Blixt™: An Available Bandwidth Measurements’ Approach for High-Speed Mobile Networks

Al-Hussein Hameed Jasim

Computer Science and Engineering, master's level (120 credits)
2018

Luleå University of Technology
Department of Computer Science, Electrical and Space Engineering
Luleå University of Technology
Department of Computer Science, Electrical and Space Engineering
Erasmus Mundus Master’s Programme in Pervasive Computing & Communications for Sustainable Development (PERCCOM)

Al-Hussein Hameed Jasim

BLIXT™: AN AVAILABLE BANDWIDTH MEASUREMENTS’ APPROACH FOR HIGH-SPEED MOBILE NETWORKS

Examiners:  Professor Eric Rondeau (University of Lorraine)
            Professor Jari Porras (Lappeenranta University of Technology)
            Associate Professor Karl Andersson (Luleå University of Technology)

Supervisors:  Associate Professor Karl Andersson (Luleå University of Technology)
               Specialist, Network and Protocols Niclas Ögren (InfoVista)
This thesis is prepared as part of a European Erasmus Mundus programme PERCCOM - Pervasive Computing and Communications for Sustainable Development [1].

This thesis has been accepted by partner institutions of the consortium (cf. UDL-DAJ, n°1524, 2012 PERCCOM agreement).

Successful defense of this thesis is obligatory for graduation with the following national diplomas:

- Master in Complex Systems Engineering (University of Lorraine)
- Master of Science in Technology (Lappeenranta University of Technology)
- Master of Science in Computer Science and Engineering, specialization in Pervasive Computing and Communications for Sustainable Development (Luleå University of Technology)
ABSTRACT

Luleå University of Technology
Department of Computer Science, Electrical and Space Engineering
Erasmus Mundus Master’s Programme in Pervasive Computing & Communications for Sustainable Development (PERCCOM)

Al-Hussein Hameed Jasim

Blixt™: An Available Bandwidth Measurements’ Approach for High-Speed Mobile Networks

Master’s Thesis
84 pages, 21 figures, 2 tables, 1 appendix

Examiners: Professor Eric Rondeau (University of Lorraine)
Professor Jari Porras (Lappeenranta University of Technology)
Associate Professor Karl Andersson (Luleå University of Technology)

Keywords: Network Performance, Active Measurements, Available Bandwidth, 4G/LTE

Mobile networks are in the process of becoming the world’s leading medium for data traffic. This challenge has raised the bar for Quality of Services (QoS) provided by the mobile network operators. It requires methods and tools to verify the Service Level Agreement (SLA) and benchmark competitors by comparing metrics of QoS, e.g. the round-trip time and available bandwidth. For this purpose, Blixt™ has been developed, which is a property of InfoVista Sweden AB. Blixt™ is an Android application which measures the available bandwidth and the round-trip time for the latest generation of mobile networks. Blixt™ approach relies on a time-stamping protocol commonly known as Two-Way Active Measurement Protocol (TWAMP). This research work discusses how the packet probing parameters affect the accuracy of measurements and the level of intrusiveness. The performance of the technique was experimentally tested and compared to other tools and methods, namely, iPerf3, nPerf and FTP test.
ACKNOWLEDGEMENTS

First and foremost, I want to express my sincere gratitude to my thesis supervisors, Associate Professor Karl Andersson (Luleå University of Technology) and Niclas Ögren (networks specialist at InfoVista) for their continual support, valuable feedback, knowledge sharing, and guidance throughout the research.

I would like to thank the PERCCOM consortium for giving me the opportunity to participate in this program. Special thanks to the professors, Eric Rondeau, Jean-Philippe Georges, Jari Porras and Karl Andersson, for the outstanding organization of the academic and cultural activities. All thanks for the professors from ITMO University and Leeds Beckett University for the seminars and hosting the summer school.

I would also like to take this opportunity to thank Dimitar Minovski (PhD candidate) for the inspiring discussions.

Many thanks to my family who always supports and believes in me and in whatever I undertake.

Before concluding, I would like to thank all of my cohort mates, Ahmed, Anara, Asif, Carl, Daniyal, Farniba, Furkat, Geraldine, Germain, Ida, Jonathan, Kristian, Madhubala, Marcel, Mina, Mukhammadjon, Phoebe, Roman, Xhesica and Tawseef, for the memorable moments and the wonderful time we spent along the journey.

Skellefteå, August 15, 2018
Al-Hussein Hameed Jasim
# TABLE OF CONTENTS

1 Introduction ........................................................................................................................................... 6
   1.1 Introduction..................................................................................................................................... 6
   1.2 Problem Definition ......................................................................................................................... 8
   1.3 Research Motivation ....................................................................................................................... 9
   1.4 Research Aim and Questions .......................................................................................................... 10
   1.5 Research Scope ............................................................................................................................. 10
   1.6 Research Methodology .................................................................................................................. 11
   1.7 Research Contributions .................................................................................................................. 11
   1.8 Thesis Outline ............................................................................................................................... 12

2 Background And Related Work ................................................................................................................ 13
   2.1 Bandwidth-Related Metrics ............................................................................................................. 13
      2.1.1 Capacity .................................................................................................................................. 13
      2.1.2 Available Bandwidth ............................................................................................................... 13
      2.1.3 Throughput ........................................................................................................................... 15
      2.1.4 Bulk Transfer Capacity ......................................................................................................... 15
   2.2 Applications of ABM ...................................................................................................................... 16
   2.3 Developers’ View of Network QoS .................................................................................................. 16
   2.4 Measurements Methods .................................................................................................................. 17
      2.4.1 Passive Measurements ........................................................................................................... 17
      2.4.2 Active Measurements ............................................................................................................ 18
      2.4.3 Hybrid Measurements ............................................................................................................ 20
   2.5 Packet Probing Models ..................................................................................................................... 20
      2.5.1 Probe Gap Model ................................................................................................................... 20
      2.5.2 Probe Gap Model ................................................................................................................... 21
   2.6 Packet Probing Techniques ............................................................................................................... 22
      2.6.1 The Variable Packet Size ....................................................................................................... 23
      2.6.2 The Packet Pair/Train Dispersion ......................................................................................... 23
      2.6.3 The Self-Loading Periodic Streams ....................................................................................... 24
      2.6.4 The Train of Packet Pairs ..................................................................................................... 24
2.6.5 The Packet Tailgating ................................................................. 25
2.7 Filters .......................................................................................... 25
  2.7.1 The Moving Average Filter ..................................................... 25
  2.7.2 The Exponentially Weighted Moving Average Filter ............... 26
  2.7.3 The Vertical Horizontal Filter ................................................ 26
  2.7.4 Kalman Filter .......................................................................... 27
  2.7.5 Fuzzy Filter ............................................................................. 27
2.8 Comparative Studies .................................................................... 28

3 ABM in Cellular Networks................................................................. 31
  3.1 Cellular Networks ....................................................................... 31
  3.2 LTE .............................................................................................. 32
  3.3 QoS in LTE .................................................................................. 34
  3.4 LTE Features ............................................................................... 35
  3.5 Bandwidth Estimation Approaches for Mobile Networks .......... 36
    3.5.1 Packet Probing Method .......................................................... 36
    3.5.2 Packet’s RTT ......................................................................... 36
    3.5.3 Download Based Technique .................................................. 37
  3.6 ABM State-Of-The-Art Tools For High-Speed Cellular Networks .... 38
    3.6.1 Gping-Pair ............................................................................ 38
    3.6.2 Correlation Coefficient Method ............................................. 39
    3.6.3 NEXT .................................................................................... 40
    3.6.4 NEXT-V2 .............................................................................. 41
    3.6.5 NEXT-FIT ............................................................................. 41
    3.6.6 A Proxy-Based Approach ...................................................... 41
    3.6.7 PathQuick ............................................................................ 42
    3.6.8 TPG ...................................................................................... 42
    3.6.9 PathML ................................................................................ 43
  3.7 The Challenges and Uncertainties ................................................ 44

4 Implementation: Protocols And Technologies ................................. 47
  4.1 A One-Way Active Measurement Protocol (OWAMP) .............. 47
  4.2 A Two-Way Active Measurement Protocol (TWAMP) .............. 48
4.2.1 TWAMP Full Mode .......................................................................................... 49
4.2.2 TWAMP Light Mode ..................................................................................... 50
4.3 Android OS and Supported APIs ...................................................................... 50
4.4 Blixt™ Algorithm and Technique ................................................................... 52
4.5 Blixt™ Measurement Method ........................................................................... 52
   4.5.1 Blixt™ Client .............................................................................................. 54
   4.5.2 Blixt™ Server ............................................................................................ 55
4.6 Packet Probing Parameters .............................................................................. 56
   4.6.1 Probing Packet Size .................................................................................. 56
   4.6.2 The Number of Probing Packets ............................................................... 57
   4.6.3 Transmission Time Interval ....................................................................... 57

5 Experiments, Results and Discussion .................................................................. 58
   5.1 The Measurements ....................................................................................... 58
      5.1.1 Packet Probing Parameters Study ......................................................... 60
      5.1.2 Performance Evaluation ....................................................................... 65
   5.2 Discussion ..................................................................................................... 68

6 Conclusion and Future Work ............................................................................... 70

References ........................................................................................................... 72

Appendix
1. Results from Experiments: iPerf 3 Throughput Measurements
LIST OF FIGURES

Fig. 1. Mobile data traffic per application ................................................................. 7
Fig. 2. Tight link and narrow link in multi-hop network .............................................. 14
Fig. 3. Hybrid measurements example ......................................................................... 20
Fig. 4. Turning point in PRM ...................................................................................... 22
Fig. 5. Network topology for Gping-pair experiments ................................................. 39
Fig. 6. A representation of RF limitations in cellular networks ..................................... 45
Fig. 7. TWAMP architecture, full mode ....................................................................... 49
Fig. 8. Android architecture ....................................................................................... 50
Fig. 9. Train of packets ............................................................................................... 53
Fig. 10. Blixt™’s algorithm class diagram ................................................................. 55
Fig. 11. Blixt™’s client UI ........................................................................................... 56
Fig. 12. Mobile coverage at Skellefteå Kraft Arena .................................................. 58
Fig. 13. Blixt™ measurements network topology ....................................................... 59
Fig. 14. Blixt™’s measurements for different values of packet’s size ......................... 60
Fig. 15. Blixt™ measurements versus FTP throughput for different number of probing
        packets: a) 20 Packet; b) 500 packets; c) 1000 packets; d) 4000 packets .......... 62
Fig. 16. Blixt™ measurement for different values of TTI: a) 500 ms; b) 1000 ms; c) 2000
        ms; d) 5000 ms; e) 10000 ms .............................................................................. 64
Fig. 17. iPerf 3 measurement network topology ........................................................ 65
Fig. 18. iPerf 3 measurement network topology ....................................................... 66
Fig. 19. Intrusiveness level compared to FTP ............................................................. 66
Fig. 20. Blixt train of packets vs FTP download data traffic ....................................... 67
Fig. 21. Sustainability analysis of Blixt™ .................................................................... 69

LIST OF TABLES

Table 1. Taxonomy of available bandwidth-related measurement tools [13] ............... 30
Table 2. The packet probing parameters’ values used for the experiments .................. 64
### LIST OF SYMBOLS AND ABBREVIATIONS

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABM</td>
<td>Available Bandwidth Measurement</td>
<td></td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
<td></td>
</tr>
<tr>
<td>BART</td>
<td>Bandwidth Available in Real Time</td>
<td></td>
</tr>
<tr>
<td>BTC</td>
<td>Bulk Transfer Capacity</td>
<td></td>
</tr>
<tr>
<td>FDD</td>
<td>Frequency Division Duplexing</td>
<td></td>
</tr>
<tr>
<td>FTP</td>
<td>File Transfer Protocol</td>
<td></td>
</tr>
<tr>
<td>IoT</td>
<td>Internet of Things</td>
<td></td>
</tr>
<tr>
<td>LTE</td>
<td>Long Term Evolution</td>
<td></td>
</tr>
<tr>
<td>MIMO</td>
<td>Multiple Input Multiple Output</td>
<td></td>
</tr>
<tr>
<td>OFDMA</td>
<td>Orthogonal Frequency Division Multiple Access</td>
<td></td>
</tr>
<tr>
<td>OWAMP</td>
<td>One-Way Active Measurement Protocol</td>
<td></td>
</tr>
<tr>
<td>PGM</td>
<td>Probe Gap Model</td>
<td></td>
</tr>
<tr>
<td>PRM</td>
<td>Probe Rate Model</td>
<td></td>
</tr>
<tr>
<td>QoE</td>
<td>Quality of Experience</td>
<td></td>
</tr>
<tr>
<td>QoS</td>
<td>Quality of Service</td>
<td></td>
</tr>
<tr>
<td>RTT</td>
<td>Round-Trip Time</td>
<td></td>
</tr>
<tr>
<td>SLA</td>
<td>Service Level Agreement</td>
<td></td>
</tr>
<tr>
<td>TCP</td>
<td>Transmission Control Protocol</td>
<td></td>
</tr>
<tr>
<td>TTI</td>
<td>Transmission Time Interval</td>
<td></td>
</tr>
<tr>
<td>TWAMP</td>
<td>Two Way Active Measurement Protocol</td>
<td></td>
</tr>
<tr>
<td>UDP</td>
<td>User Datagram Protocol</td>
<td></td>
</tr>
<tr>
<td>VPS</td>
<td>Variable Packet Size</td>
<td></td>
</tr>
<tr>
<td>3GPP</td>
<td>Third Generation Partnership Project</td>
<td></td>
</tr>
</tbody>
</table>
1 Introduction

This chapter introduces a brief introduction about the latest facts and figures concerning the high-speed mobile networks and their high demand. It also describes the problem intended to be solved, the research aim and the questions that come with as well as the motivation behind this research. This chapter also proceeds with the methodology followed and the contributions made by the research and what delimitates it. The thesis outline concludes this chapter.

1.1 Introduction

The phenomenal evolution of mobile data as a result of the rapid progress in smart handset industry, tablets, mobile broadband modems, and the development of mobile applications in an unprecedented manner overwhelms current cellular networks. These advanced portable devices and user friendly applications, lead to an enormous demand on the networks which in turn needs to expand their capacity to meet the challenge [2] [3].

During the last decade, the world has witnessed an exceptional development in cellular network technologies from High-Speed Packet Access (HSPA) to the 4G Long Term Evolution-Advanced (LTE-A) system [4]. At the end of 2017, LTE became the predominant mobile access technology in the world [5]. According to Ericsson mobility report, the number of LTE subscriptions continues to grow significantly, and it is predicted to reach 5.5 billion by the end of 2023, which means it will reach about 60 percent of the total number of subscriptions. The launch of the fifth generation (5G) networks is closer than ever. The mobile network operators in the United States are expected to be among the first to commercially launch these networks at the end of this year or the middle of 2019 [5].

Ericsson’s latest forecast continues with the predicted data traffic per application, video applications are expected to form 73 percent of the total mobile data exchange in 2023. The forecast is based on the current data and the data collected previously which are validated with considerable network measurements.
People's preference for the networks which are cost-effective and richer in content over traditional telephony services such as the SMS and voice calls, is the reason for increased mobile data traffic. The main motivation associated with this shift is the need for more bandwidth with less latency [6]. The emergence of LTE which was followed by LTE-A has made it possible to have higher capacity and minimized delay in accessing services (in all of their forms, data, voice and video) wirelessly, at anytime, anywhere [6]. The promising advancement contributes to making Internet of Things (IoT) more accessible to all by using real-time applications via mobile devices to control machines [6], especially through 4G networks and possibly over 5G networks shortly. With an annual growth rate of 30%, it is expected that the number of cellular IoT connection will reach 3.5 billion by 2023 [5].

LTE networks are heterogeneous cells with different Quality of Service (QoS) metrics, such as signal strength, coverage range, bandwidth, access time, etc. In order to maintain the QoS, the available bandwidth has to be measured as an essential metric to monitor the network performance. The bandwidth demand is increasing, and users' Quality of Experience (QoE) with mobile contents has become pivotal [7]. Monitoring the network performance has become a fundamental interest for service providers, vendors, and users. The users may have an interest to get information about the QoS status before sending
data over the network, and it is more likely to be the same for the service providers, who want to improve the routing plan and/or troubleshooting. There are certain parameters which give an idea of the QoS and level of performance the user can expect [8], for instance, link or path capacity, available bandwidth, loss ratio, utilization level, to name a few. The information of the current available bandwidth of a link or a path in the network is important in many cases and scenarios, for example, service level agreement (SLA) verification, congestion control, streaming applications, intelligent routing, admission control, network monitoring and management [9]. For some applications, getting information about the amount of available bandwidth of a network path can be beneficial for adapting the transmission rate and to share the bandwidth fairly [10]. For instance, self-congestion might occur in case of increasing the transmission rate of a video beyond the availability offered by the physical layer which leads to frame delays [11].

1.2 Problem definition

The present tools for estimating the available bandwidth are designed and developed for paths which have controlled links [9]. The characteristics of wired networks are different from those in wireless networks in many aspects, such as the rapid fluctuations and instability. Therefore, an assessment of a number of tools has been done to determine their efficiency in cellular networks. The problem is that most of these tools underestimate the available bandwidth [14]. Hence, most of the tools developed to estimate available bandwidth are insufficient [9], since the probing process and the post analysis to deliver the estimation require some time and do not feature real-time measurements which seems to be the main drawback of those tools e.g. Pathload [12] over mobile networks. The radio conditions usually suffer momentary variations. Thus, if the tool takes a large time to produce an estimation, it increases the probability that it does not reflect the current bandwidth status.

Another challenge to meet is to minimize the intrusiveness level and to prevent the measurements from affecting the QoE of subscribers since the measurements are performed in a shared, live commercial networks. Intrusiveness could be defined as an indicator of how much of the available bandwidth the tool uses [10]. If the amount of the external traffic the tool injects into the network is relatively high compared to the available
bandwidth during the estimation process, the tool or technique can be classified as an intrusive tool [13]. To calculate the intrusiveness $I$ as a percentage, the following formula is used:

$$I(\%) = 100 \times \left( \frac{C_p}{C} \right)$$

(1)

where $C_p$ represents the probing traffic, while $C$ is the link or path capacity. The measurements obtained by the currently available tools are clearly affected by wireless networking conditions, for instance, the adaptation rate, link speeds changes, network functionality complexity [16] and the contention with other traffic which results in inaccurate estimations, a varying convergence time, as well as a high level of intrusiveness. These obstacles make these tools impractical, as it requires accurate, and fast measurements. In addition, there is an issue related to high-speed links is the time precision. As the time intervals between packets get reduced, the measurements become more sensitive to errors.

1.3 Research motivation

The network performance usually measured with parameters such as bandwidth, packet loss, delay and jitter. In cellular networks, the available bandwidth has an impact on both, the QoS and QoE for most of the mobile services. Obtaining precise estimations of the available bandwidth of a network provides important information for network operators in order to make some adaptations in network strategy. It also plays a pivotal role in QoS deployment in a network and can highly improve a set of various network applications and technologies [18]. For instance, it is possible to configure routing table depending on the information of the available bandwidth for the paths in overlay networks. It is also important for peer-to-peer applications before starting the session. Network operators provides subscription-based services to their customers and usually the charge is in accordance to the available bandwidth provided. Service Level Agreements (SLAs) signed between service providers and customers in most instances use available bandwidth of the network as a service definition [18]. It also can be used to benchmark the competitors (the mobile networks operators) to evaluate their services. Additionally, available bandwidth is essential in avoiding congestion and planning intelligent routing systems [18]. The network
performance is of pivotal importance to both the customers and the service providers. This kind of measurements give an insight for network operators on how good their network is, and how it performs so they get an idea of what kinds of services they can deliver to their customers. Additionally, network service providers need to maintain their network and fix any issue may emerge. In special cases, there is a probability that the traffic is routed in the wrong way due to a fault in the network or when they introduce a new service or application to a network, it is needed to check its efficiency and performance before launching it to users.

Signal strength is not the only factor to consider, there are different factors (as an example, signal fading and admission control policies), that affect the available bandwidth [7]. Similarly to the signal strength indicator, this makes it needed to have bandwidth measurement tool on cellular phones, to show the estimated bandwidth instantaneously. There is a small amount of research have been done in the area of bandwidth measurement for the latest generations of cellular networks due to the limitations of earlier generations of mobile phones and/or complexity of bandwidth measurement. The upcoming generation of high-speed networks is about to emerge, motivating bandwidth measurements methods to expand the compatibility to extreme high-speeds [17].

1.4 Research aim and questions

The general aim of this research work is to develop an Available Bandwidth Measurements (ABM) approach for high-speed cellular networks with higher accuracy and less intrusiveness than the current methods.

The research questions are:

1. How to measure the available bandwidth in high-speed cellular networks?
2. What are the parameters that affect the accuracy of the measurements and the level of intrusiveness?

1.5 Research scope

This research focuses on measuring the available bandwidth of the downlink of LTE networks, since the downlink is the predominant in this type of networks as the data traffic
volume passing through this path can be more than ten times (it uses carrier aggregation technique) as that of the uplink due to the asymmetry of the system.

1.6 Research methodology

As the research is carried out with an Experimental-based approach, this master's thesis follows Quantitative research method. Generally, this method is used when a theoretical approach is infeasible or inadequate. Since there is no theoretical basis to build upon, the experimental approach was chosen [19].

InfoVista Sweden AB provided a problem definition description to use the minimal resources to measure the available bandwidth and reduce the intrusiveness. In order to attain the aim of research, the research questions have to be answered. This work started by gathering information about the models, techniques and technologies that could be used to measure the available bandwidth as well as reviewing the comparative studies of various tools. The literature review helped to clarify the path should be taken and to choose the protocol, technique for the testing tool and the effective parameters to achieve higher accuracy of measurements. Blixt™ tool has been modified for the experimental purposes to find out the optimal values of the parameters to be chosen for the tool which leads to more accurate and less intrusive measurements. The experiments have been carried out over an LTE commercial network (Telia) and for different values of the parameters. After the data being collected, the graphs have been plotted to find the correlation between the parameters’ values, measurements accuracy and the intrusiveness level.

1.7 Research contributions

This thesis discusses how to improve Blixt™ (as a method to measure the available bandwidth in high-speed cellular networks owned by InfoVista Sweden AB) in terms of handling the carrier aggregation.

It also investigates the correlation between the accuracy of the measurements and the main packet probing parameters. Moreover, it presents a comparison with other available tools and approaches in terms of accuracy and intrusiveness.
1.8 Thesis outline

This section briefly presents the contents of the next chapters of this report.

Chapter 2 covers a literature study of the available bandwidth concept and its applications, as well as the types of networks performance measurements. It also discusses the models and techniques used for this purpose and concludes with a review of previous comparative studies of available bandwidth measurements (ABM) tools.

Chapter 3 is dedicated to the studies related to the measurement of available bandwidth in cellular networks. It starts with a brief introduction to LTE networks and their characteristics, in addition to the challenges and uncertainty in this type of networks. Some of state-of-the-art tools are also briefly presented along with the challenges at the end of the chapter.

Chapter 4 introduces an overview of the techniques and the protocol on which the proposed solution is based, as well as the development environment and the API used. The algorithm and the measurements procedure of the solution are also provided.

Chapter 5 discusses the conducted experiments and an analysis of the results. In addition, it contains a comparison between the proposed approach and other tools in terms of both, accuracy and intrusiveness.

Chapter 6 concludes the thesis outcomes, provides a sustainability analysis and discusses the future work.
2 Background and Related Work

The main concepts related to this field of research are introduced in this chapter. This chapter defines what the available bandwidth is, describes the types of measurements, networks packet probing models and techniques. Additionally, it summarizes some of the comparative studies that evaluated a set of tools used for available bandwidth-related measurements.

2.1 Bandwidth-related metrics

It is important to differentiate the concepts which are pertinent to bandwidth metrics, because these differences lead to different roles depending on which metric is relevant to a specific application [13].

2.1.1 Capacity

The capacity is defined as the highest possible transmission rate at a link or from the source to destination (i.e. end-to-end capacity) [20], in other words, it is the number of bits which are successfully sent over the channel in a given time. The capacity can also be linked to the width of the communication channel and how fast the bits can be transferred.

Mathematically, if \( N \) represents the number of links (hops) in a path, then the end-to-end capacity is:

\[
C = \min_{i=1,...,N} C_i
\]  

(2)

Where \( C_i \) is the capacity of the \( i \)th link [21]. The capacity is referred to as the narrow link.

2.1.2 Available bandwidth

The available bandwidth of a link is the residual or unused capacity in a specific time period [22]. G. Aceto et al. define the available bandwidth as “the maximum rate that a new packet flow can impose on a path without affecting cross-traffic (i.e. other flows sharing path resources)” [34].
Unlike the capacity which remains constant for a relatively long time (unless there is an update to the link or a routing change), the available bandwidth is considered as a metric that varies with time [23] since the load caused by the traffic also affects it, in addition to the implicit transmission technology and the propagation channel. For this reason, the capacity does not need to be measured as fast as compared to available bandwidth.

To put the available bandwidth into a mathematical model of a link \( l \) in a period of time interval \([10, 24]\), \( U_t \) is the average usage of link \( l \) where \( U_t \) varies between 0 to 1 and \( C_t \) refers to the capacity of link \( l \). Then, \( A_t \) is the average available bandwidth of the same link \( l \) is specified by the unused portion of the capacity,

\[
A_t = (1 - U_t)C_t
\]  

(3)

By considering a path of \( N \) links, the bandwidth definition can be extended as follows:

\[
A = \min_{l=1,\ldots,N} A_t
\]  

(4)

\( A \) represents available bandwidth of the path which is the lowest available bandwidth of all \( N \) links. The link which has the minimal available bandwidth of a path is referred to as the *tight link*, which determines the available bandwidth of that link [25].

As it was mentioned earlier, the available bandwidth changes over time which makes it required to be measured quickly [22] especially in the case of the applications that use this metric to adapt their transfer rate. The figure below (2) illustrates how the capacity differs from the available bandwidth.

Fig. 2. Tight link and narrow link in multi-hop network
2.1.3 Throughput

Throughput can be defined as the amount of data that has actually been transmitted through a link or a network in a given time. T. Heikkinen defines it as “a measure of how many user data bits per time unit (usually seconds) can be forwarded by a network or system” [87]. It is also referred to as the achievable bandwidth.

The actual throughput that the user experiences is called application throughput. To illustrate this point further, downloading a file by a user from a remote server can be considered as an example. It is possible to measure the application throughput in this example by dividing the size of the file by the time taken to download the file. For instance, if the calculated throughput of the file transfer is equal to 150 kbps, then the application throughput can be around 140 kbps due to retransmissions and headers [10] [87].

2.1.4 Bulk Transfer Capacity

This metric is the measure of the achievable bandwidth by a TCP connection [22]. According to [22], the Bulk Transfer Capacity (BTC) is “the maximum throughput obtainable by a single TCP connection.”. All the algorithms of the TCP congestion control should be implemented in the link as specified in RFC 2581. BTC is defined by RFC 3148.

Available bandwidth and BTC are principally not the same metric since the BTC is TCP-specific, whilst the available bandwidth is a protocol-independent metric. Furthermore, the available bandwidth presumes that the traffic load keeps its condition and it only estimates the spare capacity a link or path can provide before the saturation happens while the BTC mainly determined by the way TCP allocates the bandwidth to different TCP traffic. To explain it further, suppose a single TCP connection saturates a path with capacity C. Hence, the available bandwidth of that path would be zero as a result of the saturation, but in the case that the BTC connection has the same RTT as the concurrent TCP connection, the BTC would be about half of the capacity [22]. Since the BTC measurement tools exploit all of the available bandwidth during the measurements, they can be classified as
intrusive.

Traceroute Reno, or TReno [27] [28], is one of the most well-known tools for measuring the BTC. It utilizes the Time To Live (TTL) for forcing the targeted-host to send back replies that it exploits to imitate TCP acknowledgements. Therefore, it is classified as a non-cooperative tool.

2.2 Applications of ABM

M. Jain and C. Dovrolis [29] explicitly discussed the applications where ABM could be beneficial.

Firstly, in SLAs verification, since the service is usually provided in terms of the available bandwidth, it is important for the customers to be able to check whether they actually get what they pay for. For network planners, having effective tools to measure the capacity as well as the available bandwidth is on top of their priorities. Another ABM application is the rate adaptation, especially in streaming applications. Video streaming applications could benefit from modifying the encoding schemes and adapt their transmission rate.

Along with the congestion control limitations, ABM can be used to control the adaptation process. ABM can also be used to perform admission control tests to avoid the intricacy of flow state in the network nodes. Another example of ABM applications is the selection of the best possible server based on ABM.

Lastly, ABM can be utilized to optimize the overlay networks in terms of both routing and QoS.

2.3 Developers’ view of network QoS

Certainly, the ultimate goal for developers is to deliver responsive, user-friendly applications to the users. U. Goel et al. [30] explained the correlation between network delays and applications performance. Network latency is a major concern despite the advances in the hardware industry and the efficiency of software processing. Network
latency does not always take advantage of this advancement.

Internet Service Providers (ISPs) normally plan their networks to minimize routing costs, which is not necessarily in line with the minimal path latency. To be more specific, ISPs may direct traffic onto less expensive but complex routes, which increases node counts and end-to-end latency. In cellular networks, ISPs may also delay transmissions until the conditions of carrier channel are suitable by modifying the scheduler's configuration. In addition, the bandwidth in mobile networks is constrained, which is the main motivation for ISPs to deploy mechanisms to shape the traffic of peer-to-peer video streaming traffic to increase the usable bandwidth for mobile users. Traffic shaping helps in inducing high latency that impedes the performance of real-time content applications like group communication, live video and collaboration tools.

2.4 Measurements methods

There is a variety of methods which are used for the network performance measurements: passive measurements by observing the traffic passes through the network, active measurements by inserting test packets, or a combination of both.

2.4.1 Passive measurements

Passive measurement is “performed by observing existing traffic without perturbing the network” [18]. In other words, it means tracking the behavior of the traffic by monitoring the packet flow without modifying it [31]. It needs access to the intermediary hops and it requires to process the entire load of the link to extract the network path information. It is possible to be implemented by combining additional intelligence into network equipment in order to differentiate and record the qualities and quantities of the traffic flow. These statistics can be collected without adding any traffic into the network. The details of the information collected are determined by the network metrics which are intended to be measured, the amount of data passes through the device and how the metrics are being collected and processed. Tcpdump and wireshark are among the most well-known passive measurement tools.
The assumption that it owns all the network nodes along the path is the main drawback of using the passive measurement method as well as the high amount of data generated which needs to be processed. Consequently, passive monitoring can only be used in a well-planned and controlled testbed [32]. Another issue is that the data collected need to be analyzed; online analysis is not feasible due to the large amount of data. In addition, there are also privacy matters to consider if the data capturing is done by a public operational network.

On the other hand, there are some advantages in passive measurements. These approaches do not need to generate extra traffic, hence they avoid network interference and provide a precise representation of the traffic [10]. In terms of accuracy, passive measurements are more accurate, for instance, it is possible to obtain an accurate measurement of the available bandwidth by observing the link usage on routers. By monitoring the buffers of the router along the path, the packet loss can be measured precisely.

2.4.2 Active measurements

Active measurement relies on injecting probe packets into the network to measure the targeted metric [31]. It implies the transmission of sequenced packets, known as probe packets, which are intended to get affected similarly to the typical packet flows by the selected path. The only aim of probe packets is to get some insight on how the network traffic is processed by the network. Moreover, with the active measurements there is a possibility of adjusting the probe packets with special properties such as packet size, bitrate, etc. [20]. The pattern of the probe packets used for estimating the available bandwidth is carefully designed, the inter-departure times and packet sizes should be precisely chosen [34]. The probe packets are time stamped at both end points, the sender and the receiver. The available bandwidth can be estimated by analyzing the cross-traffic and the effects of link capacity on the probes [33]. It is more favorable to use this type of measurement techniques to measure the available bandwidth, in spite of the fact that active methods inject external traffic into the network which might disturbs the QoS [18]. Since the active measurement does not require to fully access to network nodes, it can be considered as a major advantage of using it. It can also be used to simulate a considerable
number of users, which makes it feasible to estimate the number of simultaneous users a web server can serve.

A constrain of active measurement is that it may interfere with the network by inserting additional probe traffic into the network [31]. This is the reason why the active probing measurements need to be planned well before being carried out. The disturbance caused by the probe traffic may lead to some network QoS changes, which in turn provide erroneous values, this is why it is considered to be one of their main drawbacks [8]. In addition, the issue of intrusiveness has been addressed by much work, researchers put efforts trying to minimize the impact on the QoS of the network by reducing the number of packets sent into the network. This is the case in most of SLA verification measurements since the measurements are done frequently in a shared network, where both, the probe traffic and customer traffic share the same capacity. All active measurement tools are intrusive to some degree as they inject additional traffic in the network. Moreover, if an active measurement tool uses a small probe traffic, it would deliver estimations in a very short time [8].

2.4.2.1 Probe packet

The way of how the probing packet is structured plays a major role in measuring the available bandwidth. For this reason, different structures of probing packets were proposed.

V. Mohan et al. [10] define the probe packet as an additional packet which can be in any format depending on the information required to be extracted. A small UDP packet that contains a little or no payload at all and a timestamp could be considered as a basic example of a probe packet. Probe packets and their properties need to be selected precisely to simulate the actual network traffic. For instance, to measure network delay, it is not a good choice to use of ICMP packets since they are not going to be treated as normal traffic as they put to lower priority in most routers. In order to get a clear view of the network delay, UDP packets should be used instead.
2.4.3 Hybrid measurements

Hybrid measurement is a combination of active probing with passive monitoring [10]. The figure below (Fig. 3) shows an example of a hybrid measurement, considering a case of sending packet probes over a network and monitor their statuses by using passive ways during the process. This method makes it possible to track the probes’ path and measure the intermediate as well as the path delays which is not possible to do by using only active probing.

Fig. 3. Hybrid measurements example

As figure 3 shown, it is required to have an administrative access to the intermediate routers and this is why it is not favorable for Internet scale measurements. Hybrid measurements share all the same issues as passive and active measurements as they consist of both of them. It can be applied in testing how the network would be impacted by launching a new service, both from network operator’s and the end-user’s point of view.

2.5 Packet probing models

There are two main models, namely, the Probe Gap Model (PGM) and Probe Rate Model (PRM) which the bandwidth estimation tools are based on.

2.5.1 Probe gap model

Also known as cross traffic estimation based model which estimates the amount of cross
traffic at the bottleneck router by utilizing the variance in time-gap between successive probe packets [33]. It assumes that the bottleneck link is both the narrow link and the tight link (which in reality not necessarily is the case), and it also assumes that the capacity of the bottleneck is known in advance [35]. The available bandwidth $A$ is computed by subtracting the measured cross traffic $C_m$ from the bottleneck link capacity $C$ (which is known based on the assumption):

$$A = C - C_m$$

(5)

This model is implemented in IGI [36], Delphi [37] and Spruce [38].

The main feature of this model is that it that the estimation delivery time is short, and the probe packets size is relatively small. The drawback is that the probe packets are very susceptible to the variation of the cross traffic and the characteristics of the channel.

### 2.5.2 Probe gap model

The PRM model relies on the concept of self-induced congestion. It assumes that cross traffic follows a fluid model, FIFO (First In, First Out) queuing happens at all routers in the path [18]. PRM also assumes that there is a single tight link, but it does not assume that is the same tight link as the narrow link, and it does not assume that the capacity of the link is known.

To illustrate the concept of this model, if a sender sends probe packets to a receiver with a rate $R$, and less than the available bandwidth $A$, the packets would experience similar delays. On the other hand, if $R$ is higher than $A$, then the packets would experience increasing delays and queue in the network [39]. The aim is to find the point at which the delays start to increase. At this point, the available bandwidth is equal to the probing rate. This model is implemented in Pathload [12], pathChirp [40], and the Bandwidth Available in Real-Time (BART) [41].
The usefulness of this model is that it is easy to utilize, and the accuracy is good enough. However, the method uses large probing traffic and/or a long time for the measurement might affect the network. According to Bergfeldt et. al., PRM tools are commonly more accurate than PGM tools [9] but have higher overhead because of the iterative nature [42]. Tools evaluation studies have shown that in the presence of multiple congested links, the PRM tools seem to be more robust [17].

### 2.6 Packet probing techniques

The current bandwidth measurement techniques for measuring available bandwidth and capacity in single hops and along the path are described in this section. There are five major techniques: Variable Packet Size (VPS) probing, Packet Pair/Train Dispersion (PPTD), Self-Loading Periodic Streams (SLoPS), Trains of Packet Pairs (ToPP), and packet tailgating technique. VPS measures the capacity of links, while PPTD estimates the path capacity. SLoPS and ToPP are used for measuring the end-to-end available bandwidth.
2.6.1 The Variable Packet Size

This technique tries to deliver estimation of the capacity of individual hops (link) [10]. It achieves this by transmitting variable sized probe packets from one node to all other nodes in the path. Measuring the RTT from the sender to each node along the path as a function of the packet-size is the key element of this technique. This method was firstly tested by Bellovin in 1992 and then followed by V. Jacobson in pathchar [43] tool. It was later used in tools such as pchar and Clink.

This technique utilizes the TTL field in the IP header in order to forcibly expire the packets to at a specific node [22]. The probing packets would be discarded by the router of that node, and it returns error messages (ICMP time exceeded) to the sender and these ICMP packets are used to measure the RTT to that node. The RTT to each router has three delay components in both paths, the forward and the reverse: propagation delays, queuing delays and serialization delays. The propagation delay is the time it takes for bits of the packet to cross the link and it does not depend on the packet size. The queuing delays can happen in the buffers of network nodes when there is disputation at the input or output ports of these nodes. Finally, the serialization delay is time to transmit the packet on the link and it is equal to the ratio of the packet of size $S$ to the transmission rate $R$ of the link.

The downside of VPS is it might face underestimation errors in case of measuring the capacity over a path that contains store and forward layer 2 switches [44] due to the serialization delays introduces by these devices and without generating the ICMP replies.

2.6.2 The Packet Pair/Train Dispersion

This technique is based on sending multiple equally-sized packets from along the path and measure the dispersion of the packets at the destination [10]. The dispersion of the packet increases due to the narrow link of the path. Hence, the available bandwidth can be measured. The packet pair technique uses a pair of packets, while the packet train technique uses multiple packets, which is the main difference between these two techniques [10]. The dispersion of a packet pair (or train) represents the time measured from the last bit of the first packet to the last bit of the last packet., The tools (SProbe, pathrate, nettimer and bprobe) implement this technique.
2.6.3 The Self-Loading Periodic Streams

SLoPS technique is used to estimate the available bandwidth. The key element of this technique is to send a sequence of equally-sized packets at an increasing rate and to observe how the one-way delay changes [10]. The indication of the congestion in the path’s tight link is when the delay increases. SLoPS uses a repeated search method in order to find the optimal rate in which the delay does not increase and the congestion does not occur.

2.6.4 The Train of Packet Pairs

ToPP is somewhat similar to the SLoPS method but it uses different packet structure and works on minimizing the latency of the measurement. It uses an algorithm to measure the available bandwidth based on the injected probe packets [45]. It sends groups of packet pairs with an increasing rate to the node. As compared to a packet pair, a packet train is generally less sensitive to random variations. Unlike SLoPS, ToPP’s sending rate is linear, while SLoPS uses a binary search method to adjust the rate [10]. As a matter of fact, the differences between the two techniques are mainly related to the statistical processing of the measurements. ToPP is also capable of estimating the capacity of the tight link of the path which is not the case with SLoPS.

To explain ToPP further, consider a link with available bandwidth $A$, capacity $C$, and the average cross traffic rate $R_c$ is equal to $C - A$. The increasing rate at which ToPP sends the packet pairs is $R_s$ [13]. In the case $R_s$ becomes larger than $A$, the rate at the receiver $R_r$ will be:

$$R_r = \frac{R_s}{R_s + R_c} C$$  \hspace{1cm} (6)

Or

$$\frac{R_s}{R_m} = \frac{R_s + R_c}{C}$$  \hspace{1cm} (7)

The available bandwidth $A$ is estimated when $R_s \approx R_m$. From the slope $\frac{R_s}{R_m}$ in Equation 7, it
is possible to estimate the capacity $C$. In multi-links paths, the $\frac{R_L}{R_m}$ curve may present some changes because the queuing at the links have higher available bandwidth than $A$.

### 2.6.5 The packet tailgating

Lai and Baker [10] present a hybrid technique which represents a combination of VPS and packet pair techniques. This technique aims to estimate the path capacity two phases. In the first phase (sigma phase), measurement of the properties of the whole path is done. The second phase (tailgating phase), measures the properties of the individual links (hops). The tailgating concept is based on sending a large packet (tailgated) and followed by a small packet (tailgater) for each link individually. The smaller packet will constantly queue behind the larger packet until the larger packet’s TTL expires. In this case the tailgater will carry on to the destination without queuing. Another assumption is that the larger packet will not face queuing, whereas the smaller packet is always queued. STAB [47] tool uses packet tailgating technique along with packet chirps to measure the available bandwidth.

### 2.7 Filters

The accuracy of available bandwidth-related measurements remains a challenge. Many researchers used filter to improve the estimation accuracy, for instance, Exponential Weight Moving Average Filter (EWMA), Vertical Horizontal Filter (VHF), Kalman Filter and Fuzzy Filter. These filters were applied to different measurements tools.

#### 2.7.1 The moving average filter

This filter was used to smooth the estimations of pathChirp. It delivers the estimation by averaging the last window size $W$ observations [48]. The challenge is finding the right size of $W$ for the dataset. If $W$ is too wide, the filter shows slow reactions due to the real changes being leveled out. On the other hand, if $W$ is too small, the peaks of noise are not eliminated, and the result tends to follow the estimated values.
2.7.2 The exponentially weighted moving average filter

J. Navratil and R. Cottrell attempted to apply EWMA filter in Abing [49] tool (formerly known as ABwE) and it resulted in two versions of Abing with tunable parameters. Sommers et. al. also applied the EWMA filter on another tool, Yaz [50].

This filter uses the observed values $O_v$ and the new estimation $E_n$ is calculated as follows in equation 8:

$$E_n = \alpha E_{n-1} + (1 - \alpha) O_v$$  \hspace{1cm} (8)

The authors of Abing firstly set the value of $\alpha$ to 0.9, which provides heavy smoothing, while the coefficient $\alpha$ is equal to 0.75 in the updated version of the tool [48]. On the other hand, Yaz needs to use less $\alpha$, around 0.3. The challenge with EWMA filter is in the choice of the smoothing factor. The filter gives more stable but slow estimation when a large $\alpha$ is used; by keeping $\alpha$ small, the agility is achieved. It cannot be said that agility or stability is desirable at all times. It is preferable to adapt and set the value of $\alpha$ according to the changes in the network, then equation 8 can be reformulate:

$$E_n = \alpha_n E_{n-1} + (1 - \alpha_n) O_v$$  \hspace{1cm} (9)

2.7.3 The vertical horizontal filter

This filter introduced by Adam White in August 1991 [51]. It is used mainly in economics; hence this filter was borrowed from the financial world. It tells whether the prices are in a congestion phase or in a trend phase. Emanuele Goldoni et. al. [48] were the first to apply both VHF and EWMA filter to smooth the measurement of the pathChirp, which resulted in producing ASSOLO [52].

They also compared the VHF with the EMWA filter. Emanuele Goldoni used VHF to calculate the smoothing factor $\alpha$ on the arrival of packets which makes $\alpha$ is an adjustable parameter. VHF is a modified EWMA filter based on the same principles of the previously mentioned filters.
2.7.4 Kalman filter

In 1960, RE Kalman published an article which aimed to solve the problem of discrete-data linear filtering [53]. This filter is very powerful in predicting the future status [54]. Kalman filter estimates the process in two phases (based on a form of feedback control): In the first phase, it estimates the state of process and then gets the feedback. There are two groups of Kalman filter’s equations, measurement update equations and time update equations. The measurement update equations are used for the feedback while the latter ones are used for projecting the current state and error variance measurements. The time update equations and the measurement update equations can be considered as a predictor and corrector equations [54].

Svante Ekelin et. al. applied Kalman filter to a bandwidth measurements tool called BART [41], and it shown promising results, in terms of both the accuracy and the time it takes to estimate. The authors also presented a way of using temporal properties of the Kalman filter to estimate the available bandwidth. They also went further and published a research on applying the Kalman filter in another tool which is Cusum [55].

2.7.5 Fuzzy filter

The Fuzzy filter was used in several fields of image processing [56]. The main idea is to get the of a measured value and use other values around it. At the same time, it needs to reduce the effects of the noises which could shift the average result and lead to incorrect values. D. T. Tran and A. N. Nguyen [54] presented two techniques to be used which are the Probor filter and Fuzzy filter to enhance the accuracy of the available bandwidth measurement tools which follow the PRM.

The Fuzzy filter filters the measured values. The density of the data and the length of data segments determine the efficiency of the filter. Hence, the use of this filter is recommended when the data distribution information is available. The prober filter is better used when there is a need to improve the accuracy of the available bandwidth measurements in real time applications, and it is essential to understand the model of the network traffic [54].
2.8 Comparative studies

There has been some work that discuss the evaluation of tools and comparisons based on specific terms.

Ait Ali et. al. compared and analyzed set of tools in terms of accuracy, response time and [18]. They also discussed the measurement delays and uncertainties, pointed out the weaknesses of some of probe pattern which have an impact on the accuracy. Spruce, Pathload, IGI, pathChirp were compared. Experiments show that Pathload has a higher level of intrusiveness and it can be very slow in some cases. IGI estimations are inaccurate and pathChirp overestimates the available bandwidth. Spruce offered better performance in the terms of to the criteria selected as it delivers the most accurate and the fastest estimations with low intrusiveness.

Another study was presented by Goldoni and Schivi [57]. They compared some of the well-known tools in a real environment, links of 100 Mbps where used in the experiments. They concluded that Pathload and Yaz provide the most accurate estimation despite the convergence time and the significantly high intrusiveness [58]. On the other hand, ASSOLO showed better performance due to its high accuracy, short time of estimation process, and the low intrusiveness.

Aceto et. al. compared the performance of compared various ABM tools (pathChirp, Pathload, Yaz and ASSOLO) in terms of accuracy in both wired and wireless networks. In the case of wired network, they found out that the cross-traffic has an impact on the accuracy of some of the tools (Pathload and Yaz are not included) [58]. The performance in the wireless scenario was worse. The results were not stable, which made it difficult to tell which tool has an acceptable performance.

Labit et. al. [8] evaluated set of tools (pipechar, IGI, Abing, pathchirp and Spruce) used to estimate the path’s available bandwidth in a real environment. All of these tools failed to show good results as most of them hugely overestimate the available bandwidth. Overestimation is worse than the underestimation, for instance, using pathChirp for congestion control application will result in huge congestion problem. As a conclusion,
they were not convinced by any of these tools.

Bergfeldt et. al. [9] studied the ABM over a High-Speed Downlink Packet Access (HSDPA) in UMTS. They analyzed the performance of ABM tools in a wide area mobile network. They used three tools (Pathload, pathChirp, and BART) for their experiments. The experiments were carried out in a commercial mobile network. They observed that it is feasible to get reliable estimations under certain conditions. pathChirp has been shown a good performance record, and it showed an interesting real-time characteristics. Pathload has also shown to deliver accurate measurements in different scenarios. Both, BART and pathChirp suffer from overestimation in multi-hop path scenario. They concluded that in the case of the available bandwidth is affected by cross traffic, pathChirp overestimates the available bandwidth, while BART seemed to handle this condition better.

X. Zhang [59] studied the results of measurements by using different measurements tool and found out that each tool had a favorable application. In those ranges, the measurement tools can obtain accurate measurements. Hence, these application ranges can be used to select the proper tool and thus get more accurate results.

Table one summarizes a set of ABM tools [13].
<table>
<thead>
<tr>
<th>Tool</th>
<th>Author</th>
<th>Metric</th>
<th>Technique</th>
<th>Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bing</td>
<td>Beyssac</td>
<td>End-to-end capacity</td>
<td>PPTD</td>
<td>ICMP</td>
</tr>
<tr>
<td>Bprobe</td>
<td>Carter</td>
<td>End-to-end capacity</td>
<td>Packet pairs</td>
<td>ICMP</td>
</tr>
<tr>
<td>Clink</td>
<td>Downey</td>
<td>Link capacity</td>
<td>VPS</td>
<td>ICMP</td>
</tr>
<tr>
<td>Cprobe</td>
<td>Carter</td>
<td>End-to-end available bandwidth</td>
<td>Packet pairs</td>
<td>ICMP</td>
</tr>
<tr>
<td>IGI</td>
<td>Hu</td>
<td>End-to-end available bandwidth</td>
<td>SLoPS</td>
<td>UDP</td>
</tr>
<tr>
<td>iPerf</td>
<td>NLANR</td>
<td>Achievable TCP throughput</td>
<td>Parallel TCP</td>
<td>TCP, UDP</td>
</tr>
<tr>
<td>pathchar</td>
<td>Jacobson</td>
<td>Link capacity</td>
<td>VPS</td>
<td>UDP</td>
</tr>
<tr>
<td>pathChirp</td>
<td>Ribeiro</td>
<td>End-to-end available bandwidth</td>
<td>Self-loading packet chirps</td>
<td>UDP</td>
</tr>
<tr>
<td>Pathrate</td>
<td>Dovrolis-Prasad</td>
<td>End-to-end capacity</td>
<td>Packet pairs and trains</td>
<td>UDP</td>
</tr>
<tr>
<td>Pathload</td>
<td>Jain - Dovrolis</td>
<td>End-to-end available bandwidth</td>
<td>SLoPS</td>
<td>UDP</td>
</tr>
<tr>
<td>Pchar</td>
<td>Mah</td>
<td>Link capacity</td>
<td>VPS</td>
<td>ICMP</td>
</tr>
<tr>
<td>Pipechar</td>
<td>Guojun Jin</td>
<td>End-to-end available bandwidth</td>
<td>Packet pair</td>
<td>ICMP</td>
</tr>
<tr>
<td>Netperf</td>
<td>NLANR</td>
<td>Achievable TCP throughput</td>
<td>Parallel TCP</td>
<td>TCP</td>
</tr>
<tr>
<td>nettimer</td>
<td>Lai</td>
<td>End-to-end capacity</td>
<td>Packet pairs</td>
<td>TCP</td>
</tr>
<tr>
<td>SProbe</td>
<td>Saroiu</td>
<td>End-to-end capacity</td>
<td>Packet pairs</td>
<td>TCP</td>
</tr>
<tr>
<td>Treno</td>
<td>Mathis</td>
<td>BTC</td>
<td>Emulated TCP</td>
<td>UDP, ICMP</td>
</tr>
<tr>
<td>Ttcp</td>
<td>Muuss</td>
<td>Achievable TCP throughput</td>
<td>TCP connection</td>
<td>TCP</td>
</tr>
</tbody>
</table>
3 ABM in Cellular Networks

This chapter is dedicated to the cellular network, mainly the Long Term Evolution (LTE). It presents state-of-the-art tools which are used for estimating the available bandwidth in cellular networks along with various approaches. It also highlights the challenges that the tools face when measuring the bandwidth in this kind of environment which is subject to quick fluctuations.

3.1 Cellular networks

The mobile technology has evolved through times and further labelled into generations. Cellular networks carry a constantly increasing amount of data with their pervasive deployments and data rates, which have increased significantly in the last decade. However, networks such as HSPA and LTE contain link layer protocols that are quite different from the ones used in wired and Wi-Fi networks. Thus, it is crucial to understand the behavior and the properties of cellular networks.

Considering the fast development of wireless technologies with different properties in addition to the user’s preferences for seamless mobility, the future networking environments seem to be heterogenous [60]. A variety of access technologies will be offered by a number of independent service providers forming a multiple Radio Access Technology (RAT) environment with overlapping wireless coverage. Also, mobile devices are typically equipped with two or more access technologies. However, it is not an easy task to ensure a proper QoS and to handle the mobility in a heterogeneous network. Typical problems that are encountered include latencies when executing vertical handovers, timely dealing with fluctuations in network performance, and properly assessing the experienced quality of service.

The services delivered by the applications of cellular networks, are usually interactive and deployed in clouds [61]. These applications support a broad range of tasks, such as social gaming, live video applications, augmented reality, and communication services. Applications of real-time video analysis, live speech translation or other computationally
intensive tasks are expected to take the advantage of the processing power and the cloud-based datasets in such innovative applications. The application response will greatly meet with the network performance, as the interactions between mobile devices and servers increases.

Mobile networks, 4G, LTE and the upcoming 5G in particular, are proceeding to become the dominant communications services which provide a high quality of services [26]. This growth comes in response to the rising popularity and spread of mobile devices among individuals, business and consumers of government services. QoS assurance to the customers is the key element in the successful establishment of such wireless communication technologies in international level in addition to the appropriate design and resource provisioning.

3.2 LTE

In November 2010, the International Telecommunications Union (ITU) has ratified the International Mobile Telecommunications-Advanced (IMT-A) standards, and the 4G communication systems are currently deployed worldwide [4].

LTE is an initiative of Third Generation Partnership Project (3GPP) and the true evolution comes with the progress work of 3GPP release termed as LTE-A under 3GPP [64]. 3GPP is the standard developing body that looks into the LTE/LTE-A specifications. In general, LTE networks consist of three subsystems: The User Equipment (UE), the Radio Access Network (RAN), and the Core Network (CN). UEs are cellular phones with communication capabilities carried by the users. The RAN is a set of antennas that ensure the connectivity between a UE and the CN of the operator. It consists of multiple base stations called Evolved Node B (eNodeB). eNodeB is responsible for handling the radio communications with multiple devices in one cell and carries out handover decisions and radio resource management. The backbone of the cellular network represents the centralized CN which is responsible for the connection to the Internet [65].

The concept merely lies here to support more devices on IP based services. The whole LTE network is based around packet switched services and was primarily developed to provide
higher data rate services, lower delay for interactive services and higher spectral efficiency to the user. LTE-A systems use numerous antennas called Multiple-Input Multiple-Output (MIMO) technology to transmit data in the physical layer on the transmitter and the receiver. This feature enables the fast wireless services, which improves the efficiency of the network. The advanced antenna techniques, adaptive modulation and coding with the usage of Orthogonal Frequency Division Multiple Access (OFDMA) helps LTE to achieve a significant throughput and spectral efficiency enhancement. The distribution of functions for radio access network between 3G Radio Network Controller (RNC) and NodeB is now simplified in LTE with base station or eNodeB. LTE deploys OFDM and SC-FDMA for downlink and uplink transmission respectively. OFDM allows support for both time and Frequency duplexing modes (TDD & FDD). It relies on the rapid adaptation to channel conditions and employs rate adaptation and hybrid soft combining techniques.

Carrier aggregation technique has also been added in this release which gives the opportunity to the users to download data from different sources simultaneously, that empowers them with higher download speeds. In LTE networks, it is possible to migrate from a network that is built to enable some data to an IP network which is built to support different data services including voice and video over one common architecture. In case the mobile requires to migrate from one cell to another, related information needed to be gathered and collected to prepare for the handover process. Neighbor networks and resources used to discover the status of the network such as the frequency, bandwidth and the channel number, are all part of it. The gathered information by the user equipment (UE) and/or the network, provides different information about the metrics for the user, such as available network coverage, security, bit error rate, etc. The handover process makes it possible to switch from one communication cell to another without interrupting the call. Throughput is important for link (cell) selection decision when handover happens. It has to find which link provides higher throughput to select.

LTE-A allows more than 20 MHz bandwidths for carrier aggregation and this enables to provide supplementary service like MIMO as toolbox for innovative features like relaying which aims for better capacity and coverage [63]. The 4G network can provide up to 100 Mbps (LTE-A standard) and up to 128 Mbps for mobile users [4].
As an important part of LTE's Radio Resource Management (RRM), LTE packet scheduling plays a vital role in enhancing the data rate and in supporting the QoS metrics requirements of the mobile system [62]. LTE packet scheduler is responsible for intelligently allocating radio resources to the UEs, and the QoS settings in such that the LTE network abide to its performance specifications [66]. The eNodeB MAC sub layer is responsible for scheduling transmission over both uplink and downlink. The period of time that the scheduler takes is called the Transmission Time Interval (TTI). It differs based on the mobile system, and it has been reduced in the recent technologies. In LTE, it is only 1 ms. Hence, the data received by the UEs is actually a burst during the TTIs in which the resources have been allocated, whereas nothing to receive during the TTIs with no resources been allocated. Fairness scheme (or proportionally fair) is usually followed in the scheduling process which takes into consideration the channel quality, and the data needs. It considers the previous resources allocation and the current potential bandwidth of all the competing users. By doing so, it can balance the resources distribution among the users, despite of their channel quality, and it helps in optimizing the aggregated throughput of the BS.

The transmitted packet goes through several phases until it reaches the user's device. Firstly, the packet is transmitted from the internet to the core network of the mobile operator, which in turn forwards it to the BS that the user is connected to. At this stage, the packet been stored in a buffer located in the BS. It stays there until the scheduler allocate resources to that user. Depending on various factors, such as the signal strength, it would either be sent alone or grouped with other packets that are stored in that buffer.

### 3.3 QoS in LTE

According to 3GPP standardization releases, QoS provides the mobile operators with a group of tools to start the service and differentiate the level of service provided to subscribers which allows to offer different services to the user [64]. These are services could be, prepaid connections and roaming services.

LTE exploits different methods for radio resource management in order to maximize the cell throughput while maintaining fairness for users, QoS and other services. To collect the
radio interface data from the networks, drive testing is the method used for this purpose in different locations under the coverage of the cellular network. Hence, it allows the mobile operators to make an information networks maps.

Different metrics can be measured and analyzed, such as the user throughput, signal strength, delays, call drops. These values are mainly determined by the scheduling algorithm and resource allocation. One of the major information is the Reference Signal Received Power (RSRP) which essential for handover procedure and cell selection. It is average power (in W) of the resource components that carry cell-specific related signals in that measurement frequency band [67]. Channel Quality Indicator (CQI) is Another radio measurement of user throughput. Wideband CQI is the averaged CQI for the system bandwidth as a whole. LTE defines 16 CQI levels for different coding Schemes and modulation. WCQI can assist us an indication for data rate that the system supports for a certain link conditions. The bitrate per symbol rate obtainable in an LTE network depends on the SINR for both the links.

3.4 LTE features

LTE has a number of features that render traditional ABM methods inadequate. The most salient of these features are as follows:

- In LTE, the radio channel is a shared resource between all users in a cell. An FTP file transfer to one user in a cell (for example, the testing device) will significantly affect other users in that cell, as will any other traditional drive test activity.
- It is possible for multiple operators (carriers) to share the same radio access network. This puts requirements on parallel testing, as subscribers of different network operators might, for example, share the radio network but use separate core networks.
- High data rates: 100 Mbps, is the theoretically possible transfer rate to be attained (theoretically). It can be a challenge to fill up the channel with data for the purpose of measuring the actual bandwidth of the channel; system links, and from the server to the client, has to be carefully tuned to manage such high transfer rates. Due to
the UE’s limited CPU processing performance, testing applications will face problems in handling the data and filling the bit-pipe which is restricted by the battery performance.

- Rich configuration possibilities: An LTE network can employ a large array of different MIMO configurations, and the scheduler used in this technology has very powerful and flexible mechanisms for maximum utilization of the radio path (both uplink and downlink). Traditional ABM techniques do not adapt to such rapid variations in the link capacity.

3.5 Bandwidth estimation approaches for mobile networks

There are three main categories that the ABM tools in mobile connection fall within: packet probing based tools, packet’s RTT, and the download-based tools.

3.5.1 Packet probing method

This approach is based on sending probing packets from one node to another and measure the delay of packet arrival at the receiver [7]. The available bandwidth $A$ can be computed as:

$$A = \frac{S}{t_2 - t_1} \quad (10)$$

Where $S$ is the packet size, $t_2$ is the delay of arrival between the two packets at the destination, $t_1$ is the delay at the sender. Some approaches assume that $t_1$ is equal to zero, as the packets are sent back-to-back. This approach is considered as one of the most well-known ones.

3.5.2 Packet’s RTT

It is another common approach for measuring the available bandwidth by using the packet’s round-trip-time [7]. RTT is a two-way delay, that the packet experiences from sender to receiver and back. It is crucial to manage the RTT in order to deliver a better service of the interactive applications over the network.
By using ICMP packets, this approach is based on measuring the time the ping request packet takes to traverse a network path from the source to the destination and the ping reply to get back to the sender. It also includes the delay at the receiver to send the reply back. There is a general assumption that both the uplink and downlink are symmetrical, since the RTT includes the time for both. And so, the time a packet requires to traverse from the source to the destination is about half of the RTT. Pathchar is one of the tools that uses this approach. The available bandwidth $A$ can be computed as:

$$A = \frac{S}{RTT + D_p}$$

Where $S$ is the packet’s size as previously stated, and $D_p$ is the propagation delay which differs based on the wireless medium. This technique also uses ICMP packets of increasing sizes to estimate the capacity of the path. The reason why to use multiple packets is based on the assumption of at least one or a few packets will not face any queuing delay or encounter the minimum delay. This method only requires installing the software on the sender side which is considered as an advantage.

### 3.5.3 Download based technique

It is one of the commonly used approach to estimate the bandwidth is by uploading or downloading a large chunk of data. There are many ABM tools (such as Netperf2 and Ookla speedtest) use this method to measure the achievable bandwidth. It computes the achievable bandwidth by dividing the file size by the time it takes to upload or download it. Finding the suitable file size remains as the main challenge of this technique. If the size is too large, then it is considered as a waste of resources since the measurement takes place at the end of the test. However, if the size is too small, then it only measures the bandwidth under a given load which does not mean the maximum achievable bandwidth of that path which also results in wasting more network quota than the need. This technique is undesirable to be used in mobile networks as it endures high overhead in terms of Internet quota.
3.6 ABM state-of-the-art tools for high-speed cellular networks

Despite the fact that Wi-Fi networks and mobile networks are both wireless, using prior tools which were originally designed for Wi-Fi networks is chancy to deliver accurate estimations of the available over cellular networks since they behave differently from Wi-Fi networks [68]. In fact, the authors of [69] concluded that WBest [15] is impractical for the 3G networks due to inaccurate estimations after evaluating WBest over the downlink of a (nowadays, old-fashioned) 3G 1xEV-DO network. Mike P. Wittie et. al. [70] presented a platform to measure and characterize mobile networks performance as experienced by the user. It was named as MIST (Mobile Internet Services Test). The following subsections introduce some of the recent approaches of measuring the available bandwidth in cellular networks.

3.6.1 Gping-Pair

M. Zhong et. al. [7] propose Gping-Pair (Gateway Ping Pair), which based on packet’s RTT approach. It uses a software component only on the client end to estimate the bandwidth of cellular networks. The authors claim that Gping-Pair is capable of measuring the achievable bandwidth between the client and the ISP gateway, which is the 1st gateway of the cellular network provider. This link is subject to the admission control and different interference; therefore, it is distinctly possible to become the bottleneck of the whole connection to the Internet. Ping program was used to find the ISP gateway by checking the connectivity to a server. They tried to check the IP address of each node along the path to the server by increasing the TTL, and the first gateway is the first hop that replies with an ICMP message. They also used considerable variance in the sizes of the packet to obtain accurate delay details. It has to be noted that it should not be too large to avoid fragmentation. Gping-Pair probes the network with n ping pairs of 64 and 1064 bytes sizes (this approach uses two RTT values at 64 bytes and 1064 bytes.), and the default value of n was set to 10 considering the overhead probability. The minimum RTTs can be calculated from the ICMP echo replies; hence the achievable bandwidth AB is computed as:

\[ AB = \frac{2 \left( S_1 - S_2 \right)}{RTT_1 - RTT_e} \]  

(12)
They assumed symmetric path and the one-way delay is the half of the RTT; $S_l$ and $S_s$ are the large and small packet sizes respectively for the ping pair. The minimum $RTT_l$ and $RTT_s$ are the RTT measurements, the large and the small ones respectively. Download-based measurement was used for benchmarking. The figure below (Fig. 4) shows the topology used for the experiments.

![Network topology](image)

**Fig. 5.** Network topology for Gping-pair experiments

### 3.6.2 Correlation coefficient method

Y. Takano et al. [71] proposed a method that measures the correlation coefficient between the different transmission rates at which it sends the measurement packets and the packet loss rate at the destination to correctly estimate the available bandwidth.

Considering the case when the transmission rate is less the available bandwidth, a policer generates random packet loss. Meanwhile, the receiving rate becomes provisionally stable. In contrast, if the transmission rate becomes more than the available bandwidth, the policer generates more packet loss because of the available bandwidth management. Hence, the packet loss is correlated to the transmission rate when the latter exceeds the available bandwidth. Thus, they proposed a method to improve the measurement of the available bandwidth in a network based on the correlation information. To emulate a mobile
network, they used a Linux based network emulation tool [3] called Netem. Netem mimic the behavior of mobile network by storing packets in a queue and generate delays and packet losses by using Linux packet scheduler. The proposed method was implemented as an improved version of Pathload and they named it Pathload-P.

3.6.3 NEXT

NEXT (New Enhanced Available Bandwidth Measurement Technique) was proposed by A. K. Paul et al. [72]. To estimate the available bandwidth, they used pathChirp’s excursion detection algorithm. Their proposed probing method features a probing train structure that has a region where packet rates are sampled more often than in other regions. Another algorithm was proposed which is for adjusting the rate in every cycle in a way that the expected available bandwidth would fit into the packet dense regions. The rate adjustment algorithm uses two spread factors, to minimize the number of packets sent over the network. One of the spread factors is the “ratio of successive packet inter-spacing times within a chirp”.

The sender node of NEXT constantly sends probing streams of length $P$ packets, each packet has a different sending rate by controlling the inter-packet gaps. The NEXT receiver records the arrival time of each packet and it calculates the queuing delay based on the information obtained from the time stamp of the sender. In the case of the packet’s rate is lower than the available bandwidth, no queuing delay occurs. On the other hand, if the packet’s rate is higher than the available bandwidth, then queuing delay happens. The receiver detects the turning point in which the queuing delay starts to increase constantly and use it to estimate the available bandwidth.

NEXT, similarly to pathChirp, it also estimates the end-to-end available bandwidth by using a number of packet chirps $p$ (numbered as $P = 1, 2, \ldots, n$) from source to destination. Each train of packets has a given number of packets depending on the rates and the spreading factor. The difference between pathChirp and NEXT is the shape of the traffic and the spacing form between the packets, in spite of that both methods use a train of packets of increasing delays.
To implement and evaluate the performance of their approach, they used a simulated environment. They also compared it with other ABM tools, such as pathChirp, Spruce, and IGI. They implemented each of these methods in the ns-2 network simulator. Measurements showed that NEXT delivers better estimations than pathChirp in terms of accuracy and the intrusiveness level.

### 3.6.4 NEXT-V2

The authors of the previously introduced NEXT, presented an extended version of it to improve its estimations, which is NEXT-V2 [39]. In this updated form of NEXT, they modified the algorithms of excursion detection the packet loss recovery. NEXT-V2 estimates the available bandwidth based on the analysis of the one-way queuing delay detecting the turning point.

### 3.6.5 NEXT-FIT

NEXT-V2 was not the latest version of NEXT, A. K. Paul et al. extended their work and introduced NEXT-FIT [73]. NEXT-FIT is a method for measuring the available bandwidth over LTE networks. NEXT-FIT used long probe-packet trains. They introduced a “*parameter-independent curve-fitting*” technique instead of pathChirp’s excursion detection algorithm to estimate the available bandwidth from a one-way queuing delay (similar to NEXT-V2) and it was evaluated in a commercial 4G/LTE mobile communication network of a Japanese mobile operator.

### 3.6.6 A proxy-based approach

In the end of November 2016, A. K. Paul et al. [74] proposed another approach to measure the available bandwidth. The proposed method is a proxy-based. They designed a probing measurement scheme for Android smart-phones which follows the concept of merging the packet probing method with application data transfer. By utilizing a proxy technique, the proposed scheme transfers data by piggybacking practice in the probing packets and thus no need to change both the “*existing client applications and servers as opposed to kernel based implementation*” in [75].
Another contribution of this work was accomplishing the proposed scheme without the need of rooting the Android device to get the super-user privileges. They face a major issue which is how to eliminate the unfavorable impact on the performance of the application without affecting the performance of the measurement tool, and it is also required to suppress the overheads of active measurement.

3.6.7 PathQuick

PathQuick is based on PRM model. It aimed to study and reflect on complicated behavior of queuing delays over operational networks. The general concept of PathQuick is summarized as a sender sends a UDP packet train probing packets to a receiver. At the sender, each of these packets is timestamped and carries it which the receiver uses along with its own timestamps to deliver an estimation of the available bandwidth and reports it to the sender by sending a UDP packet. PathQuick3 uses the concept of the “curve fitting along with the nonlinear least squares method”.

The authors of PathQuick3 [68] mentioned that it represents the successor version of their previous methods PathQuick [77] and PathQuick2 as both, PathQuick and PathQuick2 were designed for wired networks in the first place. They evaluated PathQuick3 against pathChirp over an LTE network in the downlink direction and they found out that pathQuick3 estimations are more accurate.

3.6.8 TPG

Train of Packet Groups (TPG) [76] method was developed based on PathQuick. The idea behind it is to send a sequence of packets called as a train of packets from the sender node which is located in the base station side to the mobile device in order to measure the downlink available bandwidth. The train of packets has a several groups of packets. The overall packet size of the packet train is set to increase linearly. They set the value of inter packet-group gap to match the TTI value of the link. The delays increase when congestion, hence, by determining the packet-group from which the delays start to increase (the turning point), it is possible to estimate the available bandwidth. They evaluated the method in real testbed which is a commercial FDD-LTE system.
The measurement procedure is as follows:
Firstly, packets are being encrypted, and the IP headers are being compressed by Packet Data Convergence Protocol (PDCP), then the packets are segmented or concatenated in the Radio Link Control (RLC) to a specific size known as Transport Block Size (TBS). The calculation of TBS is done by the scheduler in MAC layer by taking into account the present signal quality. TBS becomes small when the radio signal is poor, in contrast to the case when the signal is strong which results in big TBS. TTI is the interval in which the scheduler calculates an optimum value of TBS and measures the signal quality. The scheduler calculates the TBS every 1ms since TTI in LTE is set at 1ms. The MAC Protocol Data Unit (MAC-PDU) which is segmented, or concatenated data called is sent from the eNodeB to the UE through the physical layer in every 1ms (TTI).

### 3.6.9 PathML

In 2017, N. Sato et al. [78] proposed a method that uses machine learning techniques based on a “data-driven paradigm with a large amount of data” to estimate the available bandwidth. Since the task is to predict the available bandwidth (output) from queuing delays (input), they used supervised learning. Four machine learning algorithms were selected, which are: kernel ridge regression, random forests, convolutional neural network, and support vector regression. They implemented the system with these techniques by using Scikit-learn [79] and TensorFlow [80] which are machine learning libraries. The experiments were conducted over an operational LTE network (Japan's primary mobile operator) and they found out that the convolutional neural network was the most accurate algorithm. They also evaluated their method against PathQuick3.

The learning procedure is as follows:
They started with preparing the training dataset which is a large amount of observed queuing delays pairs at a receiver and corresponding available bandwidth values. After that, the machine learning and by using the dataset, generated a predictor to predict the values of the available bandwidth based on the observed queuing delays. The predictor uses patterns based on the relationship between available bandwidth and the queuing delays and, for instance, when there is a small amount of packet, the transition point refers
to low available bandwidth and in contrary to the large number of packets. It is possible to get the available bandwidth estimated value by adding a new set of queuing delays into the predictor.

### 3.7 The challenges and uncertainties

Mobile network performance is a challenging interest because of its broadness of various factors that affect and interact with the QoE. For instance, the service quality that a user receives from the provider can depend on the network access technology, the available hardware resources of the device, how far is the user from the cell tower, session code execution overhead [81]. It also depends crucially on the radio environment, which is subject to very rapid fluctuations. For example, Rayleigh fading conditions change on a millisecond basis, as do scheduling and cross-traffic (such as data from other users). Further, as the user moves, the performance is subject to change over time, the number of UEs that use the cell of a the eNodeB also changes due to the same reason.

In LTE downlink, radio quality of the wireless channel varies quickly over time due to multi-path fading, interference, and signal attenuation (path loss) which depends on the propagation distance. It is evident that the allocation of radio resources among users depends highly on properties of transmitted traffic, signal strengths, activities of other users (since it is a shared-resources network) etc. Therefore, the queuing scheme tends to be more complicated than just a simple FIFO concept, and the assigned capacity for an individual user could also be affected by rapid fluctuations [33].

In contrast to wired links, the throughput of cellular networks is hardly equal to the data rate, due to variety of factors that were mentioned previously. Wireless networks have several characteristics such as transmission retries, variable packet error rate, contention and wireless link rate adaptation [83]. Moreover, the wireless medium is also a shared channel, and the available bandwidth changes with the number of hosts competing for the resources the channel offers [84]. Therefore, bandwidth estimation in cellular networks is more challenging. To aggravate the problem, the ISP use buffer configurations and different admission control policies [82] according to their operational strategy or business
model. Most of the operational LTE networks uses a proportional fair scheduler that takes the variations of radio quality and number of UEs into consideration, but still it remains as one of the major reasons of the measurements’ accuracy degradation is the behavior of the packet scheduler at the LTE link layer which changes dynamically. The packet scheduler causes severe disturbance to the queuing delays observed at a receiver [68]. It is difficult to calculate the cross-traffic impact on the available bandwidth as there is no simple formula to do so, since it mainly depends on the varying wireless channel conditions, scheduling and modulation techniques, packet size, pre-configured QoS metrics, etc. [39]. The challenges also include the interference in addition to resource constraints (power and data) on mobile end points. Real-time nature exacerbates the problem which makes it important to balance the accuracy of the measurements, the time it takes to deliver an estimation and amount of traffic used to probe the network.

![Fig. 6. A representation of RF limitations in cellular networks](image)

The ABM tools face three main challenges in high-speed networks which are:

- Finding a precise fine-scale inter-packet gaps at the sender, accurately timestamping packet arrival at the receiver and dealing with the noise [85]. The susceptibility to the disturbance by the small-scale buffering related noise. In practical, ABM tools have been shown poor performance [17] in real environment evaluations in contrary to what they show in simulation testbeds. The fine-scale buffering between the transmitter and the receiver in a shared resources system introduces noise in the gap between packets, which is one of the major factors in real environment that can affects the analysis of one-way delays or the inter-packet gaps at the receiver for such bandwidth estimation. It has more impact in
high speed networks since the gaps and delays are much smaller in scale. Furthermore, the available bandwidth estimation techniques need to deal with noisy data observed from the measurements which is, for instance, caused by inaccurate time-stamping.

Randomness could also be considered as one of the issues. One of the practical approaches to overcome and deal with the issues of randomness is to repeat the measurements multiple time and average the results. This approach was used in pathChirp and Spruce, or to decide based on the majority, as it was done in Pathload which reports an undecided bandwidth. This approach was also used in BART from repeated measurements and to monitor the variations of the available bandwidth.

In addition to the three challenges (and the randomness), it also faces two main challenges in cellular networks which are:

1. Handover procedures and admission control policies: it is an impacting factor that is controlled by ISPs which responsible for balancing the load and bandwidth reservation. It depends on the instantaneous number of users.
2. Mobility: The mobility of uses causes huge signal fluctuation which in turn causes bandwidth degradation, whereas it is relatively stable when the mobile device does not move or moves in uniform velocity.

To sum it up, bandwidth of cellular networks is noticeably changeable in comparison with Wi-Fi and wired networks. This is why that most of the bandwidth estimation tools fail in mobile networks.

Thus, an accurate bandwidth estimate is often obtained at the expense of high data costs and delays that may become invalid rapidly.
4 Implementation: Protocols and Technologies

In this chapter, the technologies and the protocol that Blixt™ relies on are introduced. In addition, Blixt™ algorithm and class diagram are also presented. The packet probing parameters are also discussed.

4.1 A One-Way Active Measurement Protocol (OWAMP)

OWAMP is defined in RFC 4656. The aim of it is to measure one-way delay by providing a precise and interoperable mechanism [87]. This protocol takes security threats in consideration, since the protocol traffic is hard to be detected and manipulated as it uses ordinary UDP packets which makes the measurements hard to be interfered. It is difficult for attackers to change the time stamps undetectably, since it is possible to encrypt the test packets. As it is defined in the RFC, the architecture of OWAMP architecture is divided in five different roles to make it more flexible. The five roles are:

1. Session ender: The sending node of the testing session.
2. Session receiver: The receiving node of the testing session.
3. Server: It is responsible for managing the testing sessions, the session states configuration in the session endpoints, and returning the results of the testing session.
4. Control client: It sends the requests to initiate the testing sessions, and it also triggers the start or termination of testing sessions.
5. Fetch client: It sends the requests to fetch the results of completed test sessions.

The OWAMP protocol is divided into two sub-protocols which are test protocol and the control part. The first one is responsible for sending the UDP test packets from the sender to the receiver. The IP addresses and port numbers are predefined in the session initialization [10]. The second protocol is responsible for initiating, starting and stopping the session. It also fetches the test results from the receiver. The principle of the protocol is as follows: the sender node sends test packets to the receiver along with the packets’ timestamps. The sequence numbers and TTLs are recorded on arrival at the receiver. The clocks of both the sender and the receiver need to be synchronized since
OWAMP measures the one-way delay by comparing the timestamps on the sender’s and receiver’s node.

4.2 A Two-Way Active Measurement Protocol (TWAMP)

Two Way Active Measurement Protocol (TWAMP) is defined in RFC 5357. TWAMP is predominantly utilized in active measurement for measuring two-way network’s metrics such as packet loss, latency, packet reorder and duplication. It obviates the need for special proprietary protocols and provides interoperability. It is not required to synchronize the clocks of the participation hosts to measure the RTT.

Similarly to OWAMP, TWAMP also consists of two inter-related protocols, test and control, but there are some differences between OWAMP and TWAMP. The session reflector replaces the session receiver which it can create and send test packets when it receives the test packets from a session sender. In contrary to the session receiver, the session reflector does not collect any information from the test packets, since the RTT can only be measured after the reflected test packet has been sent back and received by the session sender. There is another exception, which is the server cannot return the results of a test session since the session reflector does not collect any delay information. Thus, there is no need for a fetch client and that is why it disappears from the architecture of TWAMP. The control client is responsible for establishing a test session with the Server through non-standard means since they are located on the same host. The sender sends test packets to the reflector once the session is established, and then the session reflector reflects back the packets, so that the sender can collect RTT data.

Figure 7 shows the architecture of TWAMP. TWAMP endpoints can be categorized as either the session-sender or the session-reflectors responder nodes. The session-sender node is responsible for generating test traffic and processing the returned traffic to measure the network performance. It also rules of the measurement process and controls all aspects of the test. The reflector node processes and returns test traffic received from the sender node. Besides looping back the traffic, the reflector (sometimes known as responder) also inserts and processes timestamps and sequence numbers in the response packet for more accurate one-way measurement of performance.
There are three modes in TWAMP-Test protocol: encrypted, authenticated and unauthenticated. The control protocol uses the TCP protocol to establish the session on the default port 862 which is the preferred one.

There are two modes to define TWAMP, the full mode and the light mode.

### 4.2.1 TWAMP full mode

The TWAMP full mode test is designed to work in a client-server relationship, where the test control may be managed by separate devices from the devices that will be sending and receiving the test traffic [91]. There are two protocols linking the devices in question:

TWAMP-Control and TWAMP-Test. Test sessions are initiated, started and stopped using TWAMP-Control and test packets are exchanged between TWAMP nodes through TWAMP-Test. The test is initiated by a controller requesting a test run to a server, which then initiates the appropriate processes to reflect traffic from the sender. When the session is confirmed, the sender sends the test packets under the control of the client.
4.2.2 TWAMP light mode

The main purpose of designing TWAMP light mode is to implement the TWAMP standard in components that act as active responders to TWAMP controllers [91].

In TWAMP light mode, TWAMP control protocol is eliminated, and the session-sender, control-client and server are combined into one host: the controller. The function of the session-reflector is executed by a separate host: the responder. The responder, or in this case the session-reflector alone, simply reflects the coming packets back to the controller. Simultaneously, the responder copies required data and generates timestamp values and sequence numbers without knowledge of the state of the session. This is the simplest form of deploying TWAMP with a session-reflector.

4.3 Android OS and supported APIs

Google’s Android OS has gained a high market presence which can be represented as solution stack for the mobile devices, system applications, and the middleware [92]. Android OS platform can be subdivided into following 5 layers [93] as shown in figure 8.

![Android architecture](image)

**Fig. 8.** Android architecture
Android has a set of C/C++ libraries. The main core libraries can be accessed by the developers through the application framework layer. The core libraries provide most of the functionality are mainly available in the core libraries of Java.

Android provides 3 sets of APIs, namely Software Development Kit (SDK) with Java Support, Native Development Kit (NDK) with native support and render script support. SDK provides the necessary libraries required by the developers to create applications compatible with Android device using Java. Java allows easy development of applications and has other advantages such as language level security and portability of programs developed in SDK being easily run on different Android devices. The source code of the applications developed using SDK is compiled to bytecode on a developer machine. On the launch of the application developed, Dalvik VM as an Android execution engine interprets the bytecode, or it uses the JIT compiler for the compilation of byte code into machine instructions and hence execute it.

NDK is a tool set which provides usage of native code in the Android application. Hence it allows the developers to create and compile code in C/C++ for an Android platform. Android NDK allows us to include native code for some critical parts of the application. It also allows the reuse of C/C++ codes. In Android NDK, the code developed is compiled in a machine code and it is then packaged into an execution file format (.apk). The native code developed is executed through the Java Native Interface (JNI). The usage of native code is restricted to use functionalities provided by Android. Hence usage of JNI allows access to all functionalities through SDK API from the native code. For NDK applications developed to be able to run over various CPU architecture, the developer needs to develop and built different versions of native library for the targeted application binary interface.

The Radio Interface Layer (RIL) provides an access to the hardware’s radio by using the Android telephony services API android.telephony which was used by Blixt. The RIL has two main parts which are Vendor RIL and RIL Daemon. The latter component initializes the first one. The API level of the device used for performing Blixt™ measurement is 25 as the Android code name is Nougat and the version is 7.1. Android OS limits the types of measurements that can be taken from the radio environment.
4.4 Blixt™ algorithm and technique

Blixt™ relies on a time-stamping protocol which is TWAMP (see section 4.2). Other time-stamping protocols could have been used; the reason for selecting TWAMP was that it is a standard protocol in the field which has a simple implementation and is easily extendable and that is what Blixt™ does, it uses an improved version of TWAMP.

The main idea of the design is based on a device communicating with an ABM server, where the server reflects the packets back to the sender, including timestamps and other information included in the packets. The device can then easily be configured to test the performance of different parts of the network by accessing different servers.

The packet train transmissions are designed to make full use of the maximum available bandwidth, without limiting the throughput rate by the slow start or low-load scheduling mechanisms. Data is sent in short, intense bursts (“chirps”) with much longer pauses in between. The use of short bursts meets the requirement of a high enough temporal resolution (with minimal intrusiveness) which means at least once in a while, it can be expected that optimal radio conditions may prevail throughout a data burst (provided that the network configuration and the device’s position permit this in the first place). With multiple packets sent back-to-back and scheduled in consecutive TTIs, it is ensured that the ABM tool has the network’s full available capacity allocated to it at least for some TTIs in the middle of the burst. The reason for sending multiple contiguous packets is that if only one packet is sent, it is likely that one TTI will not be filled, or will be scheduled via TTI, which means that the full bandwidth available in any TTI will not be used. The peak load is high enough to reach the network’s theoretical maximum, while the average load is kept low. This scheme allows us to sound out the available bandwidth while still making minimum use of network resources. The algorithm adapts to network configuration parameters: the amount of data sent is adjusted according to the network’s maximum throughput, while keeping the level of intrusiveness to a minimum at all times.

4.5 Blixt™ measurement method

Data bursts are sent at one-second intervals (by default). In between these bursts, whose
duration is always a small fraction of a second, nothing is sent. Each data burst consists of a number of packets sent back-to-back, collectively referred to as a packet train.

![Train of packets](image)

**Fig. 9.** Train of packets

According to TWAMP specifications, the packets are time-stamped four times, the first one $t_1$ when the session-sender sends the packets to the session-reflector while $t_2$ represents the time stamp at the session-reflector (server) when it receives the packets. Similarly, the reflector time-stamps the packets when it reflects them back to the client $t_3$ and the last time-stamp $t_4$ is when the client receives the reflected packets from the server.

By using this information about the time-stamping, the packet size and the number of packets send to the server, it becomes feasible to estimate the available bandwidth for both the downlink $D_i$ and the uplink $U_i$ as well as the RTT by using the following general formulas:

$$D_i (bps) = \frac{\text{Packet size (bytes)} \times 8 \times \text{no. of packets per train}}{t_4 \text{ of the last packet} - t_4 \text{ of the first packet}}$$  \hspace{1cm} (13)

$$U_i (bps) = \frac{\text{Packet size (bytes)} \times 8 \times \text{no. of packets per train}}{t_2 \text{ of the last packet} - t_2 \text{ of the first packet}}$$  \hspace{1cm} (14)

$$\text{RTT (sec)} = t_4 \text{(average)} - t_1 \text{(average)} - (t_3 \text{(average)} - t_2 \text{(average)})$$  \hspace{1cm} (15)
An implementation of Blixt™ for a two-host scenario consisting of a session sender and a session reflector is done. ABM tool developed for Android mobile devices acts as a session sender by generating probing packets. The TWAMP Session Sender was targeted towards an Android OS platform whereas the TWAMP server or the session reflector was targeted towards Linux machine.

### 4.5.1 Blixt™ client

Blixt client is an Android application which initiates the TWAMP session by sending UDP packets as a probe traffic to the session reflector based on PRM model. To have higher accuracy of the timestamps, native API (NDK) is better to use. Blixt client source code is written in C++, and it uses Linux virtual machine (Ubuntu 16.04 LTS on virtual box) to build Blixt library that is needed to be added to the Android application code which is written in Java (this procedure used for the modifications and development in the research). NDK compiles the C++ code and into a native library and adds it to the executable apk’s package by using Gradle.

The IP address and the port number associated for the session are carried by the tool to the session reflector of TWAMP as an input. The size of packet, number of packets per train to be sent with the interval time can be specified. The application receives the reflected train of packets from the TWAMP server to compute the available bandwidth. A graph is used to display the measurement’s values. It displays the values in Mbps for both uplink and downlink available bandwidth. The graph is based on GraphView library [94] for Android which is used for creating diagrams. To get started with this library, it requires to get its dependency (available in the website, download and getting started) included in Gradle build (modular app) in Android studio.

Figure 9 shows the class diagram of Blixt™’s LTE downlink algorithm and its inheritance. (The diagram was generated with Doxygen which is a tool for generating documentation.)
Fig. 10. Blixt™’s algorithm class diagram

*Blixt algorithm* implements the idea of Blixt which means it contains the equation used for the estimation. *Blixt algorithm generic* is responsible for adapting the probing parameters based on the bandwidth factor, while *Blixt algorithm LTE downlink* contains the information related to this specific access technology, such as the probing parameters values. This class diagram shows a simplified version of Blixt™ design algorithm and it only shows the LTE downlink case.

Blixt™ client displays the network type that the device is connected to by using telephony API (LTE in this scenario). The device was used for the test is Sony XZ consisting of Qualcomm® Snapdragon™ 820, 64-bit Quad-core processor and it supports LTE (4G) Cat 9 (device specifications according to SONY mobile).

### 4.5.2 Blixt™ server

Blixt™ server is a Linux-based server operates on a single port as defined in TWAMP standard for exchanging TWAMP test packets. The server accepts incoming UDP packets on this port from the various clients. The server receives a train of packets from the client while timestamping on reception of packets and buffers them until the last packet in the train is received. On reception of last packet in train, it reflects back the whole train of packets with an interval as defined in the received packet information. Hence the server merely acts as a reflector by doing no calculations on the received train of packets as opposed to OWAMP.
4.6 Packet probing parameters

Blixt™ has been modified in order to test out the effect of changing the values of the parameters on the measurement’s accuracy. The figure below shows the user interface with the packet probing parameters as inputs.

![Blixt™ Client UI](image)

**Fig. 11.** Blixt™’s client UI

4.6.1 Probing Packet Size

The size of probing packets is one of the important parameters to consider. In the case of using a large packet size, the time interval increases and results in interference with the cross traffic as a consequence. If the packet size is too small, that would not be an ideal solution as well. The packet dispersion decreases when the length of the packet decreases and makes the measurements more sensitive to the case in which the first packet faces a longer delay than the rest. The experiments with small probing packets show that it has high sensitivity to the interference which makes it reasonable to send fairly large probing packets.
4.6.2 The number of probing packets

As it was mentioned in chapter 3, the internet traffic consists of data burst which means short glance is not enough to observe the average traffic load. That is the reason for sending a somewhat large number of packets to probe the network. It should also be considered that sending too many packets can cause larger queues and increases the network load in addition to packet loss.

4.6.3 Transmission time interval

The concept of drive testing is to collect Receive Signal Level (RSL) measurements, from different regions which differ in distance sizes. Binning is a procedure to shape the coverage area. Mobile operators measure the coverage area based on bin sizes which are 50x50, and 25x25 meters of resolution. If the speed of the car is 90 km/h, this means it is required to measure the available bandwidth every 1s,

\[ T_t = \frac{(25 \times 3600)}{(90000)} = 1 \text{s} \]

Where \( T_t \) is the transmission time interval.
5 Experiments, Results and Discussion

The chapter presents the experiments have been done to observe the impact of tuning the packet probing parameters on the measurement performance. This chapter also presents an evaluation of Blixt™ and comparison of its estimation with traditional methods of measuring the bandwidth. This chapter is concluded with a discussion and sustainability analysis.

5.1 The measurements

The experiments were performed over a live commercial 4G/LTE networks (where conditions may change over time), for two mobile operators, tele2 and Telia (the results provided are from the experiments performed over Telia network). The site where the measurements have been conducted was chosen based on the information obtained from Bredbandskartan.se website which provides information about the broadband access in Sweden. The measurements were chosen to be performed at Skellefteå Kraft Arena which is a sport venue. The experiments were mainly performed in the late evenings to have higher bandwidth.

Fig. 12. Mobile coverage at Skellefteå Kraft Arena
The ground truth of the available bandwidth is unknown since it is not feasible to access the intermediate nodes of the mobile operator network. For this reason, the maximum achievable FTP throughput was used as a ground truth of the true available bandwidth. It has been also used by other researchers along with TCP [78]. FTP download test provided by TEMST™ pocket was used. In TEMST™ pocket, there automated testing of services, by using previously written scripts are used, for instance, Ping, HTTP, FTP, etc. It is possible to start and stop a script with menu commands. The FTP server is provided by the academic computer club, Umeå University, Sweden, and it is available at the following link ftp://ftp.acc.umu.se. The file used for the test is Arch Linux which is Linux distribution for computers based on x86-64 architectures with a size of ~ 550MB. Firstly, the throughput was measured the by performing the FTP test and then the available bandwidth of the downlink was measured by Blixt™. Then the procedure was reversed to observe if any changes occur.

As it is not feasible to design a controlled environment to measure the available bandwidth in mobile networks and to assure the consistency of the results, the solution was to conduct multiple experiments under similar conditions (Same device, time and location). The measurements were alternated between Blixt and FTP tests to align the assessment results.

In the first set of experiments, it was tested how the available bandwidth changes by tuning the values of the three main packet probing parameters (one at a time) which are: the size of packet, number of packets/train and the Transmission Time Interval (TTI).

![Fig. 13. Blixt™ measurements network topology](image)

The second set of the experiments was the evaluation of Blixt™ versus FTP test (TEMS™ pocket used for this test), iPerf 3 and nPerf.
5.1.1  Packet probing parameters study

1. The packet’s size

The experiments which aim to test out how the available bandwidth varies by changing the packet’s size were performed. The number of packet and the TTI were kept constant, and equal to 190 packets, 1000 ms consecutively. Figure 10 shows that the larger the packet size is the higher the available bandwidth estimations are.

![Figure 10](image)

**Fig. 14.** Blixt™’s measurements for different values of packet’s size

The reason for these observations is due to the active probing method’s behavior, as the packet probes increases, it gets closer to the actual bandwidth. The observation is that the packet size which is equal to 1400 bytes gives better measurements than the lower values of the packet sizes when it compared to FTP throughput.

2. The number of packets per train

The number of packets is a crucial parameter for determining the accuracy and the level of intrusiveness which are the major concerns regarding the available bandwidth measurement in high-speed cellular networks. The number of packets
was chosen to vary between 20 to 4000, while the packet size and the TTI were remained as constants and equal to 1400 bytes and 1000 ms consecutively. The measurements were also compared to FTP throughput. The observation is that as the number of packets increases, the available bandwidth increases too, and its average gets closer to the FTP throughput, but not in a linear form, which means it is possible to have an acceptable estimation when the number of packets is between 130 and 500. To keep the level of intrusiveness as minimum as possible it is recommended to use the lowest number of packets to probe the network which gives accurate measurements. Figure 11 shows how the available bandwidth varies by changing the number of packets versus the FTP throughput.
Fig. 15. Blixt™ measurements versus FTP throughput for different number of probing packets: a) 20 Packet; b) 500 packets; c) 1000 packets; d) 4000 packets

The disadvantages of sending long packet trains are: overhead, delay and increased risk for packet loss (to name few).

3. Transmission time interval

TTI was the third tested parameter in order to observe its effect on the measurements’ performance. Different values were chosen, starting from 500 ms up to 10000 ms, the packet size was 1400 bytes and the number of packets was 190
packets during all the experiments for observing how the TTI affects the measurements accuracy. Figure 12 shows how the measurements change as a result of changing the value of TTI.

(a)

(b)

(c)
Fig. 16. Blixt™ measurement for different values of TTI: a) 500 ms; b) 1000 ms; c) 2000 ms; d) 5000 ms; e) 10000 ms

It was observed that as the TTI increases, measurements points become less, and the available bandwidth values drop down as well due to the reduced probing traffic over the same period of time. Table 2 shows the values used for the experiments.

Table 2. The packet probing parameters’ values used for the experiments

<table>
<thead>
<tr>
<th>Packet size (Bytes)</th>
<th>Number of packets</th>
<th>TTI (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>20</td>
<td>500</td>
</tr>
<tr>
<td>1000</td>
<td>190</td>
<td>1000</td>
</tr>
<tr>
<td>1200</td>
<td>500</td>
<td>2000</td>
</tr>
<tr>
<td>1400</td>
<td>1000</td>
<td>5000</td>
</tr>
<tr>
<td>2000</td>
<td>4000</td>
<td>10000</td>
</tr>
</tbody>
</table>
5.1.2 Performance evaluation

The second part of the experiments was about evaluating the performance of Blixt™ against FTP tests (by using TEMS pocket for this test), iPerf 3 and nPerf. The benchmarking was limited to the commonly known and widely used tools since the details of how ABM tools are implemented and how they perform the measurement are not broadly available.

The optimal values of the packet probing parameters which give better measurements were chosen first phase of measurements to be used in the evaluation experiments. The values chosen were as follows: the packet size is 1400 bytes; the number of packets is 190 and the TTI is equal to 1000 ms.

IPerf [95] is an open source software and widely used for measuring the network path performance and it “has become an unofficial standard in the research networking community” [16]. IPerf originates TCP/UDP data packets between client and server hops to measure the throughput of the network. IPerf usability and performance are often studied in several research works, and the authors of [26] used it in estimation the throughput of 4G networks. Another benchmarking tool was used for the comparison, which is nPerf [96]. nPerf is a software which is available on Android and aims to measure the internet quality. This application is edited by a French based company called nPerf SAS. In order to test out the performance of iPerf 3, magic iPerf was used which is an Android application as an iPerf 3 client. IPerf 3 server was installed and ran over an Amazon EC2 instance, which is located in (eu-central-1, Frankfurt) region to be accessible over the internet with a public IP. Figure 14 illustrates iPerf 3’s experiment setup:

Fig. 17. iPerf 3 measurement network topology
The experiments show that Blixt gives the most accurate measurements compared to iPerf 3 and nPerf and it has the lowest level of intrusiveness (about 10 times less than the FTP test) among the tested tools.

**Fig. 18.** iPerf 3 measurement network topology

The results show that Blixt closely follows the FTP throughput compared to iPerf 3 and even nPerf. Figure 18 shows the level of intrusiveness introduced by each tool compared to FTP test.

**Fig. 19.** Intrusiveness level compared to FTP
It should be noted that the level of intrusiveness is relative to the intrusiveness of the FTP test which assumed to be 100% as it exploits the whole bandwidth. Moreover, these measurements consider the carrier aggregation scenario, Blixt™ forms less than 5% of intrusiveness based on the experiments performed prior to the design by InfoVista and it was validated in single carrier scenario. The figure below shows the difference between the data usage over time of the FTP test in comparison with the very short Blixt™’s bursts.

Fig. 20. Blixt train of packets vs FTP download data traffic
5.2 Discussion

Based on the result, Blixt™ estimation show that it follows the FTP throughput pattern in most of the measurement values. In very few points, it was observed a bit lower values because the tests were not done simultaneously (even if two devices were used, that does not help since it measures the available bandwidth which means the remaining bandwidth). To illustrate this point, the eNodeB tends to select lower bit rate in the case of the mobile’s RSSI is low, in order to prevent large bit errors.

Lightweight packet train used to achieve the goal of accurate and fast estimation. As the length of packet trains increases, the estimations become more reliable. At a train length of 130-190 packets, the estimations become stable enough to give a result that is accurate enough to be of value to the application.

To achieve accurate estimations, the active measurement techniques need strong time-related constraints. The constraints are:

1. Accurate timing in sending the probing packets which have to match the inter-departure constraints.
2. Arrived probing packets need to be time-stamped accurately.
3. The synchronization of the sender and the receiver needs to be accurate in order compare the clock between the two nodes.

One of the most important considerations to be discussed is the feasibility of sustainability. Blixt™ helps in reducing the carbon emission by performing three measurements in the same time (Available Bandwidth for both uplink and downlink, and the Round-Trip Time RTT) which means only one drive-test is needed to perform all these measurements. It also reduces the energy consumption both in cell and the network, by reducing the data process compared to other tool that uses other protocols such us File Transfer Protocol FTP. It also helps in enhancing the subscribers’ QoE by minimizing the probing data that utilize the shared resources network, i.e. the cellular network. It reduces the cost when benchmark the competitors.

The figure below shows the different aspect of where this solution can contribute to the sustainability.
Fig. 21. Sustainability analysis of Blixt™
6 Conclusion and Future Work

The accuracy of the ABM method is determined by the packet size, the number of packets in the packet train, the gaps between two packet trains as well as by the instantaneous data rate, which is chosen to correspond to the maximum bandwidth according to current System Information parameters, UE capabilities, and other settings. The use of less packets per train or smaller packet sizes would reduce the time that the tool takes to estimate, but in the cost of reducing the accuracy. The accuracy of measurements has a correlation with the amount of probe samples that have been delivered into the data pipe per iteration. The more samples are injected into the connection, the higher accuracy is provided by the given measurement method. However, injection of probe samples into the network could also decrease of service quality or cause congestion in the network. That is why it is important to find some optimal value of amount of probe samples per iteration that will be big enough to provide an acceptable bandwidth accuracy and Small enough to not consume a lot of network resources. Blixt™ solves the problem by performing the required level of test (getting enough data) and keeping measurement intrusiveness to an absolute minimum.

Blixt™ creates short traffic bursts. These bursts last for a very short period, with relatively large empty periods between the packet trains. Therefore, the average probing packets rate represents a small portion of the overall available bandwidth.

The main challenge in active probing techniques is minimizing the amount of data to be sent over the network and still getting the most accurate estimations while keeping the level of intrusiveness low. It can be hard to say that a tool is intrusive if it only sends small amount of UDP bursts to measure RTT in the case that the available bandwidth is not low. For instance, if the tool consumes more than 20% of the available bandwidth in a shared resources environment, it can be said that it is intrusive.

For future work, using machine learning algorithms to predict the available bandwidth based on the previous measurements could be considered, hence reducing the interfering with the network which results in less intrusiveness. In order to success in this approach, the algorithm needs to be trained with large set of data to achieve better accuracy, the data
should be good enough and representative.
References


36. D.-G. Herculea, M. Haddad, V. Capdevielle, and C. S. Chen, “Poster: Network-


Appendix 1. Results from Experiments: iPerf 3 Measurements

The screenshots below show the measurements performed by using iPerf 3, both for the iPerf 3 client (Magic iPerf) and for the server which I ran on an AWS EC2 instance.

On the client side, `-c` means that this node works as a client, then it is followed by the IP address of iPerf 3 server. `-t 40` means that the measurements perform for 40 seconds, and `-i 1` means that it measures every 1 second. For the server, it is required to install iPerf 3 packages, and to run the server, `iperf3 -s` command was used.