CHALMERS
UNIVERSITY OF TECHNOLOGY

UNIVERSITY OF GOTHENBURG

Instantaneous Bandwidth Approximation in 5G Networks

Master's thesis in Computer science and engineering

ALBIN ASPLJUNG
UZAIR AHMAD

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2024

Master's thesis 2024

Instantaneous Bandwidth Approximation in 5G Networks

ALBIN ASPLJUNG
UZAIR AHMAD

UNIVERSITY OF
GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2024

Instantaneous Bandwidth Approximation in 5G Networks
ALBIN ASPLJUNG
UZAIR AHMAD

© ALBIN ASPLJUNG, UZAIR AHMAD, 2024.

Supervisor: Romaric Duvignau, Department of Computer Science and Engineering
Advisor: Magnus Castell, Johan Nicander, Jörgen Sjögren, Ericsson
Examiner: Gerardo Schneider, Department of Computer Science and Engineering

Master's Thesis 2024
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Gothenburg, Sweden 2024

Instantaneous Bandwidth Approximation in 5G Networks
ALBIN ASPLJUNG
UZAIR AHMAD
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

The estimation of available bandwidth in real-time is a challenging problem which has resulted in various proposed solutions for various network architectures. Because network conditions can change quickly, it can be difficult to estimate the available bandwidth without measuring it. Instead of directly measuring data speeds, which can use up the network's capacity and be slow, another approach is to use counters that exist in 5G base stations. These counters track network activity and offer a simpler way to estimate bandwidth without putting extra load on the system.

This thesis aims to develop a model for estimating bandwidth in base stations within 5G networks. It also explores the selection of appropriate counters for this purpose, examines potential correlations between them, and investigates how various traffic scenarios impact the accuracy of bandwidth estimation. A quantitative methodology is used to this end. There has been previous research done on the topic of bandwidth estimation and prediction, however not as much has been done using counters as input.

The results show that certain counters related to data volumes and resource block symbols appear to be particularly well suited for estimating bandwidth. The correlations themselves suffice to be simple linear relationships and are for the most cases accurate. The models estimate differently well for different traffic scenarios. The scenarios with two devices transmitting data overall have higher error rates than comparing to the one device scenarios.

Keywords: 5G, Base station, Bandwidth, Data Throughput, Network Traffic, Bandwidth Estimation.

Acknowledgements

We want to thank Romaric Duvignau, who, as our academic supervisor, helped guide us through the thesis as well as giving us insightful input into our work.

We also want to thank Johan Nicander as well as the Expericom team consisting of Magnus Castell, Jörgen Sjögren, Annika Ericson from Ericsson for helping us solving practical problems and giving us access to tools and equipment necessary to drive our thesis forward.

Albin Aspljung, Uzair Ahmad, Gothenburg, 2024-10-24

Abbreviations

5G

3GPP 3rd Generation Partnership Project

5QI 5G Quality of Service Identifier

CU Central Unit

DRB Data Radio Bearer

DTX Discontinuous Transmission

DU Distributed Unit

EPS Evolved Packet System

FDD Frequency Division Duplex

FR1 Frequency Range 1

FR2 Frequency Range 2

HARQ Hybrid Automatic Repeat Request

LTE Long Term Evolution

NSA Non-Standalone

OFDM Orthogonal Frequency-Division Multiplexing

PCell Primary Cell

PDCCH Physical Downlink Control Channel

PDSCH Physical Downlink Shared Channel

PDU Protocol Data Unit

PSCell Primary Secondary Cell

PUSCH Physical Uplink Shared Channel

QAM Quadrature Amplitude Modulation

QPSK Quadrature Phase Shift Keying

RAN Radio Access Network

RSRP Reference Signal Received Power

RSRQ Reference Signal Received Quality

SA Standalone

SCell Secondary Cell

SDU Service Data Unit

TDD Time Division Duplex

UE User Equipment

Machine Learning

LSTM Long Short-Term Memory

RNN Recurrent Neural Network

SVM Support Vector Machine

Other Abbreviations

EWMA Exponentially Weighted Moving Average

MAC Medium Access Control

MANET Mobile Ad hoc Network

MRE Mean Relative Error

QoS Quality of Service

QUIC Quick UDP Internet Connection

SCTP Stream Control Transmission Protocol

SMA Simple Moving Average

TCP Tranmission Control Protocol

UDP User Datagram Protocol

Contents

List of Figures

List of Tables

1

Introduction

Mobile networks are complex systems whose performance depends on a multitude of factors, particularly for 5G networks. Due to this complexity it is difficult to accurately assess the bandwidth in base stations that exist in mobile networks. The bandwidth can be assessed experimentally by for example downloading a large packet of data. This however has the downside of also using up a lot of the network's resources. Therefore some alternate way of instantaneously estimating the bandwidth without using the network's resources could prove beneficial. This thesis will be done in cooperation with the networking and telecommunications company Ericsson.

Base stations in 5G mobile networks are transceivers that function as the main points through which devices can communicate with the rest of the network. These base stations have internal counters that keep track of a wide variety of various statistics. These counters work well as a basis for an estimation algorithm since they cover a lot of statistics that might potentially affect the bandwidth. Since these counter values only exist locally in the base stations, no additional network resources are used compared to for example experimentally estimating the bandwidth, because the bandwidth can be locally computed in the base station.

There are many different ways to define bandwidth. To ensure clarity and consistency, whenever bandwidth is mentioned in this paper, it is defined as the maximum amount of bits that one particular device can transmit through a base station per second.

This paper aims to function as a preliminary analysis on the topic and in turn to allow future research to be built upon the findings of this research. Therefore not much focus is put on creating a model which is able to efficiently adapt to traffic scenarios, but rather to investigate whether or not there exist counters and correlations at all.

1.1 Aim

The goal of this thesis is to develop an algorithm which, by using counters that exist in 5G base stations, estimates the bandwidth for a device transmitting data through a base station in a 5G mobile network. This consists of creating an analytical and statistical model for the estimation of bandwidth as well as implementing it. Since there are very many available counters to be used as input from the 5G base station,

it is not trivial to find which parameters work for the algorithm. Therefore the main part of the work is to find which counters primarily correlate to the bandwidth, as well as investigating how they correlate.

1.2 Research Questions

Here the questions that are intended to be answered throughout this thesis are presented.

1. Which counters are more suitable to use as a basis for bandwidth estimation?
2. How do counters correlate to bandwidth?
3. How do different network scenarios influence bandwidth estimation?

1.3 Challenges

The primary challenge lies in finding counters that appear to correlate to bandwidth as well as finding how they correlate. There are thousands of possible counters that can be used that track a multitude of statistics relating to various parts of the 5G network, meaning that the search space is very large for performing a manual analysis.

With the added aspect of the unpredictable nature of traffic, finding a model which performs well on many different traffic scenarios is not trivial. Certain counters might be better at estimating certain traffic scenarios, whereas they perform worse on others. Therefore creating a model that ubiquitously estimates accurately is a complex task.

1.4 Limitations

In order to keep the thesis work feasible, several limitations need to be imposed. First of all it would be unfeasible to try to use all available input values to model as there is a large amount of available inputs to analyze. Instead, efforts are made through analysis to identify inputs that exhibit a significant impact on the resulting bandwidth, utilizing these findings as the foundation for the estimation algorithm.

Only values from the counters are used and no external factors or measurements are used as input to the estimation algorithm. It is therefore assumed that the internal counters contain enough information through the values already available.

Another limitation is to not look into the hardware of base stations and see how they affect our results, but instead assume that the values of the counters are correct.

Possible inference and physical conditions that might affect the radio part of the network are not considered. This can include for example weather conditions or obstructing objects that affect the quality of radio signals between a device and a base station.

A fair share division of resources, i.e. that all devices are entitled to the same amount of bandwidth, is assumed.

The analysis is limited to considering only the high-band frequencies, which exist at the higher end of the frequency spectrum and is exclusive to 5G. This is mainly to simplify the setup and experimentation.

1.5 Summary of Methodology

A quantitative methodology is used through the collecting of data points after which an analytical and statistical approach is employed in this thesis. This approach involves systematically gathering data from various 5G network counters under different traffic conditions to understand their relationship to the bandwidth. The experiments are performed for different scenarios for uplink/downlink, one or two devices, and using continuous or realistic streams. A statistical method is used to identify the most relevant counters for estimation. After correlations have been found and models are created, the performance of these models is evaluated to find which one appears to correlate closest to the bandwidth.

2

Background

In order to alleviate understanding of this paper and the setup used, an overview of relevant information relating to 5G networks, their architecture as well as related concepts are presented here.

2.1 TCP

TCP (Transmission Control Protocol) is a reliable protocol used to send data over networks like the internet. Before sending any data, TCP sets up a connection between the sender and receiver using a three-step handshake process. Once connected, TCP breaks the data into smaller packets and sends them one by one. Each packet is acknowledged by the receiver to confirm it arrived safely. If a packet gets lost, TCP will resend it to ensure all data reaches the destination correctly. After sending all the data, TCP closes the connection.

TCP adjusts the speed at which it sends data using congestion control mechanisms. Initially, it sends data slowly to test the network capacity, and then it increases the speed exponentially, doubling the amount of data sent with each successful transmission. This is called Slow Start. Once the network reaches a threshold, TCP reduces the sending rate after which it increases the sending rate linearly. There are version of TCP that handle it differently. For example in CUBIC, the congestion window grows cubically instead of linearly.

2.2 iPerf3

iPerf3 is a tool which is primarily used to measure throughput between two devices. This is done by having one of the devices host an iPerf3 server, and the other one being an iPerf3 client. The client can connect to the server and perform the measurements. It supports sending packets with three different protocols, namely SCTP (Stream Control Transmission Protocol), TCP and UDP (User Datagram Protocol). It is also capable of running both tests for uplink bandwidth, where the client transmits data, and downlink bandwidth, where the server transmits data. iPerf3 also allows users to control the amount of data sent during bandwidth tests. Users can specify data packet size and adjust testing parameters like duration and stream count, which refers to the number of parallel connections between the client and server that are used to transmit data during a bandwidth test [1].

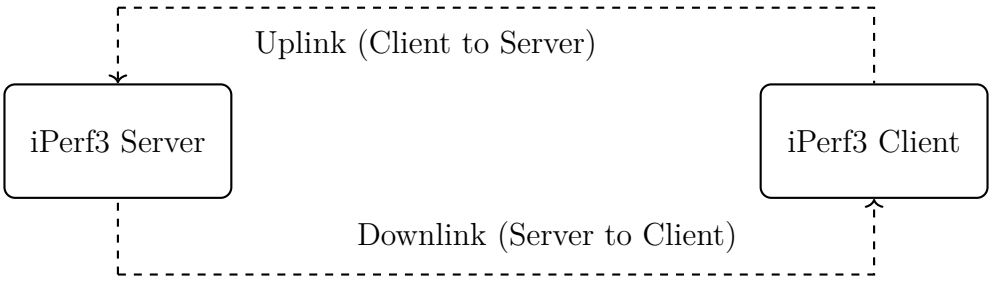

Figure 2.1: Illustration of the iPerf3 Client/Server Model.

2.3 Bandwidth

Bandwidth is typically defined as the maximum amount of data that can be transmitted during some time period. Bandwidth can be measured for many different things, for example in a router, in physical mediums such as wires or radio waves, or on a path.

There are different types of bandwidth, such as physical bandwidth - also known as channel capacity - and available bandwidth. Physical bandwidth is a measure of the maximum amount of bits that can be processed during a time period. This is essentially only affected by physical considerations. For example the bandwidth of a link is only dependent on the physical medium through which it flows. Physical bandwidth can however be more complicated to establish in a router or some hardware as there are more factors that can affect it. Available bandwidth, in contrast to physical bandwidth, is on top of physical considerations also affected by current traffic. Available bandwidth is essentially the maximum of data that can be sent given the current traffic. The added aspect of considering the current traffic further complicates the process of estimating the available bandwidth.

Whenever bandwidth is mentioned in this paper, it is defined as the maximum amount of bits that one particular device can transmit through a base station per second. A distinction is made between up- and downlink bandwidth. Uplink is when the data is transmitted from the device, and downlink is when it is received by the device. This definition is essentially a variant of available bandwidth.

iPerf3 is used to estimate the bandwidth in this paper. When using iPerf3 with TCP, due to the congestion control of TCP, it has the effect of dynamically changing the rate of transmission based on traffic conditions in order to prevent congestion in networks. This has the added effect of essentially tracking the available capacity on the network path between the device it is sent from to the receiver, since it attempts to transmit as much data as possible until it reaches a bottleneck.

2.4 4G Networks

4G networks represent the fourth generation of mobile broadband technology, providing faster speeds, lower latency and improved overall performance compared to

previous generations. With 4G, download speeds can reach up to 1 Gb/s and 500 Mb/s for upload [2].

LTE (Long-Term Evolution) is the wireless standard developed by 3GPP (3rd Generation Partnership Project) used for 4G networks. It allows backward compatibility for earlier technologies and it has become universally available, commonly for areas that do not yet have access to 5G technology [3].

2.5 5G Networks

5G networks are cellular networks, meaning that the coverage is divided into small geographical cells. They allow mobile users within these cells to get internet coverage. 5G networks have significantly increased performance and capacity compared to previous generations. For example, 5G reaches its peak data rate at 20 Gb/s and a time delay under 1 ms whereas 4G reaches its at 1 Gb/s and has a delay of 10 ms [4]. Due to the increased capacity it is expected to be able to handle more concurrent users, have higher reliability and provide higher connectivity regardless of geographic region [5].

5G networks are typically divided into different parts that are responsible for certain functionalities of the network, such as RAN (Radio Access Network), Packet Core and the Transport Network.

2.5.1 RAN

RAN is a crucial part of the 5G network, made up of antennas and radio signals. It is the first part that devices connect to when using 5G. The radio part of of RANs can support various frequencies. These are mainly divided into three different spectrums: low-band (under 1 GHz), mid-band (1 GHz - 6 GHz) and high-band (over 24 GHz) [6]. Low-band and mid-band together are also known as FR1 (Frequency Range 1) and high-band as FR2 (Frequency Range 2).

These spectrums have various differences, for example relating to the frequency range as well as the division duplex, e.g. FDD (Frequency Division Duplex) and TDD (Time Division Duplex). TDD works by dividing the time into slots, where the slots are distributed between up- and downlink. This means that during the slot that is allocated for a direction of transmission, the direction type has access to the full capacity of the link, including the full frequency range. FDD on the other hand divides the frequency range into slots, so that up- and downlink get a fraction each of the frequencies to use, however they are available to use continuously [7]. FR1 can use both FDD and TDD, whereas FR2 only uses TDD to transmit packets.

5G networks utilize OFDM (Orthogonal Frequency-Division Multiplexing) to split the spectrum, allowing multiple users to share the capacity of the base station. Its important to note that OFDM focuses on how the spectrum is divided among users, enabling simultaneous communication without interference [8]. It essentially grants a range of frequencies for some time period that one specific user has access to during that duration.

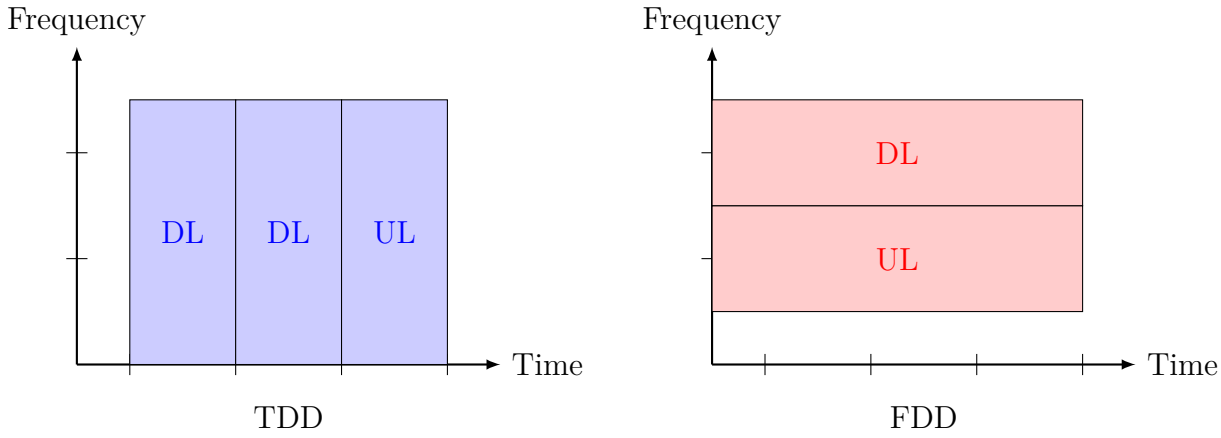

Figure 2.2: Illustration of TDD and FDD duplexing. TDD assigns time slots exclusively to up- and downlink for the full frequency range, while FDD allocates frequency slots, allowing simultaneous access but dividing the frequency range between uplink and downlink.

In 5G networks, the base station is the part that lets the user interface with the core network. As previously mentioned, 5G networks are cellular networks, meaning that they are divided into small geographical areas also known as cells. Each cell has one base station that manages the traffic within its coverage area. Since cells can overlap, a geographic point may be covered by more than one cell meaning the traffic can be handled by multiple base stations.

Base stations consist of two smaller units: DUs (Distributed Units) and a CU (Central Unit). Each base station can contain many DUs but only one CU. DUs are, among many other tasks, directly responsible for the communication with the radio unit, as can be seen in Figure 2.3. The CU however, among other tasks, has a more overarching responsibility of handling data flow and directing data packets.

In a base station there exists a baseband which is responsible for connecting the radio units, i.e. antennas, to the core network. These basebands include counters, which keep track of various statistics in 5G networks. These statistics can include information such as radio signal strength, buffer queues, amount of dropped packets and energy consumption.

In the base station, there is also a scheduler which is responsible for allocating physical layer resources for UEs (user equipment) for both up- and downlink, though separately. The scheduler exists in the base station and is very closely connected to the performance of UEs in the 5G network, since it decides who, when and for how long each user has some resources allocated to them. The scheduler allocates resources for the users of the base station, deciding who gets to use the resources and when. There is no 3GPP standard for scheduling mechanisms and this is instead left to the implementer [9].

RSRP (Reference Signal Received Power) and RSRQ (Reference Signal Received Quality) are indicators of signal strength and quality in 5G networks. RSRP measures the power level of the reference signal that a device receives from the network.

This is a key factor in determining how strong the connection is which impacts the overall performance of the network. RSRQ measures how good or clear the signal is. It takes into account both the strength of the signal and the level of interference or noise in the network. Lower RSRQ values mean better signal quality and less interference.

2.5.2 Core Network

The 5G core network is central to the operation of 5G networks, managing both the control and data planes. It handles crucial tasks such as authorization, authentication and security, ensuring secure access and data privacy for users. One of its advanced features is network slicing which allows for the creation of customized virtual networks tailored to different services or user needs.

There are two types of 5G: SA (Standalone) and NSA (Non-Standalone). The main difference between these two types is that NSA allows easier migration to 5G networks, since it is uses the packet core of a 4G LTE network, rather than a 5G Packet Core used by SA.

2.5.3 Transport Network

The function of the Transport Network is to connect all of the nodes in a 5G network with the core. It indirectly facilitates end-user connectivity through the RAN.

There are three main parts of the transport network, namely the fronthaul, midhaul and backhaul, as simplified in Figure 2.3. These are responsible for the transport between different specific parts of the network. Fronthaul is responsible for the transport between the radio unit and the DU, midhaul for the transport between DU and CU, and backhaul between CU and Packet Core.

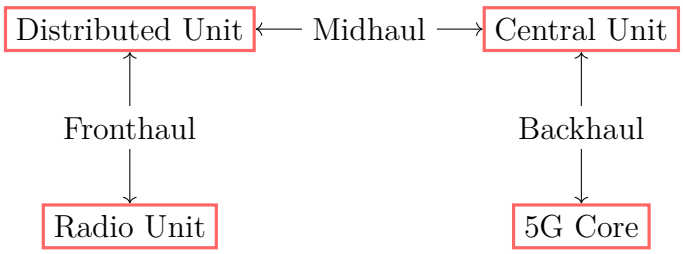

Figure 2.3: 5G Transport Network Overview

2.5.4 5QI

The concept of 5QI (5G Quality of Service Identifier) allows for the prioritization of traffic based on specific service requirements. This mechanism classifies data flows into different QoS (Quality of Service) classes, enabling differentiated treatment based on the flows needs. By assigning different QoS levels some data flows can receive higher priority and better network resources than others, ensuring that services like video streaming or real-time applications get the appropriate bandwidth and latency [10].

2.6 Counters

Counters are metrics used to keep track of occurrences or measurements of various aspects of performance, traffic and other network-related topics. They are used to help identify issues like slow response times or high error rates in base stations.

In order to understand the definitions of the selected counters used in this thesis, some useful terminology and concepts are here presented.

- **PDCCH**

  PDCCH (Physical Downlink Control Channel) is mainly responsible for transmitting scheduling details to individual UEs. This includes providing information on resource allocations for both uplink and downlink communications, as well as control messages.

- **PDSCH**

  PDSCH (Physical Downlink Shared Channel) is the physical channel shared by multiple devices used in 5G networks to transmit data from a base station to a device.

- **PUSCH**

  PUSCH (Physical Uplink Shared Channel) is the physical channel shared by multiple devices used in 5G networks to transmit data from the devices to a base station.

- **MAC**

  MAC (Medium Access Control) is a sublayer of the Data Link layer in a network responsible for managing access to a shared communication medium, such as a wired or wireless network.

- **PDU**

  A PDU (Protocol Data Unit) Session is a logical link that connects a UE to a data network, enabling the device to receive services from the network. It is specifically a PDU if the data entity is to or from a lower protocol layer entity.

- **SDU**

  SDU (Service Data Unit) is essentially the same as PDU, but specifically if the data entity is to or from a higher protocol layer entity.

- **DRB**

  DRB (Data Radio Bearer) is used to carry the data associated with an EPS (Evolved Packet System) bearer.

- **PCell**

  The PCell (Primary Cell) is the main cell and is the one used to initiate initial access for a UE.

- **SCell**

  An SCell (Secondary Cell) is an additional cell that works alongside the PCell to provide extra support for a user in carrier aggregation scenarios.

- **PSCell**

  A PSCell (Primary Secondary Cell) is the cell used to initiate initial access for UEs in secondary cell groups.

- **Type A and B**

  Type A and B are two different mappings of bits used by for example PUSCH, PDSCH and PDCCH.

In order to alleviate understanding of the counters mentioned in this paper, Table 2.1 summarizes the definition and unit of measurement for each counter.

Table 2.1: Counters and Their Meaning

Name	Unit	Definition
D_PDCCH_TYPE_A	n/a	This counter measures the total number of resource block symbols used by the PDCCH for scheduling tasks in the downlink and uplink, specifically using mapping type A.
D_PDSCH_TYPE_A	n/a	This counter measures the total amount of resource block symbols for PDSCH that are of type A.
U_PUSCH_TYPE_B	n/a	This counter tracks the total number of PUSCH resource block symbols used for data transmission with mapping type B.
D_MAC_SUCCESS	Bytes	This counter measures the total MAC PDUs volume in bytes for downlink, including both PCell, SCell, and PSCell traffic along with padding.
U_MAC_SUCCESS	Bytes	This counter measures the total MAC SDUs volume in bytes for uplink, including both PCell, SCell, and PSCell.
D_MAC_DRB	Bytes	This counter measures the total volume of MAC SDUs in bytes initially transmitted for each DRB, excluding retransmissions. It includes data from both PCells and SCells if carrier aggregation is used.
U_MAC_RESTRICTED	Bytes	This counter measures the total volume in bytes of MAC SDUs for uplink that are part of a burst. It is only included if it is restricted by air interface.

2.7 Simple Moving Average

SMA (Simple Moving Average) is a fundamental concept in time series analysis, used to smooth out short-term fluctuations and highlight long-term trends or cycles. They involve calculating the average of a certain number of consecutive data points

in a dataset. This technique is particularly useful for identifying patterns and trends over time, especially in contexts where data points fluctuate rapidly.

The SMA for a time series data can be calculated using the formula:

$$SMA(x, t, N) = \frac{1}{N} \sum_{i=0}^{N-1} x_{t-i}$$

where:

- x is the time series data array that the SMA is computed for.
- t is the time index for which the SMA is being computed.
- N is the number of data points in the SMA window.

It is important to note that when applying the SMA to a time series of length n, the resulting time series produced by the SMA will have length $n - N + 1$, where N is the window size, under the assumption that $N > 0$.

2.8 Exponential Moving Average

EWMA (Exponentially Weighted Moving Average) is used to smooth out short-term fluctuations in data, similar to the SMA. However, the EWMA gives more weight to recent data points, making it more responsive to faster changes.

The EWMA is calculated using the following recursive formula:

$$EWMA_0 = x_0$$
$$EWMA_t = \alpha x_t + (1 - \alpha) EWMA_{t-1}$$

Where:

- x_t is the current data point,
- α is the smoothing factor that controls how much weight is given to recent data.

2.9 SelectKBest

SelectKBest is a feature selection method used to identify the K best features from a dataset by evaluating how well they correlate with a target variable using a scoring function. It relies on statistical measures to determine the importance of features. SelectKBest can use statistical tests like the chi-squared test, ANOVA F-test, or mutual information score to score and rank features based on their relationship with an output variable. It essentially applies the scoring function to all variables, and then it selects the K features with the highest scores to form the final feature subset [11].

Chi-squared Test

The chi-squared test is a statistical hypothesis test used to determine whether there is a significant association between two categorical variables. In the context of feature selection, the relationship between a categorical feature and the target variable is assessed. The chi-squared test assesses the independence between two categorical variables by comparing observed frequencies with expected frequencies. The test calculates the chi-squared statistic, which measures how much the observed frequencies deviate from the expected frequencies.

The chi-squared test uses the chi-squared distribution to evaluate the significance of the observed deviations from the expected values. The p-value obtained from the chi-squared statistic indicates whether the observed frequencies significantly differ from the expected frequencies. A low p-value (typically less than 0.05) suggests that the observed frequencies are significantly different from the expected frequencies, indicating a potential relationship between the variables.

The Chi-squared test can be used in SelectKBest. It does this by checking the relationship between each feature and the target variable.

The Chi-squared statistic (χ^2) is calculated using the following formula:

$$\chi^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i}$$

Where:

- O_i = observed value
- E_i = expected value
- n = number of categories or observations

2.10 Related Works

Research on bandwidth estimation has been previously conducted. There has however not been a lot done on 5G networks or counters found in base stations specifically. The similar topic of future bandwidth prediction appears to have been more studied than bandwidth estimation. Machine learning solutions have been prevalent in research regarding bandwidth prediction. The topic of bandwidth measurement has also been researched.

Bandwidth Measurement

The measurement of bandwidth has been studied in various ways. A 1999 paper [12] looks at some contemporary bandwidth measuring techniques and discusses how they are flawed in various ways, for example incapability of adjusting quickly enough to changes in traffic, insufficient accuracy or poor scalability. Another work from 2002 [13] looks at some algorithms for measuring bandwidth that have low intrusion. It also discusses different issues with bandwidth measurement.

Bandwidth Estimation

This topic has been researched in MANETs (Mobile Ad hoc Networks), which is a type of wireless network that does not require pre-existing infrastructure, in order to find how different factors affect the bandwidth. A 2002 paper [14] and a 2010 paper [15] both analyze and discuss different factors that influence the resulting bandwidth in such networks.

Different methods for bandwidth estimation have been explored, including statistical and traffic-related approaches. A 2004 paper [16] investigates a traffic-based method investigates a traffic-based method using TCP's congestion control for bandwidth estimation, while also discussing enhancements for better accuracy.

A 2014 paper [17] addresses the challenges of bandwidth estimation in wireless networks and discusses flaws in existing techniques. It then proposes a more efficient method for mobile networks that avoids the resource-intensive approach of transmitting large packets of data.

A 2008 paper [18] discusses the importance of fast convergence times for bandwidth estimation in applications such as multimedia streaming. Previous estimation techniques are mentioned as having long convergence times. The purpose of this paper is to create a tool for bandwidth estimation which is well suited for wireless networks and can instead estimate with fast convergence times.

Bandwidth Prediction

LSTMs (Long Short-Term Memory) have been used to train on previous bandwidth values to predict future bandwidth in the paper. A 2020 paper [19] explores the use of LSTM neural networks and Bayesian fusion for real-time mobile bandwidth prediction. It is able to improve prediction accuracy compared to contemporary state-of-the-art prediction models.

Values from counters in 5G base stations were used to train Transformers to predict future bandwidth, as shown in a 2022 paper [20]. It is able to use a machine learning model trained on counter values to predict the bandwidth a few seconds into the future. It has therefore shown that using counters as a basis to solve bandwidth related problems is a suitable way of approaching the problem.

SVM (Support Vector Machine) classifiers have been used to detect network scenarios as shown in a 2023 study [21]. It succesfully showed that training models for different scenarios is able to improve prediction accuracy. RNNs (Recurrent Neural Networks) have also been used to estimate bandwidth and predict when handoffs between 4G and 5G occur in a paper from 2022 [22]. It therefore shows that despite the mobile nature of 5G, there are solutions that are able to accurately predict bandwidth.

3

Methodology

This chapter presents the methodology employed to investigate the relationship between counters and bandwidth using an internal Ericsson 5G network. Experiments are conducted across various traffic scenarios, including uplink and downlink conditions, device counts and types of traffic in order to analyze their effects on bandwidth.

3.1 Overview

The approach to understanding the correlation between 5G counters and the resulting bandwidth is statistical and analytical. This is done by performing experiments on a 5G network provided by Ericsson. The process is divided into several concrete steps.

The experiments are divided into 8 different scenarios, two for training (uplink and downlink respectively) and six for validation. The details of the experiments are explained in section 4.3. The experiments are used to gather the data on which the analysis can be performed.

Once the datasets are created from the experiments, the analysis can begin. The first step is to find which counter appears to have the best correlation to bandwidth for the two training scenarios. SelectKBest is used to this end, which outputs the 5 counters that appear to correlate best.

For each training scenario and counter found using SelectKBest, a model is created which assumes that there exists a linear relationship between derivative of the counter value and the bandwidth. The models are then run on the validation scenarios, where they are evaluated by an estimation metric to rank the models and find which one is most accurate.

3.2 Counter Selection

The selection of counters is a pivotal step in estimating bandwidth in 5G networks. Counters, which are statistical data collected from base stations, offer insights into network performance, usage, and other parameters. An investigation into their potential impact on bandwidth estimation is necessary to select the most relevant counters.

Due to the large amount of available counters, it is important to find a smaller selection to perform analysis on for the sake of feasibility. At the same time, to improve the chance of finding counters that correlate to bandwidth, it would be essential to perform the selection in a way to improve the chance of the counters having a correlation to bandwidth.

Existing knowledge and expertise in the field of 5G networks are utilized to make informed guesses about which counters are likely to be relevant for bandwidth estimation. Experts in the field, such as supervisors at Ericsson, researchers, and hardware engineers working with base stations, are consulted to gain insights into which counters they consider important based on their experience and understanding of 5G networks. An internal document containing formulas that track various performance and capacity aspects for 5G networks is also investigated in order to find counters suspected of having some relevance to bandwidth through the formulas they use.

Once a reasonably sized dataset of 43 counters is collected, including counters believed to be somehow correlated to bandwidth, a statistical method is employed to find which of these counters appears to have the best correlation to bandwidth. The method used for this analysis is SelectKBest, a feature selection algorithm that scores each counter based on its statistical significance and correlation to bandwidth. It uses the chi-squared test as a scoring function. Counters with the highest scores are identified as the most relevant to bandwidth estimation. It is run on the two training scenarios shown in 4.1 to find the 5 counters that appear to correlate best for each scenario, simplifying the amount of analysis that needs to be done. All of the counters found for these two scenarios are then together used as a basis for the analysis of all the models for both up- and downlink.

3.3 Counters' Correlation to Bandwidth

Understanding the connection between these counter values and bandwidth is crucial for optimizing network performance. Counters provide insights into different aspects of network performance, and their values could potentially correlate to the bandwidth. This section explores how the investigation into the impact of counters on bandwidth is conducted.

The main hypothesis that is to be tested for this part of the analysis is whether there exists a linear relationship between the counters and the bandwidth. By performing this step of the analysis it can be established whether a linear relationship is sufficient to correlate some counter value to the bandwidth.

One model is created per training scenario and counter found in the previous step. This means that two models are created per counter, one for uplink and one for downlink according to the following process.

Under the assumption that the measured bandwidth for the training scenarios is relatively constant, an average bandwidth can be calculated. In the case that the counter values appear to increase linearly, an average for the derivative of the counter

values can be calculated as well. Then the constant needed to multiply the counter derivative to reach the average bandwidth can be calculated.

An SMA can then be applied to smoothen out any short-term fluctuations, the window size of which can be manually adjusted. An EWMA is also used separately on the linear models in order to investigate the effect of the type of moving average on the estimating accuracy. The window size for all models trained on the same training scenario are the same to avoid any potential differences that can be caused by having varying window sizes between the models. Then the models found for a certain traffic scenario can be evaluated using some metric to find out which one appears to correlate the best to the measured bandwidth.

The models created for the V1b and V2b scenarios are also investigated to see how the effect of dividing the output by the current amount of users sending traffic affects the accuracy of estimation.

For each model that is created for the training scenarios, the same model is run without changes for all corresponding validation scenarios in order to verify their potential correctness. This means that the uplink training scenario is validated using the three remaining uplink scenarios, and the same process for downlink. This is process is presented in chapter 5.

The algorithms' parameters of the best performing models for each traffic scenario presented in chapter 6 are made explicit in Table 3.1. The parameters a and b correspond to the following constants in this linear expression:

$$a \cdot \left(\frac{x_t - x_{t'}}{t - t'} \right)$$

where x_i represents the value of counter x at time i, t is the time of the current measurement, and t' is the time of the previous measurement. The value n is the manually adjusted window size.

Table 3.1: Model Settings for Algorithm

Model	a	n
Model 1 Downlink	$7.0952 * 10^{-6}$	9
Model 2 Downlink	$7.1432 * 10^{-6}$	9
Model 3 Downlink	$9.1630 * 10^{-3}$	9
Model 5 Downlink	$1.0416 * 10^{-3}$	9
Model 6 Downlink	$6.6272 * 10^{-5}$	9
Model 7 Downlink	$3.2828 * 10^{-4}$	9
Model 1 Uplink	$2.2910 * 10^{-4}$	7
Model 3 Uplink	$7.4043 * 10^{-6}$	7
Model 4 Uplink	$7.4049 * 10^{-6}$	7
Model 5 Uplink	$2.6694 * 10^{-3}$	7
Model 6 Uplink	$1.7895 * 10^{-4}$	7
Model 7 Uplink	$7.2589 * 10^{-5}$	7

3.4 Applying Models to Real-Time Estimation

This section aims to showcase the updated functionality of the models for real-time use, demonstrating the results through pseudo code and thereby the algorithm implemented to make these adjustments. This algorithm is created to handle all the various models presented in chapter 6 by being able to change various parameters to fit the model inherent characteristics. It also applies an SMA to the result.

In order to make the models shown in chapter 6 work in real time, i.e. by taking in one value at a time, it is essential to continually store the amount of previous values that correspond to the window size of the SMA. This functionality can be simply implemented using a queue. However, to ensure that the algorithm performs like the models, it does not output a coherent value until the amount of values corresponding to the window size has been entered.

Algorithm 1 shows the pseudo code required in order to run bandwidth estimation using the different models derived in chapter 6. It takes in two inputs, namely x and s, along with the parameters n and a, which are described here:

- x - the inputted counter value
- s - amount of seconds passed since last inputted value
- n - window size
- a - coefficient in linear relationship

It is important to note that the value of s is arbitrary the first time the function is called, since there are no previous values.

This rough step-by-step overview shows the functionality of the algorithm:

1. Calculate the difference per second between the current and previous values
2. Calculate the model-specific linear expression
3. Push the expression's result to the back of the queue
4. If the queue has enough values to fill the window, return the average of all the values and pop the front value

Algorithm 1 Bandwidth Estimation Algorithm

Require: $n > 0$

$n \leftarrow$ window size
$a \leftarrow$ multiplicative factor
$Q \leftarrow$ empty queue
$p \leftarrow -1$ ▷ Represents the previous value
function BandwidthEstimation(x, s)
    if $p = -1$ **then**
        $p \leftarrow x$
        return -1
    else
        $t \leftarrow a((x - p)/s)$
        Push t to back of Q
        $p \leftarrow x$
        if length of $Q = n$ **then**
            $m \leftarrow$ calculate mean of all elements in Q
            Pop front element of Q
            return m
        else
            return -1
        end if
    end if
end function

3.5 Evaluation Of Our Estimation Method

This section delves into the metric that will be used for evaluating the performance of the bandwidth estimation models, namely MRE (Mean Relative Error). This metric provides a quantitative basis for assessing the accuracy of predictive bandwidth estimations against actual bandwidth measurements.

MRE is a widely used metric for measuring the similarity between two time series. The formula for MRE is given by:

$$\mathrm{MRE}(A, B, n) = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{A_i - B_i}{A_i} \right|$$

where A_i represents the value of a data point for a time series A at observation i, B_i represents the value of a data point for a time series B at observation i, and n is the total number of observations. MRE provides an average of all relative errors, which allows comparison of how accurate different models are. It is however important to note that MRE can be misleading in its comparison, since a model which is slightly inaccurate overall may produce the same result as a very accurate model which has a few values that are very inaccurate.

4

Experimentation

In this chapter, the setup for the experiments is presented and explained, which specifically consists of the devices used as well as the 5G network and base station that are used. The experiments themselves are also discussed and categorized, in order to improve understanding and reproducibility of the report.

4.1 Setup

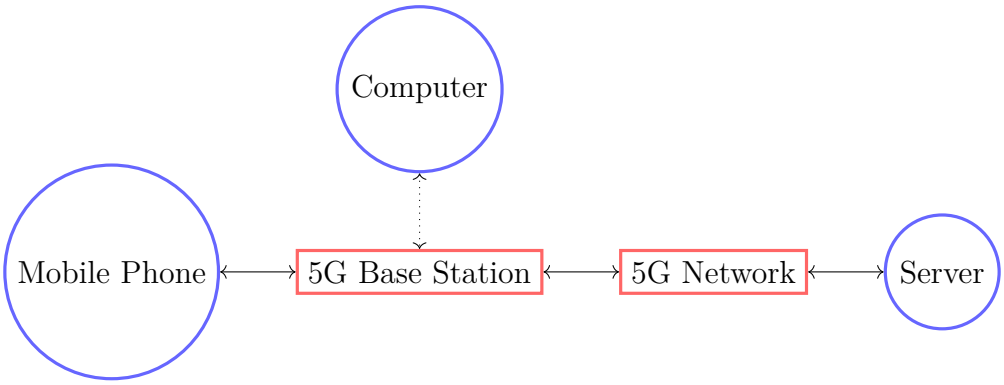

Figure 4.1: 5G Setup Overview

The setup consists of a 5G base station, one or two mobile phones (depending on the traffic scenario) capable of connecting to the 5G high-band, a computer and a server as shown in Figure 4.1. The mobile phones run Android 14. The experiments are performed in the CASE Lab at Chalmers University of Technology, which was used to connect to the 5G network used. The 5G network on which the experiments are run is the Expericom 5G network, which is an Ericsson-internal 5G network. It is primarily used for experimenting and testing. The computer is remotely connected to the base station, through which the counter values can be extracted. Then in order to get the bandwidth of the base station, a server that exists on the 5G network is used from which data can be downloaded and uploaded to get the uplink and downlink bandwidth on the path between the two devices. iPerf3 will be used to accomplish this task. By running an iPerf3 server on the server node, an iPerf3 client on the mobile phone can connect to the server, allowing the bandwidth to be measured between these two devices. Then the bandwidth can be measured at

regular intervals once a second, while extracting the counter values at the same time. This allows the counter values as well as the bandwidth they are mapped to be retrieved.

4.2 5G Network

The mobile network used supports both 5G and 4G. It uses the Ericsson Indoor Radio Unit 8848 and the Ericsson Radio Dot 4479 for the mid-band frequencies. Ericsson Radio 2203 B3 is used for LTE, and Ericsson AIR 1281 handles the high-frequency bands N257 and N261, where the supported FR2 frequency ranges are from 26.50 to 29.50 GHz. A 10 Gb/s fiber optic link connects the network to Ericsson Lindholmen for remote monitoring.

The base station used for the experiments uses an NSA setup meaning it uses a 4G Packet Core and LTE carrier for control-plane traffic. During the experiment, three out of four 5G high-band cells were intentionally deactivated, meaning that all user-plane traffic was handled by that single 5G high-band cell. The nodes for mid-band are deactivated during the experiments so that only the high-band node is available.

4.3 Experiments

The experiments are performed by having one or two phones connected to the iPerf3 server, while a script is simultaneously running on the computer that retrieves values once every second. The physical setup can be seen in Figure 4.2. They will be started at the same time to ensure that the values are synced. The retrieval of counter values through the computer does not always return the values instantaneously, and will sometimes take longer than a second to return the value. However the time stamps for the counter value retrieval is printed, which allows values to be synced to their corresponding bandwidth value by seeing how far after the time stamp of the first counter retrieval was. These retrievals will be run for some period of time, after which they can be synced so that the counter values are mapped to the bandwidth values.

When using two devices, the second phone is introduced into the experiment at specific intervals to evaluate its impact on the overall bandwidth. The second device is alternately turned on and off at predetermined time periods to create additional traffic. This setup enables the analysis of counters relate to bandwidth during various traffic scenarios.

Figure 4.2: Physical Setup

4.4 Experimental Scenarios

The experiments can be performed with some variation, for example by altering the amount of users, changing the type of data that is sent, or by using up- or downlink. In the case of the realistic data stream, video streaming is used for downlink, and a video call is used for uplink. The overview of the different scenarios is shown in 4.1. In terms of the naming of the scenarios, T and V signify a training and validation scenario respectively. 1 indicates that the scenario is for downlink and 2 for uplink. The three validation scenarios are then further named using a, b or c.

Table 4.1: Experimental Scenarios

Scenario	Description
T1	Downlink, One Device, Continuous Stream
V1a	Downlink, One Device, Realistic Stream
V1b	Downlink, Two Devices, Continuous Stream
V1c	Downlink, Two Devices, Realistic Stream
T2	Uplink, One Device, Continuous Stream
V2a	Uplink, One Device, Realistic Stream
V2b	Uplink, Two Devices, Continuous Stream
V2c	Uplink, Two Devices, Realistic Stream

For realistic applications, the latest version of YouTube uses QUIC (Quick UDP Internet Connection) for video streaming. Developed by Google, QUIC is a transport layer protocol that operates over UDP. Google Meet uses UDP for video calls relying on WebRTC for real-time communication [23].

Scenarios T1 and T2: This experiment consists of using a single device to send a continuous data stream while counter values are retrieved and bandwidth is measured once a second. The continuous stream of data is generated using iPerf3 with the TCP setting. This test is run until 120 counter data points have been gathered and is performed for both up- and downlink. This is the simplest case that will be investigated.

Scenarios V1a and V2a: This experiment consists of using a single device to send a realistic data stream while counter values are retrieved and bandwidth is measured once a second. The device runs iPerf3 with the TCP setting to measure the throughput. The choice of realistic data stream depends on whether the test is run as up- or downlink. For downlink a video is streamed from Youtube with 1080p resolution continually until 120 counter data points are acquired. For uplink a Google Meet call was used where the device was streaming its camera feed to another user in the meeting that existed outside of the 5G network. Since the throughput of iPerf3 is able to be significantly higher than the capacity required for the realistic applications, they essentially function as background noise since they can not compete with the TCP stream for the available bandwidth.

Scenarios V1b and V2b: This experiment consists of using two devices to send simple continuous data streams using iPerf3 with the TCP setting while counter values are retrieved and bandwidth is measured once a second. This case will be used to see how the bandwidth is affected by a new user starting to stream data. To evaluate how the addition of a new user affects bandwidth, the second device is activated and deactivated at specific intervals. The second device starts generating traffic after 30 seconds, after which it is turned off and on every 30 seconds until 120 counter data points have been acquired. This test is performed for both up- and downlink, where both devices are transmitting data in the same direction.

Scenarios V1c and V2c: This experiment consists of using two devices to send realistic data streams using video streaming while counter values are retrieved and bandwidth is measured once a second. iPerf3 is run on one of the devices with the TCP setting to measure the bandwidth. Depending on if the test was up- or downlink a different application is used to create the realistic data stream. Since the throughput of iPerf3 here too is able to be significantly higher than the capacity required for the realistic applications, they essentially function as background noise.

For downlink a Youtube video with 1080p resolution is streamed on both devices. The device running iPerf3 had the video streaming continually. After 10 seconds the second device starts streaming its video, and it is then alternately turned off and on every 30 seconds until 120 counter data points have been acquired.

For uplink two separate Google Meet calls are used to prevent the two devices from transmitting downlink data to each other, since the receiver of the video streams exists outside of the 5G network. The device running iPerf3 continually streamed its camera, while the second device is turned on after 10 second and then alternately turned off and on every 30 seconds.

5

Empirical Analysis

In this chapter the concrete steps of analysis performed to reach the models presented in chapter 6 are shown. The SelectKBest algorithm is employed to identify the most relevant counters, after which a model is created for each counter and training scenario. The process of analysis begins by creating linear relationships between the counters and the bandwidth for both training scenarios. These models are then further improved by using an SMA to smooth out fluctuations and improve accuracy.

5.1 Linear Relationships for Training Scenarios

This section covers the analysis for the two training scenarios. The analytical process is here exemplified by the downlink case for one device with a continuous stream, i.e. scenario T1. In Figure 5.1 one of the counters chosen by SelectKBest is plotted, where its accumulative nature is shown. This means that rather than showing the measured value for the last second, it continually keeps incrementing.

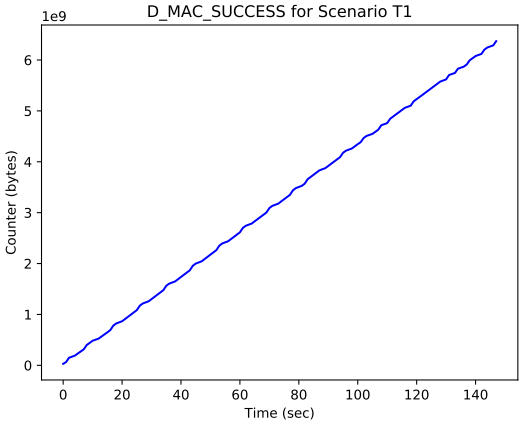

Figure 5.1: D_MAC_SUCCESS for Scenario T1

Since the counters are accumulative, in order to get the change per second which in this case is of more interest, the difference between all succeeding values are calculated. In Figure 5.2 the change per second for the same counter is shown. Despite its fluctuating nature, the fluctuations themselves appear relatively constant, since there are no overall increases over time, only periodical increases.

It is important to note that when the counter values are fetched during the creation of the dataset, it is not always possible to fetch them once a second. Therefore, in some instances there might be more than a second between two values. In order to solve this case when calculating the change per second, the difference between the current and previous value are divided by the amount of seconds passed between the two values. This essentially gives an average change over those seconds, which could possibly have differed slightly from the actual values.

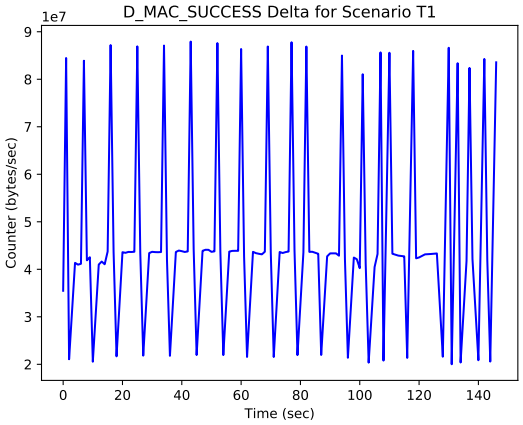

Figure 5.2: Change in D_MAC_SUCCESS for Scenario T1

In Figure 5.3 the throughput for the same traffic scenario as the previous plots is shown, with the average throughput highlighted. The iPerf3 throughput appears relatively constant, with some minor fluctuations. This is especially clear since most throughput values are very close to the average throughput.

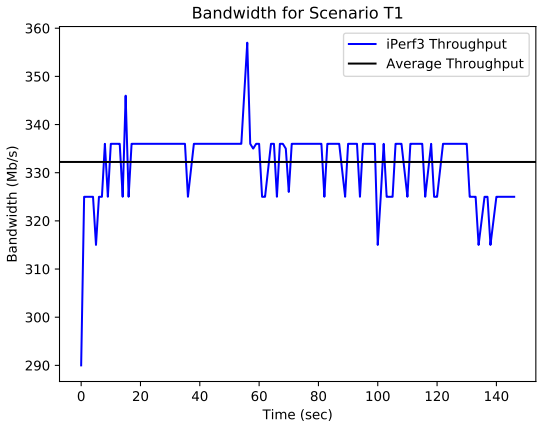

Figure 5.3: Bandwidth for Scenario T1

Since both the change in the counter values and the throughput appear to be relatively constant, it might be possible to find some linear relationship between them.

By calculating the average counter value $\overline{x}$ and the average throughput $\overline{t}$, the coefficient k, which the counter value can be multiplied with in order to potentially result in the bandwidth, can be calculated.

$$k = \frac{\overline{t}}{\overline{x}}$$

In Figure 5.4 the line labelled as Model is calculated using the following linear formula:

$$b = kx$$

As can be seen, the Model-line both significantly under- and overshoots due to its fluctuations, though the bandwidth appears to track the average of the model.

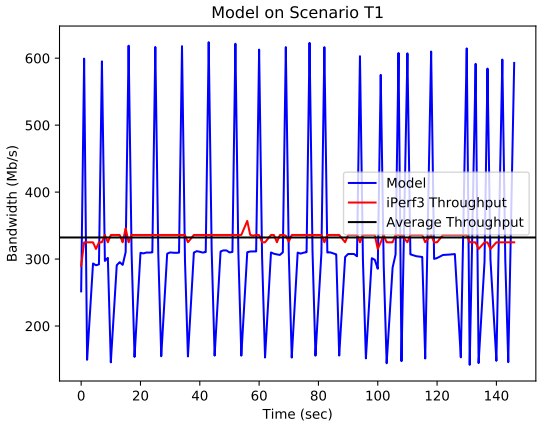

Figure 5.4: Model for Scenario T1

In order to alleviate the fluctuations as well as the under- and overshooting, an SMA is used to smoothen out the curve. The model with the SMA is presented in Figure 5.5 with a window size of 9. In this case all models trained on the same training scenario have the same window size in order to more easily compare the results of the models between each other. The window size could further be investigated to find which size would most improve the MRE values of the models.

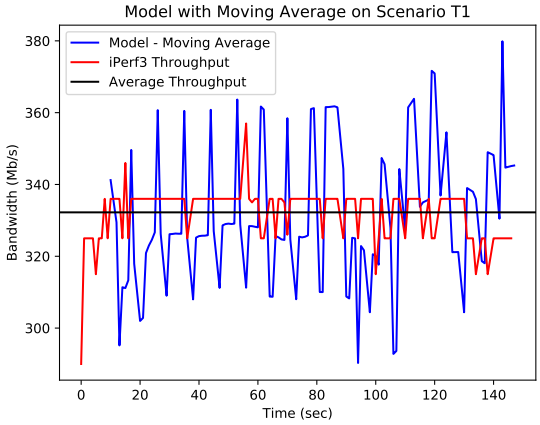

Figure 5.5: Model with SMA for Scenario T1

This process is performed on the two training scenarios for all of counters found using SelectKBest. Their MRE values are then calculated, after which the model with the lowest value is found to have the best correlation. The models with lowest MRE for each scenario are presented in Table 6.16.

5.2 Linear Relationships for Validation Scenarios

Figure 5.6 shows how the model presented in the previous section trained on scenario T1 performs on the validation scenario V1a. MRE is then calculated for the model for the validation scenario. Each model created from the training scenario is then run on the three corresponding validation scenarios, after which the model for each scenario with the lowest MRE value is chosen as the best performing model, also presented in Table 6.16.

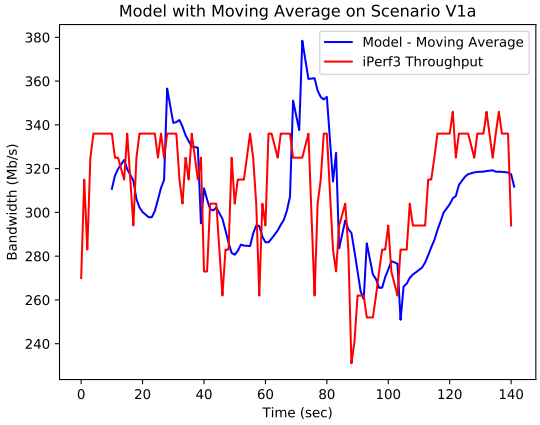

Figure 5.6: Model with SMA for Scenario V1a

6

Results

In this section the results of our models' performance are presented for the various traffic scenarios. The results include model accuracy, error rates and comparisons. Results showcasing the difference between SMA and EWMA are also included.

For the sake of readability in the results of the following scenarios, the following is defined:

$$\Delta x = \frac{x_t - x_{t'}}{t - t'}$$

where x_i is the value of counter x at time i, t is the time of the current measurement and t' is the time of the previous measurement.

Table 6.1 shows which models use which counters. The models are further appended with Uplink or Downlink depending on if they are trained on an up- or downlink scenario respectively.

Table 6.1: Models and their Counters

Model	Counter
Model 1	D_MAC_SUCCESS
Model 2	D_MAC_DRB
Model 3	U_MAC_SUCCESS
Model 4	U_MAC_RESTRICTED
Model 5	D_PDCCH_TYPE_A
Model 6	D_PDSCH_TYPE_A
Model 7	U_PUSCH_TYPE_B

6.1 Scenario T1

Here the models from the training scenario T1, which is the downlink traffic case with one device and a continuous stream, are presented. The counter that Model 4 uses does not change for this scenario and was therefore not used. Table 6.2 shows the constants calculated for all the models trained on scenario T1.

Table 6.2: Parameters for Models in Scenario T1

Model	Coefficient
Model 1	$7.0952 * 10^{-6}$
Model 2	$7.1432 * 10^{-6}$
Model 3	$9.1630 * 10^{-3}$
Model 5	$1.0416 * 10^{-3}$
Model 6	$6.6272 * 10^{-5}$
Model 7	$3.2828 * 10^{-4}$

All models use a window size of 9 for the SMA. They are together presented in Figure 6.1. Model 1 Downlink is found to have the lowest MRE value and is shown in Figure 6.2. The difference between the estimated and actual value is shown in Figure 6.3, and the percentage error between the estimated and actual value is presented in Figure 6.4. The MRE values for each model is shown in Table 6.3.

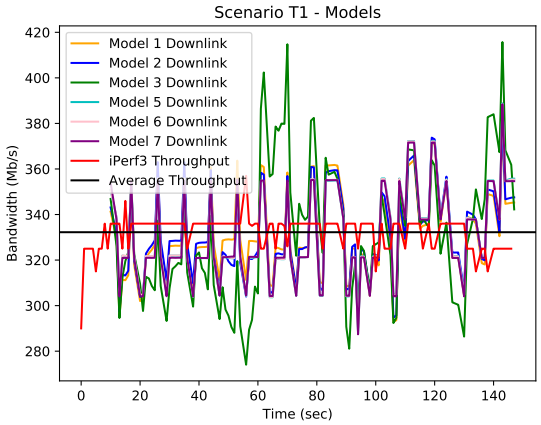

Figure 6.1: Six models for scenario T1

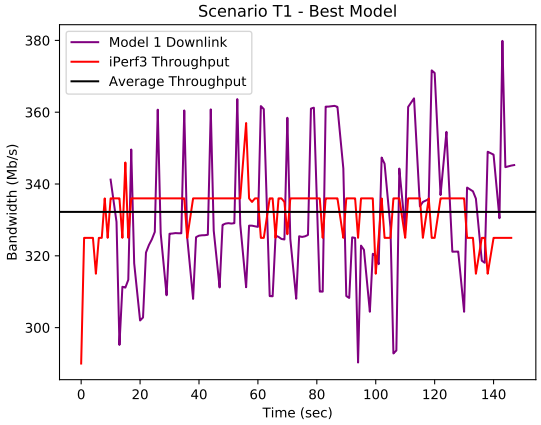

Figure 6.2: The model with lowest MRE value for the T1 scenario

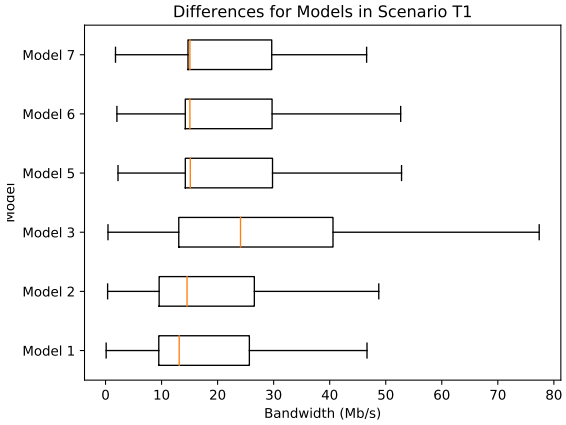

Figure 6.3: Differences between estimated and actual value for all downlink models for the T1 scenario

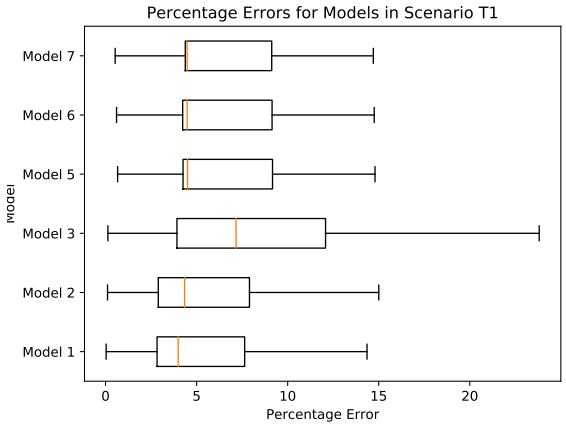

Figure 6.4: Percentage error between estimated and actual value for all downlink models for the T1 scenario

Table 6.3: MRE for Models in Scenario T1

Model	MRE
Model 1	0.053087
Model 2	0.054904
Model 3	0.081455
Model 5	0.059487
Model 6	0.059302
Model 7	0.059454

6.1.1 Scenario V1a

Here the models from the training scenario T1 are run on the data from Scenario V1a, which is the downlink traffic case with one device and a realistic stream.

The results of the models are together presented in Figure 6.5. Model 1 Downlink is found to have the lowest MRE value and is shown in Figure 6.6. The difference between the estimated and actual value is shown in Figure 6.7, and the percentage error between the estimated and actual value is presented in Figure 6.8. The MRE values for each model is shown in Table 6.4.

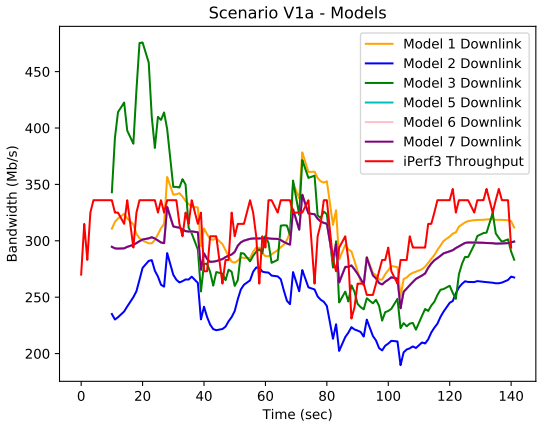

Figure 6.5: Six models for scenario V1a

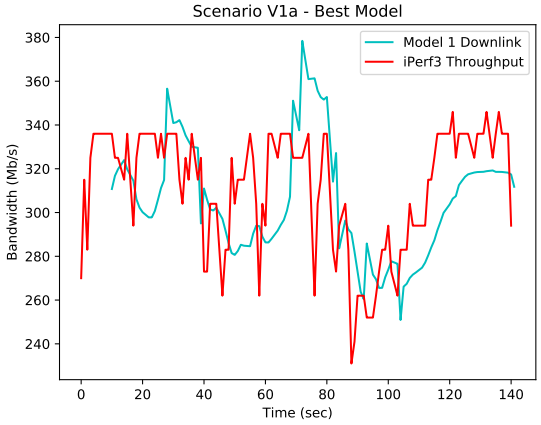

Figure 6.6: The model with lowest MRE value for the V1a scenario

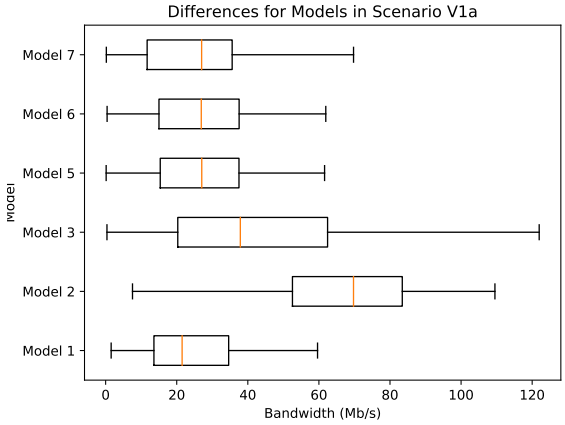

Figure 6.7: Differences between estimated and actual value for all downlink models for the V1a scenario

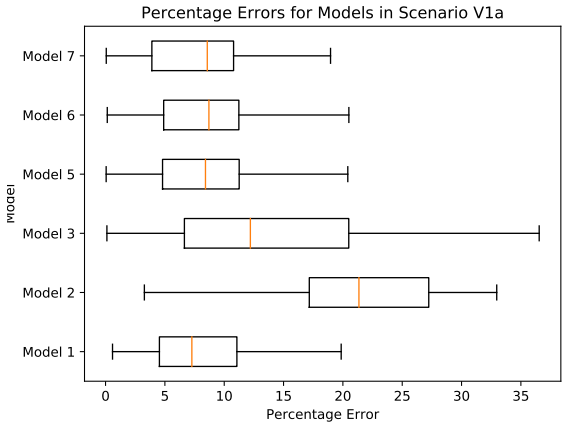

Figure 6.8: Percentage error between estimated and actual value for all downlink models for the V1a scenario

Table 6.4: MRE for Models in Scenario V1a

Model	MRE
Model 1	0.079717
Model 2	0.21211
Model 3	0.13965
Model 5	0.082495
Model 6	0.083162
Model 7	0.086743

6.1.2 Scenario V1b

Here the models from the training scenario T1 are run on the data from Scenario V1b, which is the downlink traffic case with two devices and continuous streams.

The results of the models are together presented in Figure 6.9. Model 7 Downlink is found to have the lowest MRE value and is shown in Figure 6.10. The difference between the estimated and actual value is shown in Figure 6.11, and the percentage error between the estimated and actual value is presented in Figure 6.12. The MRE values for each model is shown in Table 6.5.

By dividing the outputted value of the models by the amount of users actively sending data, the output of the models is shown in Figure 6.13. Model 3 has the lowest MRE value and is plotted in Figure 6.14. The MRE values for these models are presented in Table 6.6.

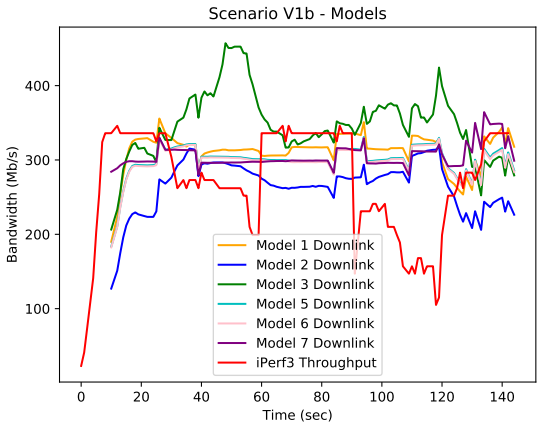

Figure 6.9: Six models for scenario V1b

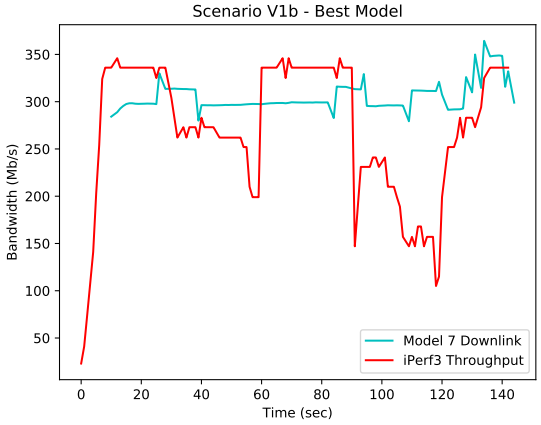

Figure 6.10: The model with lowest MRE value for the V1b scenario

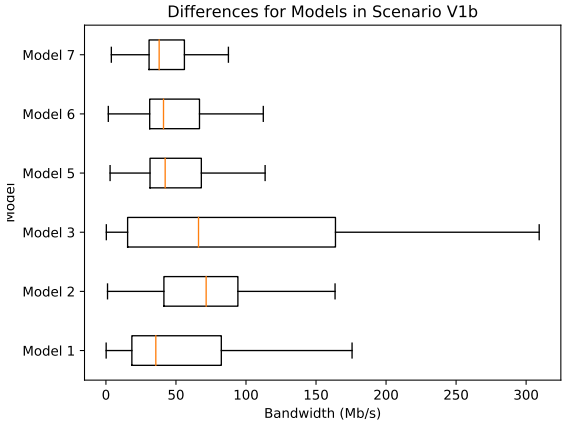

Figure 6.11: Differences between estimated and actual value for all downlink models for the V1b scenario

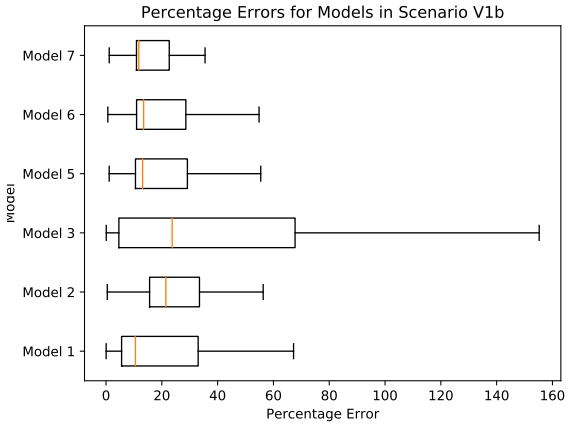

Figure 6.12: Percentage error between estimated and actual value for all downlink models for the V1b scenario

Table 6.5: MRE for Models in Scenario V1b

Model	MRE
Model 1	0.27609
Model 2	0.31863
Model 3	0.42601
Model 5	0.28165
Model 6	0.28069
Model 7	0.26092

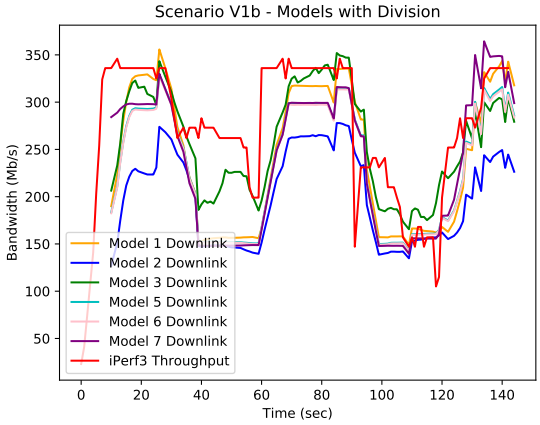

Figure 6.13: Six models divided by device amount for the V1b scenario

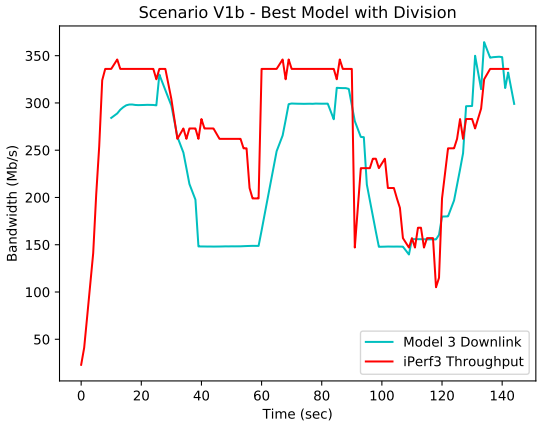

Figure 6.14: The model divided by device amount with lowest MRE value for the V1b scenario

Table 6.6: MRE for Models divided by Device Amount in Scenario V1b

Model	MRE
Model 1	0.18219
Model 2	0.29161
Model 3	0.14183
Model 5	0.20682
Model 6	0.20952
Model 7	0.19614

6.1.3 Scenario V1c

Here the models from the training scenario T1 are run on the data from Scenario V1c, which is the downlink traffic case with two devices and realistic streams.

The results of the models are together presented in Figure 6.15. Model 1 Downlink is found to have the lowest MRE value and is shown in Figure 6.16. The difference between the estimated and actual value is shown in Figure 6.17, and the percentage error between the estimated and actual value is presented in Figure 6.18. The MRE values for each model is shown in Table 6.7.

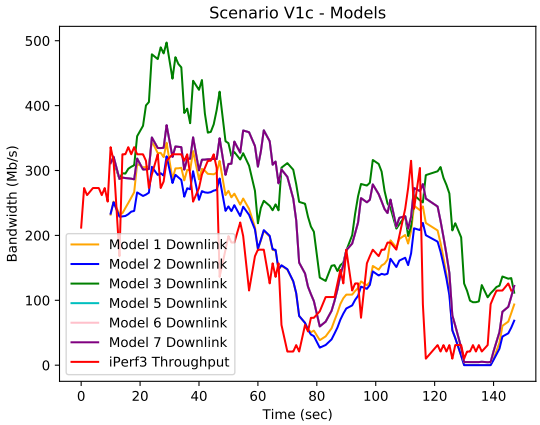

Figure 6.15: Six models for scenario V1c

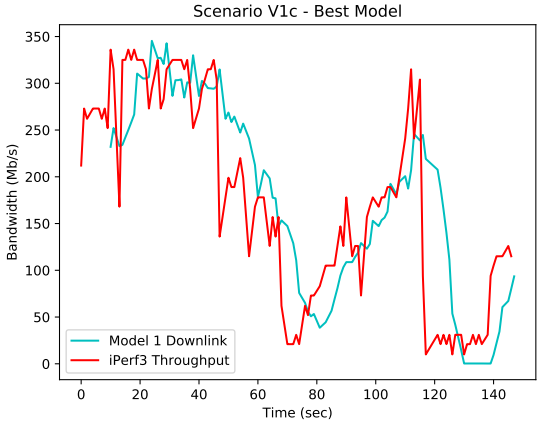

Figure 6.16: The model with lowest MRE value for the V1c scenario

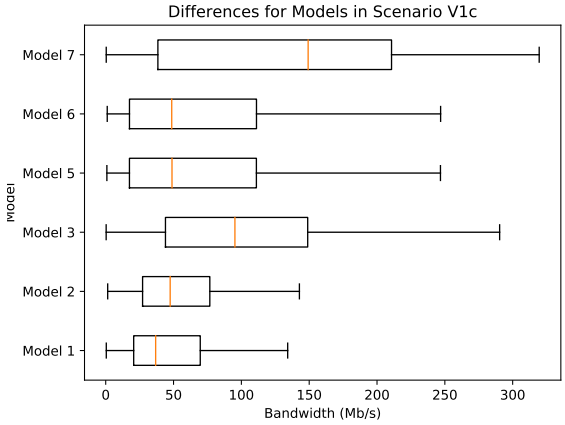

Figure 6.17: Differences between estimated and actual value for all downlink models for the V1c scenario

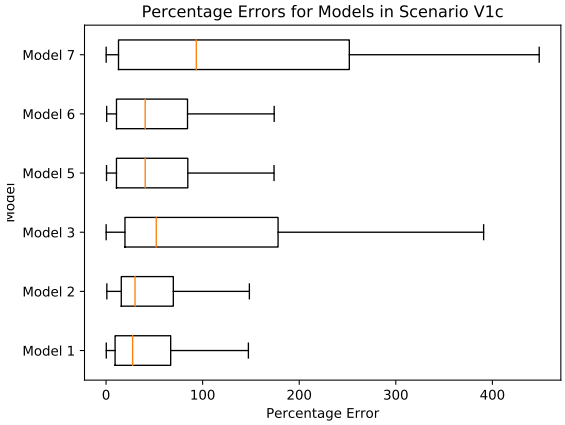

Figure 6.18: Percentage error between estimated and actual value for all downlink models for the V1c scenario

Table 6.7: MRE for Models in Scenario V1c

Model	MRE
Model 1	0.92619
Model 2	0.92729
Model 3	2.2195
Model 5	1.3727
Model 6	1.3733
Model 7	3.1548

6.2 Scenario T2

Here the models from the training scenario T2, which is the uplink traffic case with one device and a continuous stream, are presented. The counter that Model 2 uses

does not change for this scenario and was therefore not used. Table 6.8 shows the constants calculated for all the models trained on scenario T2.

Table 6.8: Parameters for Models in Scenario T2

Model	Coefficient
Model 1	$2.2910 * 10^{-4}$
Model 3	$7.4043 * 10^{-6}$
Model 4	$7.4049 * 10^{-6}$
Model 5	$2.6694 * 10^{-3}$
Model 6	$1.7895 * 10^{-4}$
Model 7	$7.2589 * 10^{-5}$

All models use a window size of 7 for the SMA. They are together presented in Figure 6.19. Model 7 Uplink is found to have the lowest MRE value and is shown in Figure 6.20. The difference between the estimated and actual value is shown in Figure 6.21, and the percentage error between the estimated and actual value is presented in Figure 6.22. The MRE values for each model is shown in Table 6.9.

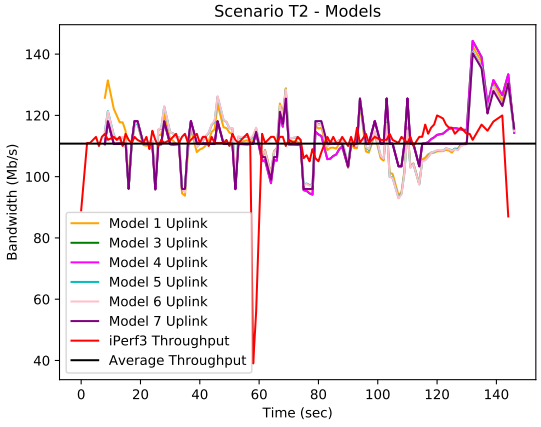

Figure 6.19: Six models for scenario T2

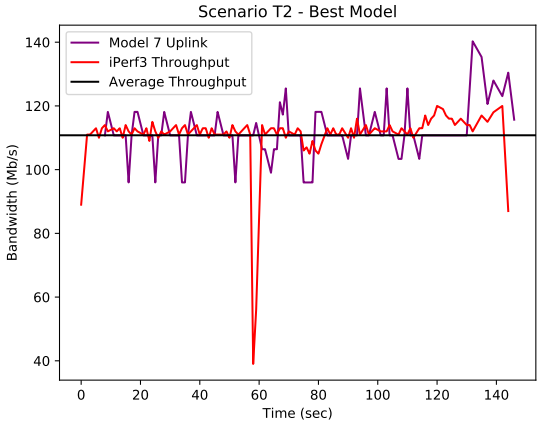

Figure 6.20: The model with lowest MRE value for the T2 scenario

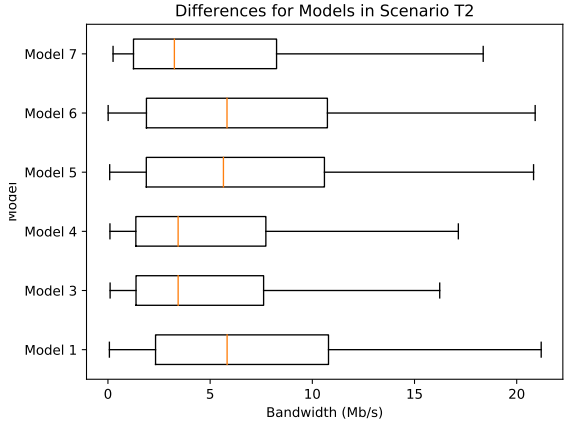

Figure 6.21: Differences between estimated and actual value for all uplink models for the T2 scenario

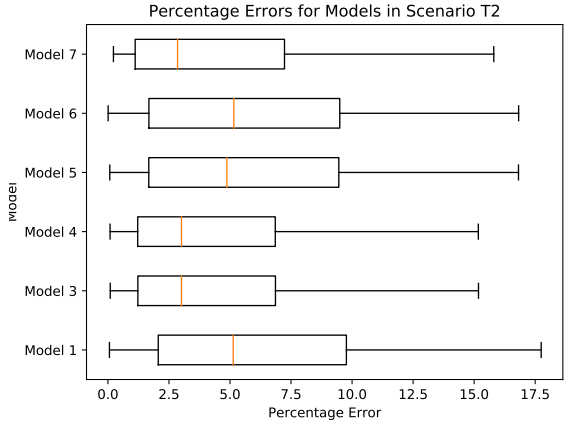

Figure 6.22: Percentage error between estimated and actual value for all uplink models for the T2 scenario

Table 6.9: MRE for Models in Scenario T2

Model	MRE
Model 1	0.091961
Model 3	0.078548
Model 4	0.078557
Model 5	0.088151
Model 6	0.088426
Model 7	0.078043

6.2.1 Scenario V2a

Here the models from the training scenario T2 are run on the data from Scenario V2a, which is the uplink traffic case with one device and a realistic stream.

The results of the models are together presented in Figure 6.23. Model 3 Uplink is found to have the lowest MRE value and is shown in Figure 6.24. The difference between the estimated and actual value is shown in Figure 6.25, and the percentage error between the estimated and actual value is presented in Figure 6.26. The MRE values for each model is shown in Table 6.10.

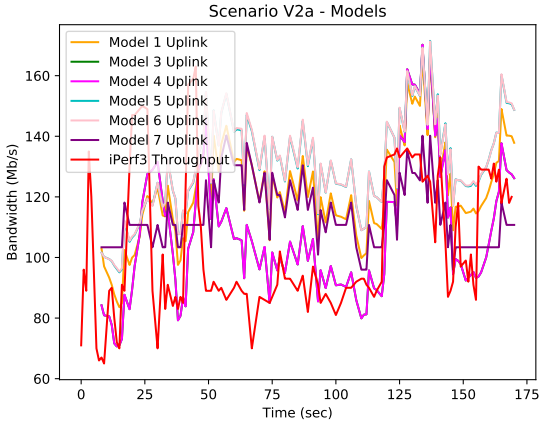

Figure 6.23: Six models for scenario V2a

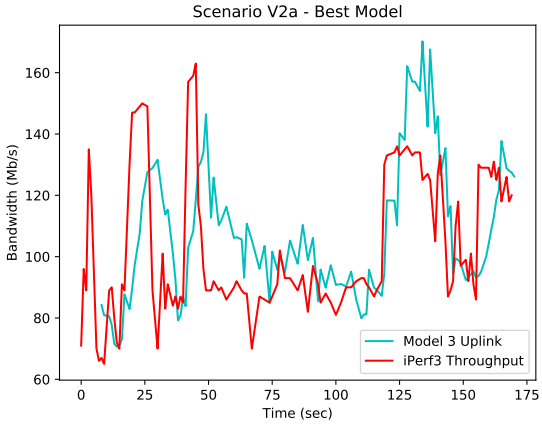

Figure 6.24: The model with lowest MRE value for the V2a scenario

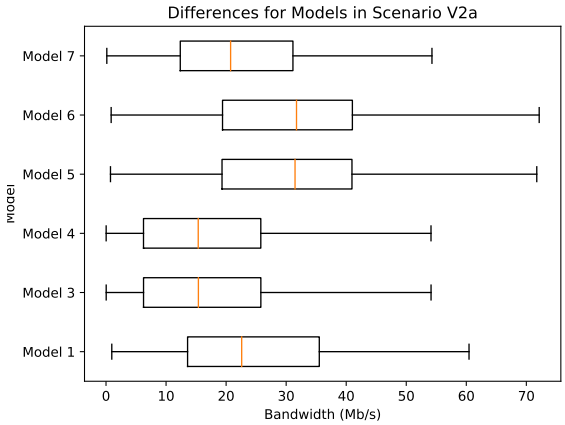

Figure 6.25: Differences between estimated and actual value for all uplink models for the V2a scenario

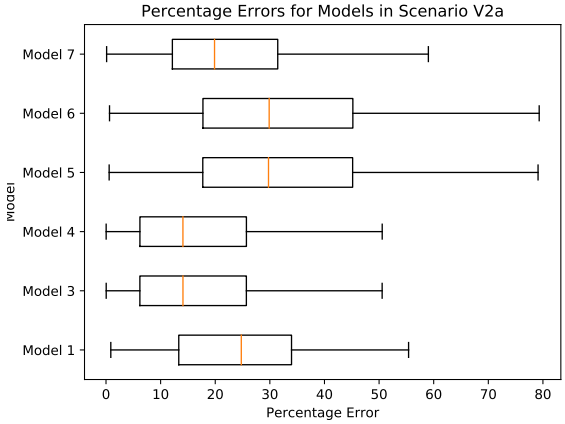

Figure 6.26: Percentage error between estimated and actual value for all uplink models for the V2a scenario

Table 6.10: MRE for Models in Scenario V2a

Model	MRE
Model 1	0.25587
Model 3	0.17034
Model 4	0.17035
Model 5	0.32251
Model 6	0.32328
Model 7	0.22849

6.2.2 Scenario V2b

Here the models from the training scenario T2 are run on the data from Scenario V2b, which is the uplink traffic case with two devices and continuous streams.

The results of the models are together presented in Figure 6.27. Model 4 Uplink is found to have the lowest MRE value and is shown in Figure 6.28. The difference between the estimated and actual value is shown in Figure 6.29, and the percentage error between the estimated and actual value is presented in Figure 6.30. The MRE values for each model is shown in Table 6.11.

By dividing the outputted value of the models by the amount of users actively sending data, the output of the models is shown in Figure 6.31. Model 4 has the lowest MRE value and is plotted in Figure 6.32. The MRE values for these models are presented in Table 6.12.

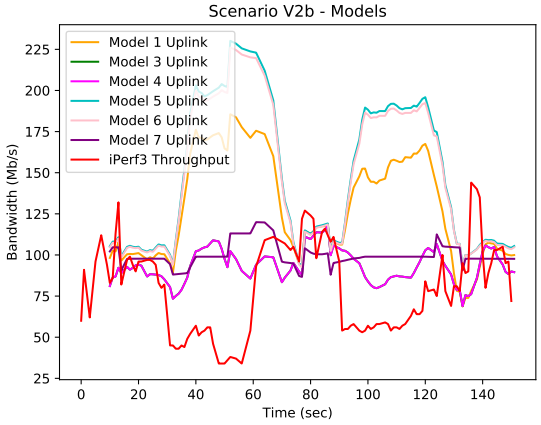

Figure 6.27: Six models for scenario V2b

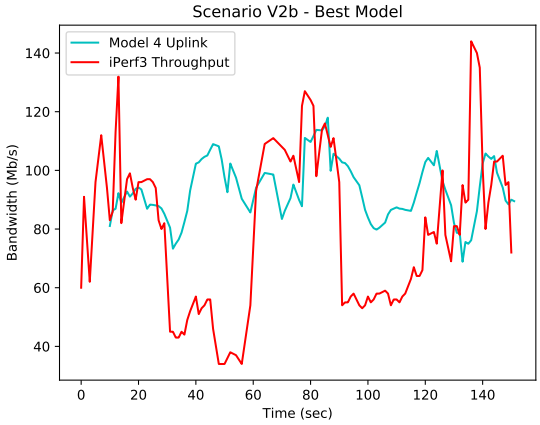

Figure 6.28: The model with lowest MRE value for the V2b scenario

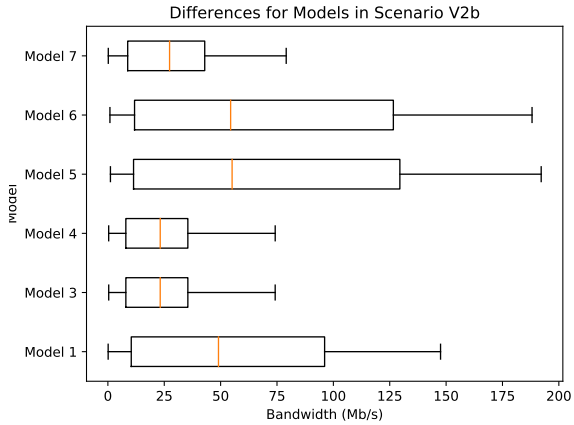

Figure 6.29: Differences between estimated and actual value for all uplink models for the V2b scenario

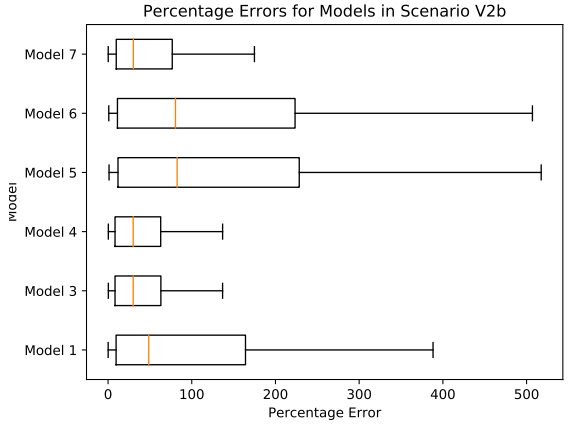

Figure 6.30: Percentage error between estimated and actual value for all uplink models for the V2b scenario

Table 6.11: MRE for Models in Scenario V2b

Model	MRE
Model 1	1.0023
Model 3	0.43650
Model 4	0.43633
Model 5	1.2974
Model 6	1.2689
Model 7	0.50522

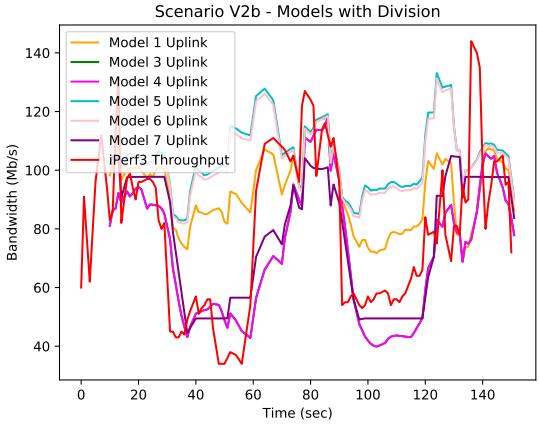

Figure 6.31: Six models divided by device amount for the V2b scenario

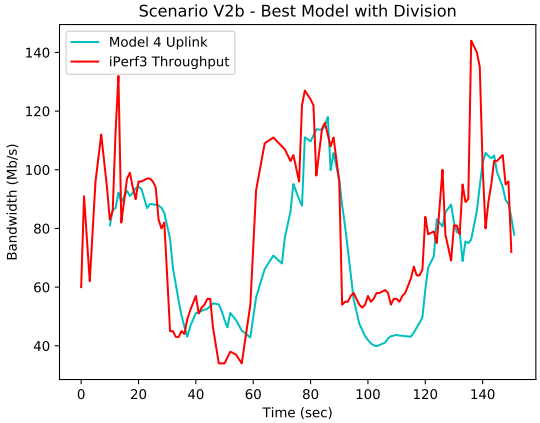

Figure 6.32: The model divided by device amount with lowest MRE value for the V2b scenario

Table 6.12: MRE for Models divided by Device Amount in Scenario V2b

Model	MRE
Model 1	0.34710
Model 3	0.18936
Model 4	0.18932
Model 5	0.50380
Model 6	0.48822
Model 7	0.19439

6.2.3 Scenario V2c

Here the models from the training scenario T2 are run on the data from Scenario V2c, which is the uplink traffic case with two devices and realistic streams.

The results of the models are together presented in Figure 6.33. Model 3 Uplink is found to have the lowest MRE value and is shown in Figure 6.34. The difference between the estimated and actual value is shown in Figure 6.35, and the percentage error between the estimated and actual value is presented in Figure 6.36. The MRE values for each model is shown in Table 6.13.

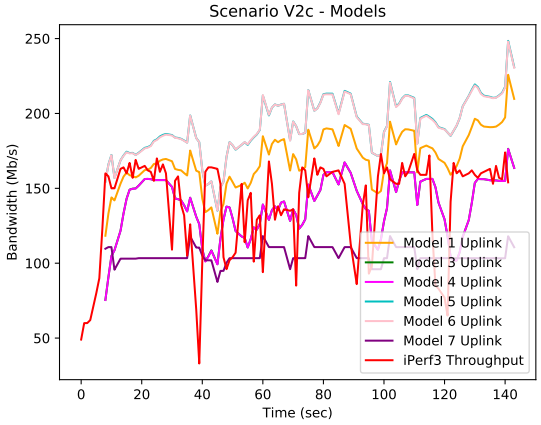

Figure 6.33: Six models for scenario V2c

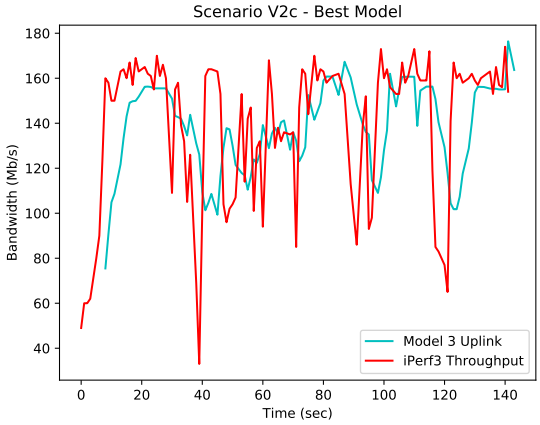

Figure 6.34: The model with lowest MRE value for the V2c scenario

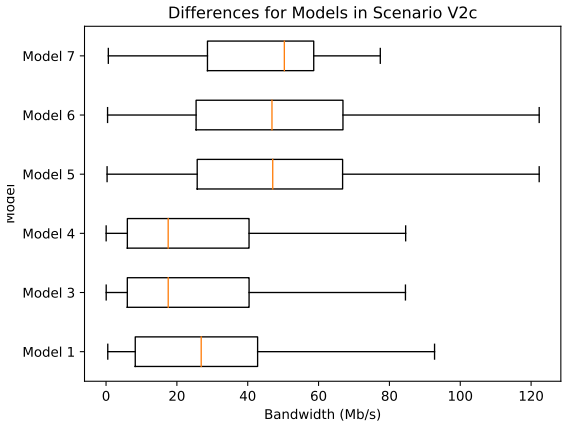

Figure 6.35: Differences between estimated and actual value for all uplink models for the V2c scenario

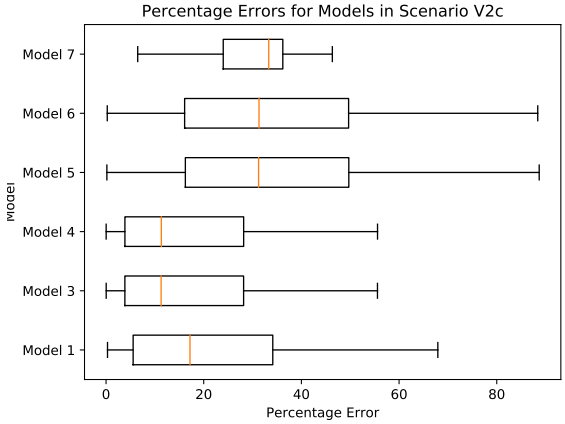

Figure 6.36: Percentage error between estimated and actual value for all uplink models for the V2c scenario

Table 6.13: MRE for Models in Scenario V2c

Model	MRE
Model 1	0.29678
Model 3	0.21304
Model 4	0.21306
Model 5	0.43380
Model 6	0.43257
Model 7	0.30518

6.3 Comparison

In Figure 6.37 the box plots display the differences between the predicted and actual bandwidth values for each downlink model applied to the three validation scenarios. These box plots show how much the estimated values deviate from the real measurements. The plots highlight the range of errors, including the median, which helps to understand how well the model performs across different scenarios.

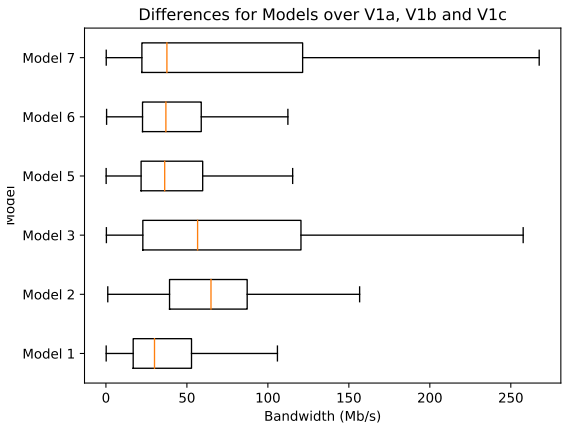

Figure 6.37: Box plots of differences for the downlink models validated over scenarios V1a, V1b and V1c

In Figure 6.38 the errors for the same models as presented in Figure 6.37 are shown, however they are presented in terms of how many percent they differ rather than by difference of pure bandwidth.

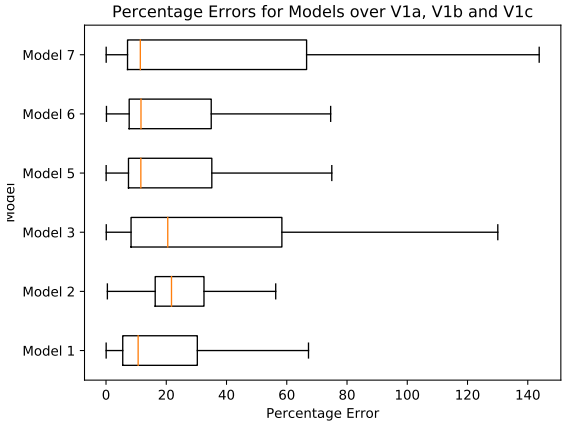

Figure 6.38: Box plots of percentage errors for the downlink models validated over scenarios V1a, V1b and V1c

In Figure 6.39 the box plots display the differences between the predicted and actual bandwidth values for each uplink model applied to the three validation scenarios.

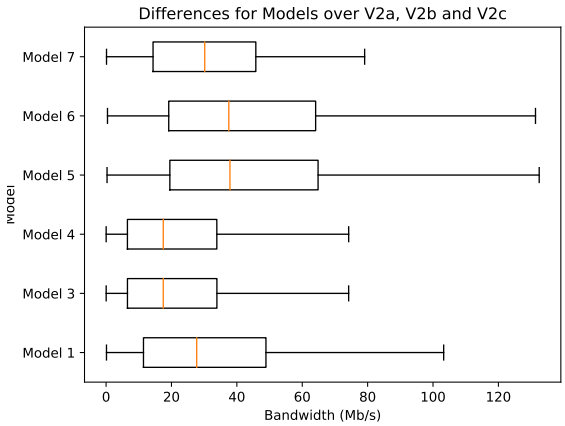

Figure 6.39: Box plots of differences for the uplink models validated over scenarios V2a, V2b and V2c

In Figure 6.40 the errors for the same models as presented in Figure 6.39 are shown, however they are presented in terms of how many percent they differ rather than by difference of pure bandwidth.

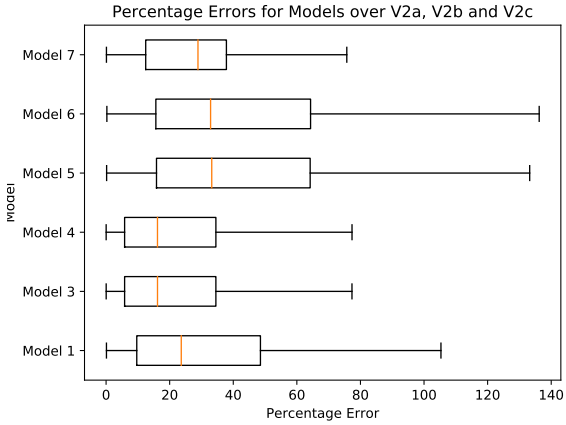

Figure 6.40: Box plots of percentage errors for the uplink models validated over scenarios V2a, V2b and V2c

6.4 Moving Average Comparison

Here the comparison in performance between the models using EWMA and SMA are presented. The downlink models using SMA had a window size of 9, and the uplink models used a window size of 7. For all models using EWMA, their smoothing factor was 0.2. In Table 6.14, the comparisons for the downlink models are shown, and in Table 6.15 the comparisons for uplink are shown. The averages are calculated over the validation scenarios.

Model	EWMA Difference	EWMA Percentage	SMA Difference	SMA Percentage
Model 1	43.865	37.164	36.494	25.118
Model 2	66.567	43.054	54.764	27.026
Model 3	78.235	88.158	72.894	67.448
Model 5	52.526	50.788	46.534	34.004
Model 6	52.596	50.791	46.520	33.908
Model 7	73.002	112.70	71.141	110.12

Table 6.14: Comparison of EWMA and SMA for all downlink models

Model	EWMA Difference	EWMA Percentage	SMA Difference	SMA Percentage
Model 1	35.010	49.169	36.561	52.891
Model 3	22.121	26.779	19.552	25.498
Model 4	22.135	26.792	19.556	25.500
Model 5	48.159	65.692	51.325	71.708
Model 6	47.632	64.807	50.791	70.760
Model 7	31.025	34.499	31.772	36.155

Table 6.15: Comparison of EWMA and SMA for all uplink models

6.5 Summary

To briefly summarize the model for each scenario that was found to have the lowest MRE value, these findings are concisely presented in Table 6.16. This table presents the nature of the linear relationship between the counter used as input for the model as well as the window size for the SMA required to run it.

Scenario	Model	Formula	Window Size
T1	Model 1 Downlink	$y = 7.0952 * 10^{-6} \Delta x$	9
V1a	Model 1 Downlink	$y = 7.0952 * 10^{-6} \Delta x$	9
V1b	Model 7 Downlink	$y = 3.2828 * 10^{-4} \Delta x$	9
V1c	Model 1 Downlink	$y = 7.0952 * 10^{-6} \Delta x$	9
T2	Model 7 Uplink	$y = 7.2589 * 10^{-5} \Delta x$	7
V2a	Model 3 Uplink	$y = 7.4043 * 10^{-6} \Delta x$	7
V2b	Model 4 Uplink	$y = 7.4049 * 10^{-6} \Delta x$	7
V2c	Model 3 Uplink	$y = 7.4043 * 10^{-6} \Delta x$	7

Table 6.16: Summary of model with lowest MRE for each scenario

7

Discussion

This chapter delves into the implications of the experimental findings and the performance of the proposed bandwidth estimation models. Topics such as the counters identified by SelectKBest, the correlations used by the models and the performance of the models for the various traffic scenarios are discussed in order to establish further insights.

7.1 Counters

Each model with the lowest MRE value for each traffic scenario uses one of these counters: `D_MAC_SUCCESS`, `U_PUSCH_TYPE_B`, `U_MAC_RESTRICTED` and `U_MAC_SUCCESS`. However, when looking at the rest of the counters found using SelectKBest, three additional counters are also included, namely: `D_MAC_DRB`, `D_PDCCH_TYPE_A` and `D_PDSCH_TYPE_A`.

All of these counters have in common that they track either resource block symbols or data volumes, both being directly connected to the bandwidth definition. As there is inherently a relationship between the amount of data sent and the bandwidth, it is not surprising that the counters are able to correlate to the bandwidth.

It is interesting to note that for the V1b scenario, which is a downlink scenario, the best model found uses an uplink counter, as can be shown in Figure 6.10. In itself this is not impossible, since iPerf3 was used with the TCP setting this means that for example ACKs were sent uplink meaning that there would have been some uplink traffic, though significantly less than the downlink traffic. However since none of the models run on the V1b scenario appear to show a clear correlation, meaning that it could simply be arbitrary that an uplink counter was chosen. For the three remaining downlink scenarios T1, V1a and V1c, Model 1 had the lowest MRE score, meaning that it is probably the best correlated counter for downlink.

For the uplink scenarios, all the models with the lowest MRE value use uplink counters. This result seems intuitive and makes sense since it is directly expected that uplink traffic is to affect uplink counters. It is possible that downlink counters could prove to have some correlation to uplink traffic for our scenarios since a TCP stream was used to estimate bandwidth.

Some counters and formulas mentioned in Appendix A are also investigated, but without any clear results. Even after using these formulas with the available data,

the analyses do not produce clear results. This lack of clear results indicates that further investigation is required to refine the models and improve the accuracy of the bandwidth estimation.

7.2 Correlations

The models presented in chapter 6 are all relatively successful in being able to correlate some counter to the bandwidth except for scenarios V1b and V2b, i.e. the scenarios with two devices running continuous streams. The correlations themselves are very simple as they are all able to estimate bandwidth by simple linear relationships with a multiplied coefficient. From this it can be established that there exists a simple and clear correlation between certain counters and bandwidth, at least when only generating traffic with one device. This also seems logical since the counters track data amounts of various kinds, the flow rate of which is inherently connected to the bandwidth. Therefore it makes sense that these correlations could be that simple. This is particularly clear since for the lowest MRE models for all scenarios except V1b and V2b the estimated bandwidth appears to track the measured bandwidth quite well and manages to react to sudden drops and increases in bandwidth.

The choice of window size for the SMA can be important to smooth out the curve enough to improve the accuracy by removing large fluctuations. At the same time making the window size too large could lead to longer divergence times and delays in changes of the estimated value. Therefore there exists a trade-off in the choice of window size that must be taken into consideration when using the models. In the case of the models created in this paper, the window size was kept the same for all models trained on the same training scenario. This is primarily because by using a SMA the trend of the estimation becomes more accurate and easier to analyze, and by keeping it the same for all models trained on the same scenario it is easier to compare the results between the models as well. Therefore it is very possible that the current window sizes are not optimal, which could be further investigated.

7.3 Traffic Scenarios

When looking at the downlink scenarios, the scenarios using two devices, i.e. V1b and V1c appear to have a larger error rate than the T1 and V1a scenarios. In Figure 6.11 the maximum difference is roughly between 100 and 175 Mb/s, except for model 3 which is around 300 Mb/s, and in Figure 6.17 it is approximately between 150 and 300 Mb/s. Comparing this to the plots for T1 in Figure 6.3 and V1a in Figure 6.7 their difference is significantly lower. For T1 it is roughly between 15 and 25 Mb/s and for V1a it is around 60-70 Mb/s for most models except for two models that are around 110-120 Mb/s. It makes sense that the models would perform better for the one device scenarios since they are trained on a one device scenario. The difference for T1 is trivial since the models are trained on that scenario. This also means that any relationship between the amount of devices and the bandwidth is

not represented in the models.

The same overall trend can be seen for uplink where the models perform better on the one device scenarios rather than the two device scenarios. For T2 the maximum difference is around 15-20 Mb/s as can be seen in Figure 6.21. For V2a in Figure 6.25 it is around 50-70 Mb/s. For V2b, the models using uplink counters are at around 75 Mb/s, whereas the downlink ones are between 150 and 200 Mb/s as shown in Figure 6.29. This is interesting because when looking at the plot of the models for V2b in Figure 6.27, models 1, 5 and 6 appear to be inversely correlated, since when the bandwidth decreases, the models increase. These three models interestingly use downlink counters, meaning that the downlink traffic appears to mirror uplink bandwidth. For V2c the models differ between 80 and 120 Mb/s as shown in Figure 6.35, which illuminates the fact that the error for the two device scenarios is higher than the one device scenarios.

7.4 Expanding to Several Max Bandwidth Users

Under the assumption that there is a fair share division of capacity in the base station, i.e. that multiple devices are entitled to the same amount of resources because they have the same 5QI, expanding the models to accurately estimate bandwidth for maximum bandwidth users appears trivial. Scenarios V1b and V2b are the only scenarios investigated that use more than one maximum bandwidth device. The primary models for these scenarios, presented in Figure 6.9 and Figure 6.27 don't correlate well to the bandwidth because their output is relatively constant, since they are only trained on a one device scenario. Therefore it is possible that the output corresponds to an aggregate bandwidth of all users in the base station, and assuming fair share each device should get the same amount of this available bandwidth. Therefore these models were altered by also dividing by the amount of currently transmitting users, which are presented in Figure 6.13 and Figure 6.31. They appear to track the bandwidth significantly better. In the case for V1b, the MRE of the best model almost halved when introducing the division of currently transmitting users. In the case for V2b, the MRE for the best model more than halved when introducing the division.

7.5 Expanding to a Virtual Speedtest

The proposed algorithm of this thesis could in turn be used to solve a similar problem, i.e. given a situation and counter values, what available bandwidth would a new device that is not yet transmitting get if it were to start transmitting data.

In the simplest case of only having one device transmitting data through the base station, our method can be used to first estimate the amount of data sent by the device. Assuming that all devices have the same 5QI, the bandwidth that a new device would be entitled to depends on the amount of data sent by the currently transmitting device. If that amount is lower than half of the physical bandwidth of the base station, then a new user is expected to be able to use the remaining

capacity of the base station. However if the transmitting user uses more than half of the physical bandwidth, then given a fair share scheduling, both users are entitled to the same amount of capacity, i.e. half of the physical capacity.

This can then be extended by also considering RSRP and RSRQ which gives an indication of the signal quality, which can in turn be used to estimate how the bandwidth of a device is affected by radio conditions as well.

To further expand this in the case of more devices transmitting data, the amount of bandwidth that can be ensured for a new device is very much dependent on whether the transmitting users are using more or less of their guaranteed share of capacity, as well as how many users are transmitting.

7.6 Moving Averages

SMA is primarily used for the majority of the results, however to get an indication of whether SMA or EWMA appear to perform better for the models, some limited tests have been created. All created models are run with both SMA and EWMA, whereupon the average difference and percentage error are calculated. For EWMA, all models used a smoothing factor of 0.2. For SMA the downlink models use a window size of 9 and the uplink models use 7. Box plots for EWMA for the different scenarios can be found in Appendix B.

For downlink, as presented in Table 6.14, SMA consistently outperformed EWMA for all models. Interestingly for uplink, as shown in Table 6.15, SMA no longer outperforms EWMA. 4 out of 6 models perform better with EWMA for uplink. There are several reasons why this could be the case. It could simply be that the window sizes and smoothing factors are not optimally selected, and therefore that the window sizes are better selected for downlink rather than uplink. It is also possible that SMA and EWMA work better for different types of trends. For example, without any moving average the fluctuations are larger for downlink models compared to uplink models. Therefore it is possible that SMA better captures the overall trend because it can easily eliminate the large fluctuations, but EWMA would be more prone to follow the quick changes, because it is more reactive to recent data points.

7.7 Applications

By having knowledge of the bandwidth for base stations, it could for instance be applied to the use case of route planning for self-driving vehicles. In the case of self-driving vehicles it is vital they have a good and stable connection in order to safely drive the route they are planning to take. However if it is possible to see that one base station ahead is very congested, it might be more reasonable for the vehicle to take a different route that appears to be able to provide a higher bandwidth, which can essentially lower the risk for faults and can increase the safety of the vehicle and those around. The method could also be used as a visual aid to drivers of normal

vehicles by showing which routes offer better connectivity.

A similar application is for drones, both autonomous and remotely controlled ones. If they are connected through 5G, it is important that they have a good connection in order to ensure them functioning correctly. Therefore in a similar way as for self-driving vehicles in that the proposed method of this paper can be used to estimate the load of different cells, which can in turn be used to ensure that the route a drone flies will have good connectivity.

7.8 Algorithm

It is worth mentioning that in order to keep the implementation of the algorithm shown in section 3.4 consistent with models presented in chapter 6, it starts calculating the bandwidth once it has stored enough values to correspond to the window size of the SMA and instead returns -1 until enough values are entered. There are however ways to change the algorithm in order to support outputs even before the window size is full. For example one could average all the previous values until the amount corresponding to the window size has been reached, essentially performing an SMA with an incrementing window size. It is not guaranteed that this would produce good results, since without an SMA, the output is very fluctuating, and gradually increasing the window size might not correspond well to the actual bandwidth for these values. Since the window size is relatively small and with the assumption that one value is inputted every second, not many seconds of values will be lost, meaning it will probably not result in major practical problems.

7.9 Ethical Considerations

Since what this thesis investigates is quite abstract, it is difficult to discuss the ethical considerations of being able to produce the bandwidth value itself. However, the ethical considerations become relevant when looking at the potential applications of bandwidth estimation as well as the implications these could have.

One can look at the application case mentioned above on using the estimation algorithm to plan routes for self-driving vehicles. In this case, a stable connection is vital for the vehicle to ensure its full functionality. Without it the risk of some malfunction to occur increases, which is not desirable due to it being in traffic around other cars. It can probably be assumed that the self-driving vehicles have other mechanisms in place in order to minimize harm done if something unexpected occurs. However if the vehicle takes a route that uses a very congested base station with barely any capacity left, it could create problems if there is a reported accident ahead of which the vehicle could have been warned if there is more capacity available.

7.10 Future Work

Since one limitation imposed on this work is to only look at the high-band frequencies, possible future work could relate to investigating possible relationships between counters and bandwidth for the mid-band frequencies instead.

It could also be of interest to look into whether the different counters are more accurate for certain intervals than others. If so, one could attempt to combine different counters to create a better estimation, for example in the case of one type of counter being particularly good at estimating decreases and another at estimating increases in bandwidth.

This thesis is done under the assumption that all devices are entitled to the same amount of bandwidth resources, however it could be of interest to look into 5QI and prioritization of packets to see how the knowledge of the 5QI classes of the devices could be used to estimate bandwidth for more realistic scenarios.

Additionally, using measurements like RSRP and RSRQ could improve these estimates by considering real-world factors like interference and signal coverage. For example a weak signal or high interference could reduce available bandwidth making the model more realistic by accounting for these physical challenges.

It could be of interest to investigate how the scheduler and its policy could be used to estimate bandwidth as well as how different policies affect the resulting bandwidth, since it is integral to deciding how packets are handled.

8

Conclusion

This thesis aims to explore the topic of bandwidth estimation for base stations in 5G networks using counters. This is investigated by looking into which counters appear to be best suited to use, how specifically they correlate to the bandwidth, as well as looking into different traffic scenarios. These research questions are concluded here:

1. **Which counters are more suitable to use as a basis for bandwidth estimation**

   For the two training scenarios, 5 different counters are selected using SelectKBest as the counters that potentially have the greatest correlation to the bandwidth. For all of the counters selected for these scenarios, there is an overlap that results in only 7 counters out of a possible 43. This means that these 7 counters consistently appear to show potential to correlate to bandwidth.

   These counters directly link to resource block symbols or data volumes which makes them naturally useful for estimating bandwidth, since they can be directly connected to the amount of data transmitted through the base station and thereby to the bandwidth definition.

2. **How do counters correlate to bandwidth**

   The analysis shows that the counters are able to correlate with bandwidth through simple linear relationships where the bandwidth is estimated by multiplying the derivative of the counter values by a coefficient. This means that as the values of the counters increase or decrease, the bandwidth adjusts accordingly in a predictable manner. This linear relationship makes sense because the counters that are found measure different aspects of data volumes, which is directly related to the definition of bandwidth.

   However, in the two scenarios V1b and V2b which are the only to contain more than one maximum bandwidth user, the bandwidth does not appear to be affected by the amount of devices transmitting data. This could be because the models track an aggregate bandwidth of all the devices connected to the base station, meaning that it rather outputs how much capacity is used overall. However assuming a fair share division of capacity between devices, simply dividing the output by the amount of currently transmitting users significantly improved the estimation accuracy.

3. **How do different network scenarios influence bandwidth estimation?**

Network conditions can affect how well bandwidth can be predicted. For example there appears to be a slightly larger error for scenarios with two devices compared to one device.

To conclude, the results of this thesis show that bandwidth estimation using 5G counters is not only possible, but can be found to correlate simply.

Bibliography

[1] *iPerf user docs*, https://iperf.fr/iperf-doc.php.

[2] T. Trainer, "4g: Fourth generation of mobile communication technologies," in *Telecom Trainer*, 2024, Online. [Online]. Available: https://www.telecomtrainer.com/4g-fourth-generation-of-mobile-communication-technologies/.

[3] A. Gillis, *Lte (long-term evolution)*, 2023. [Online]. Available: https://www.techtarget.com/searchmobilecomputing/definition/Long-Term-Evolution-LTE.

[4] Y. Hao, "Investigation and technological comparison of 4g and 5g networks," *Journal of Computer and Communications*, vol. 9, pp. 36–43, 2021.

[5] J. A. Adebusola, A. A. Ariyo, O. A. Elisha, A. M. Olubunmi, and O. O. Julius, "An overview of 5g technology," in *2020 International Conference in Mathematics, Computer Engineering and Computer Science (ICMCECS)*, IEEE, 2020, pp. 1–4.

[6] *5g spectrum bands explained low, mid and high band*, https://www.nokia.com/thought-leadership/articles/spectrum-bands-5g-world/.

[7] GSMA, *5g tdd synchronization and fdd overview*, 2021. [Online]. Available: https://www.gsma.com/connectivity-for-good/spectrum/wp-content/uploads/2021/03/5G-TDD-Synchronisation-QA.pdf.

[8] Telcoma, *5g multiple access techniques*, 2021. [Online]. Available: https://telcomaglobal.com/p/5g-multiple-access-techniques.

[9] *Scheduling*, https://www.3gpp.org/technologies/scheduling.

[10] *What is 5QI in 5G?* https://www.5gworldpro.com/uncategorized/what-is-5qi-in-5g.html.

[11] K. D, *Optimizing performance: Selectkbest for efficient feature selection in machine learning*, https://medium.com/@Kavya2099/optimizing-performance-selectkbest-for-efficient-feature-selection-in-machine-learning-3b635905ed48, 2023.

[12] K. Lai and M. Baker, "Measuring bandwidth," in *IEEE INFOCOM'99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No. 99CH36320)*, IEEE, vol. 1, 1999, pp. 235–245.

[13] G. Jin, "Algorithms and requirements for measuring network bandwidth," 2002.

[14] D. D. Perkins, H. D. Hughes, and C. B. Owen, "Factors affecting the performance of ad hoc networks," in *2002 IEEE International Conference on*

Communications. Conference Proceedings. ICC 2002 (Cat. No. 02CH37333), IEEE, vol. 4, 2002, pp. 2048–2052.

[15] D.-p. Wu, R.-y. Wang, and S. Huang, "The research on the factors affecting network bandwidth in mobile ad hoc networks," in *2010 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM)*, IEEE, 2010, pp. 1–4.

[16] A. Capone, L. Fratta, and F. Martignon, "Bandwidth estimation schemes for tcp over wireless networks," *IEEE transactions on Mobile Computing*, vol. 3, no. 2, pp. 129–143, 2004.

[17] M. Zhong, P. Hu, and J. Indulska, "Revisited: Bandwidth estimation methods for mobile networks," in *Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks 2014*, IEEE, 2014, pp. 1–6.

[18] M. Li, M. Claypool, and R. Kinicki, "Wbest: A bandwidth estimation tool for ieee 802.11 wireless networks," in *2008 33rd IEEE Conference on Local Computer Networks (LCN)*, IEEE, 2008, pp. 374–381.

[19] L. Mei, R. Hu, H. Cao, *et al.*, "Realtime mobile bandwidth prediction using lstm neural network and bayesian fusion," *Computer Networks*, vol. 182, p. 107 515, 2020.

[20] T. Azmin, M. Ahmadinejad, and N. Shahriar, "Bandwidth prediction in 5g mobile networks using informer," in *2022 13th International Conference on Network of the Future (NoF)*, IEEE, 2022, pp. 1–9.

[21] M. Zhang, X. Jiang, G. Jin, P. Li, and H. Chen, "Capradar: Real-time adaptive bandwidth prediction for dynamic wireless networks," *Computer Networks*, p. 109 865, 2023.

[22] L. Mei, J. Gou, Y. Cai, H. Cao, and Y. Liu, "Realtime mobile bandwidth and handoff predictions in 4g/5g networks," *Computer Networks*, vol. 204, p. 108 736, 2022.

[23] *Webrtc protocol*, `https://getstream.io/glossary/webrtc-protocol/`.

A

Appendix 1

In Table A.1 and Table A.2 lists and descriptions of the specific counters used are provided for uplink and downlink slot utilization.

- **PUSCH**

  PUSCH is the channel shared by multiple devices used in 5G networks to transmit data from the devices to a base station. It serves as a crucial pathway for users to send their information, supporting various applications such as uploading files or sending messages. PUSCH employs advanced modulation techniques to efficiently utilize the available spectrum and ensure reliable transmission of data.

- **PDSCH**

  PDSCH is the channel shared by multiple devices used in 5G networks to transmit data from a base station to a device. It plays a vital role in delivering data and content from the network to users supporting services such as web browsing and video streaming. PDSCH utilizes scheduling and resource allocation algorithms to optimize the delivery of data to multiple users simultaneously maximizing network efficiency and user experience.

- **HARQ**

  HARQ (Hybrid Automatic Repeat Request) is a mechanism within 5G networks that makes sure that data is received correctly. It works by combining error detection and correction with retransmission of erroneous data units.

- **QPSK**

  QPSK (Quadrature Phase Shift Keying) is a method used in digital communication to send data over radio waves. It works by changing the phase of the signal to represent different combinations of bits. Unlike some other methods that change both the phase and the amplitude. QPSK only adjusts the phase. Each different phase represents a specific combination of bits. With QPSK, 2 bits can be encoded for each change in phase, allowing more data to be sent efficiently.

- **QAM**

QAM (Quadrature Amplitude Modulation) is a type of digital modulation process, which essentially can convey two bit-streams simultaneously by changing the amplitudes of these bit-streams. The bit-streams are sent on the same frequency, but with a phase shift of 90°.

There are several types of QAM, such as 16-QAM, 64-QAM and 256-QAM, which allow more bits per symbol to be sent, where the preceding number corresponds to the amount of combinations of bits per symbol it can send.

- **DTX**

DTX (Discontinuous Transmission) refers to a technique used in wireless communication systems where a device temporarily switches off its transmitter when there is no data to send. This helps conserve battery power and reduces interference on the network. In relation to counters, DTX impacts the measurement of network activity because during periods of discontinuous transmission the device does not contribute to the overall traffic load. Therefore, counters tracking network usage or activity need to account for these periods of inactivity to provide an accurate representation of network performance and resource utilization, especially when DTX leads to no reception for a device. By understanding when DTX is active and inactive, network operators can better interpret counter data and optimize network resources accordingly.

Table A.1: Uplink Counters and Their Meaning

Name	Definition
US_x	Amount of successful HARQ transmissions for various modulations in uplink, including QPSK, 16-QAM, 64-QAM and 256-QAM
UN_x	Amount of occasions when no HARQ is received in the base station's uplink when the uplink grant is meant for HARQ transmissions of transport block that uses various modulations, including QPSK, 16-QAM, 64-QAM and 256-QAM, and DTX is considered the reason for no reception
UU_x	Amount of unsuccessful HARQ transmissions for various modulations in uplink, including QPSK, 16-QAM, 64-QAM and 256-QAM
UP	Total duration of slots where the base station's scheduler can schedule on PUSCH (each slot is 125ms)

Table A.2: Downlink Counters and Their Meaning

Name	Definition
DS_x	Amount of successful HARQ transmissions for various modulations in downlink, including QPSK, 16-QAM, 64-QAM and 256-QAM
DN_x	Amount of occasions when downlink HARQ feedback for a HARQ transmission using various modulations, including QPSK, 16-QAM, 64-QAM and 256-QAM, could not be detected from UE
DU_x	Amount of unsuccessful downlink HARQ transmissions for various modulations, including QPSK, 16-QAM, 64-QAM and 256-QAM
DP	Total duration of slots where the base station's scheduler can schedule on PDSCH (each slot is 125ms)

Using these counters, two formulas have been provided by our industrial partners that can aid in the estimation of bandwidth, one for uplink in Equation A.1 and one for downlink in Equation A.2. These formulas calculate the slot utilization, i.e., how large part of all available slots that are being used for PUSCH and PDSCH respectively. The set M consists of the values {QPSK, 16-QAM, 64-QAM, 256-QAM}, which is here shortened as {Q, 16, 64, 256}.

$$\text{SlotUtilUp} = \frac{\sum_{x \in M} US_x + UN_x + UU_x}{UP} \tag{A.1}$$

$$\text{SlotUtilDown} = \frac{\sum_{x \in M} DS_x + DN_x + DU_x}{DP} \tag{A.2}$$

The eight following tables present the counter values, corresponding bandwidth and calculated slot utilization for four traces, for which the counter definitions are presented in Table A.1 and Table A.2. One time step in each tables is one second.

The main purpose of the two equations for slot utilization is to investigate if multiplying their output by the theoretical maximum for up- and downlink would correspond to the bandwidth. This is plotted in Figure A.1 and Figure A.2, where an SMA has been applied to the actual bandwidth to potentially improve vision of the actual bandwidth trend.

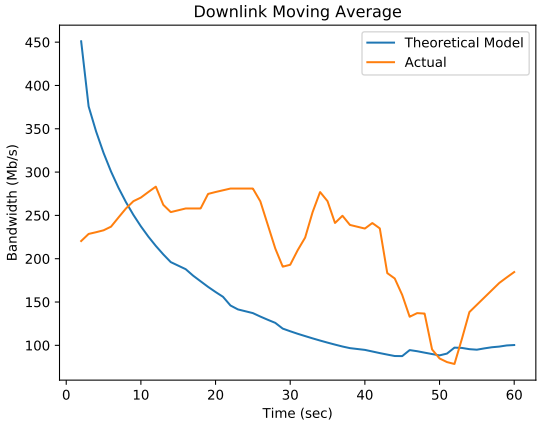

Figure A.1: Simple slot utilization model for a downlink trace

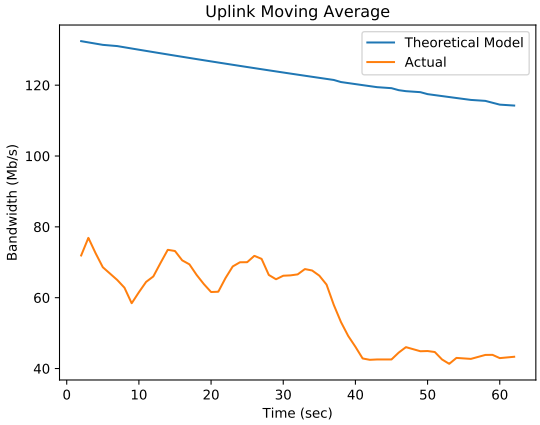

Figure A.2: Simple slot utilization model for an uplink trace

The tables below contain all measurements performed to test the two formulas mentioned.

Table A.3: Uplink Trace 1

Time (s)	US_Q	US_{16}	US_{64}	US_{256}	UN_Q	UN_{16}	UN_{64}	UN_{256}	UU_Q	UU_{16}	UU_{64}	UU_{256}	UP	Bandwidth (Mb/s)
0	51675	0	130528	133900	0	0	5	2565	442	0	16197	38	832000	105
1	51675	0	131927	134096	0	0	5	2565	442	0	16197	38	833600	136
2	51675	0	133340	134283	0	0	5	2565	442	0	16197	38	835200	147
3	51675	0	134778	134440	0	0	5	2565	442	0	16197	38	836800	126
4	51675	0	136210	134608	0	0	5	2565	442	0	16197	38	838400	147
5	51675	0	137669	134749	0	0	5	2565	442	0	16197	38	840000	136

Table A.4: Uplink Trace 1 Slot Utilization

Time (s)	Slot Utilization (%)
0	40.30649038461539
1	40.42046545105566
2	40.534602490421456
3	40.647705544933075
4	40.76097328244275
5	40.87380952380953

Table A.5: Uplink Trace 2

Time (s)	US_Q	US_{16}	US_{64}	US_{256}	UN_Q	UN_{16}	UN_{64}	UN_{256}	UU_Q	UU_{16}	UU_{64}	UU_{256}	UP	Bandwidth (Mb/s)
0	70195	0	222387	145250	31	0	5	4145	688	0	21933	56	1305600	62
1	70211	0	223796	145250	31	0	5	4145	688	0	22104	56	1307200	73
2	70217	0	225232	145250	31	0	5	4145	688	0	22259	56	1308800	62
3	70231	0	226652	145250	31	0	5	4145	688	0	22424	56	1310400	62
4	70437	0	227893	145250	31	0	5	4145	688	0	22573	56	1312000	73
5	70965	0	228823	145250	31	0	5	4145	688	0	22714	56	13136006	52

Table A.6: Uplink Trace 2 Slot Utilization

Time (s)	Slot Utilization (%)
0	35.592064950980394
1	35.67059363525092
2	35.749006723716384
3	35.82738095238095
4	35.90533536585366
5	35.98332825822168

Table A.7: Downlink Trace 1

Time (s)	DN_Q	DN_{16}	DN_{64}	DN_{256}	DU_Q	DU_{16}	DU_{64}	DU_{256}	DS_Q	DS_{16}	DS_{64}	DS_{256}	DP	Bandwidth (Mb/s)
0	1372	0	88585	2994	0	0	0	0	6	0	2362	0	957700	208
1	1372	0	94504	2994	0	0	0	0	6	0	2542	0	963800	213
2	1372	0	100439	2994	0	0	0	0	6	0	2707	0	969900	227
3	1372	0	112339	2994	0	0	0	0	6	0	3007	0	982100	235
4	1372	0	118292	2994	0	0	0	0	6	0	3154	0	988200	245
5	1372	0	124250	2994	0	0	0	0	6	0	3296	0	994300	242

Table A.8: Downlink Trace 1 Slot Utilization

Time (s)	Slot Utilization (%)
0	9.952908008771013
1	10.522722556547002
2	11.085472729147336
3	12.19000101822625
4	12.73203804897794
5	13.267424318616111

Table A.9: Downlink Trace 2

Time (s)	DN_Q	DN_{16}	DN_{64}	DN_{256}	DU_Q	DU_{16}	DU_{64}	DU_{256}	DS_Q	DS_{16}	DS_{64}	DS_{256}	DP	Bandwidth (Mb/s)
0	83870	0	241625	23552	0	0	0	146	793	0	7929	30	3196401	282
1	84675	0	246759	23552	0	0	0	146	793	0	8090	30	3202501	277
2	85921	0	251435	23552	0	0	0	146	794	0	8267	30	3208601	283
3	85972	0	257306	23552	0	0	0	146	794	0	8445	30	3214701	309
4	88966	0	260254	23552	0	0	0	146	797	0	8600	30	3220800	294
5	90322	0	264846	23552	0	0	0	146	798	0	8751	30	3226901	278

Table A.10: Downlink Trace 2 Slot Utilization

Time (s)	Slot Utilization (%)
0	11.198375923421374
1	11.367521821226598
2	11.536024578936427
3	11.703887857688787
4	11.871118976651763
5	12.03771048445552

B

Appendix 2

Here the models trained on the two training scenarios use an EWMA with a smoothing factor of 0.2. In Figure B.1 the box plots display the differences between the predicted and actual bandwidth values for each downlink model using EWMA applied to the three validation scenarios.

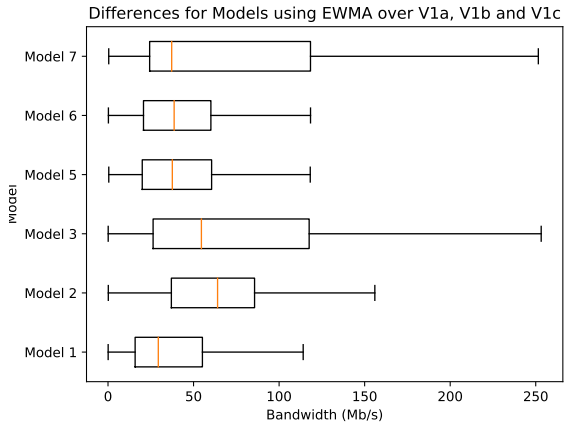

Figure B.1: Box plots of differences for the downlink models using EWMA validated over scenarios V1a, V1b and V1c

In Figure B.2 the box plots display the percentage error between the predicted and actual bandwidth values for each downlink model using EWMA applied to the three validation scenarios.

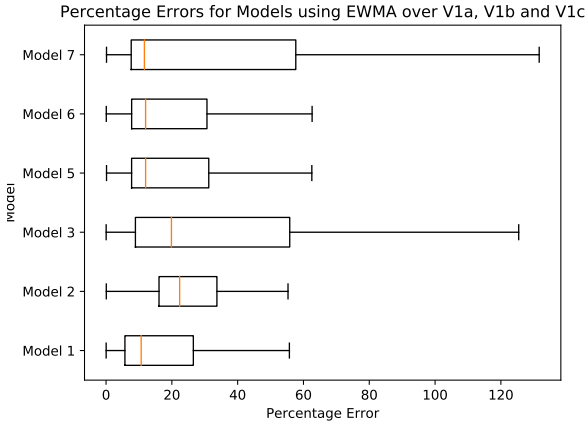

Figure B.2: Box plots of percentage errors for the downlink models using EWMA validated over scenarios V1a, V1b and V1c

In Figure B.3 the box plots display the differences between the predicted and actual bandwidth values for each uplink model using EWMA applied to the three validation scenarios.

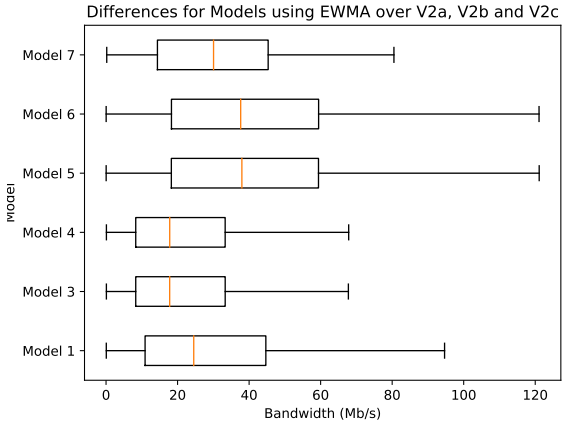

Figure B.3: Box plots of differences for the uplink models using EWMA validated over scenarios V2a, V2b and V2c

In Figure B.4 the box plots display the percentage error between the predicted and actual bandwidth values for each uplink model using EWMA applied to the three validation scenarios.

Figure B.4: Box plots of percentage errors for the uplink models using EWMA validated over scenarios V2a, V2b and V2c