logical channels each associated with a different network layer. Physical channels, which are associated with the physical layer, are mapped directly on certain PRBs as is shown in Figure 2.10. Note that this figure shows the mapping for a 1.4 MHz carrier. The figure looks slightly different for wider bands as in this case the synchronisation signals are only present in the central six PRBs. Examples of physical channels include the Physical Broadcast Channel (PBCH) for broadcasting a limited number of parameters essential for initial access of the cell such as the downlink system bandwidth [62] or the Downlink Shared Channel (DL-SCH) for carrying downlink user plane data.

On top of the physical channels, transport channels, which are associated with the Medium Access Control (MAC) layer, are mapped. Logical channels, which are associated with the network layer, are in turn mapped on the transport channels.
Figure 2.10: The structure of the physical layer.
2.6 MAC Layer

The most notable functions of the MAC layer in LTE are Automatic Repeat Request (ARQ), multiplexing/demultiplexing, scheduling and random access.

ARQ is a mechanism that deals with erroneous transmissions by acknowledging frames upon correct reception. For each frame that is transmitted a timer is set. When an acknowledgement is received the corresponding timer is removed. When a timer expires the frame is retransmitted. LTE uses a combination of ARQ and FEC called Hybrid ARQ (HARQ). More specifically multiprocess HARQ is used with up to eight Stop-and-Wait (SAW) ARQ processes. This means that there can be up to eight data buffers for which when a block is sent the process will stop until feedback is received. When one process is unable to utilise the transmission medium another can do so making sure that all available resources can be used.

Another important function of the MAC layer is scheduling. The scheduler is responsible for distributing the available radio resources among the UEs that are connected to the eNB. In LTE the scheduler is located in the eNB. The scheduling algorithm in LTE is left to the eNB implementation, only the signalling to support the scheduling, like communicating the status of the data buffers at the UE, is standardised. The scheduling algorithm at the eNB should take the QoS requirements of the different UEs, the sizes of the buffers, the quality of the radio link, etc. into account.

The MAC layer has a number of additional responsibilities. These include random access for UEs that do not yet have a connection with an eNB and uplink timing alignment.

2.7 Network Layer

On the network layer communication is organised in so-called RABs. An RAB is a logical connection between the UE and the E-UTRAN and all communication between the UE and the E-UTRAN is performed over a RAB. There are bearers that carry signalling traffic and data bearers that carry user plane data. Bearers that carry control traffic use the Radio Resource Control (RRC) (between the UE and eNB) and Non Access Stratum (NAS) (between the UE and MME) protocols while user plane bearers carry IP packets.

With each RAB QoS requirements like a delay budget and a loss rate are associated. These QoS requirements are taken into account by the scheduler. Table 2.3 gives an overview of the different QoS classes. Each QoS class has a identifier called the QoS Class Identifier (QCI). Furthermore a QoS class is either for Guaranteed Bit rate (GBR) services like VoIP or video streaming, or non-GBR for services like browsing or email.
Table 2.3: The different QoS classes.  

<table>
<thead>
<tr>
<th>QCI</th>
<th>Resource Type</th>
<th>Priority</th>
<th>Delay Budget</th>
<th>Loss Rate</th>
<th>Example Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GBR</td>
<td>2</td>
<td>100 ms</td>
<td>$10^{-2}$</td>
<td>VoIP</td>
</tr>
<tr>
<td>2</td>
<td>GBR</td>
<td>4</td>
<td>150 ms</td>
<td>$10^{-3}$</td>
<td>Video call</td>
</tr>
<tr>
<td>3</td>
<td>GBR</td>
<td>5</td>
<td>300 ms</td>
<td>$10^{-6}$</td>
<td>Streaming</td>
</tr>
<tr>
<td>4</td>
<td>GBR</td>
<td>3</td>
<td>50 ms</td>
<td>$10^{-3}$</td>
<td>Real time gaming</td>
</tr>
<tr>
<td>5</td>
<td>Non-GBR</td>
<td>1</td>
<td>100 ms</td>
<td>$10^{-6}$</td>
<td>IMS signalling</td>
</tr>
<tr>
<td>6</td>
<td>Non-GBR</td>
<td>7</td>
<td>100 ms</td>
<td>$10^{-3}$</td>
<td>Interactive gaming</td>
</tr>
<tr>
<td>7</td>
<td>Non-GBR</td>
<td>6</td>
<td>300 ms</td>
<td>$10^{-6}$</td>
<td>Browsing, email</td>
</tr>
<tr>
<td>8</td>
<td>Non-GBR</td>
<td>8</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>9</td>
<td>Non-GBR</td>
<td>9</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
</tbody>
</table>
Chapter 3

Self-Organising Networks

3.1 Introduction

Contemporary networks have a multitude of parameters. In order to ensure efficient and effective network operation these parameters have to be tuned individually. Doing this manually is a tedious process and requires specialised expertise. Furthermore a manual update process to deal with changes that occur on a small timescale in a timely manner is impractical. As cellular networks become more and more complex automatically planning, configuring and optimising network elements and parameters becomes imperative. Networks that are automatically planned, configured, optimised and healed are called SONs. Automatically planning, configuring, optimising and healing a network also reduces the Capital Expenditure (CAPEX) and Operational Expenditure (OPEX) of deploying new network elements and during continuing operations. This is because applying SON reduces the need for placing additional cells and other infrastructure as existing resources are used more efficiently and less manual intervention is required in order to keep the network running optimally. LTE was designed with SON in mind. For instance, UEs support measurements and procedures which can be used for self-configuration and self-optimisation [11].

Within SON there is a distinction between a number of processes [56] [11]:

- **Self-configuration** refers to automatically configuring newly deployed network nodes to get the required configuration for operation. This is usually done in the pre-operational state.

- **Self-optimisation** refers to auto-tuning the network based on performance measurements that are made while the node or network is operational.

- **Self-healing** refers to automatically diagnosing problems and detecting irreg-
ular behaviour and subsequently remedying the problem.

- **Self-protection** refers to automatically protecting a system from attacks and maintaining its security and privacy.

Interest in SON grew with the introduction of 4G networks as managing fundamental functions of the RAN like resource allocation, mobility and interference management became more and more complex. Furthermore using SON allows to exploit the full potential of new features, leading to significantly higher performance by continuously adapting them to prevailing operational conditions. The first SON functions were independent closed-control-loop algorithms operating in a single RAT mainly 4G. However, due to their attractiveness for operators, SON functions have also been retrofitted to 2G and 3G. For these SON functions, two paths have been followed. The first is SON functions operating across multiple RATs and across layers (e.g., macro and micro cells). These SON functions mainly aim at improving mobility, load balancing and offloading processes. The second direction is the design of SON functions tailored to specific 4G technologies like vertical and virtual sectorisation or 4G relay stations.

### 3.2 Principles

Figure 3.1 shows a typical SON process [5]. When the network or a network component is switched on the self-configuration phase sets the initial configuration of a number of radio parameters or resource management algorithms. Examples of parameters that are set during the self-configuration phase are pilot powers and neighbour lists. These parameters can later be subject to optimisation by a self-optimisation function. A SON function itself will typically consist of two phases: a measurement phase and an optimisation phase. During the measurement phase relevant measurements are collected from various sources. Examples of measurements are radio channel characteristics, traffic and mobility information, etc. These raw measurements are processed into Key Performance Indicators (KPIs) which provide relevant information for the optimisation step. The format, accuracy and periodicity of the measured information depends on the specific SON function. Based on the measurements that are collected during the measurement phase an updated set of radio parameters is derived during the self-optimisation phase. Examples of radio parameters that can be optimised are antenna tilts, power settings, etc. Not all SON functions will contain these three phases nor will they adhere to a strict measurement/self-optimisation control loop. Many SON functions will however follow this principle.
3.3 Use Cases

Both 3GPP and Next Generation Mobile Networks Alliance (NGMN) have defined a number of agreed SON use cases and solutions [6, 3, 4]. This list is however far from exhaustive but it contains typical examples of problems that can be addressed by SON functions and how they can be addressed. This section explains a number of these use cases (i.e., situations where SON might be beneficial), and their solutions. These use cases presented here are all related to the subject of this thesis, there are however more SON use cases.

3.3.1 Inter Cell Interference Coordination

ICIC aims at reducing interference between cells, both in the uplink and downlink directions. The addition of new a cell regularly requires replanning of frequency assignments in wider areas. In 2G systems central planning of frequencies and power is done [2]. In 4G systems this problem however becomes harder as cells become smaller and more abundant.

A special challenge with ICIC is the detection of interference problems as this cannot be measured easily. Indicators for interference problems are a significant number of call drops, handover failures and customer complaints. Furthermore drive tests can be performed.

A typical ICIC SON function will monitor interference levels and transmit powers of the ScNB and the surrounding Neighbouring eNodeBs (NeNB). The SON function will then adapt the parameters of the scheduling algorithm such that the interference is reduced.

3.3.2 Mobility Robustness Optimisation

Mobility Robustness Optimisation (MRO) is concerned with automatically setting handover parameters to optimal values. Suboptimal handover parameter settings can cause Ping Pong Handovers (PPHO), Radio Link Failures (RLF) and Handover Failures (HOF) which will result in a negative user experience and wasted network resources.
PPHOs occur when UEs are handed over to a cell where they can only stay for a short amount of time before they are handed back over to the cell they came from as is illustrated in Figure 3.2. They are caused by handovers that are triggered either too early or to a wrong cell in which the UE can only stay for a short amount of time. When the cell reselection parameters, which determine to which cell a user connects when it starts a call, are suboptimally chosen the UE will be handed over to another cell soon after it connected to a cell which has similar drawbacks as a PPHO.

Figure 3.2: A PPHO occurs when a user that handed over from A to B returns to A in a short amount of time.

RLFs can be caused by various sources. When a handover occurs too early, a RLF might occur in the target cell shortly after the UE has connected to it successfully as the handover is triggered inside a small strip of coverage inside the source cell. If the UE cannot reconnect to the source cell shortly after the RLF a call drop may occur meaning that the user has lost connectivity. Also when the wrong target cell is chosen by the handover algorithm a RLF might occur. On the other hand, when a handover is triggered too late a RLF might occur in the source cell as it is no longer possible to maintain a connection with the serving cell as its signal strength fades. Usually the UE will reconnect to a cell different from the source cell shortly after the RLF.

HOFs occur when during the handover procedure, the UE cannot connect to the target cell and has to reconnect to the source cell. Like some RLFs, HOFs are caused by handovers that are triggered too early in a small island of coverage inside the source cell.

For these failures three causes can be derived:

- Too late handover triggering
- Too early handover triggering
- Handover to a wrong cell

A typical MRO SON function will mitigate these failures by optimising the following parameters which are explained in Chapter 4.
• Hysteresis
• Time-to-Trigger (TTT)
• Cell Individual Offset (CIO)
• Cell reselection parameters

The hysteresis is an extra offset that the signal strength of a possible handover target cell has to rise above the signal strength of the serving cell before a handover is considered. The TTT is the minimum amount of time that a UE has to experience this higher signal strength before a handover is considered. The CIO is like the hysteresis but it can be configured for each neighbouring cell separately. Most of these parameters are further explained in Chapter 4.

3.3.3 Mobility Load Balancing

The objective of Mobility Load Balancing (MLB) is to improve the performance by optimising the cell reselection and handover parameters such that traffic load is more equally balanced among cells while minimising the number of handovers that are required to achieve this goal. Doing so can improve the system capacity compared to static/non-optimised cell reselection/handover parameters while not affecting the QoS of the users negatively. Load balancing can be performed between cells of the same technology as well as between cells of different technologies.

A MLB SON function will typically monitor the load in the controlled cell and exchange this information with neighbouring cells. When the MLB SON function detects that the load becomes too high it will try to offload users to neighbouring cells that are under lower load. This can be achieved by delaying or advancing the handing over of UEs between cells, for instance via CIO adjustments.

3.4 Research

SON has been the subject of a number of research papers, research projects and books. This section gives a brief overview.

3.4.1 Research Projects

The SOCRATES (Self-Optimisation and self-ConfiguRATion in wirelEss networkS) project was a project within the European Seventh Framework Programme (FP7) that ran from 2008 until 2011 [26]. The project aimed at the development of self-organisation methods to enhance the operations of LTE networks, by integrating
network planning, configuration and optimisation into a single, mostly automated
process requiring minimal manual intervention [34]. In this project a number of SON
functions for self-optimisation, self-healing and self-configuration including SON
functions for admission control, handover, load balancing and ICIC were developed.
Furthermore the project studied the simultaneous operation of SON functions in
which conflicts between SON functions that are active in the same network is studied
and proposed a framework for resolving conflicts between SON functions.

The SEMAFOUR (Self-Management for Unified Heterogeneous Radio Access Net-
works) project was the follow up of the SOCRATES project that ran from 2012
until 2015 [27]. The goal of the project was to design and develop a unified self-
management system, which enables the network operators to holistically manage
and operate their complex heterogeneous mobile networks. SON was an import-
ant aspect of this project, in particular SON functions that address different RATs
simultaneously [16]. The developed SON functions focused on dynamic spectrum al-
location and interference management, multi-layer LTE/Wi-Fi traffic steering, tackling
the problem of high mobility users and active/reconfigurable antenna systems.
Another important aspect of the project was to design and develop an integrated
SON management system which interfaces between operator-defined performance
objectives and the set of multi-RAT/multi-layer SON functions and to develop a
SON coordinator. The author was involved in both the SOCRATES and SEMA-
FOUR projects.

Also, a number of other research projects have developed SON functions or have at
least studied them. These projects include METIS [36], UniverSelf [24], SHARING
[33], SAPHYRE [31] and COMMUNE [25].

3.4.2 Literature

SON has also been studied extensively in literature as can be seen from the various
conferences and workshops that are dedicated to the topic like the International
Workshop on Self-Organizing Networks (IWSON) [29], the Self-Organising Net-
works Conference (SON [32] and the International Workshop on Self-managing and
Autonomous Networks (SAN) [28].

[30] gives an overview of SON in general. It lists a higher number of network para-
eters; parallel operation of 2G, 3G and 4G networks; and the growing number of
base stations as the main drivers for SON. It further gives an overview of different
SON architectures. These are centralised SON where optimisation algorithms are executed in a centralised location and distributed SON where optimisation al-
gorithms are executed at each eNB. Centralised and distributed SON can also be
combined in hybrid SON. The paper also lists a number of SON functions, most of
which are listed in Section 3.3.

Handover optimisation also has been the subject of papers.
3.4. RESEARCH

In [59] a self-optimisation algorithm for handover in LTE is presented. The algorithm adapts the hysteresis parameter by measuring the performance for a parameter setting $p$ as well as for parameter settings $p + \Delta$ and $p - \Delta$ where $\Delta$ is a fixed offset. After each optimisation cycle the new value for $p$ is set to the parameter setting which resulted in the best performance. The proposed algorithm was evaluated using simulations and results show that the number of dropped calls is reduced significantly.

In [44] a self-optimisation algorithm for handover oscillation control is proposed. A handover oscillation is a handover that is initiated less than 1 s after the UE connected to the cell. The goal of this algorithm is to minimise the oscillation rate which is defined as the ratio of the number of handover oscillations and the total number of handovers. It does this by adapting the hysteresis and TTT with increments that are calculated based on the difference of the current oscillation rate and the target oscillation rate. Results show that the optimisation algorithm is able to prevent handover oscillations as well as an algorithm with conservative fixed handover parameters but that the downlink and uplink throughputs are improved by 20%.

The handover self-optimisation algorithm that is presented in [61] also adapts both the hysteresis and TTT. In each optimisation period the performance is measured based on an aggregated performance measure which takes into account ping-pong handovers, radio link failures and handover failures. This performance metric is then compared to a performance threshold. If the performance is good the performance threshold is lowered to make it more strict. If the performance is bad and optimisation is possible the handover parameters are updated in such a way that the performance should be improved. The way the parameters are updated is derived from a lookup table. If optimisation is not possible the performance threshold is increased to make it more relaxed. Results show that the algorithm is able to increase the system performance significantly and that the algorithm is able to deal with changes in its environment.

The topic of avoiding unnecessary handovers, which is also the subject in this thesis, has been studied in [45]. In this paper, the authors predict the direction a user is moving to based on Global Positioning System (GPS) locations of both the user and the cells. Based on this prediction the most suitable handover target is selected. Furthermore the TTT is adapted based on the velocity of the users in order to trigger the handover at the most optimal time.

[55] tackles the problem of frequent handovers in the context of small cells. The algorithm assumes that the future path of users is known and that the best server areas of all base stations in the network are known. Based on this information the algorithm will determine the most optimal handover sequence. Results show that by doing this the number of handovers can be reduced by half.

In [57] user movement is extracted from a real data set used to train a Hidden Markov Model. The user movement data that is used consists of the cells that have been visited by the users in the past. The Hidden Markov Model uses the current
and previous location of users to predict the next possible location of the user. The predictions of the algorithm was compared to the actual behaviour of users in a data set that was collected in a campus network. Results show that the algorithm makes correct predictions in 80% to 85% of cases.

SON has also been the subject of a number of books. A notable example is [50]. This book describes the principles behind SON and gives an overview of common SON use cases. Furthermore an in-depth description of the various aspects of SON namely self-configuration, self-optimisation and self-healing; is given. The book also discusses SON for the core network, interactions between SON functions and SON for heterogeneous networks.

3.5 Evolution of SON

Together with the evolution towards 5th Generation (5G) where high flexibility results in a tremendous number of possible network settings and deployment options, SON functions also become more sophisticated and evolve towards cognitive functions. Cognition has first been introduced in the radio communication field by Mitola [52] in 2000 under the cognitive radio paradigm. Cognitive functions are aware of the user and network needs and can adapt their functioning accordingly using machine learning. They can furthermore learn from their past experience and make intelligent decisions and evolve over time based on this experience.

The SON function that is developed in this thesis also learns from the past and has therefore traits of a cognitive function.
Chapter 4

Handover

4.1 Introduction

This chapter explains the handover procedure in LTE. Handover is performed for various reasons. The most important reason is to maintain, preferably seamless, connectivity as a user moves in and out the best server area of consecutive cells. Another reason to perform handover is for balancing load between adjacent cells.

Handover in LTE is network driven and user assisted. This means that the network decides whether and to which cell to handover a UE based on measurements that are made by the UE. In case of LTE it is the SeNB that makes the handover decision. The UE however provides the SeNB with information which it can use to make the handover decision.

The actual handover decision algorithm is not standardised. Vendors are thus free to implement their own handover algorithms. 3GPP has however standardised means for the SeNB to let the UE perform measurements that can aid the SeNB in the handover process in the form of measurement reports [11].

There are two types of handover named X2 handover and S1 handover, referring to the interfaces over which they are carried out. X2 handovers are performed over the X2 interface, a direct link between two eNBs. Whenever this link is available between two eNBs, a handover can be performed without the intervention of the EPC. Whenever an X2 interface is not available between the SeNB and the TeNB the handover will be performed over the S1 interface which connects the eNB to the EPC. This means that the EPC is involved in the handover which causes an additional communication delay and communication overhead in the EPC. In this thesis X2 handovers will be assumed although the algorithms, results and conclusions will also valid in case of S1 handovers as in this case the signalling traffic will be routed through the core network leading to an even higher resource consumption.
and longer latencies.

4.1.1 Overview

Figure 4.1 depicts a sequence diagram of the handover procedure in LTE. The entire handover process consists of multiple phases. After a UE connects to a new SeNB, either because it started a new connection or because it was handed over to the SeNB, the SeNB instructs it which measurements should be made and which events should be reported (step 1). As long as it is connected to the SeNB, the UE performs the requested measurements and reports the requested events to its SeNB (step 2). The SeNB is free to do with these measurements as it desires. It could for instance collect a number of measurements in order to make a deliberate handover decision. At some point the SeNB will however make a handover decision (step 3). After a handover decision has been made the actual handover procedure starts with the handover preparation phase. In this phase the SeNB first sends a handover request to the selected TeNB (step 4). This request contains information about the connections of the UE. The TeNB will decide whether to admit the UE or not (step 5) and sends a handover request acknowledgement to the SeNB containing its decision (step 6). If the UE is admitted to the TeNB, the SeNB will on its turn send a handover command to the UE instructing it to perform a handover to the TeNB (step 7). With this step the handover preparation phase is finished and the handover execution phase starts. During the handover execution phase the UE detaches from the SeNB and connects to the TeNB (step 9). At the same time, the SeNB will forward incoming packets, destined for the UE to the TeNB which will buffer them (step 8) until the UE connects to it. After the UE has connected to the TeNB (step 11) the data path is switched from the SeNB to the TeNB (steps 12–16) and the resources related to the UE at the SeNB are freed (steps 17 and 18).

4.2 Measurement Configuration

LTE allows SeNBs to configure connected UEs to perform certain measurements [13]. This is done through the RRC Connection Reconfiguration message which is sent from the SeNB to the UE as is shown in Figure 4.2. The RRC Connection Reconfiguration message contains a number of measurement objects. Measurement objects specify on which entities the UE has to perform measurements. Each measurement object is either a single E-UTRA carrier, a single UTRA carrier, a set of GERAN carriers or a single CDMA2000 carrier frequency. Furthermore the RRC Connection Reconfiguration message contains a number of reporting configurations. Reporting configurations specify how the measured values should be reported to the SeNB. They consist of a reporting criterion which specifies whether measurements should be reported periodically or event-driven and a reporting format which specifies the measurements that should be reported to the SeNB. Measurement objects
Figure 4.1: The LTE X2 handover procedure [III].
and reporting configurations are linked to each other by measurement identities which are also present in the RRC Connection Reconfiguration message. Each measurement identity maps one measurement object to one reporting configuration. They allow a many-to-many relationship between measurement objects and reporting configurations. Next to measurement objects, reporting configurations and measurement identities, RRC Connection Reconfiguration messages also contain quantity configurations and measurement gaps. These specify which measurements have to be made by the UE and when the UE should perform measurements. The measurement configuration of a UE can be changed over time by sending additional RRC Connection Reconfiguration messages that add, remove or modify measurement objects, reporting configurations and/or measurement identities.

![Diagram](image)

Figure 4.2: The RRC Connection Reconfiguration message provides the measurement configuration to the UE [11].

The measurements that can be made depend on the measured RAT. For E-UTRAN the measurements that can be made are the Reference Signal Received Power (RSRP) and Reference Signal Received Quality (RSRQ). For UTRAN these are the CPICH (Common Pilot Channel) Received Signal Code Power (RSCP) and Energy per Chip over Total Noise Power Density ($E_c/N_0$), for GERAN the Received Signal Strength Indicator (RSSI) and for CDMA2000 the Energy per Chip over Total Interference Power Density ($E_c/I_0$). In the remainder of this thesis, these measurements (possibly after filtering) are in general referred to as measurement results.

### 4.3 Performing Measurements

Figure 4.3 shows a general overview of how measurements are performed and processed at the UE. The physical layer (layer 1) performs measurements and supplies these to higher layers (A). These measurements are filtered for a first time in an implementation defined way and are subsequently provided (B) to the layer 3 filter. The working of this filter is standardised and can be configured by the SeNB. Based
4.3. PERFORMING MEASUREMENTS

on the output of the layer 3 filter for multiple cells \((C, C')\) the reporting criteria are evaluated resulting in measurement reports \((D)\) which are sent to the SeNB.

![Figure 4.3: The measurement pipeline [11].](image)

4.3.1 Layer 1 Measurements

In each LTE subframe, reference signals are transmitted in certain resource elements as is shown in Figure 4.4. In these reference signals known signals that carry no data are transmitted. From these reference signals the UE can estimate the channel. Figure 4.4 shows the positions of the downlink reference signals in two consecutive resource blocks (one subframe in time).

One of the measurements that is performed by the UE is the RSRP. The RSRP is the linear average over the power contributions in watt of the resource elements that carry Cell-Specific Reference Signals (C-RS) [12]. It is an indicator of the signal strength of a cell and is used for cell reselection and handover.

From the RSRP the RSRQ is calculated by dividing the RSRP by the RSSI and multiplying it by the number of PRBs \(N\) over which the RSSI is measured as is shown in Equation 4.1:

\[
\text{RSRQ} = \frac{N \cdot \text{RSRP}}{\text{RSSI}}
\]  

4.3. PERFORMING MEASUREMENTS

The RSSI is defined as the linear average of the total received power in watt from all sources including interference and noise from various sources. The 3GPP specifications do not specify over how many PRBs the measurements should be made but only that both numerator and denominator are made over the same set of PRBs.

As its name suggests the RSRQ is intended to also capture the signal quality of a cell as opposed to the pure receive power as is the case with the RSRP.

The 3GPP specifications do not exactly specify how layer 1 measurements should be made apart from the aforementioned definitions. Moreover, it is not specified
CHAPTER 4. HANDOVER

Figure 4.4: Downlink reference signals. [19]
4.3. PERFORMING MEASUREMENTS

how layer 1 should filter the measurements nor at what rate they should be reported to layer 3 or even whether they have to be provided at regular intervals. The specifications only state that the measurements have to meet certain accuracy requirements [10, 18]. In practice the layer 1 measurements will be performed in periods which have an order of magnitude of 1 ms and the period between the reporting of measurements to higher layers will have an order of magnitude of 10 ms.

4.3.2 Layer 3 Filtering

In contrast to layer 1 filtering which is not standardised and whose implementation can be freely chosen by a vendor within certain boundaries, layer 3 filtering is standardised by 3GPP [11]. Each new filtered layer 1 measurement $M_n$ is filtered by the layer 3 filter using the formula given in Equation 4.2, yielding the filtered layer 3 measurement $F_n$.

$$F_n = (1 - \alpha)F_{n-1} + \alpha M_n$$  \hspace{1cm} (4.2)

This means that each new filtered value $F_n$ is the weighed average of the new measurement $M_n$ and the previous filtered value $F_{n-1}$ with respective weights $\alpha$ and $1 - \alpha$. This kind of filter is called an exponential filter because the influence of the old filtered value decreases exponentially as new measurements are added. The parameter $\alpha$ is called the smoothing factor and determines the importance of the new measurements and how fast the contribution of older measurements will diminish. Its value can be configured by the SeNB by means of a filter coefficient $k$ which can have the value 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 13, 15, 17 or 19. The value of $\alpha_{200\text{ ms}}$ is calculated according to Equation 4.3.

$$\alpha_{200\text{ ms}} = \frac{1}{2^k}$$  \hspace{1cm} (4.3)

Note that when $k$ is set to 0 layer 3 filtering is effectively disabled and the filtered layer 3 measurements are the same as the filtered layer 1 measurements.

The output rate of the layer 3 filter is the same as the input rate, meaning that for every filtered layer 1 measurement there will be a filtered layer 3 measurement. Because the rate at which new measurements arrive (and thus are output) is not standardised the smoothing factor $\alpha_{200\text{ ms}}$ should be adapted on the measurement interval such that the characteristics of the adapted filter are as if the measurement interval is equal to 200 ms. In order to do this the actual value of $\alpha_{200\text{ ms}}$ should be adapted as in Equation 4.4, yielding the actual value that is used for smoothing $\alpha$.

$$\alpha = 1 - \sqrt[1 - \alpha_{200\text{ ms}}]$$  \hspace{1cm} (4.4)

Using filter value $\alpha$ will make sure that when the sample interval is $x$ times 200 ms, the importance of the old filtered value after $x$ new measurements is the same as after a single new measurement when using the smoothing factor of $\alpha_{200\text{ ms}}$. 
4.4 Measurement Reporting

A UE will send measurement reports to its SeNB according to its reporting configuration (see Section 4.2). A measurement report contains the following information:

- The measurement identity that triggered the sending of the report.
- The measurement result of the SeNB.
- If applicable the measurement results of a number of neighbouring cells.

The sending of these measurement reports can be triggered either periodically or event-driven. In case of periodic reports a report is sent at regular intervals. The reporting period can vary from 120 ms to 1 h. Periodic reports include the RSRP and/or RSRQ of the SeNB depending on the reporting configuration and measurement results of certain neighbouring cells. Event-driven reports are sent whenever a certain event occurs. Section 4.4.1 describes how these events are triggered. When an event-driven report is triggered the UE can be configured to perform up to 64 periodic reports.

4.4.1 Event-driven Report Triggering

Event-driven reports are triggered by monitoring certain events. These events all have an associated entering and leaving condition which are inequalities involving the measurement result of either the SeNB, a neighbouring cell or both. Each event also has an associated hysteresis and TTT. The hysteresis is a threshold that is added to or subtracted from the measurement result when evaluating the entering or leaving conditions. Its purpose is to avoid that once an event is triggered or untriggered it is not immediately untriggered or retriggered again respectively. The hysteresis can be set to a value ranging from 0 dB to 15 dB in steps of 0.5 dB. The TTT is the amount of time that the entering or leaving condition has to hold before the event is triggered or untriggered. Like the hysteresis, the purpose of the TTT is to avoid that events are immediately triggered or untriggered again because of fluctuations in the measurement results. The hysteresis and TTT are individually set for each reporting configuration but the same value is used for both the entering and leaving conditions. The TTT can be set to the following values: 0 ms, 40 ms, 64 ms, 80 ms, 100 ms, 128 ms, 160 ms, 256 ms, 320 ms, 480 ms, 512 ms, 640 ms, 1.024 s, 1.28 s, 2.56 s and 5.12 s. For each measurement object, a cell individual offset can be specified per cell. This offset is added to the measurement result of the cell when evaluating the entering and leaving conditions of an event. When discussing the triggering and untriggering of events in the following sections RSRP, RSRQ or the term ‘measurement result’ refers to the output of the layer 3 filter plus the cell individual offset.

Regarding an event, cells can be either triggered or not. Initially cells are not triggered for any event. For each event, the entering condition is evaluated for the
relevant cells (i.e., cells for which it makes sense to evaluate the event) that are not triggered for the event each time there is a new layer 3 measurement. Once the entering condition holds for a continuous amount of time that is greater than or equal to the TTT that is associated with the event the cell is triggered for that event and the cell is added to the list of triggered cells. For cells that are in the triggered list of an event the leaving condition of that event is evaluated each time a new measurement is made. Once the leaving condition holds for a continuous amount of time that is greater than or equal to the TTT that is associated with the event the cell is untriggered for that event and the cell is removed from the list of triggered cells. The triggering and untriggering of cells is illustrated in Figure 4.5.

![A state diagram depicting the triggering and untriggering of cells.](image)

There are three classes of events, labelled A, B and C. A events are events that only involve LTE cells. These compare the RSRP or RSRQ of the SeNB and/or a NeNB with each other or with a fixed threshold.

B events are events that involve neighbouring cells of RAT other than LTE notably UTRAN, GERAN and CDMA2000. Because of this, comparisons between the measurement results of the SeNB and the neighbouring cell are not possible. The events therefore only compare the measurement results of the SeNB and neighbouring cells with fixed thresholds.

C events are events that compare the CSI Reference Signal Received Power (CSI-RSRP) of LTE cells instead of the RSRP or RSRQ. CSI-RSRP is calculated based on CSI Reference Signal (CSI-RS) instead of C-RS and is defined as
the linear average over the power contributions in watt of the resource elements that carry CSI-RS. C events were not part of the initial LTE standard and were introduced in 3GPP release 12.

Not all these events are of importance for this thesis neither are all events used as frequently by other algorithms. The events that are important for this thesis and the ones that are commonly used are discussed below.

**Event A1: ‘Serving becomes better than threshold’**

Figure 4.6 shows the triggering and untriggering of event A1: ‘Serving becomes better than threshold’. This event has an associated threshold to which the RSRP/RSRQ of the SeNB is compared. When the RSRP/RSRQ of the SeNB minus the associated hysteresis becomes greater than the associated threshold (A) and remains greater for the associated TTT, the event is triggered (B). The event is untriggered again when the RSRP/RSRQ of the SeNB plus the hysteresis becomes less than the associated threshold (C) and stays less for the associated TTT (D).

![Figure 4.6: The triggering and untriggering of event A1.](image)

Event A1 is mainly useful for triggering periodic reporting or enabling or disabling additional reporting configurations.
Event A2: ‘Serving becomes worse than threshold’

Figure 4.7 shows the triggering and untriggering of event A2: ‘Serving becomes worse than threshold’. This event has an associated threshold to which the RSRP/RSRQ of the SeNB is compared. When the RSRP/RSRQ of the SeNB plus the associated hysteresis becomes less than the associated threshold (A) and remains less for the associated TTT, the event is triggered (B). The event is untriggered again when the RSRP/RSRQ of the SeNB minus the hysteresis becomes greater than the associated threshold (C) and stays greater for the associated TTT (D).

Like event A1, event A2 is mainly useful for triggering periodic reporting or enabling or disabling additional reporting configurations.

Event A3: ‘Neighbour becomes offset better than serving’

Figure 4.8 shows the triggering and untriggering of event A3: ‘Neighbour becomes offset better than serving’. This event has an associated offset, called the CIO, that is added to the RSRP/RSRQ of the SeNB. When the RSRP/RSRQ of the considered NeNB minus the associated hysteresis becomes greater than the RSRP/RSRQ of the SeNB plus the associated offset (A) and remains greater for the associated TTT, the event is triggered (B). The event is untriggered again
when the $\text{RSRP/RSRQ}$ of the considered NeNB plus the hysteresis becomes less than the $\text{RSRP/RSRQ}$ of the SeNB plus the associated offset ($C$) and stays less for the associated TTT ($D$).

Unlike the other events, this event has an additional report-on-leave flag. If this flag is set a measurement report will also be sent when an eNB is removed from the list of triggered eNBs. The A3 event is often used by handover algorithms to trigger handovers, especially between cells that operate in the same frequency band.

**Event A4: ‘Neighbour becomes better than threshold’**

Figure 4.9 shows the triggering and untriggering of event A4: ‘Neighbour becomes better than threshold’. This event has an associated threshold to which the $\text{RSRP/RSRQ}$ of the NeNB is compared. When the $\text{RSRP/RSRQ}$ of the considered NeNB minus the associated hysteresis becomes greater than the associated threshold ($A$) and remains greater for the associated TTT, the event is triggered ($B$). The event is untriggered again when the $\text{RSRP/RSRQ}$ of the NeNB plus the hysteresis becomes less than the associated threshold ($C$) and stays less for the associated TTT ($D$).
4.4. MEASUREMENT REPORTING

Event B1: ‘Inter RAT neighbour becomes better than threshold’

Figure 4.10 shows the triggering and untriggering of event B1: ‘Inter RAT neighbour becomes better than threshold’. This event has an associated threshold to which the measurement result of the neighbour is compared. When the measurement result of the considered neighbour minus the associated hysteresis becomes greater than the associated threshold (A) and remains greater for the associated TTT, the event is triggered (B). The event is untriggered again when the measurement result of the neighbour plus the hysteresis becomes less than the associated threshold (C) and stays less for the associated TTT (D).

Event B1 is very similar to Event A4, the only difference is that Event A4 applies to E-UTRA neighbours while event B1 applies to inter-RAT neighbours.

4.4.2 Measurement Report Sending

Whenever an event is triggered or, in case of events A3 and A6, untriggered and the report-on-leave flag is set or when the periodic report timer expires, a measurement report is sent to the SeNB. This measurement report contains the following information:

- The measurement identity of the event that triggered the measurement re-
Measurement Result

<table>
<thead>
<tr>
<th>Neighbour MR</th>
<th>Threshold</th>
<th>Hysteresis</th>
<th>Time-to-trigger</th>
</tr>
</thead>
</table>

Figure 4.10: The triggering and untriggering of event B1.

- The RSRP and/or RSRQ result of the SeNB.
- The measurement results of the relevant neighbouring cells.

The list of neighbours which are included in the measurement report depends on the trigger that caused the measurement report to be sent. If the measurement report is sent because of a periodical trigger the list of neighbours includes the neighbours for which new measurement became available since the previous report. If the measurement report is triggered by an event, the list of neighbours contains the cells which are in the set of triggered cells for the event. Note that in case of events A1 and A2 there are no neighbours cells in the triggered list. The length of the list of neighbours is limited up to a certain maximum. In case there are more cells to report than can be included in the message, only the strongest 32 cells are included.

### 4.5 Handover Decision

The algorithm for making the decision to handover a user is not standardised by 3GPP and can be freely chosen by the vendor of the equipment. The algorithm that takes this decision is implemented at the eNB. In order to obtain the required information to take a proper handover decision the algorithm can use the
measurement reporting mechanism that is described in the previous sections. Many
handover algorithms use event A3 to trigger a handover.

Handover algorithms and handover optimisation algorithms have been studied in a
number of papers.

A very common and simple handover algorithm is presented in [46]. This algorithm
configures a single A3-event with a certain hysteresis and TTT. When this event
is triggered and a measurement report is sent to the SeNB a handover to the cell
with the highest RSRP will be initiated. Usually this is the cell that triggered the
event as at first there will only be one cell in the list. When the selected target cell
refuses to admit the UE or when the handover fails other cells can however be chosen
as more measurement reports are sent. The hysteresis and TTT avoid handovers
being triggered too soon and consequently PPHOs and RLFs. The performance of
this algorithm is studied in [47]. Results show that this algorithm performs quite
well in normal circumstances but that its performance can be improved upon. This
algorithm is often used as the basis for applying SON. In this case the hysteresis
and TTT parameters are automatically tuned based on measurements according to
some preset performance targets as is done in [61] and [63].

4.6 Handover Preparation

Once the handover decision has been made, the SeNB will send a handover request
to the chosen TeNB. The handover request contains information about the UE
including its bearers. Upon receiving the handover request, the TeNB will make a
decision about whether the UE will be admitted to the cell or not. The admission
control algorithm that makes this decision is, like the handover algorithm, not
standardised by 3GPP and its implementation is free for the vendor to choose.
Typical admission control algorithms however typically take the load in the cell and
the requirements of the user into account. The TeNB will send a handover reply
back to the SeNB including the handover decision. The SeNB can send handover
requests for the same UE to multiple candidate TeNBs at once, this is called multiple
preparation. In this case, when the handover to the preferred TeNB fails, the UE can
be handed over to one of the other prepared TeNBs more quickly as the handover
preparation does no longer have to be performed. The excess handover requests
can be cancelled later on by the SeNB once they are no longer needed.

4.7 Handover Execution

Once the SeNB has received a handover reply from the TeNB it sends a handover
command to the UE. This handover command is a RRC Connection Reconfigura-
tion message that includes mobility control information. This message instructs the
UE to perform a handover to the selected TeNB. The mobility control information
contains the ID of the TeNB, its carrier frequency and bandwidth, the amount of
time the UE should try to connect to the TeNB and other information that is re-
quired to connect to the TeNB. When the UE receives this message it disconnects
from its current SeNB and performs the random access procedure with the TeNB.
This is called a hard, or break before make, handover. Once this procedure is suc-
cessfully completed, the TeNB which is now the new SeNB will send configuration
information to the UE which is confirmed by the UE after which data transmission
can resume. If the random access procedure does not succeed after the time that
is included in the handover command, a HOF is said to occur. In this case the
UE aborts the random access procedure at the TeNB and tries to reconnect to the
SeNB. If this is successful, the SeNB might command the UE to perform a handover
to another cell, for instance, if multiple preparation is used, one of the other TeNBs
to which a handover request was sent. If this fails the connection is lost.

4.8 Handover Completion

After the UE has successfully connected to the TeNB, the data path is rerouted
from the old SeNB to the TeNB which is now the new SeNB. Before this switch,
all downlink data was sent to the old SeNB and then forwarded to the TeNB while
the uplink data is forwarded immediately by the TeNB.
Part II

SON Function and Results
Chapter 5

SON Function Overview

5.1 Introduction

In case of a dense deployment of cells and/or when user velocities are high, frequent handovers and short stays will occur. These frequent handovers and short stays will cause a degradation of user experience and network performance as the high handover interruption time, and the radio conditions during handover have a negative impact on user experience. In addition there will be an increased signalling and data overhead in the core network due to the handover signalling and data forwarding. In this thesis a SON function is developed and assessed that deals with this problem. The goal of this SON function is to classify users according to their mobility behaviour and, based on this classification, steer them in an intelligent way such that the handover rate is reduced while the resource consumption is not severely impacted. Users can be steered within the same layer (macro, micro, pico cells) and [RAT] [LTE UMTS ...] but also between different layers and/or possibly [RAT] depending on their mobility behaviour and the availability of layers and [RAT]s to steer them to. This chapter gives a high-level overview of the developed SON function. Subsequent chapters will explain the components of the SON function in more detail and the algorithms that are used by these components in detail and will assess their performance.

5.2 Short Stay Reduction

The problem of short stays can be mitigated in a number of ways.

Users that move at a relatively high velocity can be steered towards macro cells. In this way they could stay connected to their SeNB for a longer amount of time.
before making a handover which would reduce the handover frequency. On the other hand, users that move at a slow velocity could be steered to smaller cells for better coverage and throughput. This has the additional benefit that the load in the surrounding macro cells is reduced. This is however not a direct goal of the proposed SON function.

Another way in which the handover frequency can be reduced is by avoiding unnecessary handovers. These are handovers that make the user switch to a cell in which it stays for only a brief amount of time and that are not required to maintain connectivity. These can occur when users pass through the edge of a cell or through isolated patches of a cell. Figure 5.1 illustrates such a situation. The user whose trajectory is indicated by the arrow travels from cell A to cell B. During this travel it briefly has the best coverage in cell C. A naive handover algorithm that bases its decisions only on signal strength would first handover the user from A to C and shortly thereafter from C to B. In case it is possible for the user to maintain connectivity with A until it can be handed over to B, the brief period during which the user will experience better throughput in C will not outweigh the drawbacks of the additional handover. In this case a handover can be avoided by not handing over the user to C but immediately from A to B.

For a typical handover algorithm that only takes information coming from the user itself into account in order to determine the best handover strategy, it is difficult to avoid unnecessary handovers. The handover algorithm could however take historical information coming from other users into account and try to predict the future behaviour of users. As users usually travel through cells along the roads that run through it there are only a limited number of possible trajectories that can be followed. Furthermore not every trajectory through a cell is as likely to be followed
by a user. For instance, vehicular users that follow a highway through a cell and enter it from one side will very likely stay on it and exit it at the other side.

5.3 Rationale

The rationale behind the developed SON function is:

Users that follow similar trajectories as other users that were active in the past will likely have the same mobility behaviour.

The SON function will collect measurements from users that allow it to match users that follow similar trajectories through a cell. It will match the measurements from currently active users to measurements from users that were active in the past. When it finds such a match between the active user and a user that was active in the past it will assume that the future behaviour of the currently active user will be similar to the behaviour of the user that was active in the past. Based on the information that was collected by the user that was active in the past the SON function estimates the future behaviour of the currently active user and makes decisions on how to steer the user.

5.4 Architecture

The general architecture of the developed SON function is shown in Figure 5.2. It consists of three main components: the Trajectory Identifier, the Trajectory Classifier and the Traffic Steerer. The SON function will be placed at the eNB and will operate on a per-cell level. This allows the gradual deployment of the SON function in the network.

The Trajectory Classifier is the core of the SON function. It is responsible for classifying users according to the trajectory they follow through the cell. It does this by comparing measurement traces that are made by currently active users (active traces) to measurement traces that were made by users that were active in the past (reference traces), and as such identifying matching traces of measurements. The measurements that are used for composing the measurement traces are the consecutive measurement reports that are sent by a UE to its SeNB (see Section 4.4). For the matching, the Modified Dynamic Time Warping (MDTW) algorithm which is discussed in Chapter 7 is used. The used measurements are part of the LTE specifications and are guaranteed to be available for all users in contrast to, for instance, GPS data which would require non-standardised support from the UE and would therefore not always be available.
Figure 5.2: The SON function consists of three components: the Trajectory Identifier, the Trajectory Classifier and the Traffic Steerer.
The Trajectory Identifier is responsible for identifying the traces that will serve as reference traces. This means selecting new reference traces as well as removing old, obsolete reference traces. When a user arrives in a cell, either by starting a call or because it is handed over to the cell during an ongoing call, it becomes, with a certain probability, either a tagged user or an untagged user. Tagged users collect measurements for reference traces, and are therefore not subject to steering by the SON function. This is done in order to avoid incomplete and biased reference trajectories as the Traffic Steerer will influence the behaviour of the users. When a tagged user leaves a cell, either by stopping its call or because it is handed over to another cell, the measurements that were collected by it are turned into a reference trace. This reference trace is then added to the set of reference traces. In order to avoid that this set grows without bounds, a maximum size is imposed on it.

The Traffic Steerer is responsible for making a traffic steering decision once a sufficiently reliable match of the trajectory of a currently active user with a reference trace has been made by the Trajectory Classifier. By assuming that events that occur to one user will also occur to another user that follows a similar trajectory through the cell, the Traffic Steerer can decide whether it is beneficial to hand over the user to another cell or to keep it in the current cell. Events mean new measurements and thus possible changes in signal quality and consequently achievable throughput. Before the Traffic Steerer decides to steer the user to another cell, or not, it will extrapolate and project events that occurred to the reference user in the period after the match on the active user, in order to make an estimate of the future achievable throughput of the active user in each of the neighbouring cells as well as in the serving cell itself. Based on these predictions of the future achievable throughput of the UE in the serving cell as well as in neighbouring cells, a traffic steering decision is made.

The decisions that are taken by the Traffic Steerer are communicated to the handover algorithm that is active in the cell, which is responsible for triggering the actual handovers. Apart from executing the commands coming from the Traffic Steerer, the handover algorithm is also responsible for handing over users for which the SON function did not provide instructions as the SON function will only deal with users for which it sees an opportunity to optimise the handover behaviour.
Chapter 6

Simulation Modelling

6.1 Introduction

To assess that the components of the SON function that is developed in this thesis as well as the SON function as a whole work properly and as intended, simulations are performed. This chapter gives an overview of the simulation scenario that is used in these simulations. As the SON function is based on the assumption that in reality users follow predictable traffic patterns and that the pattern of received signal strengths along the followed trajectories is unique, it is not desirable to use random mobility models and path loss data that is derived from a simple model. Creating a simulation scenario that meets these requirements is however not a trivial task. Therefore, the simulation scenario that will be used to evaluate the proposed SON function is taken from real network data. The simulation scenario was created by Technische Universität Braunschweig (TUBS) [48]. It is a realistic scenario based on data that was collected in the city of Hanover, Germany. The simulation data provides information like base station locations, transmit powers, user movement, isotropic path loss and antenna patterns. The simulation data was collected from various sources like OpenStreetMap [30], mobility simulators like Simulation of Urban MOBility (SUMO) [35], real base station locations.

6.2 Simulation Area

The Hanover scenario covers an area of 20 km by 24 km around the city of Hanover, Germany. A margin with a width of 4 km is not considered when collecting metrics and serves in order to minimise border effects. This means that the effective area from which measurements are collected is 12 km by 16 km. This area is centred around a major city and covers a wide variety of environments ranging from rural
over suburban to urban as is shown in Figure 6.1.

Figure 6.1: The Hanover scenario covers a wide variety of environments ranging from rural over suburban to urban [9].

6.3 Base Stations

The Hanover scenario features both macro and micro base stations. Macro cells have outdoor antennas that are generally placed above rooftops. They have a transmit power ranging from 40 dBm to 46 dBm. There are 195 macro base stations in the Hanover scenario. The base stations transmit in the 1800 MHz frequency band at 46 dBm. The antennas are directional Kathrein 742212 antennas and have a realistic placement and orientation. This means that their placement is dense in the city centre and less dense outside the city centre. Figure 6.2 shows the placement of the macro base stations in the Hanover scenario as well as their orientation. The best server areas of the base stations are indicated using different colours. Within a cell the brightness of the colour indicates the RSRP at the location.
Figure 6.2: The placement of the macro base stations in the Hanover scenario. As can be seen the cells are more dense in the city centre and less dense outside the city centre.
Like macro base stations, micro base stations are also placed outdoor but their antennas are placed below the rooftops, for instance mounted on a lamp post. Their transmit power is lower, ranging from 30 dBm to 40 dBm. There are 28 micro base stations in the Hanover scenario. They are located in a residential and business area in the city centre of Hanover and have directional Kathrein 742212 antennas. The micro base stations in the Hanover scenario transmit at 30 dBm to 33 dBm. Figure 6.3 shows the placement of the micro base stations in the Hanover scenario. The best server areas of the base stations are indicated using different colours and the RSRP is indicated by the brightness of the colour. The location of the area in Figure 6.3 is indicated by the rectangle in Figure 6.2.

Figure 6.3: The placement of the micro base stations in the Hanover scenario.
6.3. BASE STATIONS

6.3.1 Isotropic Path Loss

For each base station a path loss map is available that indicates for each geographical location how much the transmit power of the base station is attenuated due to propagation through space. This path loss map has a resolution of 10 m by 10 m, intermediate values are obtained using linear interpolation. The path loss data was collected from a ray tracing model. This model takes direct path, specular reflections up to the second order, diffuse scattering and diffraction into account. The path loss predictions were based on 2.5D building information (i.e., including the height of buildings) and 3D building models.

6.3.2 Antenna Models

The path loss predictions for the base stations are made for isotropic antennas. This means that they do not take the gain of the antenna into account. Figure 6.4 illustrates the difference between an isotropic antenna and a directional antenna. As can be seen in this figure the isotropic antenna radiates evenly in all directions while in case of the directional antenna most of the transmitted power is directional in a specific direction. The gain of a directional antenna is defined as the ratio of the power radiated by the antenna in that direction divided by the intensity of an isotropic antenna. The function that expresses the gain of an antenna for the direction is called the antenna pattern. Figure 6.5 gives an illustration of an antenna pattern.

![Isotropic and Directional Antennas](image)

Figure 6.4: The difference in path loss when using an isotropic antenna versus a directional antenna.

The actual path loss at a given location can be obtained by adding the
CHAPTER 6. SIMULATION MODELLING

Figures 6.5: An example of an antenna pattern [9].

isotropic path loss at that point and the gain of the antenna (in the logarithmic domain).

6.4 Users

User mobility is an important aspect in the context of this thesis. As the SON function that is developed in this thesis (see Chapter 5) relies on the fact that users behave according to realistic patterns it is not possible to use random mobility models like random walk. Instead a mobility model is required where there is a certain degree of correlation between the places a user visited in the past and the places where it will travel to. Users should make logical decisions about the route they follow and not make strange decisions like all of a sudden travel back to where they came from and so on.

The Hanover scenario features two kinds of users: vehicular users and pedestrian users. Vehicular users are users that move along the major roads at moderate to high velocities depending on the road they are on. Their mobility patterns are generated using the SUMO mobility simulator [35]. This simulator realistically simulates the behaviour of users including lane changes, overtaking by cars, acceleration and deceleration. Pedestrian users move at a lower pace and are situated in the city centre. They also move according to realistic patterns.

As the developed SON function operates on a per-cell level, some simulations require the trajectory that the user follows through a particular cell but not outside the cell. Whenever this is necessary a simple handover algorithm is used to split the mobility traces in separate parts per cell. This handover algorithm configures a single event A3 with a certain hysteresis and TTT. When this event is triggered
first time a handover will be performed and the trace is split at that point. The
hysteresis and \( T_{TTT} \) that are used are respectively 3 dB and 256 ms.

## 6.5 Simulator

The simulations to evaluate the developed SON function were performed using a
custom built simulator. This simulator is written in C++ using the OMNeT++
simulation library [23]. The simulator is a dynamic system level simulator that
simulates users, base stations and their behaviour and interactions. The architecture
of the simulator is shown in Figure 6.6.

![Figure 6.6: The architecture of the simulator.](image)

For each simulated user there is a corresponding component in the simulator. The
UE component is for every individual user responsible for:

- Call setup and tear down
- Generation of traffic
- UE behaviour as described in the 3GPP standards
  - CQI reporting
  - Measurement reporting
  - Transmission and reception of data traffic
Like for every UE, there is also a corresponding component in the simulator for every individual eNB. This component is responsible for:

- Admission control
- Handover
- Scheduling
- eNB behaviour as described in the 3GPP standards

All components that represent users and base stations are connected to the transmission medium (TM). This component is responsible for the transmission of radio messages between UEs and eNBs. Whenever a UE or an eNB sends a radio message, it will calculate the path loss, receive power and SINR and determine whether the message will arrive or not. The transmission medium will also delay the delivery of the messages at their destination according to the transmission delay. The mobility of the users is also implemented in this component. Therefore a mobility model is included in the transmission medium. This mobility model can be changed easily and will, for each user determine its position and in which direction it is travelling and at which speed. This information is in turn used to determine the path loss that is experienced by the users.
Chapter 7

Trajectory Classification

7.1 Introduction

This chapter presents the Trajectory Classifier, one of the three main components of the developed SON function. The Trajectory Classifier is the core component of the SON function. Its purpose is to match traces coming from currently active users to reference traces that were collected from users that were active in the past and finding the reference trace that matches the active trace the best. The Trajectory Classifier will not just match the entire active trace with the entire reference trace. Instead, it will match a trailing part of the active trace (i.e., a part of the active trace starting at some measurement and ending at the last measurement) with an interval somewhere in the reference trace (i.e., starting at some measurement of the reference trace and stopping at some other measurement later in time of the reference trace) as is depicted in Figure 7.1. The rationale behind this is that the most recent past of the active user has the highest influence on its future behaviour.

7.2 Measurements

The key to handing over users to the most suited target cell is being able to identify users that follow similar trajectories through a cell and to distinguish between users that follow different trajectories. In order to do so the Trajectory Classifier bases itself on measurements that are derived from the standardised measurement reports that are sent by the UE to its SeNB as a consequence of a reporting configuration that is configured by the SeNB (see Chapter 4). From these measurement reports the Trajectory Classifier derives the signal strengths experienced by the user of its SeNB and its NeNBs. Each new measurement report results into an update of the
CHAPTER 7. TRAJECTORY CLASSIFICATION

Figure 7.1: The Trajectory Classifier matches a trailing part of the active trace with an interval of the reference trace.

signal strength information of the surrounding eNBs. These updates themselves form a time series which the Trajectory Classifier uses to match active users to reference users.

7.2.1 Measurement Configuration

The measurements that are used by the Trajectory Classifier are derived from measurement reports that are triggered by events A1, A2 and A4. Event A1 monitors when the RSRP of the SeNB rises above a certain threshold. Event A2 monitors when the RSRP of the SeNB drops below a certain threshold. Event A4 monitors when the RSRP of a NeNB rises above a certain threshold. When a UE enters a cell, the Trajectory Classifier configures multiple instances of these events for various thresholds. Note that additional events might be configured by other functions at the SeNB for instance the handover algorithm. The Trajectory Classifier will however only consider measurement reports that were triggered by the events it configured. This is done because the Trajectory Classifier has no control over the measurements it did not configure and taking measurement reports from different events or even events with different hysteresis, TTT or threshold values into account greatly complicates the algorithm. The thresholds that are used range from a preconfigured minimum value to a preconfigured maximum value and lie at regular intervals. As explained in Chapter 4, the UE sends a measurement report when an event is triggered for a certain cell. These measurement reports contain the list of eNBs for which the event is triggered. The events do not necessarily have to lie at regular intervals. Unequally spaced events could be beneficial to trigger more
7.2. MEASUREMENTS

... events in areas of the cell where the signal strength does not change a lot and to reduce the number of triggered events in areas where it does. Unequally spaced events are however not in the scope of this thesis as they would complicate things. The choice for RSRP over RSRQ was made as it is not influenced by factors like the interference in the cell which can change over time. Using the RSRQ would mean that the measurements that are made by two different users at different points in time could be very different even though the users follow the same trajectory.

For each UE that is connected to it the SeNB maintains a list containing an entry for each eNB the UE is aware of, including the SeNB itself. For the SeNB this list contains the set of thresholds of the events A1 that were triggered for it. When a measurement report that was triggered by an event A1 is received, the threshold associated with the event is added to the set. When a measurement report that was triggered by an event A2 is received, the threshold associated with the event is removed from the set if present, otherwise the measurement report is ignored. For each NeNB this list contains the set of thresholds of the events A4 that were triggered for it. When a measurement report that was triggered by an event A4 is received, the threshold associated with the event is added to the set of each NeNB that is included in the measurement report and removed from the set of each NeNB that is not included in the measurement report. After receiving a measurement report from a UE a new measurement is created by constructing a list of all eNBs the UE is aware of and the highest threshold that is associated with it in the corresponding set. The list of thresholds is used instead of the actual RSRP measurements for two reasons. Firstly, only the actual value for the SeNB is included in the measurement report. The values of the NeNBs are not included in the measurement report and thus are not known to the SeNB. Furthermore, as updates are only sent each time a threshold is crossed, the RSRP can have any value within the interval between two consecutive thresholds shortly after a measurement report, meaning that even though the RSRP value of the SeNB is known right after a measurement report is received, it can be quite different some time after receiving the measurement report.

When more than one measurement report arrives within a short time span (1 s) only a single new measurement is produced, encompassing the updates from all measurement reports that arrived in that short time span. The rationale for this is that the order of measurement reports that arrive within a short time is often arbitrary. There are often also a large number of these measurement reports. For instance when a user passes around a street corner the RSRP of one or more eNBs might suddenly spike, causing multiple thresholds to be crossed at once. If all the measurement reports would result in separate measurements, the measurement series of two users following the same path might be quite different due to the arbitrary order in which the measurement reports arrive. This problem is avoided by applying aggregation.

Event driven reporting was chosen over periodic reporting as it avoids unnecessary signalling. It will also reduce the amount of measurements at the SeNB reducing the computational cost to process them. Furthermore it will make it easier to deal with differences in user velocities as users that follow the same trajectory at different velocities will still report the same events. Event driven reporting however...
also has disadvantages in comparison to periodic reporting as it deliverers coarser grained measurements which will result in less accurate matches. Periodic reporting is however out of the scope of this thesis.

7.2.2 Example

Figure 7.2 illustrates the collection of measurements of a user that travels through a cell. All cells in this example are assumed to be omnidirectional and it is assumed that there are four different [RSRP] thresholds. The coloured concentric circles show the thresholds of the different events A1, A2 and A4 of the SeNB (red) and three surrounding NeNBs (green, blue and orange). The circles are numbered from the outside to the inside with the numbers 1 through 4. The path that is followed by the UE is drawn in black. The best server area of the SeNB is indicated by the grey background colour. Places where special events occur are labelled and explained below.

1. The user is handed over to eNB 1 by eNB 2. eNB 1 (the new SeNB) will configure the A1, A2 and events A4 for thresholds 1, 2, 3 and 4 that are required by the Trajectory Classifier and possibly other events that are required by other network functions like the handover algorithm. The events A1 for thresholds 1 and 2 of the SeNB and the events A4 for thresholds 1 and 2 of eNB 2 are triggered a bit later than a [TTT] after the user entered the cell resulting in six measurement reports that are sent to the SeNB. Assuming that the measurement report that is triggered by event A1 for threshold 2 is sent first, followed by the measurement report triggered by event A4 for threshold 1, then the measurement report triggered by event A4 for threshold 2, then the measurement report triggered by event A2 for threshold 3, then the measurement report triggered by event A1 for threshold 1 and finally the measurement report triggered by event A2 for threshold 4, the measurement reports contain the following information:

<table>
<thead>
<tr>
<th>Event</th>
<th>Threshold</th>
<th>eNB</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>A4</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>A4</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>A2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>A1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>A2</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

Assuming the measurement reports arrive at the SeNB within the aggregation time span, the SeNB will aggregate the information in the measurement reports and extract the following information:

<table>
<thead>
<tr>
<th>eNB</th>
<th>1: 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>eNB</td>
<td>2: 2</td>
</tr>
</tbody>
</table>
Figure 7.2: A user (indicated by the black arrow) crosses various thresholds while travelling through a cell.
2. When the user crosses threshold 2 of eNB 2 nothing is reported as measurement reports for NeNBs are only sent when an event A4 is triggered (i.e., when a NeNB becomes better than the threshold, not when it becomes worse). In this case eNB 2 is removed from the list of triggered cells for threshold 2 of event A4 at the UE.

3. When the UE crosses threshold 3 of eNB 1, the corresponding event A1 is triggered (after the TTT expires) and a measurement report is sent to the SeNB containing the following information:

<table>
<thead>
<tr>
<th>Event:</th>
<th>A1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threshold:</td>
<td>3</td>
</tr>
<tr>
<td>eNBs:</td>
<td>1</td>
</tr>
</tbody>
</table>

From the received information, the SeNB updates its own RSRP level to 3 resulting in a new measurement containing the following information:

| eNB 1: | 3 |
| eNB 2: | 2 |

4. As in step 2, no report is sent when the UE crosses threshold 1 of eNB 2. eNB 2 is however removed from the list of triggered cells for threshold 1 of event A4 at the UE.

5. When the UE crosses threshold 1 of eNB 3, the corresponding event A4 is triggered and a measurement report containing the RSRP level of eNB 3 is sent. This measurement report no longer contains a record for eNB 2:

<table>
<thead>
<tr>
<th>Event:</th>
<th>A4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threshold:</td>
<td>1</td>
</tr>
<tr>
<td>eNBs:</td>
<td>3</td>
</tr>
</tbody>
</table>

After processing the information in the measurement report, the SeNB produces the following measurement:

| eNB 1: | 3 |
| eNB 3: | 1 |

6. When the UE crosses threshold 2 of eNB 3, the corresponding event A4 is triggered and a measurement report containing the RSRP level of eNB 3 is sent:

<table>
<thead>
<tr>
<th>Event:</th>
<th>A4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threshold:</td>
<td>2</td>
</tr>
<tr>
<td>eNBs:</td>
<td>3</td>
</tr>
</tbody>
</table>

After processing the information in the measurement report, the SeNB produces the following measurement:
7.2. MEASUREMENTS

As in steps 2 and 4, no report is sent when the UE crosses threshold 2 of eNB 3. eNB 3 is however removed from the list of triggered cells for threshold 2 of event A4 at the UE.

8. The UE crosses threshold 1 of eNB 4 and soon afterwards threshold 3 of eNB 1. This will respectively trigger the corresponding events A4 and A2 containing the following information:

| Event: A4 | Threshold: 1 | eNBs: 3, 4 |
| Event: A2 | Threshold: 3 | eNB: 1 |

Cell 3 is also in this list as event A4 for threshold 1 is also triggered for it. As both measurement reports are received by the eNB within the aggregation period only a single measurement is extracted from both measurement reports:

| eNB 1: 2 |
| eNB 3: 1 |
| eNB 4: 1 |

9. As in steps 2 and 4 no report is sent when the UE crosses threshold 1 of eNB 3. eNB 3 is however removed from the list of triggered cells for threshold 1 of event A4 at the UE.

10. When the UE crosses threshold 2 of eNB 4, the corresponding event A4 is triggered and a measurement report containing the RSRP level of eNB 4 is sent:

| Event: A4 | Threshold: 2 | eNBs: 4 |

The SeNB extracts the following information from this measurement report, resulting in a new measurement:

| eNB 1: 2 |
| eNB 3: 1 |
| eNB 4: 2 |

11. The UE is handed over from eNB 1 to eNB 4 by the handover algorithm.

The resulting measurement trace thus is:
7.2.3 Inter-RAT Measurements

For cells of other RATs, LTE provides event B1 which is very similar to event A4. It is triggered when the measurement result of an inter-RAT neighbour rises above a certain fixed threshold. As with event A4, a measurement report is sent whenever the event is triggered for a particular cell. This measurement report contains all triggered cells of the RAT to which the cell belongs. Events B1 can be used in exactly the same way as events A4 are used. This means that the Trajectory Classifier can also use inter-RAT measurements for determining that users follow the same trajectory or not. Extending the SON function to multiple RATs is discussed in Chapter 10.

7.3 Trajectory Matching

Based on the measurements that are collected from users, using the mechanism described in Section 7.2, the Trajectory Classifier matches measurement traces from currently active users with measurement traces from users that were active in the past. Measurements that are sent by users that follow the same trajectory can slightly differ due to slight deviations in the trajectory that is followed, different user velocities, time variations in fading, etc. Because of this, it is not possible to just compare the measurements of both the active and reference users by looking for, for instance, the longest common sub-string. In order to make this matching more resilient against slight variations in the measurement data, a modified version of the Dynamic Time Warping (DTW) algorithm is used. The DTW algorithm was chosen as it exhibits many of the properties that are required for the goal of this thesis. Firstly its purpose is to compare time series which is exactly what is obtained from the measurement reports. Secondly the two series do not need to be exactly the same. The DTW algorithm is able to match series where certain measurements differ slightly. This is a big advantage as due to the nature of the measurements there is no guarantee that two users that follow the same trajectory return exactly the same measurements. Finally the DTW algorithm is able to deal with differences in the times between measurements. This is a crucial property of the algorithm as two users will never have the exact same velocity.
7.3. TRAJECTORY MATCHING

7.3.1 Classical Dynamic Time Warping

The DTW algorithm [54] is used in signal processing to find an optimal alignment between two time series (like the measurements coming from the users). Dynamic Time Warping determines the distance between two time series \( X = (x_1, \ldots, x_M) \) and \( Y = (y_1, \ldots, y_N) \) by determining the so called optimal warping path through the cost matrix \( C \in \mathbb{R}^{M \times N} \) whose elements \( C_{m,n} \) express the distance \( c(x_m, y_n) \) between the elements \( x_m \) and \( y_n \) of the respective series. A warping path through this cost matrix is a series \( p = (p_1, \ldots, p_L) \) with \( p_1 = (1, 1) \), \( p_L = (M, N) \) and \( p_l = (m_l, n_l) \) for \( l \in \{2, \ldots, L\} \) can only be reached from \( p_{l-1} \in \{(m_{l-1}, n_l), (m_l, n_{l-1}), (m_l-1, n_l-1)\} \), i.e., a warping path is a path through the cost matrix that starts at \((1, 1)\) and goes to \((M, N)\) by either incrementing the row index, incrementing the column index or incrementing both at the same time. The cost or distance of a warping path is given by the total cost of the elements along the path:

\[
\sum_{l=1}^{L} c(x_{m_l}, y_{n_l}) \tag{7.1}
\]

The optimal warping path is the warping path that has the lowest total cost of all possible warping paths. The optimal warping path can be efficiently determined by constructing an \( M \times N \) distance matrix \( D \) in which each element \((m, n)\) contains the minimal total cost to match the leading part of length \( m \) of \( X \) with the leading part of length \( n \) of \( Y \). Each element \((m, n)\) of this matrix (except from the ones on the first row and first column) is calculated by adding the cost \( c(x_m, y_n) \) to the minimum of \( D[m-1, n], D[m, n-1] \) and \( D[m-1, n-1] \). The values of the elements of the first row and column are calculated by adding the cost to the value of the elements to the left or above respectively. Pseudocode for the dynamic time warping algorithm is given in Algorithm 1. In this code, an additional row and column are added to matrix \( D \). The elements in the first row and column of this matrix are set to infinity, except for the element in the upper left corner which is set to 0. By doing this, the elements in the remainder of the extended matrix can all be calculated in the same way.

Step-by-step Example

This section gives an example of how the classical Dynamic Time Warping algorithm works. The measurement series in this example are strings of letters. The distance function \( c(x_m, y_n) \) between two elements of the series (i.e., letters) is defined as follows: if two letters are the same and have the same casing, the distance is 0. If they are the same but have a different casing the distance is 0.25. If they are different but have the same casing the distance is 0.75 and if they are different and have a different casing the distance is 1. An example of the distances between two letters is given Table 7.1.

The distance between the strings ‘aB’ and ‘abc’ will be computed. The cost matrix
Algorithm 1 The classical Dynamic Time Warping algorithm.

1: $D := \text{array}[0..M, 0..N]$
2: $D[0, 0] := 0$
3: for $m := 1 \rightarrow M$ do
4:   $D[m, 0] := \infty$
5: end for
6: for $n := 1 \rightarrow N$ do
7:   $D[0, n] := \infty$
8: end for
9: for $m := 1 \rightarrow M$ do
10:   for $n := 1 \rightarrow N$ do
11:     $D[m, n] := c(X[m], Y[n]) + \min(D[m-1, n], D[m, n-1], D[m-1, n-1])$
12:   end for
13: end for

Table 7.1: Example of the distances between letters.

<table>
<thead>
<tr>
<th>First element</th>
<th>Second element</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>a</td>
<td>0</td>
</tr>
<tr>
<td>a</td>
<td>A</td>
<td>0.25</td>
</tr>
<tr>
<td>a</td>
<td>b</td>
<td>0.75</td>
</tr>
<tr>
<td>a</td>
<td>B</td>
<td>1</td>
</tr>
</tbody>
</table>

of these two strings is shown in Figure 7.3.

a b c

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>0.75</th>
<th>0.75</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>0</td>
<td>0.75</td>
<td>0.75</td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>0.25</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 7.3: The cost matrix.

The extended empty distance matrix is shown in Figure 7.4.

The matrix is filled from the upper left corner to the lower right corner. It does not matter if the the rows are filled first or the columns, in this example the rows are filled first. The value of the upper left element is given by $0 + \min(\infty, \infty, 0) = 0$:
The value of the second element of the first row is computed based on this newly computed value as $0.75 + \min(\infty, 0, \infty) = 0.75$: 

The third value of the first row is computed likewise:
### Chapter 7. Trajectory Classification

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>∞</td>
<td>∞</td>
<td>∞</td>
</tr>
<tr>
<td>a</td>
<td>∞</td>
<td>0</td>
<td>0.75</td>
</tr>
<tr>
<td>B</td>
<td>∞</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

The first value of the second row is calculated based on the value of the $1 + \min(0, \infty, \infty) = 1$:

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>∞</td>
<td>∞</td>
<td>∞</td>
</tr>
<tr>
<td>a</td>
<td>∞</td>
<td>0</td>
<td>0.75</td>
</tr>
<tr>
<td>B</td>
<td>∞</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

The second element of the second row is the first element whose value is based on three previous computed values: $0.25 + \min(0.75, 1, 0) = 0.25$:

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>∞</td>
<td>∞</td>
<td>∞</td>
</tr>
<tr>
<td>a</td>
<td>∞</td>
<td>0</td>
<td>0.75</td>
</tr>
<tr>
<td>B</td>
<td>∞</td>
<td>1</td>
<td>0.25</td>
</tr>
</tbody>
</table>

The value in the lower right corner is calculated likewise:
The value in the lower right corner of the matrix, i.e., 1.25, is the minimum warping distance of the strings ‘aB’ and ‘abc’. The corresponding warping path is:

Below is a more elaborate example is given for the strings ‘aBcCbabbC’ and ‘AB-CACaBbbcC’. The cost matrix, expressing the distances between the individual characters for these strings is:

```
    a   b   c
  a  ∞  0.75  1.5
  B ∞  1  0.25  1.25
```
The constructed distance matrix, containing the values to match two leading parts up to a certain point becomes:
Yielding a minimum warping distance of 2.5. The optimal warping path is indicated using arrows.

### 7.3.2 Modified Dynamic Time Warping

The major shortcoming of the classical DTW algorithm for the purpose of the Trajectory Classifier is that it matches two time series entirely. This is however not desirable. First, the more recent past of the active user is more interesting as its behaviour in the future will be influenced more by it than by its earlier behaviour. Furthermore, it should be possible to match an active user with a reference user without having to match the latest measurement of the active user with the last reference measurement as this usually is when the reference user left the cell and it should be possible to proactively make decisions before the active user also leaves the cell. In order to overcome this problem the DTW algorithm was adapted such that it is able to match any trailing part of the measurement series of the active user with any interval of the measurements of the reference user. This is one of the major contributions of this thesis.

The DTW algorithm is adapted as follows: the matrix $D$ that is used to calculate the intermediate distances is filled backwards, i.e., the DTW algorithm is applied to the reverse series. By doing this each element of the matrix contains the minimal total
cost, when only matching the trailing parts of both measurement series up to that point. During construction of the matrix, a current best match is kept and updated each time the value of an element is calculated. By doing this, by the time the upper left corner of the matrix (the element that corresponds to the first elements in both series) is reached the best matching trailing parts of the measurement series of the active and reference user have been found.

In order to match a trailing part of the measurement series of the active user with any interval of the measurements of the reference user, the end of the measurement series of the active user is shifted along the elements of the measurement series of the reference user. The MDTW algorithm is applied to the entire active series and the sub-series of the reference user starting from the first measurement up to the measurement that is aligned with the end of the measurement series of the active user.

As the MDTW algorithm is able to find matches of sub-series of different lengths, it is important that the length of the warping path is taken into account when determining the optimal warping path. If the length of the warping path would not be taken into account when comparing the costs of two warping paths and the costs would just be compared as with the original DTW algorithm, short matches would be greatly favoured over long matches even if the long matches are accurate. Long matches are however favoured over short matches as they are a much better indication for an actual coincidence of the user’s trajectories than short matches. In order to deal with warping paths of unequal length, distances between measurements and costs along warping paths will be represented using tuples $\langle v, w \rangle$. The component $v$ represents the value of the cost and is a value between 0 and 1. When $v$ has value 0 this means that there is an exact match between two measurements. When $v$ has value 1 this means that two measurements are completely different. The component $w$ represents the importance or weight of the value. The exact meaning of the importance depends on how the cost between measurements is defined (see also Section 7.3.3), but as a general rule the weight should be higher when the cost is based on more information, for instance a longer warping path. Two operations are needed by the dynamic time warping algorithm: the addition of costs and the comparison of costs. The addition of two costs is defined as in Equation 7.2.

$$\langle v_1, w_1 \rangle + \langle v_2, w_2 \rangle = \left( \frac{v_1 w_1 + v_2 w_2}{w_1 + w_2}, w_1 + w_2 \right)$$  \hspace{1cm} (7.2)

As can be seen from Equation 7.2, the value of the sum is equal to the weighed average of both addends while the weight of the sum is equal to the sum of the weights. Two costs are compared by ignoring the weights and just comparing their values as is shown in Equation 7.3.

$$\langle v_1, w_1 \rangle < \langle v_2, w_2 \rangle \iff v_1 < v_2$$  \hspace{1cm} (7.3)

Due to the way that addition is implemented, tuples will be compared based on the average cost of the elements along the warping path. Other ways for taking the length of the match into account have been considered. For instance, as longer matches are to be preferred, the cost along the warping path could be scaled down
more for long matches than for short matches. In order to test which way for scaling the cost along a warping path would yield the best result, various functions of the length $L$ of the warping path through which the cost along that warping path is divided were tested. The following functions have been considered for rescaling the cost along the warping path:

- $1$
- $\log L$
- $L$
- $L \log L$
- $L^2$
- $\sqrt{L}$
- $3\sqrt{L}$

Simulations were performed to assess which way to scale the cost along the warping path. These simulations however showed that only using $L$ yielded viable results. Using 1, $\log L$, $\sqrt{L}$ or $3\sqrt{L}$ (i.e. values that are lower than $L$) resulted in very short matches while using $L \log L$ or $L^2$ (i.e. values that are higher than $L$) resulted in matches that included almost all measurements of both the active and the reference trace. Therefore it was concluded that the best way to scale the cost along the warping path was $L$ yielding Equation 7.2 and Equation 7.3.

Scaling the cost along the warping path by the length of the warping path $L$ still does not always yield the desired result: shorter trailing parts will still be favoured over longer trailing parts as it is more likely to find a short perfect match than it is to find a longer one. In order to mitigate this problem, the initial cost $I$ is set to $(1, x)$ with $x \geq 0$. Since after a series of additions, the value component of the sum is the weighted mean of all added values, the importance of this initial cost will become less important as more values are added to it. By using a tuple with non-zero weight as the initial cost, longer matches will be favoured over shorter matches. Note that adding $(v, w)$ to the initial cost $(1, 0)$ will result in $(v, w)$, removing the influence of the initial cost altogether.

Another problem that arises when allowing the algorithm to match every combination of trailing parts is that the cost matrix can be traversed mainly horizontally or vertically, as this produces warping paths of the same length as going diagonally. Going mainly vertically or horizontally is however less favourable as this will match shorter portions of either one of the compared traces for the same length of warping path. This is not a problem with the classical DTW algorithm as it matches the entire series. In order to mitigate this problem an additional cost can be added when traversing the cost matrix in the vertical or horizontal directions, but not when going diagonally. This will ‘encourage’ the algorithm to match longer portions of both series. Like the initial cost, this additional cost, which is called the
extra cost $E$, will have the form $(1, x)$ with $x \geq 0$. Pseudocode for this is given in Equation (7.4)

\[ D[i, j] := c(X[i], Y[j]) + \min(D[i - 1, j] + E, \quad \]  \[ D[i, j - 1] + E, \quad \]  \[ D[i - 1, j - 1]) \quad (7.4) \]

The pseudocode for the MDTW algorithm is given in Algorithm 2. Note from this code that the MDTW algorithm will always return a match, together with the cost of that match. In Section 7.4.3 criteria for determining whether a match is accurate will be established.

Algorithm 2 The modified Dynamic Time Warping algorithm.

1: $m := \text{length}(X)$
2: $n := \text{length}(Y)$
3: bestMatch := None
4: bestDistance := $\infty$
5: for start := 1 → $m$ do
6:   $D := \text{array}[1..\text{start} + 1, 1..n + 1]$
7:   for $i := 1 → \text{start}$ do
8:     $D[i, n + 1] := \infty$
9:   end for
10:   for $j := 1 → n$ do
11:     $D[\text{start} + 1, j] := \infty$
12:   end for
13:   $D[\text{start} + 1, n + 1] := I$
14: for $i := \text{start} → 1$ do
15:   for $j := n → 1$ do
16:     $D[i, j] := c(X[i], Y[j]) + \min($
17:       $D[i + 1, j] + E,$
18:       $D[i, j + 1] + E,$
19:       $D[i + 1, j + 1])$
20:     if $D[i, j] < \text{bestDistance}$ then
21:       bestMatch := (start, $i, j$)
22:       bestDistance := $D[i, j]$
23:     end if
24:   end for
25: end for
26: end for

7.3.3 Measurement Distance

As the DTW algorithm (and thus the MDTW algorithm) requires a distance measure to be defined between two measurements, one is needed for the measurements
that are derived from the measurement reports. This distance measure should yield a low value (close to 0) when the measurements are similar and a high value (close to 1) when the measurements are different. Measurements are essentially a list of base stations and their associated RSRP level. This RSRP level is a number between 1 and the number of RSRP levels $N$. Not all base stations that appear in one measurement should appear in the other measurement.

If $X$ is the set of base stations in the first measurement and $m_i$ is the RSRP level corresponding to base station $i \in X$; and $Y$ is the set of base stations in the second measurement and $n_j$ is the RSRP level corresponding to base station $j \in Y$ then the distance between the first and second measurements is defined as:

$$c(X, Y) = \frac{\sum_{i \in X \cap Y} |m_i - n_i| + \sum_{i \in X \setminus Y} m_i + \sum_{i \in Y \setminus X} n_i}{N (|X| + |Y| - |X \cap Y|)}$$

(7.5)

That is for all base stations that appear in both measurements the contribution to the cost is the absolute value of the difference of the indexes of the RSRP levels. For all base stations that appear in only one of the lists the contribution to the cost is the the index of the RSRP level itself. The sum of these values is then rescaled by dividing by the maximum number of RSRP levels and the number of terms in the numerator to make sure that the result is a value between 0 and 1. The cost function between two measurements should reflect how similar two measurements are (with 0 being entirely equal and 1 being as different as possible). As the indexes of the RSRP levels have the same ordering as the RSRP levels themselves (i.e., 1 corresponds to the lowest RSRP level while $N$ corresponds to the highest RSRP level) subtracting these RSRP level indexes and dividing the absolute value of this difference by the number of RSRP levels $N$ will yield 0 if both RSRP levels are the same and $1 - \frac{1}{N}$ (which is nearly equal to 1) if one RSRP level is the highest one possible and the other is the lowest one possible. If only one measurement contains an RSRP level for a certain eNB the ratio between the index of that RSRP level and the number of RSRP level is used. This will yield a value between $\frac{1}{N}$ and 1. The rationale behind this is that when an eNB is not included in a list, its RSRP is assumed to be below the lowest RSRP level. In this cases, the higher the corresponding RSRP level of the other measurement is, the further the measurements are apart which is reflected by the outcome. As the RSRP decreases exponentially with distance, using a distance functions in the logarithmic domain is a sensible choice as equal differences will roughly correspond to equal geographical distances.

For example, consider the measurements $X$ and $Y$ in Figure 7.5 in a network where there are 10 different RSRP levels. These measurements have 2 eNBs in common, namely 1 and 2. Furthermore measurement $X$ has eNB 3 and measurement $Y$ has eNBs 4 and 5. The distance $c(X, Y)$ that is used in Algorithm 2 is given by Equation 7.6.

$$c(X, Y) = \frac{\sum_{i \in X \cap Y} |m_i - n_i| + \sum_{i \in X \setminus Y} m_i + \sum_{i \in Y \setminus X} n_i}{N (|X| + |Y| - |X \cap Y|)} = 0.2$$

(7.6)
7.4 Evaluation

In order to demonstrate that the Trajectory Classifier functions as is intended, simulations were performed. As in these simulations the locations of the users are known, it is possible to assess whether the results of the Trajectory Classifier are accurate or not. This is something that cannot be done in real-life as then the actual locations are not available.

The goal of the simulations is threefold:

1. Firstly, the influence of the cost parameters of the MDTW algorithm, namely the initial cost and the extra cost, on the performance of the MDTW algorithm under various circumstances has to be determined. Furthermore suitable values for these parameters need to be established.

2. Secondly, as the MDTW algorithm will always produce a match, suitable criteria to distinguish between accurate and inaccurate matches need to be established. It should be possible to check these criteria given the information that is available in real-life.

3. Finally, the ability of the Trajectory Classifier to identify users that follow similar trajectories and to distinguish between users that follow different trajectories has to be demonstrated.

This section will first explain how the simulation results were obtained. Next, the simulation results will be presented and discussed.

7.4.1 Methodology

This section explains how the simulation results in this chapter are obtained.

Per cell random combinations of two traces coming from users within the cell are made. These traces are obtained as described in Section 6.4. One of the traces of each pair is designated as the reference trace and the other as the active trace. Figure 7.6 illustrates such a pair of traces. This figure indicates the entire reference and active trajectories through the cell, with the start and end locations indicated.
7.4. EVALUATION

Reference Trajectory
Active Trajectory
Reference Measurement Reports +
Active Measurement Reports ×
Reference Trajectory Start Point +
Active Trajectory Start Point ×
Reference Trajectory End Point +
Active Trajectory End Point ×
Reference Coincidence Start Point +
Active Coincidence Start Point ×
Reference Coincidence End Point +
Active Coincidence End Point ×

Figure 7.6: A pair of traces with the relevant events indicated.

using blue and yellow markers respectively. Both trajectories coincide for a certain part, the start and end locations of the coincidence are indicated using red and black markers respectively. The locations where a measurement report is received are marked using small markers that have the same colour as the reference and active traces.

From the RSRP measurements, measurement reports are derived for both the reference trace and the active trace. These measurement reports are sent as a consequence of the events A1, A2 and A4 that are configured by the Trajectory Classifier. The series of measurements are in turn derived from the measurement reports.

Next, the MDTW algorithm is applied to all possible leading parts of the active trace, i.e., all subsequences of the active measurement trace, starting from the beginning and ranging up to a certain measurement. This corresponds to how the SON function will operate in real-life: each time a new measurement report is received and a new measurement is added to the series of active measurements, the MDTW is applied to the entire series of reference measurements and the series of active measurements that has been obtained so far. For each leading part of the active trace, the MDTW algorithm will yield a match. This match will indicate the start and end measurements of the matched part of the reference trace and the start of the matched part of the active trace, as well as the total cost of the match (see Section 7.3.2 for more information). Note that the matched part of the active trace ends with the last measurement that was passed to the MDTW algorithm.

For each pair of traces, functions \( \tilde{f}(\tilde{t}) = (\tilde{x}, \tilde{y}) \) and \( \hat{f}(\hat{t}) = (\hat{x}, \hat{y}) \) exist that respect-
ibly map a time $\tilde{t}$ on the coordinates $(\tilde{x}, \tilde{y})$ of the reference user at that time and a time $\hat{t}$ on the coordinates $(\hat{x}, \hat{y})$ of the active user at that time. The domains of these functions are the intervals of time from when to when respectively the reference and active users were active. For the reference user this is $[\tilde{t}_0; \tilde{t}_e]$ and for the active user this is $[\hat{t}_0; \hat{t}_e]$. The image of these functions is the set of locations that are visited by the respective users. Trajectories where the user visits the same location more than once are not considered so $\tilde{f}$ and $\hat{f}$ are bijections. This means that the inverse functions $\tilde{f}^{-1}$ and $\hat{f}^{-1}$ exist that map the locations of respectively the reference user and the active user on the time at which that the user was at that location. When a reference trace and an active trace overlap geographically from time $\tilde{t}_a$ to time $\tilde{t}_b$ in the time frame of the reference user and from time $\hat{t}_a$ to time $\hat{t}_b$ in the time frame of the active user, it is possible to map a time $\tilde{t}$ in the time frame of the reference user to a time $\hat{t}$ in the time frame of the active user by using Equation 7.7

$$\hat{t} = \hat{f}^{-1}(\tilde{f}(\tilde{t}))$$

(7.7)

Vice-versa, it is possible to map a time $\hat{t}$ in the time frame of the active user to a time $\tilde{t}$ in the time frame of the reference user using Equation 7.8

$$\tilde{t} = \tilde{f}^{-1}(\hat{f}(\hat{t}))$$

(7.8)

In Figure 7.6 $\tilde{t}_0$ corresponds to the blue plus, $\tilde{t}_e$ to the yellow plus, $\tilde{t}_a$ to the red plus, $\tilde{t}_b$ to the black plus, $\hat{t}_0$ to the blue cross and $\hat{t}_e$ to the yellow cross, $\hat{t}_a$ to the red cross and $\hat{t}_b$ to the black cross. Note that although the locations of $\tilde{t}_a$ and $\hat{t}_a$; and $\tilde{t}_b$ and $\hat{t}_b$ are the same, the times at which they are visited by respectively the reference and active users is not.

### 7.4.2 Accuracy Metric

In order to assess whether the Trajectory Classifier is able to find accurate matches, an accuracy metric is required. As the geographical locations of the users are known when doing simulations (this is not the case in real-life), it is possible to identify at which points in time the trajectories of the active and reference users coincide and at which points in time they differ and in case they coincide which geographical locations along the reference trace correspond to which geographical locations along the active trace. Using this information the accuracy of the matches that are made by the MDTW algorithm can be assessed. For each match that is produced by the MDTW algorithm it can be determined whether during the time between the start and the end of the match the paths of the reference user and the active user coincide geographically and hence whether the match should be considered accurate. Note that the MDTW algorithm always returns a match. This can be done by assessing how well the match made by the MDTW algorithm corresponds to the coincidence of the geographical paths. Figure 7.7 illustrates the accuracy metric that is used. In this figure, $s_{MDTW}$ represents the time that corresponds to the first measurement (start) of the match that was identified by the MDTW algorithm, $e_{MDTW}$ represents the time that corresponds to the last measurement (end) of the match. Furthermore,
Figure 7.7: The accuracy metric is calculated by dividing the duration of the overlapping period by the duration of the union.

$s_G$ represents the time that corresponds to the start of the geographical coincidence, $e_G$ represents the time that corresponds to the end of the geographical coincidence. The accuracy of a match is defined by Equation 7.9.

$$\frac{\max(\min(e_{\text{MDTW}},e_G) - \max(s_{\text{MDTW}},s_G), 0)}{\max(e_{\text{MDTW}},e_G) - \min(s_{\text{MDTW}},s_G)}$$  

(7.9)

This accuracy can be evaluated for either the reference user or the active user. In both cases $s_G$ is the start time of the geographical coincidence of the active trace with the reference trace which is $\tilde{t}_a$ for the reference trace and $\hat{t}_a$ for the active trace. When evaluating it for the active user, $s_{\text{MDTW}}$ and $e_{\text{MDTW}}$ are the times that correspond to respectively the first and last matched active measurement. $e_G$ is taken as the minimum of the time of the last active measurement and the end time of the geographical coincidence of the active trace with the reference trace. If the last active measurement was received at time $\hat{t}_c$, $e_G$ is given by Equation 7.10.

$$e_G = \min(\hat{t}_c, \hat{t}_b)$$  

(7.10)

The minimum of these two times is used as the MDTW algorithm cannot find matches for the part of the active trace for which it has no data yet. When evaluating it for the reference user, $s_{\text{MDTW}}$ and $e_{\text{MDTW}}$ are the times that correspond to respectively the first and last matched reference measurement. $e_G$ corresponds to the minimum of the time when the reference user was at the same position along the trajectory as the active user at the time of the last active measurement in the
CHAPTER 7. TRAJECTORY CLASSIFICATION

matched active measurement trace and the end time of the geographical coincidence of the active trace with the reference trace as is shown in Equation 7.11.

\[
e_G = \min \left( \tilde{f}^{-1} \left( \hat{f} \left( \hat{t}_c \right) \right), \tilde{t}_b \right)
\]

(7.11)

The position of the last active measurement is mapped on the reference user as the Trajectory Classifier cannot find matches for the part of the reference trace after this measurement. The minimum of these two times is used for the same reason as when matching the active user. Note that when the position of the last active measurement cannot be mapped on the reference user because at that point the geographical locations no longer coincide the minimum will return the end time of the geographical coincidence anyway.

Based on the accuracy metrics, matches are deemed accurate or inaccurate. Matches where the accuracy metric is greater than or equal to 0.8 will be considered to be accurate. Matches where the accuracy metric is less than 0.8 or where there is no geographical coincidence altogether will be considered to be inaccurate. The cutoff value of 0.8 was chosen as it requires matches to be fairly accurate but at the same time allows for some inaccuracies due to the granularity of the measurements.

7.4.3 Reliability Criteria

In this section criteria that will be used by the Trajectory Classifier for determining whether a match is accurate or not will be established. When a new measurement is added to an active trace after receiving one or more measurement reports, the active trace is matched to each reference trace in the set of reference traces using the MDTW algorithm. From all these matches the one with the lowest cost is chosen. This match is however not necessarily an accurate match as the MDTW algorithm always produces a match. Therefore criteria are needed that allow the Trajectory Classifier to discriminate between accurate and inaccurate matches such that the SON function can decide whether it will rely on the match or not for making a steering decision. It should be possible to evaluate these criteria for each result that is yielded by the MDTW algorithm.

An obvious choice for discriminating between accurate and inaccurate matches is the length of the matched parts of either the reference or the active trace. The MDTW algorithm will match a subsequence of the series of reference measurements with a subsequence (which is always a trailing part) of the series of active measurements. When two traces do not match very well, it will be easier to find short subsequences that match than long ones. Therefore, the matched subsequences will typically be short when two traces do not match very well. When two traces do match well, the matched subsequences will typically be longer as longer matches will reduce the influence of the initial cost.

When the length of a match is low it is likely to be inaccurate. When the length of a match is high it is likely to be accurate. Figure 7.8 illustrates this principle. The
length of the match (either the length of the reference subsequence or that of the active subsequence) is plotted on the horizontal axis while the number of accurate or inaccurate matches is plotted on the vertical axis. The two curves represent the number of accurate and inaccurate matches. The goal is to define a threshold for the length of matched part of either the reference trace or the active trace. A match whose length is shorter than this threshold will be considered inaccurate by the Trajectory Classifier. A match whose length is greater than this threshold will be considered to be accurate by the Trajectory Classifier. The shaded area in Figure 7.8 indicates the number of inaccurate matches. In this figure the threshold is indicated by the orange dashed line. As the match will later on be used to predict the future behaviour of the user, this threshold should be chosen carefully. When the threshold is set too low, too many inaccurate matches are considered to be good matches, resulting in bad predictions. When the threshold is set too high, too many accurate matches are considered to be bad matches, resulting in fewer users being steered than are possible. In Section 7.4.4 the optimal value for the threshold will be determined.

Figure 7.8: The threshold should be chosen such that a minimum of matches are considered accurate or inaccurate erroneously.

### 7.4.4 Optimal Cost Parameters and Length Threshold

Using the accuracy metric that was defined in Section 7.4.2 optimal values for the cost parameters as well as the length threshold by which the Trajectory Classifier decides whether a match is accurate or not will be determined. For this all combin-
ations of an extensive number of initial costs and extra costs are tested. The values
of both the initial and extra costs are chosen from the geometric series \( x_i = \left( \frac{4}{3} \right)^i \) for \( i \in \{-31, \ldots, 6\} \). By using a geometric series the values of the cost parameters
range from very small to very large. For each combination of initial cost and extra
cost the optimal length threshold \( l \) is determined. The optimal length threshold
is the threshold \( l \) for which the ratio of the number of correctly assessed matches
and the total amount of matches (called the correctness) is maximal. By optimiz-
ing the number of matches that are correctly deemed inaccurate and the number
of matches that are correctly deemed accurate a balance is struck between identi-
fying sufficiently many accurate matches while not considering too many accurate
matches as inaccurate.

Figure 7.9 shows the correctness for the different combinations of initial and extra
costs. The horizontal axes of the graphs express the exponent \( m \) of the initial cost
\( \left( \frac{4}{3} \right)^m \). The vertical axes express the exponent \( n \) of the extra cost \( \left( \frac{4}{3} \right)^n \). The top
graph shows the correctness for the optimal length threshold which is shown in the
bottom graph. High correctness values in the top graph indicate suitable parameter
values, low performance values indicate unsuitable parameter values. As can be
seen, suitable parameter combinations are those where the extra cost is rather high
\( \geq \left( \frac{4}{3} \right)^{-4} = 0.32 \) and the initial cost is not too high but also not very close to
zero. When these criteria are satisfied, the exact parameter values are irrelevant.
It is however wise not to choose values that are too close to a bad combination of
parameter values. In the remainder of this thesis a value of 0.05 \( \approx \left( \frac{4}{3} \right)^{-10.4} \) will
be used for the initial cost and a value of 0.02 \( \approx \left( \frac{4}{3} \right)^{-13.6} \) will be used for the
extra cost.

When the initial cost is very close to zero the MDTW algorithm produces inaccurate
matches as can be seen in the top graph. These matches often only include single
measurements. This is because when the initial cost is very close to zero, adding a
second cost \( c_i \) that is only slightly higher than the average cost of the costs that
are already included will not decrease the average cost along the warping path causing
the MDTW algorithm to stop. Suppose the MDTW algorithm considers adding an
additional cost \( c_{n+1} \) to an existing warping path \( c_1 \ldots c_n \), the cost of the warping
path is given by Equation 7.12. Note that this value is the first element of the tuple
that is obtained from the MDTW algorithm.

\[
\frac{I + \sum_{i=1}^{N} c_i + c_{N+1}}{I + N + 1}
\]  
(7.12)

In this equation \( I \) represents the initial cost. The cost of the warping path before
adding \( c_{N+1} \) is given by Equation 7.13

\[
\frac{I + \sum_{i=1}^{N} c_i}{I + N}
\]  
(7.13)

In order for the MDTW algorithm to include measurement \( c_{N+1} \) in the warping
path, the result of Equation 7.12 should be less than that of Equation 7.13. In this
Figure 7.9: The correctness and associated optimal length threshold for each combination of initial cost and extra cost.
Equation (7.16) \( \langle c_i \rangle \) denotes the average cost of the warping path of length \( N \).

\[
I + \sum_{i=1}^{N} c_i + c_{N+1} \over I + N + 1 < I + \sum_{i=1}^{N} c_i \over I + N
\]  \quad (7.14)

\[
Nc_{N+1} - \sum_{i=1}^{N} c_i \over 1 - c_{N+1} < I \quad (7.15)
\]

\[
Nc_{N+1} - \langle c_i \rangle \over 1 - c_{N+1} < I \quad (7.16)
\]

When a match should be made, the \( c_i \)'s are close to zero meaning that the denominator of Equation (7.16) is close to 1 and can be ignored. This means that in order for the MDTW algorithm to consider preferring the warping path that includes \( c_{N+1} \) over the warping path that does not, \( I \) has to be greater than \( Nc_{N+1} - \sum_{i=1}^{N} c_i \), i.e., the difference between \( c_{N+1} \) and the average cost along the warping path of length \( N \) should be less than \( \frac{I}{N} \). As this becomes less probable for longer matches, the MDTW algorithm will likely prefer short matches when \( I \) is small.

While the initial cost should not be too low, it should also not be too high. As can be seen in the top graph of Figure 7.9, when the initial cost exceeds \( \left( \frac{4}{3} \right)^{-5} = 0.24 \), the produced matches become inaccurate regardless of the value of the extra cost. This is because the initial cost becomes too high for the length of the matched parts of the traces and the MDTW algorithm starts including measurements that do not match. Suppose \( c_{N+1} \) is the cost that corresponds to two measurements that do not match while \( c_1, \ldots, c_N \) are. In this case \( c_{N+1} \) will be significantly higher than \( c_1, \ldots, c_N \) which lie close to zero. This means that in Equation (7.16) \( \langle c_i \rangle \) will become negligible while the denominator can no longer be ignored. In practice \( c_{N+1} \) will still be rather small (around 0.1–0.2) so the denominator will not be close to zero. Figure 7.10 shows the function \( \frac{c_{N+1}}{1 - c_{N+1}} \), which is the factor that relates \( N \) with the lowest \( I \) that will cause \( c_{N+1} \) to be added to the warping path for varying \( c_{N+1} \). It can be seen that for a sufficiently high initial cost \( I \), Equation (7.16) will hold and the cost \( c_{N+1} \) will be added to the warping path despite both measurements not actually matching which will result in bad matches.

Furthermore there is a clear diagonal line running from the lower left corner to the upper right corner in the top graph of Figure 7.9. This line does however not lie where the initial cost is equal to the extra cost but slightly biased towards a higher initial cost relative to the extra cost. A parameter combination below the line results in slightly worse results than a parameter combination above the line. This is because in the cases below the line, an extra horizontal or vertical step is added to the beginning or end of the warping path by going either down or right as is illustrated in Figure 7.11. Assuming that the costs of the extra step and the other steps are all equal along the warping path, the extra step is added to the warping
7.4. EVALUATION

path if:

\[
\frac{I + a(N + 1) + E}{I + E + N + 1} < \frac{I + aN}{I + N} \quad \text{(7.17)}
\]

\[\iff EN < I \quad \text{(7.18)}\]

with \(a\) being the cost of a step and \(E\) the extra cost. In other words, the extra step is added to the warping path when the initial cost is greater than the extra cost times the length of the match. The average length of a match lies around three which means that the difference between exponents of the extra cost and the initial cost that are on the x- and vertical axes along the line is \(\log_4 3 = 3.8\).

Optimal Length Threshold

The bottom graph in Figure 7.9 shows for each parameter combination what the length threshold \(l\) is that corresponds to the optimal value of the correctness. As can be seen, the length threshold is relatively low (around 3) for good parameter combinations. This is not unsurprising, when no matches can be made, the MDTW algorithm will only match one or two measurements. Once it is able to make longer matches, the matches will be accurate. Excluding matches that are longer than 3 will cause accurate matches to be considered as inaccurate. This value will be used in the following chapters as the value of the minimum match length parameter of the Trajectory Classifier.
CHAPTER 7. TRAJECTORY CLASSIFICATION

Figure 7.11: An extra step is added to the warping path when the initial cost is less than the extra cost times the length of the match.

Suitability

As can also be seen in Figure 7.9 the accuracy is 80–85% for parameter configurations near the upper left region of the plot. This indicates that the Trajectory Classifier is able to accurately match trajectories. Even when the cost parameters are set to suboptimal values the Trajectory Classifier is still able to correctly match more than half of the traces.

7.4.5 Influence of the Environment

In order to investigate the influence of the cell type on the optimal choice of the length threshold and the cost parameters the cells were divided in three groups urban, suburban and rural. For each cell type the optimal values are determined.

The environment of a cell might have an influence on the performance of the algorithm as well as on the optimal values of the parameters of the algorithm as the environment will have an influence on the measurements. Measurements might for instance be sent more or less frequently, there might be trajectories that are less easy to distinguish from each other and so on. When it is more difficult to distinguish between trajectories the minimum match length might for instance have to be set to a higher value as a higher confidence is required to distinguish between trajectories.

Figure 7.12 shows the results for cells that are located in an urban environment,
Figure 7.13 for cells that are located in a suburban environment and Figure 7.14 for cells that are located in a rural environment. As can be seen the optimal values for the initial and extra cost parameters and for the optimal length threshold are similar for every scenario. There is however a difference in how well the MDTW algorithm is able to discern between accurate and inaccurate matches. In an urban environment the Trajectory Classifier is slightly less accurate than in suburban environment and in a suburban environment it is even less accurate than in a rural environment. The reason for this is that in a rural environment cells are larger which means that users travel longer distances within a cell and there are fewer roads than in an urban cell. This means that in a rural cell there will be more measurements for the Trajectory Classifier to base its decisions on. At the same time there are bigger differences between the measurements that are collected on different trajectories through the cell as the measurement are performed further apart and there are fewer choices for the Trajectory Classifier to choose from.

Although the results in an urban environment are worse than in a rural environment the Trajectory Classifier is still able to accurately discern between accurate and inaccurate matches.

7.4.6 Signalling Overhead

The amount of signalling overhead that is caused by collecting measurements depends on the number of thresholds that are defined, the parameters of the events (i.e., the hysteresis and the TTT) and the radio conditions in the cell. Typically the amount of measurement reports will be limited to a couple of tens of measurements, usually spaced seconds apart. This can be seen in Figure 7.6 where the crosses indicate the locations where measurement reports are sent. For the scenario that is used in this thesis the average number of events A4 over all cells and all trajectories is 15.8 per call and per cell.

7.4.7 Other Environments

The results that were presented in this chapter so far were all obtained in the same simulation scenario. Although this scenario is based on real life data caution should be taken to assume that the results apply universally. Although this can never be shown definitively using simulations, more confidence in this fact can be achieved by evaluating the algorithm in other scenarios.

The Trajectory Classifier was evaluated in a different scenario in [58]. This evaluation was done in a more regular scenario featuring 48 hexagonal cells as is shown in Figure 7.15. Path loss calculations are performed using the Okumura-Hata model for large urban areas. Furthermore, shadow fading that is both auto-correlated in time and cross correlated with the shadow fading of other antennas is considered [49]. Measurements are collected by a user that moves according to a Manhattan
Figure 7.12: The optimal length for urban cells.
Figure 7.13: The optimal length for suburban cells.
Figure 7.14: The optimal length for rural cells.
mobility model. The velocity of the user changes at each intersection and is drawn uniformly from an interval that is centred around 5 m/s.

The algorithm was evaluated using the same accuracy metric as is described in Section 7.4.2. The obtained results are shown in Figure 7.16. In this plot the length of a match (horizontal axis) is plotted against the accuracy of the match (vertical axis) for various velocity intervals. Similar conclusions can be drawn from these results as for the Hanover scenario although in this scenario a match is considered to be accurate when its length is 5 or more.

### 7.5 Conclusion

This chapter presented the Trajectory Classifier. The Trajectory Classifier is the component of the SON function that is responsible for matching measurement traces coming from currently active users to reference traces that were collected from users that were active in the past and finding the one that matches the best. The measurement traces that are used for this are derived from measurement reports.
which are sent by the UE to its SeNB. In order to match the measurement traces the Trajectory Classifier used the MDTW algorithm. This algorithm is derived from the DTW algorithm that is used in signal processing to find an optimal alignment between two time series. The MDTW algorithm matches a trailing part of the active trace to an interval of reference traces from a set of reference traces. If a reliable match is found, it is subsequently used by the Traffic Steerer component to decide whether or not to steer the active user to another cell and, if so, to which cell.

The ability of the Trajectory Classifier to correctly match users that follow the same trajectory and to discern between users that do not follow the same trajectory was assessed using simulations. In these simulations pairs of trajectories were matched with each other and the returned match was compared to the geographical correspondence of the users. Results show that the Trajectory Classifier is able to correctly identify whether two trajectories match or not in 80–85% of cases. Furthermore the environment has an influence on the ability of the Trajectory Classifier to correctly distinguish between matching and non-matching trajectories. This influence is however marginal and the results are in any case very accurate.
Chapter 8

Trajectory Identification

8.1 Introduction

This chapter presents the Trajectory Identifier. The Trajectory Identifier is responsible for identifying the traces that will serve as reference traces. The identified reference traces will be kept in a set called the reference set. The Trajectory Classifier (see Chapter 7) will use the reference traces in this set to match an active trace to each time a new measurement is added to it.

It is important that the traces in the set of reference traces cover as much of the common trajectories through the cell as possible. At the same time the number of reference traces in the set should be kept as small as possible as the Trajectory Classifier will match an active trace to each reference trace in this set meaning that having redundant traces in this reference set will unnecessarily slow down the matching process. The Trajectory Identifier should furthermore be able to deal with changes in the environment which might render certain reference traces obsolete and/or might create a need for additional reference traces.

First the algorithms that are used in the Trajectory Identifier are detailed. Subsequently the ability of the Trajectory Identifier to select a concise set of reference traces that covers as many of the common trajectories through a cell and to deal with changes in the environment is assessed.

8.2 Reference Trace Selection

As the Traffic Steerer will influence the behaviour of the users, measurement traces that are collected by these users will be incomplete and biased reference traces. In
order to deal with this the Trajectory Identifier will randomly split users that are active in the cell into two distinct groups: tagged and untagged users. Whether a user is a tagged or untagged user will be determined by random chance. When a user arrives in a cell, either by starting a call or because it is handed over to the cell during an on-going call, the Trajectory Identifier will tag it with a certain (small) probability $p$. The probability of a new user becoming a tagged or an untagged user could also be adapted based on how well the Trajectory Classifier is able to make matches. If it is able to make accurate matches that allow the Traffic Steerer to make accurate decisions, $p$ could be decreased, otherwise $p$ could be increased. Note that a particular user can switch between being a tagged user and an untagged user within the same call when it is handed over from one eNB to another. The selection procedure is shown in Figure 8.1. The tagged users are used for the collection of reference traces that will later be used by the Trajectory Classifier to match active users to and subsequently by the Traffic Steerer to decide what the best choice for steering a user is. Tagged users will not be subject to traffic steering by the developed SON function but only to regular handovers and possibly other SON and non-SON functions that are active in the cell like MLB and MRO. Furthermore it is important that these tagged users are kept in macro cells as long as possible in order for the collected measurements to be as complete as possible. This is achieved by not handing over the tagged users between cells of different layers if it is not absolutely necessary to maintain connectivity. The small fraction of tagged users will however not experience bad performance as they will still be subject to MRO, MRO SON and other functions that are active in the cell. Untagged users will be subject to traffic steering by the Traffic Steerer. The measurements that are collected by the tagged users will be stored for later use, but they will not be used as long as the tagged user is active in the cell. When the user stops being active in the cell, either by ending its call or by being handed over to another cell, the collected measurements will become a reference trace. This reference trace is then added to the set of reference traces. Traces that are collected by untagged users will not be considered to become reference traces. Note that tagged users do not make additional measurements relative to untagged users. The only difference between tagged and untagged users is what the SeNB does with these measurements. In case of tagged users these measurements are used to create reference traces. In case of untagged users these measurements are used to steer the users.

### 8.3 Relevance Score

In order to avoid that the set of reference traces grows without bounds, a maximum size is imposed on it. This means that once this set has reached its maximum size, the next time a reference trace is added to it, another one should be removed from it. Imposing a fixed maximum size on the reference set is not the only way to deal with the problem of the reference set growing without bounds. The set could for instance be limited to traces that cover unique trajectories through the cell or to traces that are frequently used. Doing this would however increase the complexity of the Trajectory Identifier and is therefore out of the scope of this thesis.
Figure 8.1: The selection procedure of tagged and untagged users.
In order to determine which reference trace to remove when the set of reference traces has reached its maximum size and a new trace is added, each reference trace $i$ in the reference set $F$ is assigned a relevance score $S_i$. This relevance score is a value between 0 and 1 which expresses how useful a reference trace is. A relevance score close to 1 indicates that the reference trace is often matched to active traces and thus is relevant, whereas a relevance score close to 0 indicates that a reference trace is not used very often and thus is less relevant. When values are updated, in order to distinguish between values before and after the update, the superscript ‘old’ will denote the values of before the update and the superscript ‘new’ will denote the values after the update. Values that are the same before and after the update have no superscript.

When a reference trace in the set becomes more relevant, the relevance of all other reference traces should become less. For this the sum of all relevance scores is required to be equal to one (i.e., $\sum_{i \in F} S_i = 1$) at all times. Relevance is thus conserved.

Ways other than the relevance score have been tested. One of these was removing one of the two most similar reference traces when the reference set reached its maximum size. The problem with this removal strategy is explained in Section 8.5.2.

### 8.3.1 Adding a New Reference Trace

When a new reference trace $n$ is added to the set of reference traces, the initial relevance score of this new reference trace will be set equal to one over the new number of reference traces: $S_{\text{new}}^n = \frac{1}{|F_{\text{new}}|} = \frac{1}{|F_{\text{old}}|+1}$, while the relevance score of the already present reference traces $i$ will be updated to $S_i^{\text{new}} = \left(1 - \frac{1}{|F_{\text{new}}|}\right) S_i^{\text{old}}$. This is done to make sure that the newly added reference trace immediately is on a par with the other reference traces. Setting the relevance score of a new trace to 0 would cause newly added reference traces to be removed quickly after being added to the reference set and would keep traces that were in the reference set for a long time in the reference set even though they had become obsolete. It can easily be shown that the total amount of relevance is conserved. If the set of reference traces is empty, the relevance score of the first reference trace that is added to it, is set to one in order to satisfy the condition that the total relevance should be one.

### 8.3.2 Updating the Relevance Score

Whenever a match is made by the Trajectory Classifier between a reference and an active trace, the relevance of the reference trace should be increased. The relevance score of the matched reference trace is not just updated, instead, the relevance of all reference traces is updated according to their resemblance to the active trace. This is done because multiple reference traces can (partially) cover the same trajectory.
through a cell. When updating the relevance score of one of the traces that cover similar trajectories through a cell, it is desirable that the relevance scores of the other reference traces that cover similar trajectories through the cell is also increased in order to avoid increasing the relevance score of somewhat arbitrarily chosen reference traces. Therefore not only the score of the reference trace $i$ that was matched against the active trace will be updated. Instead the relevance score $S_j$ of all reference traces $j$ (which includes reference trace $i$) will be updated according to Equation 8.1.

$$S_j^{\text{new}} = \beta R_{i,j} + (1 - \beta) S_j^{\text{old}}$$

(8.1)

In this equation $\beta$ (whose value lies between zero and one) is a smoothing factor that determines the importance of a new match on the relevance score with zero meaning that a new match has no importance at all and one meaning that only the new match is taken into account. $R_{i,j}$ is the relative resemblance of the reference traces $i$ and $j$, whose definition is given in Equation 8.2.

$$R_{i,j} = \frac{D_{i,j}}{\sum_{k \in F} D_{i,k}}$$

(8.2)

$D_{i,j}$ is the normalised distance between the reference traces $i$ and $j$. This distance is obtained by applying the original DTW algorithm (see Section 7.3.1) to reference traces $i$ and $j$ and dividing the resulting distance by the length of the warping path. This result is subsequently subtracted from 1 yielding $D_{i,j}$. $D_{i,j}$ and $R_{i,j}$ lie between zero (totally different) and one (exactly the same) and satisfy the following properties:

$$D_{i,i} = 1$$
$$D_{i,j} = D_{j,i}$$
$$\sum_{j \in F} R_{i,j} = 1$$

It can easily be shown that the ‘total amount of relevance’ is conserved after updating the relevance scores.

### 8.3.3 Removing Reference Traces

Once the set of reference traces has reached its maximum size, each time a new reference trace is added to the set another one is removed from it. The reference trace that will be removed is the one with the lowest relevance score in the set of the reference traces. In case there is more than one reference trace with the lowest score one of these is removed at random. Note that when the reference set is full, a trace is first removed from it before the new trace is added. This is done because the Trajectory Identifier has no data yet on the newly added trace and cannot assess whether it is more useful than the traces that already are in the reference set. The reference set will therefore include a small fraction of traces of which the actual relevance has not been established yet. This will happen over time as matches are
made. It is assumed that, if the newly added reference trace is not relevant, the relevance score of the newly added reference trace will quickly decrease and that the reference trace is soon removed from the reference set. This is not a problem as long as the reference set consist of a sizeable fraction of the reference set consist of relevant traces which will be the case if the parameters of the Trajectory Identifier are chosen right.

When a trace is removed, its relevance is distributed among all other reference traces according to their relative resemblance to the removed reference trace. This will ensure that the total amount of relevance is conserved. A more important reason for doing this is that when there are multiple reference traces that resemble each other, the total relevance of the trajectory that is covered by them is distributed among these reference traces. This means that these traces are more likely to be removed than a trace that covers a trajectory on its own, while this trajectory might be less important. By redistributing the relevance of the removed trace over the remaining traces relative to their resemblance to the removed trace, traces that cover the same trajectory will be assigned a higher relevance score, which makes it less likely that they will be removed. When a reference trace \( n \) is removed, its relevance score is redistributed among the other reference traces \( i \) according to Equation 8.3.

\[
S_{i}^{\text{new}} = S_{i}^{\text{old}} + \frac{R_{n,i}^{\text{old}}}{\sum_{j \in F_{\text{old}} \setminus \{n\}} R_{n,j}^{\text{old}}} S_{n}^{\text{old}}
\]  

(8.3)

After removing a reference trace, the total amount of relevance is conserved.

### 8.4 Example

This section gives an illustrative example of how a reference set is updated when the relevance scores are updated and when a trace is removed and a new one is added.

Table 8.1 shows the initial state of a reference set with size 5. As can be seen, trace 1 has the highest relevance, followed by traces 3, 5, 2 and 4.

<table>
<thead>
<tr>
<th>Trace</th>
<th>Relevance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.35</td>
</tr>
<tr>
<td>2</td>
<td>0.1</td>
</tr>
<tr>
<td>3</td>
<td>0.25</td>
</tr>
<tr>
<td>4</td>
<td>0.1</td>
</tr>
<tr>
<td>5</td>
<td>0.2</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>1</strong></td>
</tr>
</tbody>
</table>
When a new match between an active trace and a trace in the reference set is made, the relevances of all traces in the reference set are updated. Table 8.2 shows the resemblances of the active trace and the traces in the reference set. As can be seen the active trace resembles trace 2 the most and trace 1 the least.

Table 8.2: The resemblances of the traces in the reference set and the active trace.

<table>
<thead>
<tr>
<th>Trace</th>
<th>Distance</th>
<th>Resemblance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.2</td>
<td>0.091</td>
</tr>
<tr>
<td>2</td>
<td>0.8</td>
<td>0.364</td>
</tr>
<tr>
<td>3</td>
<td>0.5</td>
<td>0.227</td>
</tr>
<tr>
<td>4</td>
<td>0.3</td>
<td>0.136</td>
</tr>
<tr>
<td>5</td>
<td>0.4</td>
<td>0.182</td>
</tr>
<tr>
<td>Total</td>
<td>2.2</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 8.3 shows the state of the reference set after updating the relevances with a smoothing factor ($\beta$) of 0.1. As can be seen the relevances of the traces evolve to their resemblance with the active trace.

Table 8.3: The state of the reference set after updating the relevances because of the match with the active trace.

<table>
<thead>
<tr>
<th>Trace</th>
<th>Relevance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.324</td>
</tr>
<tr>
<td>2</td>
<td>0.126</td>
</tr>
<tr>
<td>3</td>
<td>0.248</td>
</tr>
<tr>
<td>4</td>
<td>0.103</td>
</tr>
<tr>
<td>5</td>
<td>0.198</td>
</tr>
<tr>
<td>Total</td>
<td>1</td>
</tr>
</tbody>
</table>

When subsequently a tagged user leaves the cell and a new reference trace is added to the reference set, first trace 4, the trace with the lowest relevance is removed. Its relevance is redistributed over the elements of the reference set as is shown in Table 8.4 according to their resemblances to the removed trace which are shown in Table 8.5.

Table 8.4: The resemblance of the reference traces to reference trace 4.

<table>
<thead>
<tr>
<th>Trace</th>
<th>Distance</th>
<th>Resemblance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.5</td>
<td>0.172</td>
</tr>
<tr>
<td>2</td>
<td>0.9</td>
<td>0.310</td>
</tr>
<tr>
<td>3</td>
<td>0.4</td>
<td>0.138</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>0.345</td>
</tr>
<tr>
<td>5</td>
<td>0.1</td>
<td>0.034</td>
</tr>
<tr>
<td>Total</td>
<td>1.9</td>
<td>1</td>
</tr>
</tbody>
</table>
Table 8.5: The state of the reference set after deleting trace 4 and redistributing its relevance.

<table>
<thead>
<tr>
<th>Trace</th>
<th>Relevance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.351</td>
</tr>
<tr>
<td>2</td>
<td>0.175</td>
</tr>
<tr>
<td>3</td>
<td>0.27</td>
</tr>
<tr>
<td>5</td>
<td>0.204</td>
</tr>
<tr>
<td>Total</td>
<td>1</td>
</tr>
</tbody>
</table>

When afterwards the new reference trace (trace 6) is added to the reference set its relevance is set to 1 over the size of the (new) reference set. The same amount of relevance is subtracted from the other traces in the reference set as is shown in Table 8.6.

Table 8.6: The state of the reference set after adding trace 6 and adjusting the relevances.

<table>
<thead>
<tr>
<th>Trace</th>
<th>Relevance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.281</td>
</tr>
<tr>
<td>2</td>
<td>0.14</td>
</tr>
<tr>
<td>3</td>
<td>0.216</td>
</tr>
<tr>
<td>5</td>
<td>0.163</td>
</tr>
<tr>
<td>6</td>
<td>0.2</td>
</tr>
<tr>
<td>Total</td>
<td>1</td>
</tr>
</tbody>
</table>

### 8.5 Evaluation

In order to demonstrate that the Trajectory Identifier functions as intended, simulations were performed. The goal of these simulations is threefold:

- First the simulations will demonstrate that the Trajectory Identifier is able to include the most relevant reference traces in the reference set.

- Secondly the simulations will demonstrate that the Trajectory Identifier is able to deal with changes in the cell that render certain reference traces obsolete in a timely fashion.

- Thirdly the simulations will be used to establish suitable values for the parameters of the Trajectory Identifier, namely the smoothing factor *β*, the tagging probability *p* and maximum size of the reference set.
8.5. **EVALUATION**

8.5.1 **Reference Set Coverage**

In the simulations that are performed to demonstrate that the Trajectory Identifier is able to include the most relevant reference traces in the reference set the updating of the reference set is performed as would be the case in normal circumstances. These simulations run for a certain amount of time and at the end of the simulations the coverage of the resulting reference set is assessed.

In order to be able to assess the coverage of the reference set, a metric is required that reflects how well the different reference traces cover the major trajectories through a cell. When the trajectories through a cell that are frequently used by users are covered by the traces in the reference set the Trajectory Classifier will be able to make more matches and consequently the Traffic Steerer will be able to steer more users. This will reduce the number of unnecessary handovers thereby achieving the goals of the developed algorithm.

The coverage metric that will be used for this purpose expresses how many times on average a street in the reference set is visited by a user. By normalising with the number of segments, reference sets of different sizes can be compared to each other and by including the length the result is not biased towards longer segments. It will allow to compare two different reference sets within a cell but not between cells.

The roads that run through a cell can be represented by a number of nodes that are connected to each other by lines. Each node is a crossroad and the lines are the streets that connect the crossroads to each other. As users follow the roads that run through a cell, trajectories will start at some point along a road and run along a number of interconnected roads before stopping along another road as is shown in Figure 8.3. When a number of trajectories are considered, the roads are sectioned into segments by the points where a new call is started or an existing call is stopped. The number of users that pass on the part of a road between two consecutive of these points including the begin and end point of the lines is constant.

Suppose that the set of all trajectories that pass through a cell is $T$. Each element $i$ of this set is in turn a sequence of consecutive road sections $j$ that are visited by the user. The definition of the coverage metric is given in Equation 8.4.

$$
\frac{\sum_{j \in \bigcup_{i \in T} i} V_j l_j}{\sum_{i \in T} \sum_{j \in i} l_j}
$$

(8.4)

In this equation $V_j$ is the number of users that visited road section $j$ and $l_j$ is the length of road section $j$. The numerator yields the total distance travelled by users along the trajectories that are covered by the reference traces. In case a road is covered by multiple reference traces it is only counted once. The denominator yields the total distance that is covered by the traces in the reference set. When two reference traces cover the same road it is counted twice. This means that if the
reference set contains the same trajectory twice, or contains two similar trajectories that overlap for a large amount, the coverage metric will be lower than when no or few roads are covered by multiple reference traces.

Figure 8.2 shows a map of the streets in the area that is covered by a particular cell. Each street is coloured according to the number of times it is visited by a user.

As an illustrative example, consider the streets that run through a cell that are shown in Figure 8.3. Table 8.7 shows the number of users that visit each segment and the length of the segment. Table 8.8 shows the list of segment each trajectory consists of.

Suppose the reference set consists of the traces 1, 3 and 6. In this case the uniquely
8.5. EVALUATION

Figure 8.3: A map of the example cell.

visited segments are \( s_2, s_3, s_4, s_5, s_6, s_7, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{16}, s_{19} \) and \( s_{20} \). The numerator of equation Equation 8.4, \( \sum_{j \in \bigcup_{i \in T} V_j l_j} \) in that case yields 66.16 and the denominator \( \sum_{i \in T} \sum_{j \in i} l_j \) yields 35.36, meaning that the coverage metric is 1.87.

In case the reference set consists of the traces 2, 5 and 6 there is much more overlap between the reference traces than in the previous example. In this case there are much fewer uniquely visited segments, namely \( s_3, s_5, s_9, s_{11}, s_{12}, s_{13}, s_{18}, s_{19}, s_{20}, s_{21} \). The numerator of equation Equation 8.4 then becomes 56.2 and the denominator becomes 39.43, meaning that the coverage metric is equal to 1.43.

As is to be expected the coverage metric is much higher in the first case where the traces in the reference set cover more of the total list of trajectories than in the second case where there is a much higher overlap among the trajectories in the reference set and less trajectories are covered.

Simulation Results

The Trajectory Identifier has three parameters: the tagging probability \( p \), the smoothing factor \( \beta \) that determines the impact of a new match on the relevance scores of the traces in the reference set and the maximum size of the reference set \( N \).

Both the tagging probability \( p \) and the smoothing factor \( \beta \) assume values 0.0001,
Table 8.7: The number of users that visit each segment and the length of the
segment.

<table>
<thead>
<tr>
<th>Segment ($s_i$)</th>
<th>$V_i$</th>
<th>$l_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_1$</td>
<td>0.50</td>
<td>0.00</td>
</tr>
<tr>
<td>$s_2$</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>$s_3$</td>
<td>2.00</td>
<td>1.95</td>
</tr>
<tr>
<td>$s_4$</td>
<td>1.95</td>
<td>1.77</td>
</tr>
<tr>
<td>$s_5$</td>
<td>1.77</td>
<td>1.95</td>
</tr>
<tr>
<td>$s_6$</td>
<td>1.58</td>
<td>1.58</td>
</tr>
<tr>
<td>$s_7$</td>
<td>0.71</td>
<td>0.90</td>
</tr>
<tr>
<td>$s_8$</td>
<td>1.77</td>
<td>1.06</td>
</tr>
<tr>
<td>$s_9$</td>
<td>2.00</td>
<td>2.06</td>
</tr>
<tr>
<td>$s_{10}$</td>
<td>1.14</td>
<td>2.67</td>
</tr>
<tr>
<td>$s_{11}$</td>
<td>1.36</td>
<td>3.05</td>
</tr>
<tr>
<td>$s_{12}$</td>
<td>0.76</td>
<td>1.06</td>
</tr>
<tr>
<td>$s_{13}$</td>
<td>3.00</td>
<td>1.36</td>
</tr>
<tr>
<td>$s_{14}$</td>
<td>2.00</td>
<td>4.81</td>
</tr>
<tr>
<td>$s_{15}$</td>
<td>2.24</td>
<td>0.96</td>
</tr>
<tr>
<td>$s_{16}$</td>
<td>0.50</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Table 8.8: The list of segments each trajectory consists of.

<table>
<thead>
<tr>
<th>Trajectory</th>
<th>Segments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$s_2$, $s_3$, $s_4$, $s_6$, $s_{10}$, $s_{11}$, $s_{12}$, $s_{14}$</td>
</tr>
<tr>
<td>2</td>
<td>$s_{11}$, $s_{12}$, $s_{13}$, $s_{18}$, $s_{19}$, $s_{20}$, $s_{21}$</td>
</tr>
<tr>
<td>3</td>
<td>$s_7$, $s_{10}$, $s_{11}$, $s_{12}$, $s_{13}$, $s_{16}$</td>
</tr>
<tr>
<td>4</td>
<td>$s_6$, $s_{10}$, $s_{11}$, $s_{12}$, $s_{13}$, $s_{18}$, $s_{19}$</td>
</tr>
<tr>
<td>5</td>
<td>$s_9$, $s_{20}$, $s_{19}$, $s_{18}$, $s_{13}$, $s_{12}$</td>
</tr>
<tr>
<td>6</td>
<td>$s_3$, $s_5$, $s_9$, $s_{20}$, $s_{19}$</td>
</tr>
</tbody>
</table>

0.0002, 0.0005, 0.001, 0.002, 0.005, 0.01, 0.02, 0.05, 0.1, 0.2 and 0.5 while the
maximum size of the reference set assumes the values 5, 10, 15, 20 and 25. For
each parameter combination 100 simulation runs are performed where 100000 users
erenter and leave the cell that are active during their entire stay. Each user is either
designated to be a tagged or an untagged user and the reference set is updated
according to Section 8.3. After all users are processed the coverage of the final
reference set is determined and averaged per parameter combination over the 100
performed runs.

The obtained results are compared to a baseline where each time a new trace is
added to the reference set when it is full, another one which is chosen at random is removed from it instead of the one with the lowest relevance. As the relevance score is not used in the baseline the smoothing factor parameter is not relevant. The tagging probability and the reference set size are however relevant. Also for the baseline simulations 100 runs are performed per parameter combination.

The simulation results are presented in a graph per reference set size where the smoothing factor $\beta$ is varied on the horizontal axis and the tagging probability $p$ is varied on the vertical axis.

Figure 8.4 shows results for the cell 89 in Figure 6.2 for a reference set size of 5. The coverage metric for the different parameter configuration ranges from 15 to 22. The most optimal combination of smoothing factor ($\beta$) and tagging probability ($p$) lies around 0.01 and 0.001 respectively. When moving away from this point the coverage metric gradually becomes less. This is to be expected, when the tagging probability becomes too low not enough users are tagged in order to obtain a representative reference set after 100000 users entering and leaving the cell. When the tagging probability is too high, reference traces are replaced too often for the relevance scores to be updated properly. Remember that when a new reference trace is added to the reference set its initial relevance score is set to one over the total amount of reference traces in the reference set. This value will be higher than the relevance score of a number of reference traces in the set as the total of all relevance scores is always one which will be divided over $N$ reference traces meaning that a relevance score of $\frac{1}{N}$ is equal to the mean relevance score. If many new reference traces are added to the reference set without updating their relevance score afterwards before adding new reference traces, the older reference traces whose relevance score was less than $\frac{1}{N}$ will be removed first while these are not necessarily less relevant than the newly added reference traces. When the smoothing factor is too low something similar happens, the relevance scores after adding a new reference trace are not updated swiftly enough before a new reference trace is added for the least relevant trace to be replaced. When the smoothing factor is too high the relevance scores will fluctuate too much and the reference traces that matched with the last few active traces will have a too high impact on the relevance scores of the reference traces.

When the reference set is larger (10 traces), the difference between good parameter combinations and bad parameter combinations becomes more outspoken, as can be seen in Figure 8.5. This is because when the reference set is bigger it is no longer only important to select reference traces that cover trajectories that are often visited by users but also to not select multiple similar traces as doing so does not benefit the coverage. When there are only a few reference traces it is harder to select two similar reference traces by accident. As can also be seen some parameter configurations are worse than the baseline which replaces a reference trace at random when a new reference trace is added. For instance, when the tagging probability is low, the reference set might contain all irrelevant traces. Some of these traces will end up with a higher than average relevance score because the total amount of relevance is always one. It will be more difficult to remove these traces afterwards resulting in the algorithm performing worse than the baseline.
Figure 8.4: The coverage for various combination of the smoothing factor $\beta$ and the tagging probability $p$ for a reference set of 5 traces.

Figure 8.5: The coverage for various combination of the smoothing factor $\beta$ and the tagging probability $p$ for a reference set of 10 traces.
When the maximum size of the reference set is even larger (15 traces), the difference between good parameter combinations and bad parameter combinations becomes even more outspoken, as can be seen in Figure 8.6. Furthermore it can be seen that the optimal parameter configuration has shifted to a tagging probability \( p \) of 0.005. This trend was already visible for a reference set size of 10 in Figure 8.5. When the reference set is larger, replacing a single reference trace has a lower impact on the coverage of the entire set. This allows for trying out more new reference traces which leads to an even better coverage of the reference set.

![Figure 8.6: The coverage for various combination of the smoothing factor \( \beta \) and the tagging probability \( p \) for a reference set of 15 traces.](image)

At some point there is no outspoken optimal parameter configuration anymore as can be seen in Figure 8.7 and even more clearly in Figure 8.8 although there are still good and bad parameter configurations. This happens when most of the important trajectories through a cell are covered by reference traces. Making the reference set larger will result in trajectories being covered by two or more reference traces. This can also be seen from the average coverage per reference trace which suddenly is much lower when the reference set is smaller as in Figure 8.6. When a certain trajectory is covered by two reference traces it is only counted once. Note that the coverage metric for the entire set (so not per reference trace) becomes larger as the reference set grows.

The maximum reference set size \( N \) at which the reference traces will start to overlap for a great amount depends from cell to cell and on the amount of roads that run through a cell. This is shown in Figure 8.9 which shows the results for a reference set of 25 as in Figure 8.8 but for cell 134 in Figure 6.2. As can be seen these results look more similar to Figure 8.6 rather than to Figure 8.8.
Figure 8.7: The coverage for various combination of the smoothing factor $\beta$ and the tagging probability $p$ for a reference set of 20 traces.

Figure 8.8: The coverage for various combination of the smoothing factor $\beta$ and the tagging probability $p$ for a reference set of 25 traces.
8.5. Evaluation

The results in this section show that the coverage of a reference set increases as its size increases. Once the set is sufficiently large the parameters of Trajectory Identifier become of less importance. When this point has not been reached the parameters do play an important role in determining the coverage of the reference set. The maximum size of the reference set at which this happens is however not the same for all cells but depends on the environment. For most realistic cells a size of 25 or more will suffice. The choice of the maximum size of the reference set mainly depend on the computational resources that are available to the SON function as the trade off is mainly between the maximum size of the reference set and the time it takes to make a match. Setting the parameters of the Trajectory Identifier to optimal values will however help to cover more trajectories with the same number of reference traces. Based on the results it is suggested to use values around 0.05 for the initial cost and 0.02 for the extra cost.

8.5.2 Dealing With Changes in the Environment

Another important feature of the Trajectory Identifier is that it should be able to deal with changes in the environment that cause certain reference traces to become obsolete and new traces to become relevant. This is not a very difficult requirement to deal with as users are tagged randomly meaning that sooner or later users that follow relevant traces will be tagged and added to the reference set. Care has however to be taken. In earlier versions of the Trajectory Identifier, when a trace
had to be removed from the reference set, one of the two most similar traces would be
removed instead of assigning a relevance score to each reference trace and removing
the one with the lowest score. This approach had however a shortcoming: whenever
a trace becomes irrelevant because of changes in the cell, and other traces that
resemble it have already been removed, it is unlikely that this irrelevant trace will
still be removed as no newly added traces will resemble it (otherwise the trace would
not have become irrelevant). This problem is illustrated in Figure 8.10: the set of
reference traces is filled over time and eventually reaches its maximal size (denoted
by $N$). After a change in the environment, each time when a new reference trace is
added to the set another trace is removed and the total number of reference traces
remains constant. In this case it is important that these obsolete traces (denoted
by the line marked ‘Before change’) are replaced by relevant traces (denoted by the
line marked ‘After change’) in a timely fashion. When removing one of the two
most similar traces, at some point some irrelevant traces might be left behind in
the set of reference traces as there are no other traces left anymore that are similar
enough to be discarded. After a couple of changes in the environment a large part
of the set of reference traces might get filled with obsolete traces.

![Figure 8.10: After a change in the environment the Trajectory Identifier might fail
to remove all obsolete reference traces in case each time a reference trace needs to
be removed one of the two most similar reference traces is removed.](image)

In order to demonstrate that the current version of the Trajectory Identifier which
removes traces based on their relevance score, is able to deal with changes in the
environment, simulations were performed. These simulations were not performed
in the Hanover scenario, but in a more controlled environment. This simulation
environment is a rectangular area measuring 433 m by 500 m. It features wrap
around meaning that users that cross the edge at one side will reappear at the other side. There are 16 cells which are placed as illustrated in Figure 8.11. In the simulation 25 users travel in the same direction in parallel lanes each 86.6 m apart in the $x$-direction and 100 m in the $y$-direction. After a certain amount of simulation time (16000 s) the users make a left turn of 90 degrees and start travelling in that direction. This process repeats itself another 2 times causing the users to travel in 4 different directions each at an angle of 90 degrees relative to the previous direction. As the users pass through different cells they are either tagged or not. Tagged users will be added to the reference set which is managed by the Trajectory Identifier. The amount of reference traces in the reference set is plotted over time separated per direction the user from which the reference trace was derived was travelling in. The size of this reference trace set is limited to 20 traces.

![Simulation Area](image)

Figure 8.11: An overview of the simulation area.

Figure 8.12 shows the evolution of the content of the set of reference traces in the case when each time one of the two most similar traces is removed. The different background colours show when the users in the simulation follow a particular direction. After the simulation has started, the reference set is quickly filled with reference traces of direction 1 until it reaches its maximum size. From this point on a reference trace is removed from the set each time a new reference trace is added to it. When the direction of the users changes for the first time, reference traces of direction 1 are removed in favour of reference traces of direction 2. There will however be reference traces of direction 1 in the reference set during the entire time the users are travelling in direction 2 and the reference traces of direction 2 will never occupy the entire reference set. When the direction changes again, this time from direction 2 to direction 3, some of the reference traces of direction 2 in the reference trace set will be replaced by reference traces of direction 3. This time only a few reference traces of direction 2 will be replaced by reference traces of direction 3.
and the fraction of reference traces of direction 3 in the set will never become very high. At the same time there will still remain a certain amount of reference traces of direction 1 in the set, which the removal algorithm still fails to remove from the reference set. When the direction of the users changes once more, this time from direction 3 to direction 4 the fraction of reference traces of direction 2 is reduced but there still remain a few reference traces of direction 2 and reference traces of direction 3 in the reference set and the reference traces of direction 4 never occupy the entire reference trace set. This example perfectly illustrates the anticipated problem with the old removal algorithm: the algorithm fails to remove obsolete reference traces in favour of more relevant reference traces. Because there will remain a residual fraction of obsolete reference traces that can no longer be removed from the reference set as there are no other reference traces that are similar enough to them in order for them to be removed from the reference set.

Figure 8.12: When removing one of the two most similar traces from the reference set obsolete traces will remain in the reference set.

Figure 8.13 shows the evolution of the reference traces in the reference set in case each time the least relevant trace is removed. As can be seen in this figure the new algorithm is much better at removing obsolete reference traces from the reference set. Once the direction of the user changes from direction 1 to direction 2, the fraction of reference traces of direction 1 starts to decrease in favour of the reference traces of direction 2 whose fraction starts to increase. After a while the reference traces of direction 1 have all been replaced by reference traces of direction 2. Note that the rate at which the obsolete reference traces are replaced by relevant reference traces is not necessarily higher than with the old algorithm. This is also not desired: it might be that certain reference traces are not used temporarily but will be used again at a later point in time. The main advantage of removing the least relevant reference trace is that it is able to remove all obsolete reference traces and none are
left behind.

Figure 8.13: Removing the least relevant reference trace ensures that all obsolete reference traces are removed from the reference set.

Figure 8.13 also gives an indication of how long it takes for the Trajectory Identifier to react on a change. After the direction changes from direction 1 to direction 2 it takes 6100 s (1h 43m) for the last traces from direction 1 to be removed from the reference set, after the change from direction 2 to direction 3 it takes 6600 s (1h 51m) for the last traces from direction 2 to be removed from the reference set and after the change from direction 3 to direction 4 it takes 4800 s (1h 21m). This is however a very rough estimate. The time it takes for the Trajectory Identifier to react strongly depend on a number factors like the parameters of the Trajectory Classifier, the amount of users that visit the cell and the magnitude of the change. The tagging probability and the smoothing factor for updating the relevance score whenever a match is made have an obvious impact as a higher tagging probability will increase the rate at which the traces in the reference set can be replaced. Especially when a change does not affect all traces it will take some time for enough new reference traces are collected that can replace the obsolete reference traces. The smoothing factor for updating the relevance score whenever a match is made influences the rate at which reference traces are replaced as it determines how fast the relevance score of the obsolete traces decreases. The amount of users that visit a cell influences the time it takes for the Trajectory Identifier in the same way as the tagging probability. When relatively few users visit a cell it will take longer before enough tagged users have visited the cell to replace the obsolete traces while cells which are visited frequently will be able to replace their obsolete reference traces sooner. Finally the magnitude of the change will also determine how long it will take before obsolete traces are removed. When changes occur gradually, the relevance scores of obsolete traces will diminish at a lower pace as the traces will not become obsolete at once.
whereas more sudden changes will more rapidly cause the relevance score of traces to diminish. It will therefore rather take multiple hours or days before reference traces that have been rendered obsolete by a change in the environment to be removed from the reference set.

8.6 Conclusion

This chapter presented the Trajectory Identifier which is responsible for identifying the traces that will serve as reference traces. New reference traces are selected by randomly tagging new users with a certain probability. The traces coming from these tagged users become reference traces ones they leave the cell. In order for the set of reference traces not to grow without bounds old traces are removed from the set once it reaches a certain size. In order to do this each reference trace is assigned a relevance score. This score is updated each time a reference trace is used. When a trace has to be removed the trace with the lowest relevance score is selected.

Results show that the optimal parameter configuration depends on the type of the cell. However, the larger the reference set is, the more trajectories through the cell it will cover. The choice of reference set size therefore depends mostly on the computational resources that are available to the SON function. The maximum size of the reference set could even be set to a more than high enough value and the Trajectory Classifier could match an active trace to the traces in the reference set in order of their relevance and as soon as it has spent a certain amount of computational resources. This is however outside the scope of the thesis and was not studied. When limiting the maximum size of the reference set however, setting the initial and extra costs to optimal value will greatly improve the coverage.
Chapter 9

Traffic Steering

9.1 Introduction

Once a sufficiently reliable match of a currently active user with a reference trace has been made by the Trajectory Classifier, the Traffic Steerer will make a traffic steering decision. By assuming that events that occur to one user will also occur to another user that follows a similar trajectory through the cell, the Traffic Steerer can decide whether it is beneficial to hand over the user to another cell or to keep it in the current cell. In order to do this, the Traffic Steerer will make an estimate of the future behaviour of the active user using the reference trace that was matched to it. In order to do so, events that occurred to the reference user in the period after the match are projected on the active user. In the scope of this thesis, events mean new RSRP measurements and thus possible changes in SINR and consequently spectral efficiency. In this chapter the algorithms that are used by the Traffic Steerer component of the SON function to decide whether and to which target cell a user should be handed over are described. The three parts of this component are:

1. The extrapolation and projection of reference events
2. The estimation of future radio conditions of an active user
3. The making of the traffic steering decision

9.2 Event Extrapolation and Projection

Measurement events can be used to obtain a rough indication of the future spectral efficiency that could be achieved by a user in various cells, which will in turn be used to determine how the active user will be steered. Future events that will occur to
the active user can be predicted by projecting events that occurred to the reference user in the period after the match on the active user.

Figure 9.1 summarises the general idea of the event projection. The horizontal axis of this plot shows the timeline of the reference user while the vertical axis shows the timeline of the active user. The beginning of the match of the measurements of the reference and active users corresponds to $x = r_s$ and $y = a_s$ respectively, while the end of the match corresponds to $x = r_e$ and $y = a_e$ respectively. A future event of the reference user at time $x = r_x$ can be extrapolated linearly yielding the projected time of the corresponding event for the active user at time $y = a_x$ using Equation 9.1

$$a_x = \frac{r_x - r_s}{r_e - r_s} (a_e - a_s) + a_s$$  \hspace{1cm} (9.1)

Figure 9.1: The projection of an event $x$ by linear extrapolation.

### 9.3 Radio Condition Estimation

Using the ability to project future reference events on the active user, the future spectral efficiency in various possible NeNBs of the active user can be estimated. The Trajectory Classifier collects measurements of the form $(t_i, m_i)$ where $t_i$ is the time of event $i$ and $m_i$ is a list of NeNB Identifiers (ID) and their associated RSRP measurement at that time. Note that the exact RSRP value is not stored, instead, the ID of the associated Report Configuration is stored. This only allows to calculate the lower bound $q_{i,j}$ and consequently the upper bound $q_{i,j} + \delta_q$ of the interval in which the exact RSRP of eNB $j$ lies. As a new measurement is generated when the
The **RSRP** of a NeNB rises above or falls below a threshold of either an A1, A2 or A4 event it is assumed that during the interval between two consecutive measurements $i$ and $i+1$, the **RSRP** level of NeNB $j$ lies between $q_{i,j}$ and $q_{i,j} + \delta q$, i.e., the **RSRP** is known within an interval $\delta q$ which is a configuration parameter of the Trajectory Classifier. Using the known **RSRP** levels of the NeNBs within a certain interval $\delta q$, the worst-case **SINR** within the time interval $[t_i; t_{i+1}]$ for each NeNB $j$ can be estimated using Equation 9.2.

$$\text{SINR}_{i,j} = \frac{10^{\frac{q_{i,j}}{10}}}{N + \sum_{k \in I_j} 10^{\frac{q_{i,k}}{10} + \delta q}}. \quad (9.2)$$

In this equation $N$ denotes the noise figure which typically is set to a value like $4.0 \cdot 10^{-14}$ W = -103.9 dBm, the thermal noise at 20 °C. $I_j$ represents the set of cells that transmit in the same frequency band as cell $j$, not including $j$ itself and therefore interfere with it. Note that in Equation 9.2 the upper bound of the **RSRP** interval is used to calculate the interference while the lower bound is used for calculating the signal strength. This yields a pessimistic estimate for the **SINR** in that interval. The **SINR** of the SeNB is calculated according to Equation 9.3.

$$\text{SINR}_{i,\text{SeNB}} = \frac{10^{\frac{q_{i,\text{SeNB}} + \delta q}{10}}}{N + \sum_{k \in I_{\text{SeNB}}} 10^{\frac{q_{i,k}}{10}}}. \quad (9.3)$$

The **SINR** values that are obtained for the SeNB and all the NeNBs can then be mapped on spectral efficiency $S$ values using the Modified Shannon Formula [20] which is given in Equation 9.4.

$$S = \alpha \log_2(1 + \text{SINR}) \quad (9.4)$$

In this equation $\alpha$ is a fixed parameter which is set to 0.6. The spectral efficiency, which expresses the number of bits per second per hertz of spectrum can be transmitted will be used to compare the conditions in various cells. The spectral efficiency is used as it gives an indication of how efficient the radio resources in a cell can be used by a certain user. Note that this formula does not take into account the discreteness of the MCS or MIMO. The traffic steering algorithm will use these spectral efficiency estimates for making the traffic steering decisions.

The **SINR** estimates are very rough as the **RSRP** values on which they are calculated are only known between certain discrete levels. As a consequence of this and because the Shannon Formula is only an approximate formula that does not take factors like the load conditions in the various cells into account, the spectral efficiency measure is only a rough estimate of the spectral efficiency that can be achieved in the target cell. This means that the Traffic Steerer bases its decisions on very rough estimates which will have an impact on the accuracy of the decisions that are made by it. The estimates of the **SINR** of the NeNBs are however pessimistic while the estimate of the **SINR** of the SeNB is optimistic. Furthermore, the rougher the estimates the **RSRP** estimates are, the more biased the **SINR** estimates of the NeNBs will be towards lower values and the more biased towards higher values the **SINR** estimates.
of the SeNB will be. The Traffic Steerer will only steer a user if the average spectral efficiency that it is predicted to achieve is higher in the NeNB than in the SeNB. This means that although it bases its decisions on rough estimates, the biasedness of these estimates will ensure that users are only steered if it is very certain that the achieved spectral efficiency will be higher in the NeNB.

9.4 Traffic Steering Decision

The traffic steering algorithm is responsible for steering users to the appropriate cells such that the amount of short stays is reduced. At the same time the radio conditions should not be severely impacted by the steering of the users. In short, the Traffic Steerer will regularly assess whether handing over a user to another cell would allow the user to stay in this cell for a sufficiently amount of time while not reducing the spectral efficiency experienced by the user. If it finds such a cell a handover to that cell will be triggered, otherwise handovers of that user will be blocked as long as the signal quality is sufficiently high in the serving cell. On each new match that is made (for an untagged user) by the MDTW algorithm of the Trajectory Classifier and that is deemed accurate enough by the Trajectory Classifier, the Traffic Steerer will estimate the projected spectral efficiency of the corresponding user as is described in Section 9.2. This will yield a spectral efficiency estimate for each NeNB as well as for the SeNB itself that varies stepwise for each projected measurement. In Figure 9.2 an example of how this data might look is given.

Using this data the Traffic Steerer will, for each NeNB for which data is available, check whether there exists periods that satisfy the following conditions:

1. they start at the current time;
2. they have a length that is longer than a predefined minimum length;
3. during these periods the predicted spectral efficiency is at no point lower than a certain fraction of the predicted spectral efficiency of the SeNB and;
4. during these periods the average predicted spectral efficiency is above the average predicted spectral efficiency of the SeNB.

If such a period is found the user will be steered to this NeNB. If multiple of these periods are found, the user will be steered to the NeNB for which the average spectral efficiency is the highest. Note that it is possible that multiple periods exist for the same NeNB as the algorithm will iterate over the projected future measurements and will check whether the period starting from the current time until the measurement satisfies conditions (2), (3) and (4). The rationale behind condition (1) is that a handover will not be triggered prematurely but only if making a handover at the present time will immediately result in a better performance. If
Figure 9.2: The SINR and spectral efficiency are derived from the RSRP measurements.
CHAPTER 9. TRAFFIC STEERING

the NeNB remains a viable target for making a handover to, the handover will be triggered at a later measurement anyway. Condition (2) ensures that when a handover is triggered the user will be able to stay in the target cell and experience a good spectral efficiency for a sufficiently long amount of time. This time is called the minimum stay duration and is a parameter of the Traffic Steerer. A typical value for this parameter is 10 s. In order to assure that the spectral efficiency of the user will at no point become much worse than when the user would have stayed with the SeNB and that the overall spectral efficiency improves, conditions (3) and (4) are added. Note that applying these rules will implicitly make a distinction between micro and macro cells and slow and fast moving users as fast moving users will not be able to stay in micro cells for a long time and will therefore not be steered towards micro cells while it is possible to do this with slow moving users.

9.5 Example

This section gives an example of how the SON function operates. Figure 9.3 shows a match of an active user with a reference user and the subsequent steering of this user based on this match.

The left picture shows the trajectories of the reference user (purple) and the active user (green). These trajectories are the entire paths that were followed by the two users, not only the part that was travelled within a single cell. This means that the Trajectory Classifier does not has the same amount of information than the information that is depicted in the figure. The entire trajectories are drawn in order to show what happened to the users after the match and to verify the assumption that users that follow the same trajectory through a cell will likely have the same behaviour in the future. The start and end locations as well as the location where the user is steered are marked. The source cell, this is the cell whose SON function has matched the users and subsequently steered the active user, is marked in purple. The target cell, this is the cell to which the active user is steered, is marked in green. Other neighbouring cells are marked using various colours. Irrelevant cells are marked in black. Arrows are used to mark macro cells which have a directional antenna. The arrows point in the direction of the antenna. The diamonds mark the locations of the micro cells which have an isotropic antenna. The lengths and the radiuses of the arrows and circles respectively indicate the transmit power which is 46 dBm for all macro cells and 30 dBm for the micro cells. As can be seen, during the period right before the match, the reference user and the active user follow the same trajectory and the points where both users are at the time of the match coincide nearly exactly.

The plots on the right show the evolution of the RSRP and the SINR of both the reference user and the active user around the time of the match. The purple curve shows the evolution of the SeNB and the green curve shows the evolution of the TeNB. The other colours show the evolution of the other NeNB, the colours of the curves correspond to the colours of the cells on the map on the left. The parts
Figure 9.3: The user with the green trajectory is matched with the purple trajectory and is subsequently steered from the purple cell to the green cell.
of the traces that were matched by the Trajectory Classifier are marked using a red background. As can be seen, the evolution of both the $\text{RSRP}$ and $\text{SINR}$ of corresponding cells during this period (as well as some time before and after this period) is very similar. The green background indicates the minimum stay duration, this is the minimum amount of time right after the match during which a NeNB needs to have a better average spectral efficiency than the SeNB and should at no point in time have a lower spectral efficiency than the minimum spectral efficiency fraction of the spectral efficiency of the SeNB. The blue background indicates the remaining information that the SeNB has available from the reference trace. This means that at the end of the blue period the reference user either stopped its call or was handed over to another cell. The plot in the bottom right shows the information coming from the measurement reports of the reference user that is available at the SeNB and which it uses to make the traffic steering decision. Note that the periods that are spanned by the backgrounds is not necessarily the same for the reference user and the active user as the velocities of both users might differ. The Traffic Steerer is able to deal with these differences and match the users nevertheless as long as the differences are not too big.

Figure 9.4 shows a user that is wrongly steered due to a bad match. As can be seen the trajectories just cross each other but do not coincide for a substantially long period of time. This can happen when it is difficult for the Trajectory Classifier to discern between users that follow different trajectories because the measurements are very similar. In this case this is because the user is far away from the macro cells where the signal strengths do not change much. These cases however rarely occur and do not have a major impact on the performance.

Another situation where a bad traffic steering decision can be made is shown in Figure 9.5. In this case the trajectories of the reference user and active user split nearly immediately after a correct match. A situation like this cannot be avoided as the Trajectory Classifier has no information to predict that this will happen. The assumption that is made by the SON function is that these situations are rare and that users that follow the same trajectory in the recent past will continue to do so in the nearby future.

9.6 Evaluation

Unlike the Trajectory Classifier and the Trajectory Identifier the Traffic Steerer is not evaluated on its own. Instead the entire SON function consisting of the Trajectory Classifier, the Trajectory Identifier and the Traffic Steerer is evaluated as it is difficult to isolate the Traffic Steerer from the other two components. This means that the presented results do not only reflect the performance of the Traffic Steerer but also of the Trajectory Classifier and the Trajectory Identifier.
The user with the green trajectory is erroneously steered from the purple cell to the green cell based on an incorrect match with the purple reference trace.
Figure 9.5: The user with the green trajectory is erroneously steered from the purple cell to the green cell because the trajectories of the reference user and the active user partly match.
9.6. EVALUATION

9.6.1 Simulation Setup

The SON function is evaluated in two environments. The first one is the Hanover city centre which is indicated by the rectangle in Figure 6.2. This area is covered by both macro and micro cells. This means that cell sizes are small and handovers will occur frequently. The users consist of a mixture of slow moving and fast moving pedestrians. The SON function should be able to improve the performance significantly in this scenario by lowering the number of handovers and short stays.

The second scenario that is considered is a larger area outside the city centre that is only covered by macro cells. Users move through the simulation area at a moderate velocity. In this scenario the handover algorithm is able to adequately handover the users and it is difficult for the SON function to improve on the performance. The goal of this scenario is to check whether the SON function does not worsen the performance.

In both scenarios the SON function is active in all cells.

The parameters of the Trajectory Classifier and the Trajectory Identifier are set to the values that were determined in Chapter 7 and Chapter 8. The two parameters of the Traffic Steerer, namely the minimum stay duration and the minimum spectral efficiency fraction are varied in order to determine their influence on the performance of the SON function. Table 9.1 summarises the values of the parameters that are used in the performed simulations.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Trajectory Classifier</strong></td>
<td></td>
</tr>
<tr>
<td>Initial cost</td>
<td>0.05</td>
</tr>
<tr>
<td>Extra cost</td>
<td>0.02</td>
</tr>
<tr>
<td>Reliable match length</td>
<td>10</td>
</tr>
<tr>
<td>Event A4 thresholds</td>
<td>{-100 dBm, -95 dBm, . . . , -5 dBm}</td>
</tr>
<tr>
<td><strong>Trajectory Identifier</strong></td>
<td></td>
</tr>
<tr>
<td>Tagged user probability</td>
<td>0.1</td>
</tr>
<tr>
<td>Maximum reference set size</td>
<td>25</td>
</tr>
<tr>
<td>Relevance smoothing factor</td>
<td>0.1</td>
</tr>
<tr>
<td><strong>Traffic Steerer</strong></td>
<td></td>
</tr>
<tr>
<td>Minimum stay duration</td>
<td>10 s, 30 s, 60 s</td>
</tr>
<tr>
<td>Minimum spectral efficiency fraction</td>
<td>0.6, 0.8, 1</td>
</tr>
</tbody>
</table>

Each simulation lasts 20000 s and consists of two parts. The first part is the warm up period. During this period the reference sets of the cells are filled with reference traces and their relevance scores are set appropriately. The second part of the simulation is used for performance evaluation. Apart from not taking the warm up
period in account when calculating the performance there are no other differences between both parts.

For each parameter combination, 10 simulation runs are performed. The results of these simulations are averaged and the standard deviation is calculated.

The results of the simulations are compared to baselines which are run in the same scenarios. In these baselines only a multi-layer handover algorithm is active but not the \textit{SON} function. This handover algorithm handovers users based on signal strength using event A3 as is described in Section 6.4.

In order to assess the performance of the Traffic Steerer and the \textit{SON} function as a whole the following performance metrics are used:

\textbf{Number of Commanded Handovers} This is the total number of times a handover is performed. The handovers can both be triggered by the handover algorithm and by the \textit{SON} function. Although the immediate goal of the \textit{SON} function is to reduce the amount of short stays and not the amount of handovers, the number of handovers is also a good indication of how well the \textit{SON} function is able to achieve its goal.

\textbf{Number of Steered Users} This metric is only relevant in case the \textit{SON} function is active and represents the number of times a handover is triggered by the \textit{SON} function. The goal of this metric is not to assess the performance of the \textit{SON} function but merely to give an indication of which share of handovers is triggered by the \textit{SON} function.

\textbf{Short Stay Ratio} A short stay is defined as a user that only stays in a cell for a time that is less than some threshold (10 s) after it has performed a handover. The short stay ratio is defined as the number of handovers that result in a short stay divided by the total number of handovers.

\textbf{Number of Call Drops} This is the number of times a call drop occurs. Call drops occur when the \textit{UE} is no longer able to maintain connectivity and loses its connection with its \textit{SeNB}. Call drops often occur when handovers are triggered too late, i.e., when the user has already moved too far into a neighbouring cell and it is no longer able to maintain connectivity with its \textit{SeNB}.

\textbf{Number of Handover Failures} This is the number of times a commanded handover fails. A handover fails when the \textit{UE} cannot connect to the \textit{TeNB} after it has been commanded to handover to that \textit{TeNB}. Handover failures typically occur when handovers are triggered too early and it is not yet possible to connect to the \textit{TeNB}. Note that a handover failure does not necessarily results in a call drop as a user can try to reconnect to its original \textit{SeNB}.

\textbf{Number of Ping-pong Handovers} A ping-pong handover is defined as a user that is handed over from cell \textit{A} to cell \textit{B} and back to cell \textit{A} within a short amount of time (5 s). Ping-pong handovers are typically caused when handovers are triggered too eagerly inside a small area of coverage of a \textit{NeNB}.

Ping-pong handovers have the same negative impact as short stays (as a matter of fact ping-pong handovers are short stays) as they waste network resources and cause unnecessary service interruption.

**Mean Spectral Efficiency** The spectral efficiency is the amount of data that can be sent to a user per unit of time and per unit of bandwidth. It is expressed in bit/s/Hz and depends on the **MCS** that is used to send data to the user and thus ultimately on the **SINR** of the user. It is a measure of how efficiently data can be sent to a user and how many resources are spent for sending a certain amount of data to it. A high spectral efficiency means that with only little resources much data can be transmitted while a low spectral efficiency means that much resources are needed to transmit little data. This metric takes into account the data outage during handover. This means that while a user is making a handover its spectral efficiency is taken to be zero. Note that this spectral efficiency value is not the estimation that is presented in Section 9.3 but the exact value.

### 9.6.2 Results

Table 9.2 shows the results of the simulations that were performed in both scenarios for various combinations of the minimum stay duration and minimum spectral efficiency fraction parameters as well as for their baselines. The results are the average of 10 simulation runs with different random seeds. The standard deviations that are associated with the values are shown in Table 9.3. As can be seen the standard deviations contain no exceptionally high values.

When looking at the city centre scenario both the short stay ratio and the number of handovers are significantly lower when the **SON** function is enabled than in the baseline as would be expected. At the same time the mean spectral efficiency is only marginally lower when the **SON** function is enabled than in the baseline. By steering users more intelligently based on their future behaviour many unnecessary handovers are avoided and only handovers that are required to maintain connectivity and a good **QoS** are performed. This is also reflected in the number of ping-pong handovers which is lower when the **SON** function is enabled than in the baseline. The number of call drops and the number of handover failures are in both cases extremely low. As there are two different layers it is not very difficult to keep the user connected and to find a suitable target cell for it as there usually are a large number of target cells to handover a user to which do not all interfere with each other.

When looking at the different parameter combinations of the Traffic Steerer two trends can be seen. When the minimum stay duration increases the number of steered users decreases. This is to be expected as increasing the minimum stay duration makes the conditions for a user being steered to a **NeNB** more stringent which leads to fewer users being steered. When calculating the number of handovers that are triggered by the handover algorithm by subtracting the number of steered users from the total number of handovers it can be seen that this number
### Table 9.2: The results of the simulations that were performed for various combinations of the minimum stay duration and minimum spectral efficiency fraction parameters as well as for their baselines. The presented values are the average values of 10 simulation runs.

<table>
<thead>
<tr>
<th>Minimum stay duration (s)</th>
<th>0</th>
<th>0.6</th>
<th>0.8</th>
<th>1</th>
<th>1.0</th>
<th>1.2</th>
<th>1.4</th>
<th>1.6</th>
<th>1.8</th>
<th>2</th>
<th>2.2</th>
<th>2.4</th>
<th>2.6</th>
<th>2.8</th>
<th>3</th>
<th>3.2</th>
<th>3.5</th>
<th>3.7</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hanover city centre</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of commanded handovers</td>
<td>33</td>
<td>3.63</td>
<td>3.62</td>
<td>3.66</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td></td>
</tr>
<tr>
<td>Number of handover failures</td>
<td>2</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td></td>
</tr>
<tr>
<td>Number of ping-pong handovers</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td>Mean Spectral Efficiency (bit/s/Hz)</td>
<td>2.48</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td></td>
</tr>
<tr>
<td>Suburban</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of commanded handovers</td>
<td>33</td>
<td>3.63</td>
<td>3.62</td>
<td>3.66</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td>3.65</td>
<td></td>
</tr>
<tr>
<td>Number of handover failures</td>
<td>2</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td>448</td>
<td></td>
</tr>
<tr>
<td>Number of ping-pong handovers</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td>Mean Spectral Efficiency (bit/s/Hz)</td>
<td>2.48</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td>2.49</td>
<td></td>
</tr>
</tbody>
</table>

Thus, Table 9.2 shows the results of the simulations that were performed for various combinations of the minimum stay duration and minimum spectral efficiency fraction parameters as well as for their baselines. The presented values are the average values of 10 simulation runs.
Table 9.3: The standard deviation of the results that are shown in Table 9.2.

<table>
<thead>
<tr>
<th>Minimum stay duration (s)</th>
<th>10</th>
<th>10</th>
<th>10</th>
<th>30</th>
<th>30</th>
<th>30</th>
<th>60</th>
<th>60</th>
<th>60</th>
<th>Baseline</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum spectral efficiency fraction</td>
<td>0.6</td>
<td>0.8</td>
<td>1</td>
<td>0.6</td>
<td>0.8</td>
<td>1</td>
<td>0.6</td>
<td>0.8</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Hanover city centre

| | Minimum stay duration (s) | 10 | 10 | 10 | 30 | 30 | 30 | 60 | 60 | 60 | Baseline |
|---------------------------|----|----|----|----|----|----|----|----|----|----------|
| Minimum spectral efficiency fraction | 0.6 | 0.8 | 1 | 0.6 | 0.8 | 1 | 0.6 | 0.8 | 1 |

| Number of commanded handovers | 110 | 130 | 122 | 113 | 147 | 105 | 107 | 122 | 64 | 133 |
| Number of steered users | 48 | 92 | 81 | 71 | 51 | 49 | 35 | 39 | 16 | – |
| Short stay ratio | 0.003 | 0.002 | 0.003 | 0.002 | 0.003 | 0.003 | 0.002 | 0.003 | 0.002 | 0.001 |
| Number of call drops | 5 | 3 | 2 | 2 | 6 | 2 | 3 | 3 | 4 | 0 |
| Number of handover failures | 0 | 0 | 0.3 | 0.6 | 0.3 | 0.4 | 0.3 | 0 | 0 | 0 |
| Number of ping-pong handovers | 5 | 10 | 6 | 4 | 4 | 5 | 3 | 2 | 2 | 9 |
| Mean Spectral Efficiency (bit/s/Hz) | 0.01 | 0.02 | 0.006 | 0.01 | 0.01 | 0.02 | 0.01 | 0.02 | 0.01 | 0.01 |

Suburban

| | Minimum stay duration (s) | 10 | 10 | 10 | 30 | 30 | 30 | 60 | 60 | 60 | Baseline |
|---------------------------|----|----|----|----|----|----|----|----|----|----------|
| Minimum spectral efficiency fraction | 0.6 | 0.8 | 1 | 0.6 | 0.8 | 1 | 0.6 | 0.8 | 1 |

| Number of commanded handovers | 119 | 86 | 111 | 164 | 123 | 120 | 98 | 105 | 120 | 131 |
| Number of steered users | 35 | 20 | 14 | 25 | 26 | 27 | 6 | 13 | 16 | – |
| Short stay ratio | 0.005 | 0.005 | 0.005 | 0.006 | 0.005 | 0.005 | 0.006 | 0.004 | 0.005 | 0.005 |
| Number of call drops | 10 | 7 | 11 | 7 | 7 | 10 | 8 | 12 | 3 |
| Number of handover failures | 5 | 6 | 4 | 4 | 7 | 5 | 4 | 6 | 4 | 0 |
| Number of ping-pong handovers | 7 | 9 | 7 | 9 | 6 | 6 | 3 | 6 | 3 | 5 |
| Mean Spectral Efficiency (bit/s/Hz) | 0.03 | 0.04 | 0.02 | 0.03 | 0.03 | 0.03 | 0.04 | 0.02 | 0.03 | 0.05 |
CHAPTER 9. TRAFFIC STEERING

only increases slightly as the number of steered users decreases. When fewer users are steered there will be slightly more cases where the intervention of the handover algorithm is required to maintain connectivity. The SON function however avoids most unnecessary handovers. The same trend can be observed when the minimum spectral efficiency fraction increases albeit less clearly. The minimum spectral efficiency fraction has a lower impact on the number of steered users as the condition that users should achieve a higher average spectral efficiency in the target cell (condition 3) usually also implies that the spectral efficiency in the target cell is at all times close to (or even higher than) that of the serving cell.

When looking at the results for the suburban scenario, the results that are obtained from the scenarios where the SON function is enabled are either slightly worse or slightly better than the baseline, depending on the parameters of the Traffic Steerer. This is because in this scenario, users move at a relatively low speed and there are only macro cells which limits the possibilities for steering users as there is often only a single choice for handing over the user. The SON function does however not worsen the situation much. It might however be beneficial to no apply it in cells that are in an environment where there is no real reason to enable it as applying the SON function introduces some computational overhead.

9.7 Conclusion

This chapter presented the Traffic Steerer and evaluated the developed SON function as a whole. The Traffic Steerer is the component of the SON function that is responsible for making a traffic steering decision once a sufficiently reliable match of the trajectory of a currently active user with a reference trace has been made by the Trajectory Classifier. By assuming that events that occur to one user will also occur to another user that follows a similar trajectory through the cell, the Traffic Steerer can decide whether it is beneficial to hand over the user to another cell or to keep it in the current cell. Before the Traffic Steerer decides to steer the user to another cell, or not, it will extrapolate and project events that occurred to the reference user in the period after the match on the active user, in order to make an estimate of the future achievable spectral efficiency of the active user in each of the neighbouring cells as well as in the serving cell itself. Based on these predictions of the future achievable spectral efficiency of the UE in the serving cell as well as in neighbouring cells, a traffic steering decision is made.

Instead of evaluating the Traffic Steerer component on its own like with the Trajectory Classifier and the Trajectory Identifier, the developed SON function was evaluated as a whole. In order to do this simulations were performed where the SON function was deployed in all the cells in the simulation area and simulating the operation of the network. Two scenarios were studied. The first one is the Hanover city centre, an area covered by both macro and micro cells where handovers occur frequently. The second one is a suburban scenario where there are only macro cells.
Results show that the SON function is able to reduce the amount of short stays by more than 30% in the scenario where there are frequent handovers. At the same time the spectral efficiency is only little impacted and the number of call drops and handover failures remains low. In the suburban scenario the SON function is not able to improve the amount of short stays and the number of handovers as these are already low. The results obtained from this scenario however show that the SON function in this case has not negative impact on the performance of the network.
Chapter 10

Extending to Multiple RATs

10.1 Introduction

This section describes how the developed SON function could be extended to multiple RATs. As an example UMTS will be used although it is also possible to extend the SON function to other RATs.

In order to extend the developed SON function to multiple RATs, two changes need to be made to the existing SON function. Firstly, as the Traffic Steerer needs to compare radio conditions in various cells, a measure to compare radio conditions between cells that belong to various RATs is required. In Chapter 9 spectral efficiency was used for comparing the conditions in various cells. In the case of multiple RATs an indication of the actual user throughput has to be used. Secondly, the measurements that are used by the MDTW algorithm have to be extended to additional RATs. Both changes to the SON function are discussed below.

10.2 Multi-RAT Throughput Estimation

The Traffic Steerer will make a decision based on the predicted future radio conditions that can be achieved by the user in the cells in its vicinity. In the single RAT LTE case spectral efficiency was used as an indication for the radio conditions. In the multi-RAT case this is however no longer favourable as it is not possible to compare the spectral efficiency between different RATs. Instead, a throughput measure shall be used that allows comparing the predicted future radio conditions between different RATs. This however requires taking the decisions of a scheduler into account, which in turn requires knowledge of different cell properties, notably the number of active users. In order to overcome this problem, a simple scheduling
policy is assumed that distributes the available resources based on the number of users. By assuming that calls arrive according to a Poisson process and the call duration is exponentially distributed, the distribution of the number of simultaneously active users is the same as the number of customers in the $M/M/\infty$ queuing system [53]:

$$\pi_N = \frac{\rho^N e^{-\rho}}{N!}$$  \hspace{1cm} (10.1)

where $\pi_N$ is the probability that a cell is serving exactly $N$ users and $\rho$ is the ratio between the mean call duration and the mean time between call arrivals. This value will be a parameter of the SON function. Using this distribution the average expected throughput that will be achieved by a user in the neighbouring cells is estimated by:

$$E_i = \sum_{N=0}^{\infty} \pi_N T_i (N + 1)$$  \hspace{1cm} (10.2)

where $E_i$ is the expected throughput (expressed in bit/s) in cell $i$ and $T_i (n)$ calculates the average throughput that can be achieved by a user in cell $i$ when there are $n$ active users in that cell. This expected throughput, which can be calculated for the different [RATs], can then be used to compare expected throughput in cells of different [RATs]. In case of an LTE cell, assuming that resources are assigned equally among users, the throughput $T_i (n)$ of a user that can achieve a spectral efficiency $S_i$ in cell $i$ is given by:

$$T_i (n) = \frac{S_i \Delta f_i}{n}$$  \hspace{1cm} (10.3)

where $\Delta f_i$ is the bandwidth that is available in cell $i$. This means that the expected throughput that can be achieved in cell $i$ is given by:

$$E_i = \sum_{N=0}^{\infty} \frac{\rho^N e^{-\rho} S_i \Delta f_i}{N!} \frac{S_i \Delta f_i}{N + 1}$$  \hspace{1cm} (10.4)

$$E_i = \frac{e^{-\rho} S_i \Delta f_i}{\rho} \sum_{N=0}^{\infty} \frac{\rho^{N+1}}{(N + 1)!}$$  \hspace{1cm} (10.5)

$$E_i = \frac{e^{-\rho} S_i \Delta f_i}{\rho} \sum_{N=1}^{\infty} \frac{\rho^N}{N!}$$  \hspace{1cm} (10.6)

$$E_i = \frac{e^{-\rho} S_i \Delta f_i}{\rho} \left( \sum_{N=0}^{\infty} \frac{\rho^N}{N!} - 1 \right)$$  \hspace{1cm} (10.7)

$$E_i = \frac{S_i \Delta f_i}{\rho} (1 - e^{-\rho})$$  \hspace{1cm} (10.8)

In case of UMTS the SINR and the number of users $n$ that have to be served in cell $i$ will be considered as limiting factors for the throughput $T_i (n)$ that can be achieved.

UMTS users are assigned orthogonal spreading codes $c_1, c_2, \ldots, c_L$ where each $c_i$ is either +1 or −1. The length of these spreading codes $L$, called the Spreading
Factor \( SF \) is a power of 2 between 4 and 512 inclusive. When transmitting a sequence of bits \( b_1, b_2, \ldots \), where a 0 is represented by \(-1\) and a 1 is represented by \(+1\), each bit is spread by multiplying it with the spreading code of the sender and replacing it by the obtained result: 
\[ b_1 \cdot c_1, b_1 \cdot c_2, \ldots, b_1 \cdot c_L, b_2 \cdot c_1, b_2 \cdot c_2, \ldots, b_2 \cdot c_L, \ldots \]
The time to transmit a single chip \( c_i \) is constant and equal to \( T_c \). This means that the time \( T_b \) to transmit a single bit is equal to \( LT_c \).

At the receiver’s end the signal which is a superposition of the sender’s signal and a number of signals coming from interferers is despread by multiplying it again with the spreading code of the sender. Despreading the received signal will result in the superposition of the sender’s signal and the signals of interferers that were despread with the spreading code of the sender. The despread signal of the sender will have a frequency of \( \frac{1}{T_b} = \frac{1}{LT_c} \). Despreading a signal with a spreading code that is different than the one that was use to spread it will however result in a signal with a frequency that is much higher than when despreading a spread signal with the spreading code that was used to spread it. The despread signal of the interferers will therefore have a frequency of \( \frac{1}{T_c} \). The process of spreading and despreading is illustrated in Figure 10.1.

\[
\begin{align*}
\text{Sender Signal} & \quad +1 & -1 & +1 \\
\text{Sender Spreading Code} & \equiv +1 +1 +1 +1 +1 +1 -1 -1 -1 \\
\text{Spread Sender Signal} & \equiv +1 +1 -1 -1 -1 +1 +1 +1 -1 -1 \\
\text{Interferer Signal} & \equiv -1 & +1 & +1 \\
\text{Interferer Spreading Code} & \equiv +1 +1 +1 +1 +1 +1 -1 -1 -1 -1 \\
\text{Spread Interferer Signal} & \equiv +1 +1 -1 -1 -1 +1 +1 +1 -1 -1 \\
\text{Sender Spreading Code} & \equiv +1 +1 -1 +1 +1 +1 -1 +1 -1 -1 \\
\text{Decoded Sender Signal} & \equiv +1 & -1 & +1 \\
\text{Decoded Interferer Signal} & \equiv -1 +1 +1 -1 -1 +1 +1 -1 +1 +1 \\
\end{align*}
\]

Figure 10.1: The spreading and despreading of the signals of the sender and an interferer.

This means that in the frequency domain the power of the signal is spread out over a bandwidth of \( \frac{1}{T_b} = \frac{1}{LT_c} \) while the power of the interferers is spread out over a larger bandwidth of \( \frac{1}{T_c} \). By applying a band-pass filter to filter out the interference only \( \frac{1}{L} \) of the power of the interferers remains as is illustrated in Figure 10.2. If the minimum \( \text{SINR} \) that is required to decode received data is given by \( \text{SINR}_{\text{min}} \) then the minimum length of a spreading code in order to still be able to distinguish the
signal of the interferer should be:

\[ 2^{\left\lfloor (\log_{10} \frac{\text{SINR}_{\text{min}} - \text{SINR}}{10}) \right\rfloor} \]  

(10.9)

The minimum length of a spreading code is also limited by the number of active users as each user has to be assigned an orthogonal code. In the following it is assumed that all users are assigned spreading codes with the same length \( L \). If spreading codes have length \( L \), at most \( L \) users can be served at a time. This means that if there are \( N \) active users in the cell the minimum length of the spreading code is given by:

\[ 2^{\left\lfloor \log_{2} N \right\rfloor} \]  

(10.10)

If the minimum length of the spreading code is greater than 512, no connection is possible. The length of the shortest possible spreading code in cell \( i \) given the number of users \( n \) is then given by:

\[ L_i(n) = \max \left( 4, 2^{\left\lfloor (\log_{2} 10)^{\frac{\text{SINR}_{\text{min}} - \text{SINR}}{10}} \right\rfloor}, 2^{\left\lfloor \log_{2} n \right\rfloor} \right) \]  

(10.11)

As the Chip Rate \( \text{CR} \), i.e., the number of spreading code elements that can be sent per unit of time, is constant and equal to 3.84 Mc/s, the throughput of a user with a given SINR in case there are \( n \) users is:

\[ T_i(n) = \frac{\text{CR}}{L_i(n)} \]  

(10.12)

When calculating \( E_i \), there will be a point at which the limiting factor of the throughput switches from being the SINR to being the number of users. This
point, called SF\(_{\text{min}}\) is given by the following equation:

\[
SF_{\text{min}} = \max \left( 4, 2^{\left( \log_{10} 10 \right) \frac{\text{SINR}_{\text{min}}}{\text{SINR}}} \right)
\]  

(10.13)

The expected throughput is then given by:

\[
E_i = \begin{cases} 
0 & \text{if } SF_{\text{min}} > 512 \\
\text{CRe}^{-\rho} \left( \frac{1}{SF_{\text{min}}} \sum_{N=0}^{SF_{\text{min}}-1} \frac{\rho^N}{N!} + \sum_{N=SF_{\text{min}}}^{511} \frac{\rho^N}{N!} 2^{\left( \log_2 (N+1) \right)} \right) & \text{otherwise}
\end{cases}
\]

(10.14)

### 10.3 UMTS Measurements

In order to identify users that follow similar trajectories through a cell in the multilayer LTE case, the Trajectory Classifier used measurements that come from measurement reports that were triggered by event A4s. An event A4 compares the RSRP/RSRQ of each NeNB to a fixed threshold. When the RSRP/RSRQ of the NeNB rises above the fixed threshold plus a hysteresis value for a time called the TTT an event A4 is triggered. When the RSRP/RSRQ later drops below this threshold minus the same hysteresis value for the same TTT time, the event A4 is untriggered. When such an event is triggered, a measurement report that contains the NeNBs for which the event is triggered is sent to the SeNB. By configuring a number of event A4s with different thresholds, the maximum value of the RSRP/RSRQ of the NeNBs can be known at the SeNB within the bounds of the thresholds for which event A4s have been configured. The SeNB used this information to classify users according to the trajectory they follow through the cell and to steer them accordingly.

When extending the SON function to UMTS similar information is needed for the neighbouring NodeBs. For this purpose events B1 can be used. Event B1 is very similar to the event A4: the entering condition of event B1 holds when the RSCP or Received energy per chip divided by the power density in the band \(E_c/N_0\) value rises above a predefined threshold plus a hysteresis value. When the entering condition holds for an amount of time that is equal to the TTT the event is triggered. Likewise, the leaving condition of event B1 holds when the RSCP or \(E_c/N_0\) value drops below the same predefined threshold minus the hysteresis and the event is untriggered when this condition holds for the same TTT. Each time the event is triggered a measurement report is sent. As with the event A4s, a number of events B1 are configured each with different thresholds whose values are separated by the same threshold. By doing this the maximum RSCP or \(E_c/N_0\) of the neighbouring NodeBs can be known within the bounds that are determined by the thresholds. From this point on the data can be intermixed with the data coming from the event A4s and further processed in the same way.
10.4 Evaluation

The multi-RAT extensions of the SON function were also evaluated. These were done in a 5 km by 2.5 km area located near the north of the Hannover scenario where a highway passes through the scenario as is shown in Figure 10.3. This region is a suburban area which is crossed both by a highway and a high-speed railway. The scenario contains a mixture of users moving at a high velocity on the highway or in a high-speed train as well as users that move slowly through the residential areas in the covered region. These users are either static users, pedestrians or vehicular users travelling at moderate velocities. This mixture of high velocity and low velocity users allows assessing the capabilities of the SON function to distinguish between users that perform frequent handovers and users that do not. The scenario contains two RATs: LTE-1800 and UMTS-2100.

Table 10.1 shows the simulation results both for the simulation where the SON function is enabled and where it is not enabled. Per simulation, results are shown separately for handovers that are triggered by the SON function and handovers that are triggered by the Handover algorithm as well as handovers that are triggered by either the SON function or the Handover algorithm. The results show that the SON function triggers fewer handovers than the Handover algorithm (when comparing the Number of Handovers by the SON function, column 1 row 1, with the Number of Handovers by the Handover algorithm, column 1 row 2, for the simulation where the SON function is enabled). The reason for this is twofold, first of all, the SON function only steers users when it has sufficient information about a user to decide whether it is beneficial to hand it over to another cell. All other handovers are left to the Handover algorithm. Furthermore, the SON function is more conservative in triggering handovers, choosing when to handover a user more carefully. The total amount of handovers that are triggered during the simulation when the SON function is enabled is however also lower in comparison to the simulation where the SON function is not enabled (compare the total Number of Handovers in the simulation where the SON function is enabled, column 1 row 3, with the total Number of Handovers, which is the same as the Number of Handovers that are triggered by the Handover algorithm, column 2 row 3, in case the SON function is not enabled). This shows the ability of the SON function in reducing the number of handovers.
Figure 10.3: The location of the scenario within the area that is covered by the Hanover scenario data is indicated by a rectangle.
Table 10.1: The results of the multi-RAT simulations.

<table>
<thead>
<tr>
<th>Metric</th>
<th>HM SON enabled</th>
<th>HM SON not enabled</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Handovers (HM)</td>
<td>32</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>Number of Handovers (HO)</td>
<td>285</td>
<td>398</td>
</tr>
<tr>
<td>Number of Handovers (total)</td>
<td>317</td>
<td>(same as HO)</td>
</tr>
<tr>
<td>Short Stay Ratio (HM)</td>
<td>12.5%</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>Short Stay Ratio (HO)</td>
<td>31.6%</td>
<td>34.9%</td>
</tr>
<tr>
<td>Short Stay Ratio (total)</td>
<td>29.7%</td>
<td>(same as HO)</td>
</tr>
<tr>
<td>Handover Failure Ratio (HM)</td>
<td>0%</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>Handover Failure Ratio (HO)</td>
<td>29.8%</td>
<td>29.8%</td>
</tr>
<tr>
<td>Handover Failure Ratio (total)</td>
<td>26.1%</td>
<td>(same as HO)</td>
</tr>
</tbody>
</table>
Chapter 11

Conclusion

This thesis presented a SON function that aims to reduce the amount of short stays in case frequent handovers occur due to a dense deployment of cells or when the user velocity is high. This SON function consists of a number of components, namely the Trajectory Classifier, the Trajectory Identifier and the Traffic Steerer. Each of these components and the algorithms behind them were explained in detail. The different components of the developed SON function as well as the SON function as a whole have been evaluated using simulations.

The Trajectory Classifier, the component that is at the heart of the developed SON function, is responsible for classifying users according to the trajectory they follow through the cell. It does this by comparing measurement traces that are made by currently active users to measurement traces that were made by users that were active in the past, and as such identifying matching traces of measurements. The Trajectory Classifier was evaluated by making it classify a number of traces and comparing the results to the geographical correspondence of the traces. Results show that the Trajectory Classifier is able to correctly identify whether two trajectories match or not in 80–85% of cases. Furthermore the environment has an influence on the ability of the Trajectory Classifier to correctly distinguish between matching and non-matching trajectories. This influence is however marginal and the results are in any case very accurate.

The Trajectory Identifier is responsible for identifying the traces that will serve as reference traces. This means selecting new reference traces as well as removing old, obsolete reference traces. The Trajectory Identifier was evaluated by assessing how well the resulting reference set after a simulation covered the frequently used trajectories through a geographical area that is covered by a base station. Furthermore the ability of the Trajectory Identifier to adapt to changes in the environment was assessed. Results show that the Trajectory Identifier is able to select relevant reference traces and is able to adapt to changes in its environment.
Finally, the Traffic Steerer is responsible for making a traffic steering decision once a sufficiently reliable match of the trajectory of a currently active user with a reference trace has been made by the Trajectory Classifier. It does this by mapping the events that occurred to the reference user after the period of the match on the active user and, based on this data, deciding whether it is beneficial to steer the user to another cell or to keep it in the current cell. The Traffic Steerer was evaluated in combination with the other components of the SON function. Results show that the SON function is able to reduce the amount of short stays by more than 30% in scenarios where there are frequent handovers.

11.1 Future Work

The SON function described in this thesis is already extensive. Therefore care should be taken when adding additional functionality to the SON function as it would only make it more complex. There are however a number of things that might improve the decisions made by it.

An important addition might be taking information from other cells into account. Currently all information that is used by the SON function to base its decisions on is gathered within a single cell. The SON function could however be extended to take information about what happens to a user after it has been steered to a certain target into account. For this eNBs should share this information which requires an extension of the current communication abilities of eNBs.

Furthermore the information that is gathered by the SON function could be used for more than simply steering users. The information could for instance be used for taking AC decisions by letting it predict the future resource requirements of users or for load balancing.

Another subject for further investigation is the interaction between the SON function and other SON functions and processes that are active in the network. One type of process that might have an effect on the SON function are SON functions that change various parameters which influence the measurements that are used by the SON function like tilt and transmit power. The SON function should be able to deal with small changes in the measurements that are made by the users as the MDTW algorithm is designed to deal with minor differences in the measurements. Furthermore as the reference traces are continuously renewed the SON function should also be able to deal with long-term changes. The impact of these changes could however be studied to assess their actual impact on the performance of the developed SON function.

There might furthermore be conflicts between the SON function and other network functions that steer users between cells in order to achieve a certain goal like MLB functions that steer users in order to reduce the load in heavily loaded cells. Actions taken by these SON functions might conflict with the actions taken by the developed
SON function. The impact of these SON functions on the developed SON function could be studied and methods for resolving conflicts could be developed.
Samenvatting

Wanneer in cellulaire netwerken cellen klein zijn en/of gebruikers aan hoge snelheid bewegen kunnen frequente handovers optreden. Deze frequente handovers kunnen een ernstige verslechtering van gebruikerservaring en netwerkprestaties veroorzaken zoals een vermindere Quality-of-Service (QoS) door een relatief hoge periode waarin geen communicatie mogelijk is ten opzichte van de cel verblijftijd, een verhoogd aantal call drops en een verhoogde signalering- en dataoverhead in het kernnetwerk. In dit proefschrift wordt een zelf-organiserende netwerkfunctie (SON) ontwikkeld die dit probleem aanpakt. Het doel van deze SON-functie is om gebruikers in te delen op basis van hun mobiliteitsgedrag en ze, op basis van deze indeling, op een intelligente manier te sturen zodat de handoverfrequentie wordt verlaagd terwijl de QoS wordt behouden of misschien zelfs verbeterd. Door aan te nemen dat de gebeurtenissen die zich voordoen bij de ene gebruiker ook zullen voorkomen bij andere gebruikers die hetzelfde traject volgen door een cel, kan de SON-functie beslissen of het nuttig is om gebruikers te handoveren naar een andere cel of in de huidige cel te houden. Om dit te doen zal de SON-functie metingen van gebruikers verzamelen die toelaten gebruikers die een soortgelijk traject door een cel volgen met elkaar te matchen. De SON-functie zal metingen van actieve gebruikers matchen met metingen van gebruikers die in het verleden actief waren. Wanneer er een overeenkomst wordt gevonden tussen twee gebruikers neemt de SON-functie aan dat het toekomstige gedrag van de actieve gebruiker vergelijkbaar is met het gedrag van de overeenkomende gebruiker die in het verleden actief was. Op basis van de informatie die werd verzameld over de gebruiker die actief was in het verleden bepaalt de SON-functie het toekomstige gedrag van de actieve gebruiker en beslist ze over de wijze waarop de gebruiker gestuurd zal worden.

De SON-functie bestaat uit drie onderdelen, namelijk De Trajectory Classifier, de Trajectory Identifier en de Traffic Steerer. De Trajectory Classifier is verantwoordelijk voor het classificeren van gebruikers volgens het traject dat ze volgen door de cel. Hij doet dit door metingen die zijn gemaakt door actieve gebruikers (actieve metingen) te vergelijken met meting die werden gemaakt door gebruikers die actief waren in het verleden (referentiemetingen) en overeenkomstige metingen te matchen. Hiervoor wordt een algoritme gebruikt dat is gebaseerd op het bekende Dynamic Time Warping (DTW) algoritme. De Trajectory Identifier is verantwoordelijk voor het bepalen van welke metingen als referentiemetingen zullen dienen. Dit betekent het selecteren van nieuwe referentiemetingen, alsook het verwijderen van
oude, achterhaalde referentiemetingen. De *Traffic Steerer* is verantwoordelijk voor het beslissen of en wanneer een gebruiker van de ene naar de andere cel gestuurd wordt eenmaal een voldoende betrouwbare match van de metingen van de actieve gebruiker met een referentiemetingen gevonden wordt door de Trajectory Classifier.

De verschillende componenten van de ontwikkelde SON-functie alsook de SON-functie in zijn geheel werden geëvalueerd door middel van simulaties. Resultaten tonen aan dat de SON-functie in staat is om het aantal korte verblijven in een cell significant te reduceren.
Summary

In cellular networks, in case of a dense deployment of cells and/or when user velocities are high, frequent handovers will occur. These frequent handovers might cause a serious degradation of user experience and network performance like a reduced QoS due to a relatively high data outage time in comparison to the cell stay time, an increased number of call drops and an increased signalling and data overhead in the core network. In this thesis a SON function is developed that deals with this problem. The SON function classifies users according to their mobility behaviour and, based on this classification, steer them in an intelligent way such that the number of short stays are reduced while the QoS is maintained or possibly even improved. By assuming that events that occur to one user will also occur to another user that follows a similar trajectory through a cell, the SON function can decide whether it is beneficial to hand over the user to another cell or to keep it in the current cell. In order to do this, the SON function will collect measurements from users that allow it to match users that follow similar trajectories through a cell. It will match the measurements from currently active users with measurements from users that were active in the past. When it finds such a match between two users it will assume that the future behaviour of the currently active user in the period after the match will be similar to the behaviour of the user that was active in the past. Based on the information that was collected by the user that was active in the past the SON algorithm estimates the future behaviour of the currently active user and makes decisions on how to steer the user.

The developed SON function consists of three components, namely the Trajectory Classifier, the Trajectory Identifier and the Traffic Steerer. The Trajectory Classifier is responsible for classifying users according to the trajectory they follow through the cell. It does this by comparing measurement traces that are made by currently active users (active traces) to measurement traces that were made by users that were active in the past (reference traces), and as such identifying matching traces of measurements. For the matching, an algorithm that is based on the well known DTW algorithm is used. The Trajectory Identifier is responsible for identifying the traces that will serve as reference traces. This means selecting new reference traces as well as removing old, obsolete reference traces. The Traffic Steerer is responsible for making a traffic steering decision once a sufficiently reliable match of the trajectory of a currently active user with a reference trace has been made by the Trajectory Classifier.
The different components of the developed SON function as well as the SON function as a whole have been evaluated using simulations. Results show that the SON function is able to reduce the amount of short stays significantly in scenarios where there are frequent handovers.
Bibliography


[37] 3GPP. UTRA-UTRAN Long Term Evolution (LTE) and 3GPP System Architecture Evolution (SAE), October 2006.


