MOBILE INTERNET TRAFFIC MEASUREMENT AND MODELING BASED ON DATA FROM COMMERCIAL GPRS NETWORKS

DISSERTATION

to obtain
the doctor's degree at the University of Twente,
on the authority of the rector magnificus,
prof.dr. F.A. van Vught,
on account of the decision of the graduation committee,
to be publicly defended
on Wednesday December, 8th, 2004 at 16.45

by

Roger August Kalden
born on December, 26th, 1971
in Kassel, Germany

This dissertation is approved by the promoter
Prof. dr. ir. Boudewijn R.H.M. Haverkort

Summary

New mobile networks of the third and evolved second generation have been deployed that enable a mobile accessible, always available wireless Internet. To exploit the full potential from those new mobile access network technologies, and to meet high user expectations, these networks need to be well designed, dimensioned and deployed. An important basis for these tasks are accurate traffic models, and sound traffic estimates. However, such knowledge does not yet exist for mobile networks.

In this thesis, measurements from commercially deployed General Packet Radio Service (GPRS) networks are investigated to derive novel knowledge on mobile network usage and its traffic characteristics to fill this gap.

As a prerequisite, a measurement setup has been developed, allowing parallel capturing of user payload data at the level of IP packets and mobile network specific events related to session and mobility management. Based on this setup we conducted the following four studies.

First, a comprehensive study on the application and session usage in GPRS is presented. It reveals the absolute dominance of the novel Wireless Application Protocol (WAP) and Multimedia Messaging Service (MMS) services, in terms of traffic volume and usage frequency.

Second, models for WAP and MMS application flows are derived. We show that the distribution of the length of those flows is not heavy-tailed. Furthermore, we show that WAP and MMS flows are very short in terms of number of packets exchanged per session.

Third, the mobility of GPRS users is investigated. We develop a model for the so-called perceived mobility, based on GPRS mobility management events. We show that users are moderately mobile currently, leading to small impacts on performance. According to our model the cell reselection inter-arrival times can be best modeled with heavy-tailed distributions.

Fourth, we assess the self-similarity property of GPRS traffic. We separately study aggregated, web related and WAP related traffic. Based on the statistically very robust Hurst estimation method by Abry and Veitch we show that GPRS traffic arrival processes are long-range dependent (LRD).

Acknowledgements

Foremost, I would like to express my grateful thanks to my parents Beatrix and Karl-Heinz Kalden, and my sister Susanne, for supporting me all my life, showing me the right values in life, and making it possible for me to study at a university.

I would especially like to thank my advisor Prof. dr. ir. Boudewijn Haverkort, for supervising my thesis work, and believing in my ability to finish this work besides my regular work at Ericsson. He always provided valuable feedback and great support in all matters when needed.

I am also thankful to Ericsson, foremost for providing me with an excellent work environment: an inspiring work place with many challenges, lots of freedom, interesting work and best and nicest colleagues. In particular, I am thankful for supporting me in my wish to work on my doctoral thesis. Special thanks go to Fiona Williams and Norbert Niebert for believing in me, and for supporting me in setting up the MeaDoW project. I would also like to thank all my colleagues, especially the SimPerf group. This includes in particular Michael Meyer, who always gave me good feedback on my research. He also provided very helpful advice on my technical writing. Reiner Ludwig, for being responsible for getting me hired by Ericsson Research in the first place. He always provided valuable feedback on my research work, and he is an excellent example for how to work efficiently.

I highly appreciate the support I got from Vodafone in the joint Ericsson/Vodafone MeaDoW project, in particular for providing access to the gigabytes of valuable measurement data. This was the most important cornerstone in my endeavor to do my thesis.

I am also thankful to my MeaDoW project team for supporting me in this interesting research activity, spanning over two companies. At Vodafone this includes, but is not limited to, Johan Bax, Bert Haverkamp, Bart Sanders, Bianca Wouters, Willem Van de Maar and Simon de Kerpel. At Ericsson Hungary this includes, but is not limited to András Veres, István Szabó, Balázs Péter Gerõ, the MONIQ development crew and especially Tamás Varga. At Ericsson Germany this includes Jan Scheurich, and the diploma students Daniel Spelmezan and Sami Ibrahim for implementing some of the tools.

Finally, I would like to thank my good friends for being so patient with me over the last few years, when I was rather sitting at my computer instead of socializing with them, but also for distracting me from work when it was necessary. In particular to Alexandra Siemes, for being my best friend for such a long time and supporting me in good and difficult times and for focusing me on studies and work without letting me to forget the fun.

Last but not least, I want to thank my partner Andrea Rüster for being an inspiring source in my life and letting me understand to think creative and not just technical, and for being a very caring mother for our son Levi. ‘Thank you’ also to my son Maximilian Levi for being such a cute and smart baby and for giving me the final reason to finish up my thesis in due time.

Confidentiality

As the measured data used for the research in this dissertation is operator-restricted and we are obliged to preserve confidentiality for the operator and privacy of the user by law, some results needed to be abstracted. This affects in particular the deployed network infrastructure, user details, and results on absolute data *volume*. This has been done with great care to fulfill the requirements of confidentiality and privacy as well as to fulfill the requirements of a scientific research work. We confirm that the hiding of some explicit numbers or other details does not change the principle results of the work undertaken. Results not shown are, to the best of our knowledge, not in contradiction to the derived conclusions.

Table of Contents

1 Introduction

This chapter provides a motivation for the subject of this dissertation in section 1.1; describes the issues addressed and the key contributions made by this dissertation in section 1.2; and outlines the dissertation in section 1.3.

1.1 Motivation

In networking and telephony two immense developments took place in the last decade: we witnessed the success story of the Internet and secondly the enormous success of the cellular wireless telephony. The number of hosts in the Internet has massively increased along with an even larger number of people using the Internet. Independently but in parallel, this was accompanied by an explosive growth in subscribers to cellular wireless systems throughout the world. Currently, a third interesting development is taking place, which is the merger of those two developments into a wireless and mobile accessible, always available Internet.

The mobile market has been growing rapidly over the last decade. The number of mobile users is approaching 1.2 billion in 2004 with around 500 thousand new subscribers being added each day (Figure 1-1) [UMTS03].

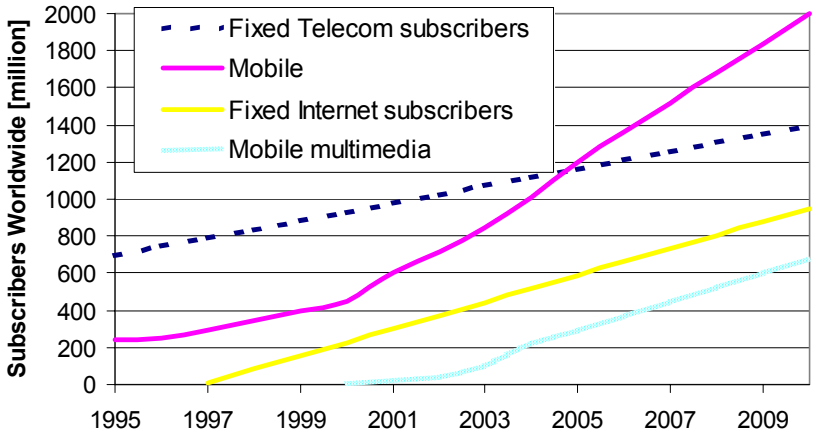


Figure 1-1: Trends in fixed and mobile Internet – source [UMTS03]

At the same time the number of server hosts in the Internet has grown to 250 million with an even higher number of Internet subscribers (Figure 1-2).

While the tremendous use of the Internet is mainly pushed by fixed line operators installing broadband access lines (like digital subscriber lines (xDSL) and cable modems), the merging of the Internet and mobile access technologies is fuelled by new packet-switched mobile networks like General

Packet Radio Service (GPRS) and Universal Mobile Telecommunications Systems (UMTS). Mobile networks of the second generation in the early 1990s already provided basic data access, which could be used to access the Internet. But only the current mobile access networks, (e.g., GPRS, UMTS), provide high enough data rates, large enough capacities and sufficient Internet Protocol (IP) integration that they can truly be considered for convenient Internet access. The practical data rates of current deployed mobile networks (e.g., GPRS) are in the order of analog fixed line modems (about 20-50 Kbit/s) but will soon be much higher. The next generation (e.g., UMTS) that is currently deployed already provides rates in the order of 384 Kbit/s and beyond. This allows the usage of Email, chat, Web browsing, and file transfer applications over mobile networks in a way the user is used to from wireline networks.

Furthermore, currently, two main service models push the wireless data usage. One is 'I-mode', developed by NTTDoCoMo and the other is 'Vodafone Live!', developed by Vodafone. The highest usage of these services can be found in Asia with about 50 million subscribers to services like 'I-mode' and 'Vodafone live!' alone in Japan in 2003 [HB04]. Compared to this, Europe is only in its beginnings, with about 500 thousand subscribers to 'I-mode' and 5 million subscribers to 'Vodafone live!' [HB04].

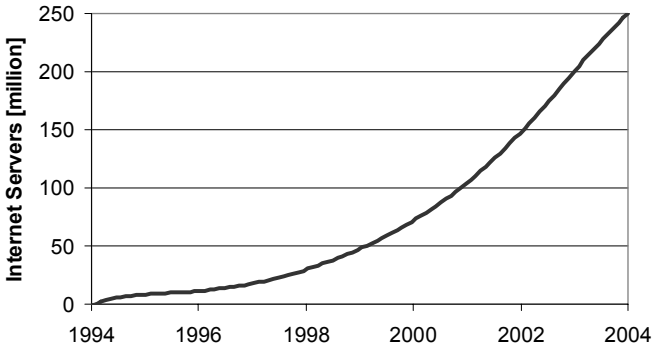


Figure 1-2: Trends in number of Internet hosts – source [ISC]

The expectation of users, concerning performance in mobile networks, is at the moment formed by their experience in wireline access networks. The user is interested in a 'well usable' service. This can be broken down into two fundamental performance metrics in traffic engineering: a high throughput and a low latency. Even though the new technologies add the novel feature of mobile data access, users still expect a high performance. To be successful with new Internet-like services over mobile access networks, the mobile network services need to fulfill those expectations. To achieve the full potential of the new mobile access network technologies, the networks need to be well designed, planned and deployed. Fundamental to the proper design, planning and deployment is the relationship between the

- quality of service (QoS) in the network,
- the deployed resources in the network,
- and the traffic characteristics and load in the network.

The QoS is described by the performance of the service and set by the expectations of the user. The deployed resources are the hard- and software invested by the operator. It is in the interest of the operator to keep the investment low. The load is a consequence of the user's activity and the application traffic. This is linked via the employed cost model to the revenue for the operator.

The objective of planning and designing data networks is to achieve a certain desired QoS, based on an assumed traffic load, while having a minimal investment. Assuming that the system can be well described and the target performance is given, often the predicted traffic load is a vague input parameter. The load needs to be known prior to calculating resource requirements. However, this information is often not available. Hence, the input parameters are based on expectations and forecasts. The key to success in network planning is the quality of the traffic estimate. A number of studies have shown how important it is to know the right traffic demand [WP98] [FL93] [FGWK98] [PKC97].

In the early days of packet-switched network planning, Markovian-based models and in particular Poisson models were matched to all traffic engineering problems, foremost, due to their nice tractability. But this resulted in underestimating link and buffer capacity requirements in the system [FL93]. We know today that packet-switched traffic has a complex fractal nature [WP98] [PF94]. And it was shown for a number of networks how the self-similarity property is linked with application statistics and protocol behavior (e.g., [CB96] [LWTW94] [PKC96] [FGWK98] [WPRT01]). Important to note is that this particular type of traffic resulted in (usually) higher capacity requirements and performance results, as compared to Markovian type traffic requirements. If not considered, delay and packet loss is increased. Therefore, it can be said that knowing the fundamental traffic characteristics is important to traffic engineering.

This importance of traffic modeling based on measurements in commercial networks was expressed by well-known researchers of the Internet community at a workshop in Schloss Dagstuhl in 1999 [CLR00]:

> "[…] The development of Internet-equivalent workloads would provide the ability to engineer better systems. It would allow for test system modifications to be done in a controlled environment without disturbing real systems. Furthermore, it would allow for more accurate benchmarking of systems. […] To achieve this capability, an understanding of how workloads are affected by spatial location/local infrastructure, cultural behavior, organizational role, time of day/week, and usage profile is needed. For example, the distribution of requests across content providers varies with cultural setting and with time of day/week, and with the interactions between the two as well. Creating a concise description or model of how workloads are dependent on these issues is a challenge. […]"

Sufficient knowledge on traffic in a network either requires that the network exists or that one can carry results over from other investigations. Therefore,

before the current mobile access networks were deployed, the traffic profile was based on vague assumptions. The anticipated application mixture consisted of Web, FTP, Email, and Telnet, and the parameters of the traffic models were based on extrapolation from wireline traffic measurements, e.g., [TR101112] [KCFD+00] [KBBM00] [SM00] [TGSL01] [KMM00] [LMW94]. This has been a valid practical approach to fill the knowledge gap on traffic demands.

However, mobile networks are not just another access technology but have some special features and characteristics which might influence the usage and consequently the traffic in the network:

(a) The unique capacity, throughput, and delay characteristics as well as the pricing structure could influence user behavior with respect to the chosen applications and the duration the applications are used. This mixture will probably be different from the wireline Internet application mixture.
(b) The operators of the new networks introduce new applications that are not used elsewhere in the Internet. For instance, in Japan such applications are ‘I-mode’ and ‘sha-mail’, while Wireless Application Protocol (WAP) and Multimedia Message Services (MMS) are promoted applications in ‘Vodafone live!’ in Europe. Such new applications can change the fundamental traffic characteristics. For instance, WAP and MMS currently do not deploy the Transport Control Protocol (TCP) but instead use the User Datagram Protocol (UDP). As Wireline networks carry predominantly TCP traffic [FML+03], a widespread usage of WAP and MMS would alter the traffic characteristics as known in wireline networks.
(c) An important feature of cellular networks is the support of mobility. We can assume that the typical usage scenario of such networks implies a mobile and nomadic usage. The user is ‘on the move’ while using such networks. That is, the user is either indeed moving, e.g., traveling by train, or he or she might be at least in a nomadic environment, e.g., waiting in an airport lounge. This assumption probably leads to at least two constraints that influence the traffic characteristics. First, the access terminal needs to be small and light weight. Second, the usage might be focused on services which are needed while being ‘on the move’. We can imagine that a correlation between the mobility of the user and the used applications exists.

It is yet unknown how much those aspects influence the traffic of mobile networks. Therefore it is essential to verify, update, and extend our knowledge about the traffic in mobile networks. Since in many countries mobile networks with data capabilities have been launched in recent years, we have today the possibility to validate our assumptions based on measurements in commercial networks.

To fill this large knowledge gap on cellular data network traffic, at least partially, is the motivation for this dissertation.

1.2 Issues addressed in this dissertation

Considering the missing information on mobile network traffic, lined out in the previous section, this dissertation provides five key contributions:

- A Measurement framework for GPRS which allows collecting Internet application relevant statistics as well as GPRS network specific events.
- Comprehensive GPRS results on application and session usage.
- Models of WAP and MMS flows, including extrapolation to future WAP 2.0 flows.
- Analysis for GPRS user mobility, including correlation of mobility and application usage, and, in particular, a model for the inter-arrival times between cell reselections.
- Assessment of the self-similarity properties of GPRS traffic and in particular for WAP traffic.

The first contribution is essential for the subsequent contributions. Therefore, we provide a specific measurement framework for GPRS comprising a larger number of tools inter-working.

The next four contributions are specific results on mobile network traffic. These results consider aspect (b) and (c) from the previous section. One aspect, which we do not address in depth, though highly interesting, is the relation between certain system parameters, marketing campaigns, tariff structures and the usage and traffic. This is aspect (a) and is left for a separate study.

Though these four areas might seem to be separate at first glance, there is an inherent relation between them, which makes the results in their entirety useful. We chose these aspects that might have an impact on the performance perception of the end user, and therefore are also of high value for the operator. While we do not investigate performance itself, but consider traffic modeling aspects, we will now outline their relation among each other and their association to performance.

First, we show the dominance of novel applications in GPRS. Second, we investigate the application flow length of the major GPRS applications. Third, we study the mobility of GPRS users and fourth we analyze the packet arrival process of GPRS traffic.

First, we show the dominance of novel applications in GPRS. This are WAP and MMS. The dominance is not only apparent in terms of users using them, but also in terms of number of flows and total bytes. Therefore the performance of these applications is of high importance when dimensioning the network. One potential influence on performance of the new applications can be seen when considering the application flow length. The performance of very short flows can be severely influenced by packet loss. We will show that in particular WAP application flows are extremely short. They are so short that it will

become critical for WAP2.0, which runs over TCP, if the application faces packet loss. Though cellular networks are not as error prone at the IP level as is often assumed, packets can get dropped while being mobile and during congestion at queues. Which leads to the next two aspects considered.

The impact of user mobility on performance can be evaluated by analyzing the correlation between mobility and GPRS usage. A high mobile usage makes it more likely that packets get delayed or lost than a stationary usage. To combat this, good seamless handover strategies are needed. However, those algorithms do not come free of charge. Therefore, assessing the mobility can help to optimize the handover strategies and adapt the transport protocols.

In order to prevent packet drops at queues due to congestion, proper queue dimensioning is needed. As many results for wireline networks have shown, it is important to understand the nature of the arrival process. This is in particular of interest, as we encounter new dominant applications in GPRS, which might exhibit specific arrival process structures. Assuming the wrong arrival process, can lead to underestimating the length of the queues. This results in higher packet loss, which has an especially high performance penalty on short flows. Therefore knowing the arrival process is an important condition to optimize performance.

1.3 Outline of this dissertation

This section motivates and describes the outline of this dissertation.

Chapter 2 provides the required background and gives further motivation for our measurement and modeling approach. We introduce the following three areas:

- The concept and architecture of wireless networks. Here we focus on their capabilities with respect to IP Internet access and how wireless networks and the Internet merge to the wireless Internet. This development motivates our measurement study.
- Second, we present some details of the Internet service architecture and the characteristics of its main applications. This is fundamental for understanding how we measure and derive application usage.
- The third major area we introduce is teletraffic theory, including the key relationship between quality of service, traffic demand and network resources. In particular, we discuss important aspects of traffic models and network measurements. Our measurement setup will be guided by these requirements.

As our measurement and modeling approach is realized in commercial GPRS networks, **Chapter 3** introduces the architecture, nodes and interfaces of GPRS. Important aspects we elaborate on are how GPRS is interconnected to packet-switched networks, how GPRS data transmission including session management is realized, and the concept of GPRS mobility management. Furthermore, the two novel GPRS applications WAP and MMS are explained in detail.

Chapter 4 is the first central part of the dissertation as it explains our measurement approach in GPRS that we use to obtain the results presented in the subsequent chapters. The objective of our measurement setup is to allow comprehensive investigations of various user behavior and application traffic aspects. We deploy two types of measurements to fulfill this. One allows us to study traffic details of all applications used over GPRS. The other measurement type allows us to capture GPRS network specific events. A unique feature of our measurement setup is the possibility to correlate these two types of measurements.

Chapter 4 also introduces our particular data traces obtained from the Gi and GPRS event measurements. The data traces are captured in some of Vodafone's first commercial GPRS networks in Europe. All subsequent results are based on these data traces.

The applications and protocols used in GPRS are analyzed and presented in **Chapter 5**. It reveals that WAP and MMS are the major applications in terms of subscriber penetration and data volume. We therefore investigate in the subsequent chapters how these new applications influence other traffic statistics of GPRS. We published earlier results about this in [KVWS03].

In **Chapter 6** we investigate the length of application flows based on UDP and TCP packet flows. We extrapolate our measurement data to include a future WAP 2.0 over TCP scenario. The remarkable result is that extremely short WAP flows dominate GPRS traffic. We outline how this result can have impact on the performance of transport layer protocols. We published a study on similar results in [KE04].

Of significant interest is furthermore to understand the user's mobility and its correlation to the application usage. These aspects together with a model for the cell reselection inter-arrival times are analyzed in **Chapter 7.** We published a study on these results in [KS04].

Chapter 8 deals with the self-similarity property of traffic arrival processes in GPRS. It introduces established testing methods of self-similar processes. Of these we use the very robust Hurst estimation method by Abry and Veitch to test aggregated GPRS traffic as well as separated WAP and Hyper Text Transfer Protocol (HTTP) traffic on its self-similar nature. We published these results in [KI04].

Chapter 9 concludes this dissertation by summarizing the main results. It also outlines further areas of research and possibilities of transferring the measurement and modeling approach to other cellular networks.

2 Background

This chapter provides an overview of the technology concepts as well as notions we build on in this thesis. The thesis is based on measurements in commercial cellular networks. Therefore, in section 2.1 the basic concept of cellular networks is presented. To understand the proposed measurement setup as well as the derived results, knowledge on the Internet protocols and services is appropriated. Hence, in section 2.2, important aspects of the Internet, are presented. Finally, section 2.3, provides an overview of important aspects of the teletraffic theory. Section 2.4 summarizes this chapter.

2.1 Cellular wireless communications

In this thesis we consider one particular cellular network service – the General Radio Packet Service (GPRS). However, many different cellular network technologies are installed around the world, today, and they all deploy a similar network architecture, which we describe in section 2.1.1. The performance of data transmission over cellular networks is strongly affected by the system characteristics. These are the determinants for the performance the user experiences and which might influence the network usage. Therefore, we consider such effects on performance in section 2.1.2. In section 2.1.3 we provide an overview on the current development of mobile networks with respect to their data transmission and IP access capabilities. We finish with section 2.1.4 on specific services in cellular wireless networks. We discuss in particular how these services will develop into a wireless Internet, which is the basis for our investigate usage scenario.

2.1.1 Cellular network architecture

Mobile networks are defined by two unique features: wireless access and mobility support. Wireless access frees the terminal from cables. The terminal communicates with the network by means of radio waves. With respect to this, Wireless Local Area Network (WLAN) systems (e.g., IEEE 802.11) and mobile networks are alike. In addition mobile networks have extensive support for mobility. The terminal device can move freely in a large area without losing connectivity to the network, and services can be used seamlessly while being mobile.

When using radio waves for communication, one has to cope with limited reachability. The signal strength decreases by at least $1/r^2$ as the receiver moves away from the transmitter. Hence, covering a wide geographical area requires frequent deployment of transmitters. Therefore, the geographical coverage area is divided into cells (see Figure 2-1). One cell is defined by the radio coverage of one antenna system. The designated mobility management protocol function in the system handles the seamless handover of ongoing calls by a mobile station, from one cell to another cell.

Figure 2-1 depicts the main components in a mobile network.[1] The network is divided into a fixed part, comprising the core network (CN) nodes and databases, as well as a radio access network (RAN) part, containing the transceiver stations as well as control units. The nodes in the core network are responsible for performing call processing and subscriber-handling related functions. The Mobile Switching Center (MSC) is similar to a normal switching node of the Public Switched Telephone Network (PSTN), plus functionality for registration, authentication, location updating, handovers, and call routing to roaming subscribers. New in mobile networks of the latest generation (2.5G and higher) is a Packet Support Node (PSN), which is responsible for handling packet switched calls and routing packet data thereof. The Home Location Register (HLR) is the most important database supporting the MSC, all administrative information of each subscriber, and the current location of the mobile is stored here. Furthermore, other supporting databases (e.g., VLR, AuC) and service nodes (e.g., SMSC) are placed in the core network. The radio network performs all radio related functions. The Base Station Controller (BSC) in the Radio Access Network handles the radio-channel setup, radio link transmission, and handovers. The Base Transceiver Station (BTS) is responsible for physical layer aspects of the radio transmission (e.g., signal modulation). The mobile station (MS) is a complex unit, providing an interface towards the user, realizing a service (e.g., speech call) and handling the communication with the relevant nodes in the network. It directly communicates with the BTS.

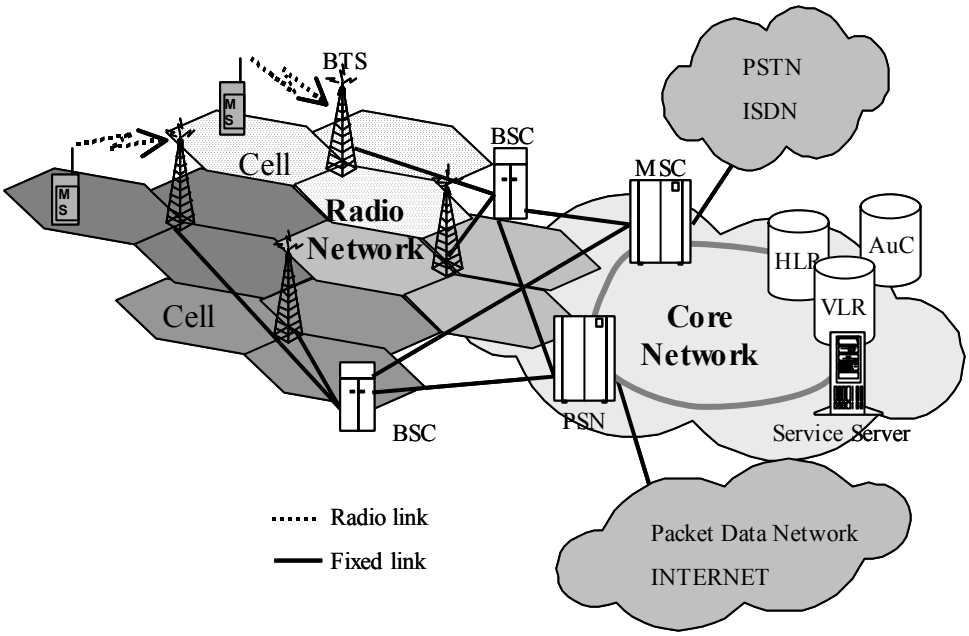


Figure 2-1: Cellular network architecture

The communication between the BTS and MS is divided into an uplink channel (from the MS to the BTS) and a downlink channel (in the opposite direction), as it is not possible to send and receive at the same time on the same frequency. The common way to realize uplink and downlink channels is Frequency Division Duplex (FDD) in which two frequencies are used; in case they are

[1] We loosely follow the GSM terminology.

realized on the same frequency Time Division Duplex (TDD) is used. In the remainder of the thesis we name all traffic towards the mobile station as downlink traffic and traffic from the mobile station as up link traffic. This description holds for traffic in the whole network and not only for the radio interface.

2.1.2 Performance considerations

The service performance experienced by users of a wireless system is strongly coupled with the transmission performance. Wireless networks have to cope with difficult conditions on the radio link, which affect the performance [XPMS02] [LKJK02]. Especially in conjunction with the TCP protocol, which is widely used in the Internet, strong interactions can occur that downgrade the system performance [RFC3155].

The system characteristics that affect transmission performance can be summarized as follows:

- High probability of packet loss (e.g., due to radio conditions and handovers)
- High and varying packet delay (jitter) (e.g., due to error recovery protocols, buffering, coding effort, etc.)
- Delay spikes (e.g., due to radio conditions and handover)
- Bandwidth (throughput) oscillation (e.g., due to radio condition changes and access contention on the radio link)

Of particular interest is how TCP can handle such situations, as TCP is the predominant transport protocol in the current Internet. It has been shown that TCP can cope with such conditions but is performing badly [XPMS02] [RFC3481] [LKJK02]. If packet loss occurs, TCP misinterprets this to be a congestion situation. As a consequence TCP reacts with drastic reduction of the sending rate. Problematic is that it takes long until TCP is back to full speed, despite the fact that the situation causing the packet loss might be long gone.

Furthermore, highly varying packet delays and handovers can cause spurious timeouts. That is, TCP does not calculate appropriated round trip time-out values (RTO) and triggers a TCP segment retransmission, even the segment arrives shortly after. This affects the TCP performance negatively. [LK00] proposes a solution for this based on time stamping the TCP segments.

Handovers can also cause abrupt condition changes. That is, the available bandwidth (throughput) is suddenly reduced or increased. The former can cause congestion, as TCP might still send with high rate; the latter causes underutilization, as TCP does not increase the sending rate quickly enough.

Bandwidth oscillation also reduces TCP performance for the same reason that TCP does not adapt fast to new situations. In [RFC3481] this is even pointed out to be the single most important factor for reduced throughput.

A number of approaches to tackle such problems have been proposed (see for summary [RFC3481]), e.g., splitting TCP connections, snooping TCP, proxy

solutions [MSH03], notification of packet loss causes, fine tuning TCP settings and options.

However, the user might still notice performance degradations, which, in the best case are only irritating, or in the worst case discouraging. In any case it can be expected to affect the user's behavior.

2.1.3 Evolution of mobile networks

Mobile communication systems have been changing in an evolutionary way every decade over the past 25 years. Based on fundamental differences in the technologies, the systems are grouped into different generations. Table 2-1 lists the current generations and their main characteristics.

Generation	Key Characteristics
1G	Analog systems Analog modulation, mostly FM Voice traffic FDMA/FDD multiple access
2G	Digital systems Digital modulation Voice traffic + very low data rate (~10 Kbit/s) TDMA/FDD and CDMA/FDD multiple access
2.5G	Digital systems Voice + low-data rates (~100 Kbit/s) Data IP (packet switched) - Internet access
3G	Digital Voice + high-data rate data (~ 1 Mbit/s) Multimedia transmission also IP Internet access

Table 2-1: Cellular network generations[2]

In the 1980s the first generation and in the 1990s the second generation cellular systems have been mainly used for voice services. First generation systems (1G) were analog, while second generation (2G) systems, which are still deployed worldwide, are digital systems. Current 2G digital systems are GSM, cdmaOne (IS-95), IS-136, and the Personal Digital Cellular (PDC) system. GSM was initially used mostly in European countries, but is used nowadays on all continents worldwide. CdmaOne and IS-136 are systems mainly deployed in North America and PDC is deployed in Japan.

But the rapid growth of subscribers in mobile networks worldwide is no longer only due to 2G systems. The 2G systems, which besides voice telephony offer narrow band circuit-switched data access, take currently an evolutionary way to high-bandwidth, packet-switched access networks. Research for third generation (3G) systems, which started already in the mid 1980s, has been triggered by the need for more efficient usage of bandwidth and also the vision to transfer the same service capabilities, available in the wireline Internet to the mobile world as well. As an important intermediate step to pave the way to 3G, some so-called 2.5th generation (2.5G) systems, for instance High Speed Circuit Switched Data - HSCSD, GPRS, IS-95b, Cellular Digital Packet Data (CDPD) and P-PDC, were integrated into existing 2G systems in the late 1990s and from 2000 on. The system evolution is depicted in Figure 2-2.

[2] Abbreviations are listed in the appendix.

The evolutionary systems developed for GSM are the packet-switched system GPRS and the HSCSD system. They are deployed in almost all GSM networks nowadays. Within cdmaOne, the evolution was from IS-95a to IS-95b, which provides much higher data rates. CDPD was a packet-switched version based on IS-136 systems introduced in North America, and in Japan the packet-switched version of P-PDC for PDC was introduced together with the very successful I-mode business model [ZAB99].

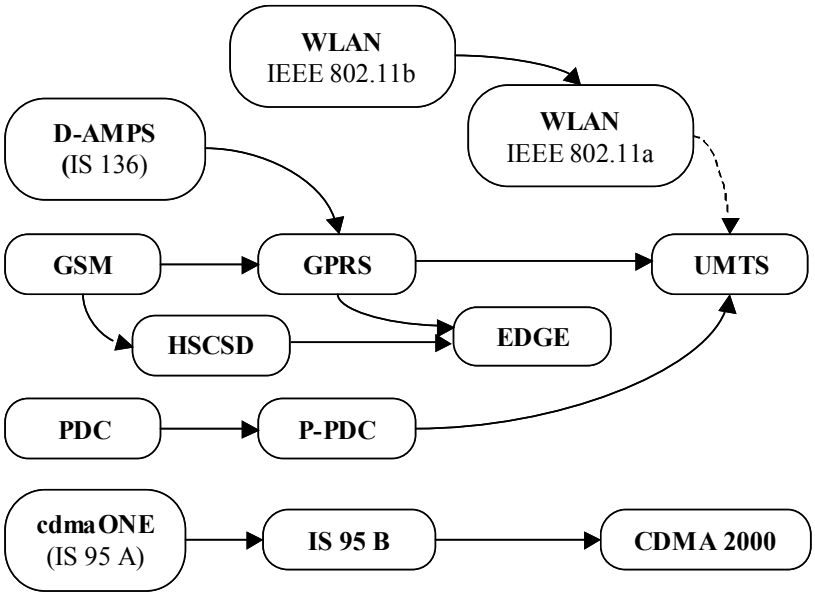


Figure 2-2: Cellular data network generations

Research on 3G systems was primarily coordinated in the International Telecommunication Union (ITU) project IMT-2000 and has lead to about 10 different proposals, of which UMTS was chosen as the evolutionary way for GSM/GPRS/EDGE systems, and CDMA 2000 was chosen as evolution for cdmaOne in North America.

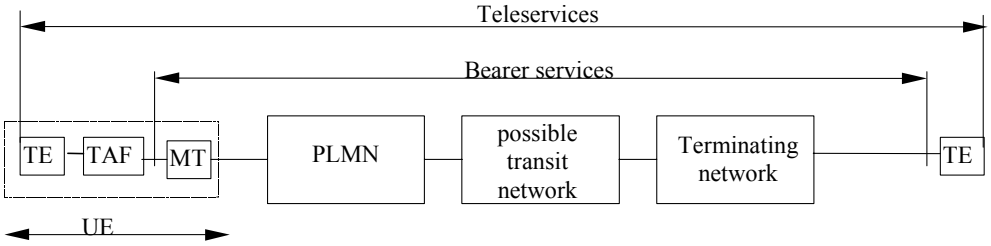
However, the demand for more bandwidth and a convergence of all networks is continuing, leading already to research on 4G and 5G systems. In parallel to the cellular system development, the WLAN system (IEEE 802.11), originally developed for office (LAN) usage, is integrated with cellular systems for 3G and 4G. Furthermore, wireless systems for very short ranges, like Bluetooth, have been developed and are being considered for integration into 3G and 4G systems.

Therefore the distinction in different generations is becoming more and more difficult, and the process of merging different access technologies even blurs the boundaries.

References for technical specification and background can be found in [Oli99] [VLLX02] [ZAB99] [Rap96]. The features of a number of actual systems are listed in Table A-1 in the Appendix.

2.1.4 Services in mobile networks

Services specified for mobile networks can be divided into bearer services and teleservices. Figure 2-3 shows the network and terminal parts over which teleservices and bearer services span [TS22.105]. They cover different protocol layers and have different termination points. Teleservices specify all parts of the service delivered to a user. Traditionally they are part of the standard of telecommunication networks. For instance, voice telephony is a teleservice that is fully specified for GSM. Bearer services, on the other hand, provide only a communication link between two endpoints. Teleservices build on bearer services to realize end-to-end communications. However, the network can also provide transparent access to the bearer services. For instance, GPRS provides a packet-switched, connectionless point-to-point bearer service for IP packets [GSM02.60].



UE: User Equipment
MT: Mobile Termination
TE: Terminal Equipment
TAF: Teminal Adaption Function
PLMN: Public Land Mobile Network

Figure 2-3: Bearer and teleservices – source [GSM2.60]

Services in mobile networks are anticipated to evolve within the three different areas: personal communication, wireless Internet and mobile multimedia access (cf. [UMTS03]).

Personal Communication – based on the telecommunication paradigm

Person to Person communication is the basis for the growth in telecommunication and will continue to be so for many years to come. Personal communication between parties will be made more expressive and enhanced with the introduction of the ability to send animated messages, chat and pictures. With higher data rate and service capabilities, live video conversations will be possible as well. This adds to the traditional teleservice incorporated in the standards.

Wireless Internet – based on the Internet paradigm

The evolution in this area is based on bearer services allowing IP-based packet-switched access to the Internet. As a first step *wireless access* to the Internet is realized. All applications that are currently used in the wireline Internet are accessible through cellular networks. This step is already realized by the introduction of 2.5G networks.

Professional users, for instance, are able to access corporate networks via Virtual Private Networks (VPN), while consumer users access the Internet for information retrieval and leisure. Additionally, new applications will be created, using on the one hand the Internet infrastructure but on the other hand being specifically tailored to the mobile access networks. WAP service is an example for this development. In cooperation with the operators it is furthermore possible to offer location-based services, which are unique to mobile networks. A location-based service provides information, especially tailored to the needs of a user, and based on the position of the user. Such a service could be 'A list of all near-by Automatic Teller Machines (ATM)'; or 'Background information to the historic building in front of me', etc.

The combination of mobile networks as access networks to wireline Internet and the development of mobile network specific services and content will lead to a network relation as depicted in Figure 2-4. We call this the *Wireless Internet*.

Mobile media – based on the media paradigm

This development builds on top of the development of teleservices and bearer services but highlights business opportunities that utilize the services. Mobile stations provide the means for services that are highly personalized, interactive, immediate, and always with the user. Media companies will utilize this as an additional unique distribution channel, creating new services.

Figure 2-5 depicts how each new generation with increased capabilities, bandwidth and used spectrum, leads to the introduction of new services [UMTS03].

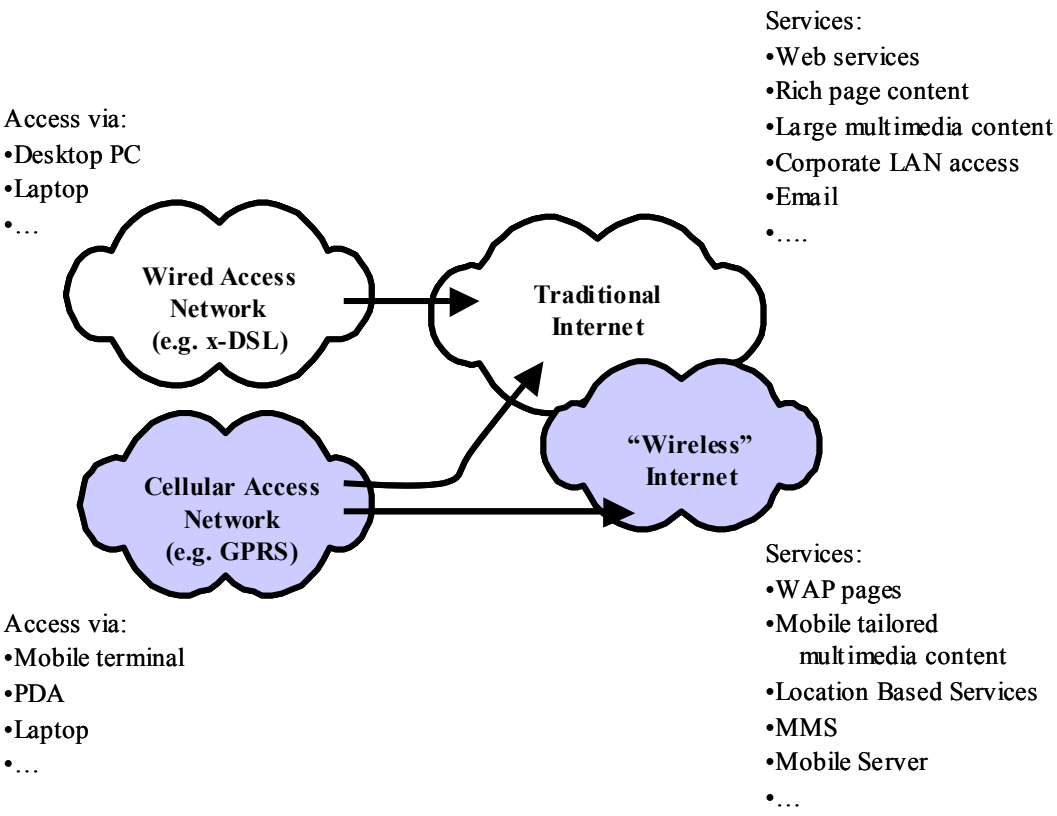


Figure 2-4: Wireless Internet

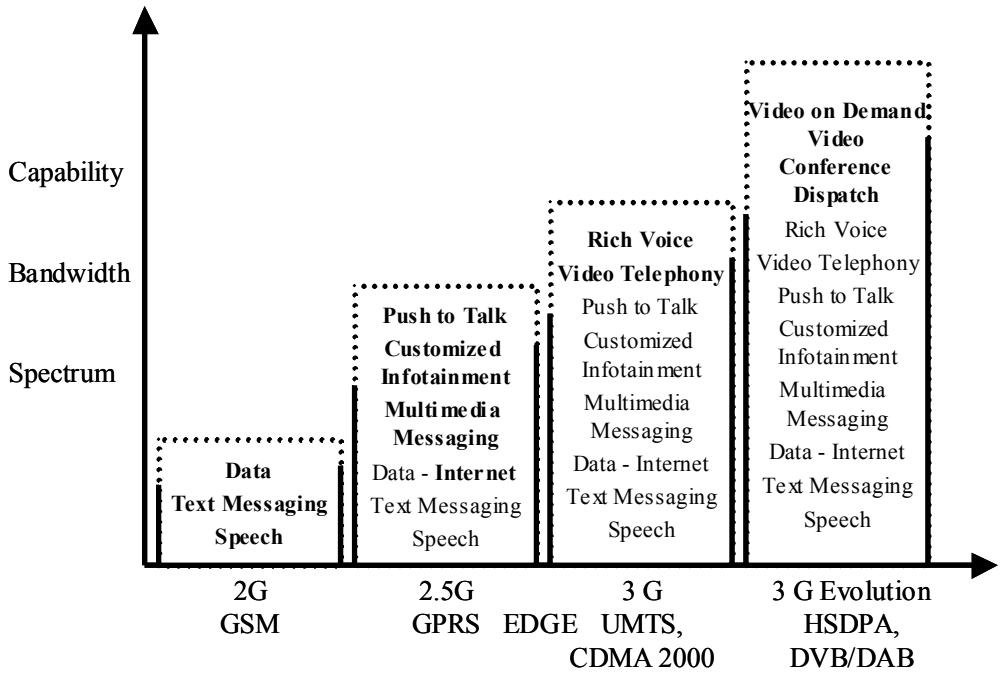


Figure 2-5: Cellular network applications development – source [UMTS03]

2.2 The Internet

Traffic modeling requires a thorough understanding of the investigated applications. Therefore, this section will survey the Internet service architecture. Section 2.2.1 presents the Internet paradigm on which the design of the Internet protocols and services is based. In section 2.2.2, the TCP/IP protocol stack is introduced. Basically all communication between applications is using this. The application service architecture is explained together with an overview of the main Internet applications in section 2.2.3.

2.2.1 The Internet paradigm

The Internet is formed by many heterogeneous networks and a very large number of connected computers worldwide (those computers are called hosts in the Internet terminology). All hosts in the Internet are reachable from any other host in the Internet. And each of them can easily access a multitude of services. This is made possible by using the common TCP/IP protocol suite, which allows a very flexible use of the Internet [Cla88] [CK74]. The basic principle of the Internet and its main protocol suite has not changed since its introduction in the 1960s, even though many new services that nobody dreamed of four decades ago have been introduced.

Three principles are cornerstones to the sustained evolution of the Internet. Firstly, the communication in the Internet is *packet switched*. No connection is established through the network and no state information needs to be stored in the network. Packets are individually routed from the source host to the destination host. This makes the network very flexible and scalable. The Internet Protocol (IP) is the unifying building block for this. Secondly, the design follows the *End-to-end connection paradigm*. That is, all intelligence goes in the end nodes, and the intermediate network(s) are primarily responsible for routing of the data packets between the end hosts [SRC84]. All functionality that is needed for a specific service (e.g., Email, Web browsing) is realized by protocols in the end hosts. This also includes functionality for reliable end-to-end communication. This is in sharp contrast to the traditional telecom approach, in which the end nodes have very little knowledge about the service and all intelligence for a service is in the network (e.g., PSTN). Thirdly, the Internet uses a *simple 4-layer protocol stack*. These are the sub-network layer, network layer, transport layer and application layer. Clear access point interfaces (API) are defined between the protocol layers.

2.2.2 TCP/IP protocol suite

The central protocol suite of the Internet comprises three unifying protocols. These are the IP, TCP and UDP protocol. In short, IP is responsible for addressing and routing packets, TCP is responsible for reliable end-to-end communication, and UDP is a lightweight connectionless transport protocol.

With the help of convergence layer protocols between subnet and network layer, IP can be run across basically any sub-network.

Figure 2-6 depicts some of the main protocols in the Internet around the central TCP/IP protocol suite.

The communication among peer hosts is horizontally performed between protocols of the same level by means of layer-*n* Protocol Data Units (PDU). For instance, the Hyper Text Transfer Protocol (HTTP) entity in one host communicates with the HTTP entity in another host by means of HTTP PDUs. The actual transmission of the layer-*n* PDUs takes place by lower layer protocols. For this, vertical communication between protocols of two layers in the same host is used. For instance, the HTTP PDU is handed to TCP for transmission. TCP segments the HTTP PDU into TCP segments before they are passed to IP. IP itself uses subnet layer protocols to transport the IP packets from one host to another host.

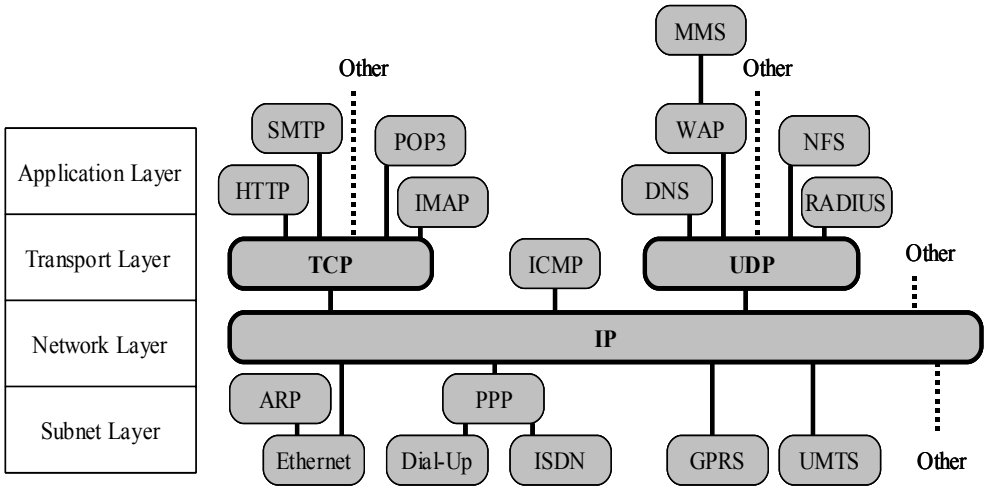


Figure 2-6: TCP/IP protocol stack

2.2.2.1 Network layer

The Internet Protocol (IP) [RFC791] is the central protocol of the Internet. IP's fundamental task is to route packets from the source host to the destination host.

0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	1 0	1 1	1 2	1 3	1 4	1 5	1 6	1 7	1 8	2 9	2 0	2 1	2 2	2 3	2 4	2 5	2 6	2 7	2 8	3 9	3 0	3 1
Version				IHL				Type of Service								Total Length															
Identification																Flags			Fragment Offset											
Time to Live								Protocol								Header Checksum															
Source Address																															
Destination Address																															
Options																													Padding		
Data																															

Figure 2-7: Internet datagram header [RFC791]

One IP packet consists of the IP header (Figure 2-7; the top row lists the byte position) and the data section. The length of the header without additional options is 20 bytes. The transport layer protocol, which is the protocol on top of the IP layer, is indicated by the protocol field. All allowed protocol numbers are defined in [RFC1700]. Table 2-2 depicts some of the most popular protocol numbers in the Internet. The source and destination address contain IP addresses, which are of 32 bit length and uniquely identify every host in the Internet. The data section contains the PDU data unit from the transport layer.

The maximum length, including the header, is 65535 bytes. See [RFC791] for an explanation of the other fields.

Protocol Number	Protocol name	Information
1	ICMP	The in-band IP signaling protocol for router status information
6	TCP	Reliable transport protocol
17	UDP	Unreliable transport protocol
50	ESP	Encapsulating Security Protocol, part of IPsec [RCFC 2401], which is used for secure communication e.g., VPN
51	AH	Authentication Header, part of IPsec.
94	IPIP	IP packets, encapsulated in other IP packets. This is used for tunneling of IP packets through a private IP address sub-network

Table 2-2: List of protocol numbers [RFC1700]

2.2.2.2 Transport layer

The transport layer handles end-to-end communication between application and services on peer hosts. Of the protocols listed in Table 2-2, only TCP and UDP are real transport layer protocols.

Transport Control Protocol

The Transport Control Protocol (TCP) [RFC793] provides means for reliable, connection oriented, end-to-end communication between end hosts. It deals on an end-to-end basis with connection setup, error recovery and flow control.

Figure 2-8 depicts the format of a TCP segment. A TCP segment is the concatenation of the TCP header of variable length plus subsequent payload. Application layer data is segmented by TCP into segments of length defined by the maximum segment size (MSS) parameter. In practice, MSS is set to 1460 bytes, which is a limit imposed by Ethernet sub-networks.[3] The source and destination port fields specify hooks to the applications on the source and destination host. The principle behind the ports will be explained in section 2.2.3

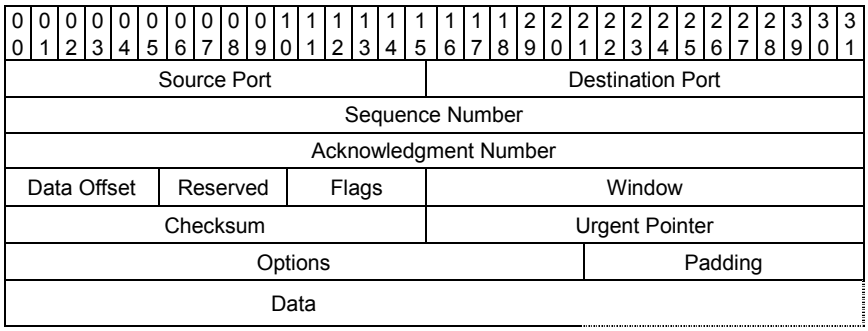


Figure 2-8: TCP header format [RFC793]

User Datagram Protocol

The User Datagram Protocol (UDP) [RFC768] is a connectionless transport protocol without error recovery, flow control and congestion control.

[3] The maximum transfer unit (MTU) for Ethernet is 1500 byte, hence the 1460 byte for the MSS.

A UDP packet consists of a short UDP header, plus the application packet from the next higher layer. The source and destination port specify a hook to the source and destination applications.

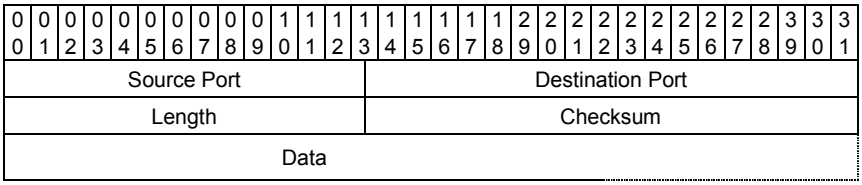


Figure 2-9: UDP header format [RFC768]

2.2.3 Internet services

Services in the Internet are realized according to the client/server approach. That is, the service is split up into two; one part executes on the client machine and one on the server machine. The client runs on the user's machine providing a front end to the service running on the server. The client interacts with the user and sends requests to the server. The server processes the requests locally and responds with reply messages to the client computer. The communication between the server and client is based on the TCP/IP protocol suite.

In Internet terminology, the service access point is called socket [Ste92]. A socket is defined by the IP address of the host and an appropriated port number. A transport layer connection between client and server is identified by a pair of sockets.[4] If a client wants to communicate with a service, it addresses the packets to the server's address and the specific service port. The application offering the service listens on that specific port. If the server application accepts the request, a connection is established between the client and the server.

In the Internet a set of *well-known ports* is used to identify specific services. All hosts that offer the same kind of service will listen on this specific port number. All well-known ports, identifying specific services, are listed in [RFC1700].[5] For example, all Web servers use port 80 for HTTP. Table 2-3 shows an excerpt from this list in [RFC1700]. The *well known ports* are in the port numbers range from 0-1023.

Beyond port number 1023 are the registered ports. They are usually not uniquely assigned to a specific service. A service can allocate any number in this range. For a client to be able to communicate with a service using a registered port it must know the used port number in advance. Table 2-3 shows some registered port numbers. These numbers are not defined in [RFC1700]. The numbers are defined in the respective application standard. But the listed numbers are only default values from the standard, often different ports are used. Especially online games and peer-to-peer file sharing programs use a whole range of ports for their communication [Joy00] [Fae02] [MC00] [KBB+03].

[4] Therefore a connection is defined by the quadruple IP address and UDP port for source and destination.

[5] These days the list is up-to-date maintained at http://www.iana.org/numbers.html [RFC3232].

We use the port number assignment to identify traffic belonging to certain applications in our analysis in later chapters.

	Port Number	Service Name	Information
Well known port numbers	**80**	HTTP	Hyper Text Transfer Protocol Transaction based protocol used for Web-services.
	25	SMTP	Simple Mail Transfer Protocol. Service used for uploading emails to the server.
	110	POP3	Service for downloading Email.
	143	IMAP4	Internet Mail Application Protocol. Advanced service for downloading emails.
	20	FTP_data	File Transfer Protocol. Used to fetch files from remote servers.
	21	FTP_control	File Transfer Protocol. Used to fetch files from remote servers. Port for control information.
	23	Telnet	Remote login service.
	53	DNS	Domain Name Service (used to look up IP address for www.domain-name.com-type names).
Registered port numbers	**9800-9803**	WAP	Wireless Application Protocol, mobile browser service.
	8080	HTTP-Proxy	Caching service for Web browsing. Typical deployed on egress routers of companies.
	1812,1813,…	RADIUS	Remote Authentication and Dial In User Service. A service used to authenticate users and assign IP numbers for dial-in type services.
	27010	HalfLife	Action game, can be played online over the Internet.
	27500, ... , 27960	Quake (III)	Action game, can be played online over the Internet.
	6346, 6347, …	Gnutella	Peer-to-peer file-sharing program. Used to share e.g., MP3 files over the Internet.
	4661,4662,4665, …	Edonkey	Peer-to-peer file-sharing program. Used to share e.g., MP3 files over the Internet.

Table 2-3: Port numbers for some services [RFC1700] [MC00] [KBB+03].

In the following sections, more details on frequently used applications are given as far as this is relevant for the analysis provided in later chapters.

2.2.3.1 Web browsing (HTTP)

The Hypertext Transfer Protocol (HTTP) is an application-level protocol for distributed, collaborative, hypermedia information systems [RFC2616]. It is the primary protocol used for *Web browsing*, as it is commonly referred to, in the Internet.

On the server side, content is encoded in markup text format (e.g., Hyper Text Markup language (HTML) and eXtended Markup Language (XML)).

The content is organized in Web pages and the Web pages are connected via hypertext links. In fact any kind of multimedia content can be linked with a Web page. All objects accessible via HTTP (e.g., HTML page, embedded pictures, JAVA programs, videos, etc.) are addressable by a Uniform Resource Locator (URL). A Web page and its embedded multimedia content can be distributed over any number of servers.

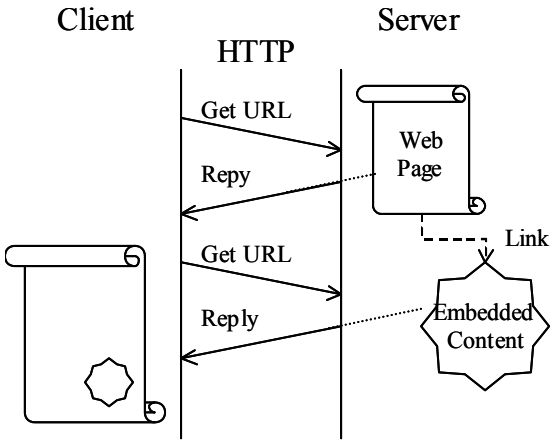


Figure 2-10: HTTP transaction model

The client (Web browser) sends a HTTP requests (Get) to the server, in order to fetch a certain Web page (see Figure 2-10). The main page is sent back with a HTTP response message (Reply).[6] If necessary, the client fetches subsequent embedded content with more HTTP request (Get) messages.

HTTP uses TCP as transport protocol and is by default addressed on port 80 on the server side. If an HTTP proxy is used for communication, port 8080 is the default port used.

In HTTP 1.0 the client uses one TCP connection for each 'Get/Reply' message pair. For instance, if the Web page consists of one main document and 3 embedded objects, 4 TCP connections are established in total. HTTP version 1.1 suggests using persistent connections if possible [RFC2616]. For persistent connections, one TCP connection is used for several 'Get/Reply' message pairs to the same server.

2.2.3.2 Email (POP3, IMAP, SMTP)

Sending and receiving emails is mainly done by the three protocols SMTP, Post Office Protocol (POP3) and Internet Message Access Protocol (IMAP). The first one is for sending and the latter two are both for receiving of emails.

SMTP [RFC2821] is the protocol used to send email messages and relay such between hosts. SMTP works in a client/server way.

An email is constructed on the client side, and transferred by the mail transfer agent from the client to the server (see Figure 2-11). The communication is started on the client side. The mail transfer agent contacts the server on the well-known port number 25 and uses SMTP for the communication between the client and the server. Underneath, TCP is used as transport protocol. The user may first construct a number of emails before they are sent out by the mail transfer agent. In this case the mail transfer agent opens one TCP connection to send all messages at one instance to the server.

[6] In correspondence with the HTTP tag 'GET' for request messages we call response messages REPLY, though this does not exist as tag word in the HTTP standard. However this makes it easier to relate to WAP in later chapters.

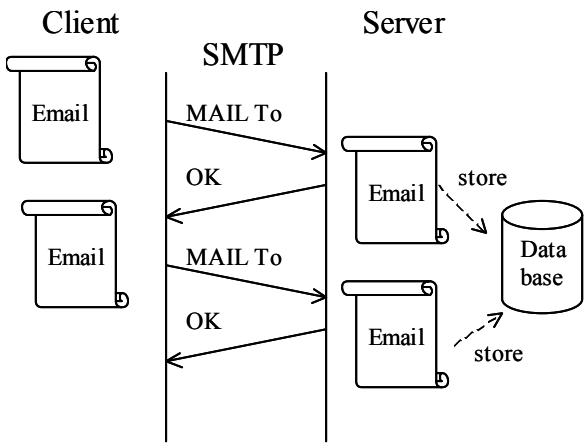


Figure 2-11: SMTP transaction model

SMTP is also used between servers for relaying the emails, in case the first server to which the client had sent the messages is not the destination server. In this case the former server is acting as a client and forwards any email message that needs to be relayed to the destination server.

POP3 [RFC1939] is intended to permit a workstation to access a server host for downloading email messages. When a client host wishes to make use of the service, it establishes a TCP connection with the server host on TCP port 110.

When the client connects to the server it requests a list of all stored email messages. Following this, it downloads all messages using one TCP connection. POP3 is designed so that all email messages are downloaded from the server at once and afterwards are deleted on the server. They are only kept thereafter on the client. Typically the TCP connection is closed right after downloading all emails. If required, the client connects again to the server to check for newly arrived emails.

IMAP [RFC3501] is, similar to POP3, a protocol to download email messages from the mail server, but it has more sophisticated functions. It supports emails to be kept on the server or to partly download the content of mail messages. In particular, it allows the manipulation of electronic mail messages (and the mailboxes) on the server in a way that is functionally equivalent to local folders. In contrast to POP3, it also provides the capability for an offline client to resynchronize with the server. A client wishing to communicate with IMAP uses TCP and connects to the server on port 143. The connection can be maintained for a longer duration in which many download, update and folder manipulations messages are exchanged.

Another option to send and receive emails is Web-mail, as it is used commonly nowadays. Web-mail is a Web page based interface to mail boxes on mail servers. Instead of using a mail transfer agent, the user uses a Web browser. All emails are displayed as Web pages.

The Web server acts as a mail transfer client and requests the emails from a mailbox. The Web server converts them to a Web page and responds this back to the client. Between the user and the Web server HTTP is used, while

between the Web server and the email server SMTP, POP3 and IMAP are used. This difference is in particular important if the traffic application mixture is distinguished, as it is done in our analysis, based on the used protocol between the client and the server. In case of webmail access all traffic looks like HTTP traffic and cannot be assigned to Email services.

2.2.3.3 File transfer (FTP)

The File Transfer Protocol (FTP) [RFC959] is used to access remotely stored files on a server. It supports file up and download as well as some file manipulations. An FTP server is accessed from a client using TCP as transport protocol on port 21. Often TCP port 25 is additionally used for data transmission. FTP allows access to a file system on a remote host. It provides many commands common to file operating systems like directory listing, renaming, downloading or deleting of files. Several files can be downloaded (from the server) or uploaded (to the server) within one FTP session, which is then based on one TCP connection.

If only file download is required, nowadays often the HTTP protocol is used to access file systems. This provides a convenient replacement for FTP, but it is limited in its functionality. In an even more restricted way HTTP also supports uploading of files. As it is the case with web-mail, this type of access blurs the boundaries between file transfer (FTP) and web browsing (HTTP), and consequently makes categorization merely according to the protocols difficult.

2.2.3.4 Access authentication (RADIUS)

The Remote Authentication Dial In User Service (RADIUS) client-server tool is used in computer networks to provide remote users authentication and accounting. It is actually defined by two protocols; authorization is defined in [RFC2865] and accounting services is defined in [RFC2866].

RADIUS applies UDP as transport protocol. The RADIUS standard recommends using port 1812 to access a RADIUS service.

Access request packets are sent to a RADIUS server, and convey information used to determine whether a client is allowed to access the network. If the client is allowed to access the network, the server replies with an accept message, which contains additional access information. Otherwise it sends a reject message. The access information also contains the IP address assigned to the client.

2.2.3.5 Mobile microbrowser applications

Especially for new handheld devices like Personal Digital Assistants (PDA), mobile terminals, etc. specifically tailored Web browsing applications have been developed, partly coming with their own protocol stack. The new protocols and specific content encoding shall help to better facilitate the highly resource-restricted wireless networks. The new wireless protocol suites often deploy a split-architecture, with a wireless network specific part in the cellular network and the established TCP/IP protocol stack towards the server side. Three examples for such specific tailored applications are I-Mode, WAP and MMS over WAP.

For example WAP uses UDP as transport protocol and proposes port numbers 9800-9803 as default port.

The communication in WAP is also transaction based (Figure 2-12). The client sends a Wireless Session Protocol (WSP) message 'Get' to the WAP Gateway, requesting a specific WAP page. The WAP Gateway translates this request into an HTTP request 'Get' and uses HTTP to request the WAP page from the Web server on behalf of the WAP client. When the WAP Gateway receives the WAP page via HTTP, it uses a WSP reply message to send the content back to the WAP client.

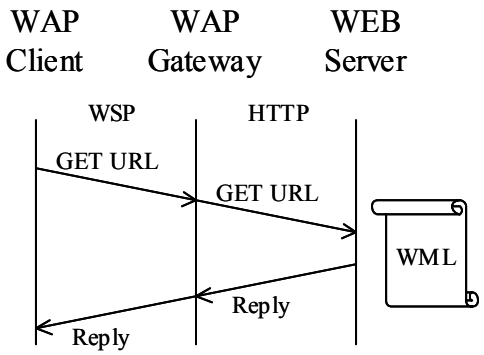


Figure 2-12: WAP 1.2 transaction model

MMS utilizes WAP as transport protocol. Therefore in order to identify MMS messages the content of WAP must be inspected. It is not enough to consider the protocol and port numbers.

Because we will encounter WAP and MMS over WAP in our analysis we will introduce them as part of the specific GPRS applications in section 3.8.

2.3 Teletraffic theory

This section discusses four aspects related to teletraffic theory. In the next section 2.3.1 we first explain how QoS, network capacity and traffic demand are linked. We will further elaborate on how teletraffic engineering can be done using simulation, analysis or experimental approaches. Subsequently, in section 2.3.2, we briefly line out how to deal with the problematic area of Internet traffic modeling. Section 2.3.3 focuses in particular on one aspect of teletraffic engineering, namely the modeling of the traffic demand. We discuss important aspects of traffic models and provide a brief overview of self-similar traffic models and heavy-tailed distributions. Modeling traffic demand accurately requires knowledge of the real traffic. This must be obtained using measurements, which we introduce in section 2.3.4. We point out important aspects to be considered in the area of traffic measurements. We conclude with an overview of measurement studies available in the Internet in section 2.3.5.

2.3.1 Fundamentals of teletraffic theory

Teletraffic theory originally was defined as mathematics for design, control and management of Public Switched Telephone Networks (PSTN), but was later extended to include data networks and incorporate Internet engineering

[WP98]. That is, teletraffic engineering is the application of mathematical modeling linking network capacity, traffic demand and realized performance [Rob01].

Teletraffic theory has been very successfully deployed in voice telecommunication for decades with the Erlang formula [Kle75]. But so far it has only played a minor role in the designing of the Internet [Rob01]. This is due to the more complex nature of the network and the traffic in data networks.

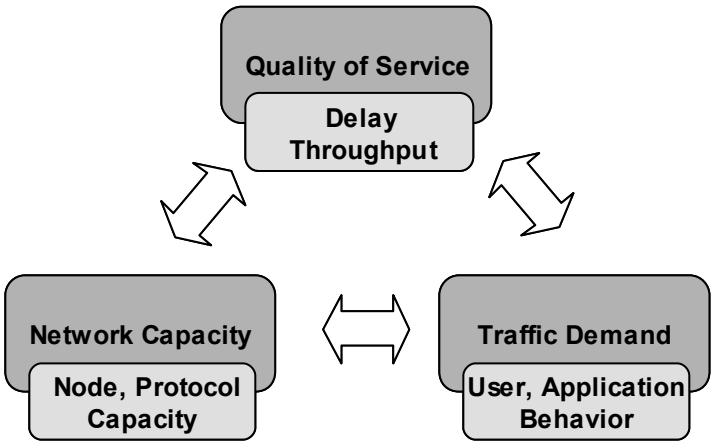


Figure 2-13: Teletraffic – QoS, traffic and capacity relationship

Figure 2-13 visualizes an important relationship in teletraffic theory. Any characteristic (QoS, Network Capacity, Traffic Demand) can be derived by knowing the other two characteristics. This relationship can be applied on any part of the network. It is applicable for a complex large network consisting of many nodes as well as individual nodes or functional units in the network. A functional unit would be for instance the packet scheduling algorithm or an outgoing link. The exact mathematical relationship depends on the part of the system it is applied on.

This leads to a multitude of possible descriptions of QoS, network capacity and traffic demand.

Examples for QoS metrics are: call blocking, packet delay, byte throughput, packet processing capability, packet error rate, etc. The network capacity metric comprises packet-processing capability, link bandwidth, scheduling algorithm, protocol specification, etc. Traffic demand is any kind of traffic imposed on to the network element to be investigated. Traffic demand can be expressed by the user and application behavior in a simple way in terms of, e.g., average data volume and in more detail by stochastic processes. But traffic demand is not only restricted to user-imposed loads on the network. It may also consider signaling load, e.g., from network operation and management or call-handling protocols. Models for traffic demand will be further elaborated in section 2.3.3.

The task of planning a network comprises many aspects. Typically the quality of service (QoS) and the traffic demand is a given input parameter for the traffic engineering task. The objective of traffic engineering is to specify an appropriate capacity in order to provide the desired QoS. Products of the

investigation are for instance the assigned link capacities and node capacities. This can be further broken down into individual aspects of the link and the node. For instance, the link might be split up into different logical channels with different assigned handling priorities. One important aspect to be considered in dimensioning the node capacity is the used buffer space for aggregated and individual flows. The buffer space has direct impact on packet delay and loss probability.

Performance analysis also uses the techniques of teletraffic theory. That is, performance analysis is also concerned with the relationship between QoS, capacity and traffic demand. The task of performance analysis is to evaluate the achievable performance of a system. It is typically used in the design process of a system or a network protocol. In case of performance analysis, the input parameters are the network capacity and the traffic demand. The result of performance analysis is the QoS that results from such a scenario.

Teletraffic engineering can be done in several ways:

- analytically
- simulation based
- experiment (measurement) based

An overview of the characteristics of the three approaches is given in Table 2-4. Traditionally, teletraffic engineering has been performed analytically. Analytical models describe the relationship in Figure 2-13 with a closed formula or a formula yielding a numerical approximation. Analytical models have a number of advantages. They are tractable and often provide deep insight into the relationship between input parameters and the results. Also, they quickly yield a result. But analytical models also require a high level of abstraction of the involved models. A complex system often needs to be reduced to a basic queueing system. The Erlang formula is one very famous example of an analytical queueing system deployed in traffic engineering. The application of queueing models is the most used analytical method in teletraffic theory. A large theoretical basis for dealing with queueing models exists, cf. [Kle75] [Kle76].

Monte Carlo simulation based investigations only became realistic over the last decade, since computers became more powerful. In simulation-based investigations, the network and protocols are described as function blocks. The traffic demand, as input function to the simulation model, is described by a stochastic model, but can be much more detailed than in the case of an analytical model. A so-called simulation engine executes the function blocks and generates the traffic according to the model. All events in the system model are simulated by the simulation engine. The result is an approximation of the real world behavior of the system. In some cases protocol layers are modeled in great detail. In Monte Carlo simulations random variables are used to describe the input process and the variable parameters of the system (e.g., the channel condition on the radio link).

Simulations are especially used for the performance analysis of complex Internet systems and protocols. Many simulations, covering aspects of Internet

protocols, are based on the freely available NS-2 simulator [NS2]. If novel systems are investigated, new simulation models need to be built. A number of proprietary simulation models handle for instance wireless systems based on, e.g., BONeS [KMM00], OPNET [OPNET] or GPRSSim [SM00].

The advantage of the simulation-based approach is that the model can cover many details in the system. It is possible to approximate the real system arbitrarily close. The danger with this possibility is that too many details hide important relationships in the investigated system. Besides the higher time effort needed to develop simulation models, it is also problematic to verify the model syntactically and semantically.

As the analytical and simulation based analysis methods are based on a number of assumptions and abstractions in the model; it is important to validate the results with real implemented systems. For this reason, another important method in traffic engineering and performance analysis is experimental investigation and measurement in existing systems. This approach is usually very time consuming, complex and costly, but also the most accurate. In the industry this approach is employed in the final stage of product development to verify the system performance and stability. A thorough system investigation should use all three approaches in a complementary way.

Criterion	Analytical Modeling	Simulation	Experimental
Stage	Any	Any	Post-prototype
Time required	Small	Medium	Varies
Tools	Analysts	Computer languages	Instrumentation
Accuracy	Low	Moderate	Varies
Trade-off evaluation	Easy	Moderate	Difficult
Cost	Small	Medium	High
Salability	Low	Medium	High

Table 2-4: Teletraffic analysis methods [Jai91]

Building a model for traffic engineering consists of two main parts. One part of the model is the system description and the other part is the traffic input description.

Though it is often very time consuming to model and verify the network system appropriately, the biggest challenge in Internet traffic engineering is to model the traffic demand. The choice of a traffic model is accompanied with high uncertainties, and choosing a wrong traffic model might lead to useless results.

2.3.2 Traffic engineering problems of the Internet

In [FP01] and [WP98] many problems around modeling and simulating aspects of the Internet are outlined. We list a few of them in the following.

The major problem is the changing nature of the Internet. In analyzing the Internet, one faces the problem that one uses input parameters from the current Internet (e.g., traffic demand or system description), and tries to predict the Internet of the future. Extrapolation from the current state of knowledge can be very difficult due to the changing nature of the traffic.

[FP01] describes two principle approaches to cope with the uncertainties. The first approach is focusing on invariants in the system description. An invariant is a facet of behavior that has been empirically shown to hold in a very wide range of environments [FP01]. That is, invariants are very robust properties that basically do not change over time and can be assumed to be the same for different investigated scenarios. In section 2.3.5, we provide an overview of some possible invariants encountered in the Internet. Finding empirical invariants stresses again the importance of using real measurements, in order to base assumptions on solid ground.

The second approach in dealing with analyzing Internet scale network scenarios is to explore a meaningful parameter space. That is, based on a-priori assumptions and measurements, the parameters should be chosen so that important key scenarios are covered. Just focusing on one value and using the result for any conclusion can be dangerously misleading. Using a whole set of input parameters and careful examination of why one observes the changes one does observe, may lead to insights into fundamental couplings between different parameters and the network's behavior [FP01].

2.3.3 Traffic models

Traffic models can be designed to be used in analytical or in simulation-based investigations. An analytical model needs to be mathematically tractable, while a model for a simulator may be more complex without a closed mathematical description.

Furthermore, traffic models can be realized at packet level, at flow level or as source model incorporating several protocol levels.

A packet level model describes the arrival process of individual packets. Typically this approach does not distinguish between users and applications but describes the aggregated traffic. Stochastic arrival processes are used to describe the packet arrival process.

Flow models do not consider individual packets but describe the traffic in terms of flow characteristics. The arrival process of the flows together with flow length and data volume within the flow is described by stochastic processes. For instance [RV00] and [KT04] provide an overview of flow models for TCP flows. Under the assumption that TCP flows tend to a steady state, they can be approximated by their average bandwidth requirement.

More complex than packet and flow models are source traffic models. Source traffic models are more detailed models describing the traffic in terms of particular *application objects*. For example, a source traffic model for HTTP describes aspects like Web session, Web page, Web objects and the statistical relation among them. The statistical relation among them is for instance the average number of Web objects in a Web page.

In some cases it is necessary to be highly accurate with the traffic pattern. In this case detailed protocol descriptions are used. That is, the traffic behavior is emulated by implementing crucial parts of the actual protocol. In particular, this approach allows considering feedback loops in the traffic generation process.

For example, the traffic of applications using TCP is self-regulated. TCP regulates the amount of traffic induced into the network, depending on the load in the network. If only the arrival process of the traffic is considered without the feedback by the TCP protocol (open loop model), the results are quite misleading. [KA98] and [AK99] show how important it is to consider closed loop TCP models. [ENNS00] provides an open loop TCP model, but also points out the usefulness of a closed-loop model.

An alternative to stochastic-based models are trace-based models. A trace-based traffic model is actually not really a model, but is the measurement log of real traffic from a commercial network. The measurement log, for instance, consists of the arrival times of packets, flows or sessions. The measurement log can be played back many times in a simulator or real system implementation. While this appears to be a highly accurate traffic representation, this approach should be used with caution. The reason is that, in general, the traffic demand is not independent of the system characteristic [AK99]. In particular, trace-based models lack feedback from the system. If the newly investigated system has very similar characteristics to the system in which the traces were taken, the trace-based approach might be applicable. But, especially if extreme load situations are considered, for instance close to the system's limit, trace-based models are not a good choice.

Building a traffic model should not only reflect some real traffic scenario, but also needs to be applicable. Therefore a traffic model should meet the following criteria:

It should be:

- Accurate
- Complete
- Tractable
- Parsimonious
- Robust
- Physically Meaningful

Accurate

That a model needs to be accurate is an obvious requirement. If all parts of the model are correct, the predictions from the traffic theoretical investigation are all correct. However, the problem lies in how to prove the accuracy of a model. Especially in complex and highly abstracted models this is not easy. In the case of traffic models, measurements are an apparent way to verify a model.

Complete

A traffic model is complete if it covers all important aspects of the traffic so that the result of the traffic theoretical investigation is deterministic. That is, the result of two investigations with the same parameter models must lead to the same conclusion. For instance a traffic model consisting just of the mean arrival rate is not complete if the type of the arrival process is not specified.

Tractable

Tractability is the basic requirement for making a traffic model useful. Mathematical methods must exist to use the model in analytical or simulative investigations. This is the main reason why Markovian models still play a dominant role in traffic engineering.

Parsimonious

If a model is parsimonious it is described by a very limited number of parameters, which makes the model much easier to use. This is highly desirable. In practical traffic engineering it is important to know how to choose parameters of a model. A parsimonious model with few meaningful parameters makes it easier to choose the parameters. Additionally, parsimonious models provide better insight into the correlation between changing an input parameter and the effect on the outcome.

Robust

A robust model is a model that is applicable in a wide range of cases. Small changes in the setup should not require a redesign of the model.

Physically meaningful

The requirement physical meaningful is related to the parsimonious requirement. The parameters of the model should have an actual meaning in the context of the model. For example, describing the arrival of FTP user sessions with a Poisson process has such a property. It is easily comprehensible, as the only model parameter describes the average number of sessions per time unit. This parameter can be adjusted by the engineer applying the model, according to real world observations. A negative example would be fitting a complex model with, e.g., 50 parameters for the same FTP session inter-arrival time distribution. In the latter case it would be difficult to map real world values to the 50 parameters.

2.3.3.1 Analytical traffic models

A wide range of traffic models has been developed [FM94] [JMW96] [HKS9]. Most of them can be rooted back to voice and circuited switched networks. These are mostly renewal and Markovian type processes. Though they are often still used for data networks, in particular the high burstiness and long-range dependence of data networks (will be introduced in section 2.3.3.2) is often inadequately modeled [PF94] [WP98]. But this new property is important to consider in traffic engineering, and therefore an increasing number of models also deals with the complexity of the new data traffic.

Next, we will provide a brief overview of stochastic processes used in traffic modeling including phase type and self-similar processes, as well as heavy-tailed distributions, as those concepts will be used in later sections.

Definition: cumulative distribution function

For any random variable X, the cumulative distribution function (CDF) is defined for all x by $F(x) = P(X \leq x)$.

In case X is discrete this implies

$$F(X) = \sum_{\{y|y \leq x\}} f(y) = \sum_{\{y|y \leq x\}} P(X = y)\,. \tag{2.3.1}$$

For empirical data, the empirical CDF is constructed as following.

Given the data x_i, $i = 1,\ldots,n$.

For the ordered set $x_{(1)} < x_{(2)} < \cdots < x_{(n-1)} < x_{(n)}$

$$F_e(x_{(i)}) = P(X \leq x_{(i)}) = \frac{i}{n}. \tag{2.3.2}$$

The (empirical) complementary cumulative density function (CCDF) is defined by

$$\overline{F}(X) = 1 - F(X)\,. \tag{2.3.3}$$

Definition: stochastic process

A stochastic process is a collection of time-indexed random variables

$$X = (X_t, t \in T), \quad X \in S.$$

The stochastic process is a discrete-time stochastic process if the index set T is countable. The stochastic process is a continuous-time stochastic process if the index set T is an interval on the real numbers. The state space is the collection of all possible values that the random variables can take on.

A realization of the stochastic process X with $X = (x_{t1}, x_{t2}, ..., x_{tn})$ is called a sample path or time series.

Definition: Phase type renewal process

Phase type process belongs to the class of renewal process' and describes the time $X(k)$ between two events. It is characterized by an underlying *finite* and *absorbing* continuous time Markov chain. The Markov chain consists of $(n\text{-}1)$ transient states plus one absorbing state. The phase type renewal process is defined by the times $X(k)$ the Markov chain resides in the transient states until it reaches the absorbing state. The underlying Markov chain is started over again to produce each $X(k)$. Therefore all $X(k)$ are independent, identically distributed. The corresponding distribution of the times $X(k)$ is called phase type distribution and is fully defined by the underlying Markov chain:

Let $\{Y(t), t \geq 0\}$ be a finite Markov chain with state space $\{1,2,\ldots,n\text{-}1,n\}$ and the generator matrix $\mathbf{Q}$. The states $\{1,..,n\text{-}1\}$ are all transient and n is the only

absorbing state of the process.[7] Further, let $q=(q_1,\ldots,q_{n-1},q_n)$ be the initial state vector, describing the probability q_i to start in state i; q_n is always zero. Let **R** be a $((n\text{-}1)\text{x}(n\text{-}1))$-dimensional matrix, with $\mathbf{R}_{ii}<0$, for $1\le i\le(n\text{-}1)$, and $R_{(n-1)i}\ge 0$, for $i\ne(n\text{-}1)$. And let $\mathbf{r}=(r_1,\ldots r_{n-1})=-\mathbf{R}*\mathbf{v1}$ be the exit-rate vector, describing the conditional intensity r_i of absorption in the absorbing state n from state I; where $\mathbf{v1}=(1,\ldots,1)'$ a column-vector of $(n\text{-}1)$ ones. The generator matrix is written

$$\mathbf{Q}=\begin{bmatrix}\mathbf{R} & \mathbf{r}\\ 0,\ldots,0 & 0\end{bmatrix}, \tag{2.3.4}$$

and **R** is called the phase-type generator.

Let $X=\inf\{t`>0:Y(t`)=n\}$ be the time until the absorbing state is reached, than X is phase type distributed with $(n\text{-}1)$-phases.

The class of phase type distributions is especially useful, as they can be used to approximate arbitrarily close all probability distributions with rational Laplace transformation. Furthermore, traffic modeling problems which have an explicit solution assuming exponential distributions are also algorithmically tractable when one replaces the exponential distribution with a phase-type distribution. Therefore many intractable queueing problems can be approximated by replacing general distributions by phase type distributions.

Special cases of phase-type distributions are hypo-exponential distributions and hyper-exponential distributions. A hypo-exponential distributed variable can be interpreted as the sum of $(n\text{-}1)$ exponential distributed variables. Another way of describing this is a sequence of $(n\text{-}1)$ exponential distributions (see Figure 2-14). A hyper-exponential distributed variable can be interpreted as a variable in which the result comes from the i-th exponential distribution with probability c_i. This can be seen as $(n\text{-}1)$ parallel exponential distributions (see Figure 2-15).

For the hyper-exponential distribution the phase-type definition yields $q=(c_1,c_2,\ldots,c_{n-1},0)$ and

$$\mathbf{R}=\begin{bmatrix}-\lambda_1 & 0 & \ldots & 0\\ 0 & -\lambda_2 & \ldots & 0\\ \ldots & \ldots & \ldots & \ldots\\ 0 & 0 & \ldots & -\lambda_{n-1}\end{bmatrix}. \tag{2.3.5}$$

The hyper-exponential distribution with $(n\text{-}1)$ stages is:

$$F(x)=1-\sum_{i=1}^{n-1}c_i e^{-\lambda_i x}\ , \tag{2.3.6}$$

[7] A transient state has an input and output transition associated. An absorbing state has only an input transition, but no output transition.

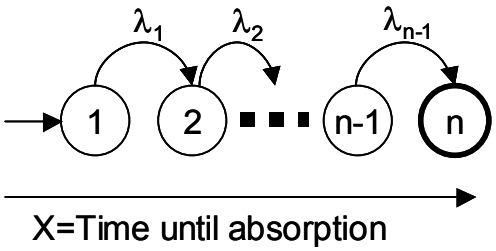


Figure 2-14: Hypo-exponential distribution

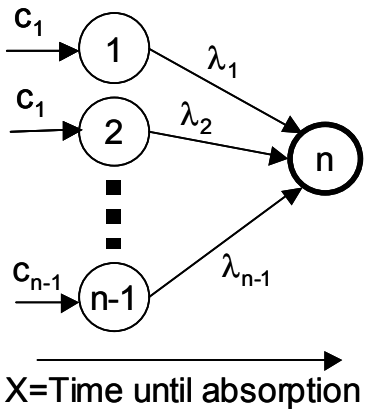


Figure 2-15: Hyper-exponential distribution

2.3.3.2 Self-similar traffic models

Internet traffic is very bursty. General speaking, plotting the time series of the data volume per unit time or the number of arriving data packets per unit time, of Local Area Network (LAN) and Wide Area Network (WAN) traffic, does not look very smooth. Furthermore, where as aggregating a large number of Poisson sources results in a smoothed-out appearance of the packet arrival process, this is not the case for Internet traffic; instead large peaks exist. While it is difficult to mathematically grasp the term burstiness (possible descriptors are index of dispersion for counts (IDC) [LWTW94], peakedness [Eck85], peak/mean ratio [EW94]), the behavior of Internet traffic can be very well described by the concept of self-similarity. [FL93] [LWTW94] [PF94], and others have shown that LAN and WAN traffic is self-similar.

Self-similarity, in a strict sense, means that the statistical properties (e.g., all moments) of a stochastic process do not change for all aggregation levels of the stochastic process. That is, the stochastic process 'looks the same' if one zooms in time 'in and out' in the process. The degree of self-similarity is expressed by the Hurst value H; large values indicate stronger self-similarity. If $H \in (0.5,1)$ the process is also long-range dependent (LRD). A LRD process shows strong correlation over a long time period. Plotting the autocorrelation function of LRD processes shows this non-vanishing correlation. On the contrary, the autocorrelation function of Poisson traffic approaches zero much more quickly.

Figure 2-16 illustrates the difference between Markovian-type traffic and self-similar traffic processes. In the top row of Figure 2-16, the aggregation of self-similar traffic is shown. As it is obvious, the traffic has visually the same burstiness when aggregated over three time scales. For comparison, in the bottom row Markovian traffic is depicted. We have chosen a Poisson process with the same initial average packet arrival rate. However the aggregated process smoothes out very quickly.

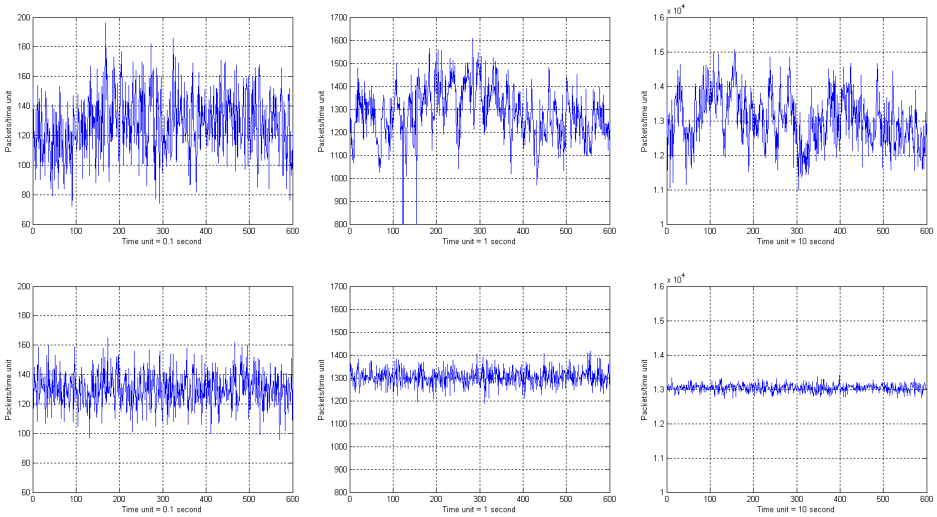


Figure 2-16: Traffic aggregation – self-similar and Poisson process

Before defining self-similar and long-range dependent traffic in more detail, we shall briefly discuss heavy-tailed distributions, as this is an important class of distributions often encountered in Internet measurements. In particular, self-similar stochastic processes are strongly interwoven with heavy-tailed distributions, as explained later.

Definition: heavy-tailed

A distribution is heavy-tailed[8] if

$$P[X > x] > ae^{-\alpha x},\ \ a > 0,\ \alpha > 0 \text{ for } x \to \infty. \tag{2.3.7}$$

Power Tail[9] distributions form an important subclass of heavy-tailed functions; defined as:

$$P[X > x] > cx^{-\alpha},\ \ \alpha > 0, c > 0 \ \text{ for } x \to \infty. \tag{2.3.8}$$

An important property of heavy-tailed distributions is that for all $k \geq \alpha$ the k-th moment is infinite. That means in particular that if $\alpha \leq 2$ the heavy-tailed distribution has infinite variance and if $\alpha \leq 1$ it has infinite mean.

Pareto distributions, which are defined as:[10]

$$P[X > x] = \frac{c}{x^{\alpha}},\ \ \alpha > 0, \tag{2.3.9}$$

are a simple but important example of power tail distributions.

---

[8] A synonym for heavy-tailed is also long-tailed or fat-tailed. Especially in the literature about Internet measurements, heavy-tailed is often used as a broader synonym when only power-tailed is meant; however, power-tailed distributions form only a subgroup of heavy-tailed distributions. Furthermore, often power-tailed distributions are only defined for $0 < \alpha \leq 2$, when used in the literature about Internet measurements.

[9] This stems from an equivalent mathematical notion. A function follows a power-law if $f(x) \sim x^{-c}$; $c > 0$. Another term for power-law is hyperbolic.

[10] This is Pareto type 1 – the simplest Pareto function. See Appendix B.

Two other important distributions, the lognormal and Weibull distribution[11] are often also referred to belong to the heavy-tail distribution class. There is, however, some debate in the literature whether they belong to the heavy-tailed class. [FP01] and [Dow01] clearly separate lognormal and Weibull from heavy-tailed distributions, where as [GF01] includes them. According to [FGMS01] the Weibull distributions belong to the heavy-tailed class if the shape parameter is less than 1. However, the lognormal distribution and the Weibull distribution decay faster than the Pareto distribution. And the lognormal distribution decays more quickly than the Weibull distribution, but both not as fast as light-tailed distributions (e.g., exponential, normal, or gamma distribution). Furthermore the Laplace transform does not exist in a closed-form for Pareto and for lognormal and Weibull, which makes this class in any case particularly difficult to use in analytical queueing analysis. Therefore we will consider the lognormal and Weibull distribution to belong to the heavy-tailed distribution class and call them 'weak' heavy tailed.

Power-tailed distributions can be identified with the help of a complementary cumulative distribution function (CCDF)-plot plotted on log-log scaled axis, in short *log-log-CCDF plot*.

Plotting a heavy-tailed distribution in a log-log CCDF yields a linear slope in the tail of the distribution (see for instance Figure 2-17). α can be estimated by various methods: for instance by the linear slope in the log-log CCDF plot, or by the Hill estimator [DdHR00]. Another way is described by [CT99] with the scaling method based on aggregated samples. The method of maximum likelihood estimation (MLE) can also be used if a particular heavy-tailed distribution is assumed. We introduce the 'scaling method' later, along with the MLE method when we need them in chapter 6.

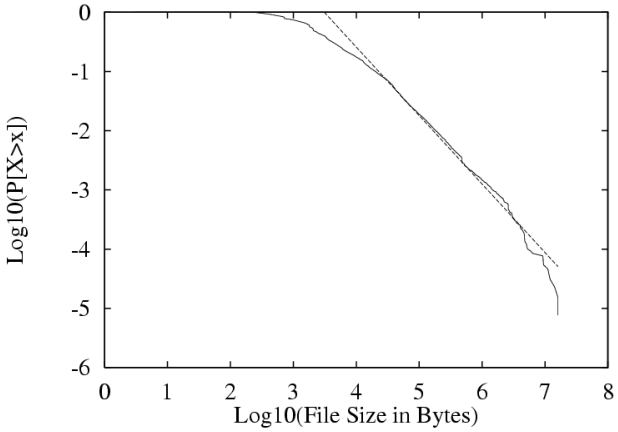


Figure 2-17: Heavy-tailed distribution
log-log scale plot α estimation – source [CT99]

In a log-log plot a Pareto distribution has a linear straight slope for the whole of the tail. A lognormal and a Weibull distribution also have a linear slope for much of the tail, but are curved downwards at the end of the tail. Therefore,

[11] See Appendix A for the definition of lognormal and Weibull distributions.

Figure 2-17 could indicate a lognormal or Weibull distribution. A light-tailed distribution (e.g., exponential) curves down much more quickly in a log-log CCDF than Pareto, Weibull and lognormal distributions. An exponential distribution can be best identified when plotted in a log-linear scaled CCDF. In this case it exhibits a straight line.

An important consequence of heavy-tailed distributions is that due to the infinity of all moments $k \geq \alpha$, no analytical form of the Laplace transformation exists. This renders in particular standard queueing theory impossible for Pareto, lognormal and Weibull distributions. Possible approaches to model these distributions for analytical analysis are the already introduced phase-type models (hyper-exponential models) [KSH03] [Ols98] or the Transform Approximation Method [HBF00]. Both methods can only *approximate* the heavy tail, but they are not really heavy-tailed and often fail to be parsimonious. In simulation-based investigations the actual distribution can be simulated. This allows arbitrarily close approximation of those functions, but requires very long simulation times. If not great care is taken in this aspect, the results greatly differ from reality [GSFM02].

Definition: covariance stationary process

A stochastic process X is covariance stationary (or weakly stationary) if the first two moments are not a function of t, that is $E(X_t)=\mu$ and $\mathrm{Var}(X_t)=\sigma$, and $\mathrm{Cov}(X_t,X_{t+k})=\gamma(k)$.

Definition: aggregated stochastic process

X is a stochastic process with autocorrelation function $r(k)$. The new process $X^{(m)}$, defined by:

$$X^{(m)} = (X^{(m)}_k, k \geq 1) \text{ with} \tag{2.3.10}$$

$$X^{(m)}_k = m^{-1}(X_{(k-1)m}+1+\ldots+X_{km}), \text{ for all } k \geq 1, \tag{2.3.11}$$

is the aggregated process of X at aggregation level m.

The corresponding autocorrelation function is denoted by

$$r^{(m)}(k);\ k \geq 0. \tag{2.3.12}$$

Note, if X is a covariance stationary process, $X^{(m)}$ is also a covariance stationary process.

Definition: self-similarity

A zero-mean covariance stationary process X is exactly self-similar if for all $m \geq 1$

$$X \approx^{D} m^{1-H}X^{(m)}, \tag{2.3.13}$$

where $\approx^{D}$ denotes equality of the corresponding finite dimensional distributions and H is the self-similarity parameter with $0<H<1$; and X is asymptotically self-similar if $m^{1-H}X^{(m)}$ converges to a non-degenerated process as $m \to \infty$. That is, the distributional equality holds for large m.

Often, only the first two moments are considered. This is the case for LRD stochastic processes. Long-range dependent processes are characterized by a hyperbolically decaying autocorrelation function. That is, they represent a class of stochastic processes which hold dependency over long time periods.

This is in contrast to Markovian processes, which are short range dependent (SRD). For them the autocorrelation decays much faster. The difference between LRD and SRD is manifested by the behavior of the autocorrelation function of the aggregated stochastic process. In the case of SRD the correlation structure of the aggregated processes degenerates for large m. That is, $r^{(m)}(k) \to 0$ as $m \to \infty$ for k=1,2,3,…, i.e., the *aggregated* process converges to second-order pure noise.

Definition: second-order self-similarity

A zero mean covariance-stationary process X, with variance σ^2 is *exactly second-order* self-similar if for all $m \geq 1$, X and $m^{1-H}X^{(m)}$ have identical second-order statistics.

That is:

$$\mathrm{Var}(m^{1-H}X^{(m)}) = \sigma^2 \tag{2.3.14}$$

and

$$r^{(m)}(k) = r(k),\ k \geq 1, \tag{2.3.15}$$

$$r(k) = 1/2((k+1)^{2H} - 2k^{2H} + (k-1)^{2H}), \tag{2.3.16}$$

with self-similarity parameter

$$H = 1 - \beta/2\,. \tag{2.3.17}$$

X is *asymptotically second-order* self-similar if for $m \to \infty$

$$\mathrm{Var}(m^{1-H}X(m)) = \sigma^2 \tag{2.3.18}$$

and

$$r^{(m)}(k) = r(k)\ k \geq 1. \tag{2.3.19}$$

Note: if $H \in (0.5,1)$, X is called LRD. Both exactly and asymptotically second-order self-similar traffic can include long-range dependency, however, as it is often the used case, we will use the term long-range dependent for asymptotically second-order self-similar traffic.

Asymptotical second-order self-similar processes have a few interesting characteristics. For instance, the autocorrelation function of an LRD process decays hyperbolically. That is (with β as in equation (2.3.17)):

$$\lim_{k \to \infty} r(k) \sim |k|^{-\beta} \quad (0 < \beta < 1)\,. \tag{2.3.20}$$

also the variance of the aggregated time series decays very slowly. That is:

$$Var\left[X^{(m)}\right] \sim m^{-\beta} \quad (0 < \beta < 1). \tag{2.3.21}$$

Another property is that the power spectrum is singular as the frequency approaches 0. That is,

$$S(w) \underset{w \to \infty}{\sim} 1/|w|^{(1-\beta)} \quad (0 < \beta < 1). \tag{2.3.22}$$

If traffic is LRD, its impact on network performance is different from SRD processes. As described in section 2.3, the task of traffic engineering is to model the relationship between traffic demand, network capacity and network performance. In the case of LRD, network metrics such as delay, throughput, loss and queue length in buffers are strongly affected. For instance, the queue length in buffers of routers are heavy-tailed [Nor94]. This leads to buffer sizes in routers that need to be larger than those predicted by traditional models with the same average arrival rate [PT98] [Mor95]. This is due to the bursty nature of the traffic and that the bursts do not smooth out by aggregation over large time scales, but instead get larger as well.

Heavy-tailed queue length implies also packet delays with a heavy-tailed distribution. Influence from this can be seen at all levels. For instance, the application object download times, which are directly experienced by the end user, have a heavy-tailed distribution [ST99] Consequently, the TCP's mechanism to estimate the round-trip times is influenced by this.

As a result, congestion situations are unavoidable for self-similar traffic, and they appear as short-lived impulses [Pop01].

In [PKC96] it is shown that only increasing the buffer sizes does not result in significant improvements in packet loss behavior. The relationship between H the buffer size, link capacity and packet loss can be summarized:

- Decrease of buffer and increase of H lead to higher packet loss
- Increase of link bandwidth and buffer space super-linearly improve performance in terms of packet loss, but they also introduce larger delays.
- For high H an increase in bandwidth and buffer space drastically decreases delay.

This concludes that for large H the bandwidth *and* buffer space need to be increased in order to keep the delay low.

The reasons for self-similarity can be split into two main areas. The user and application level is responsible for the large time scaling properties, i.e., the LRD properties, and network and transport level protocols are responsible for the small time-scaling[12] properties. In [CB96] and [WPT98] the authors showed that heavy-tailed sessions and file size lengths lead to self-similarity for large aggregation scales (long-range dependency). The authors in [FGHW99] and [PKC97] have furthermore shown that self-similarity (in particular the

[12] In the literature the scaling behavior is commonly split into large time scale scaling, which considers time orders of minutes to infinity, and small time scales which considers time scales of minutes towards zero (e.g., [FGWK98]).

multifractal) behavior at small time scales (e.g., smaller than the average round-trip time) is due to protocol interactions of TCP. The small time-scaling properties can be self-similar or more complex like multifractal.

LRD property

The LRD property is concerned with the large-scale behavior of the process. The fine-scale is not of interest. First empirical and later mathematical proofs have shown that there is an intrinsic relation between the heavy-tailedness of length distributions and the long-range dependency property.

LRD can be explained by considering individual bursts. The traffic is split into periods of packet transmission (ON-period) and periods of inactivity (OFF-period) for individual users. Such bursts can for instance describe the times of downloading a Web page (ON) and reading a Web page (OFF) or correspond to the packet trains and silent periods seen on a LAN Ethernet. Based on the self-similar traffic presented in [LWTW94], the authors in [WTSW95] showed that if the ON and OFF periods are heavy-tailed distributed, the aggregated traffic obeys the long-range dependent property.

It can be also shown that the superposition of sources which have individually an ON/OFF distribution that are heavy-tailed with $1<\alpha<2$, constitute a fractional Brownian motion process [TWS97]. A fractional Brownian motion process has the property of LRD.

The parameter α from the heavy-tailed distribution is related to the Hurst parameter by

$$H = \frac{3-\alpha}{2}. \tag{2.3.23}$$

In another model for self-similar traffic, the arrival and length of application sessions are considered. In this model, sources arrive at the link at a rate of λ, described by a Poisson process. That is, the inter-arrival times are negative exponentially distributed. Each source transmits for a random time; the length of the transmission time is heavy-tailed with $1<\alpha<2$, and infinite variance. In [PF94] [WPT98] the authors show that the arrival process for popular applications like FTP, Telnet and HTTP is indeed Poisson and the session lengths are heavy-tailed. The traffic from aggregating FTP, Telnet and HTTP is again long-range dependent. A mathematical model describing this phenomenon is the infinite source Poisson model denoted by M/G/∞ [WPRT01].

2.3.3.3 Measuring heavy-tailed distributions

To identify data from measurements, as being from a heavy-tailed distribution is very difficult. Heavy-tailedness is only defined for infinite data samples, as it requires $x\rightarrow\infty$. In reality, however, there is no such thing as infinite sample size. The data measurements are always truncated at some point in time. Further, it might be that the data values are already limited by the system. This is for example the case if the call session length is limited by the operator. That is,

one always deals with truncated distributions. An upper-truncated Pareto random variable, X, has distribution

$$P(X > x) = \frac{(x^{-\alpha} - \nu^{-\alpha})}{1 - \nu^{-\alpha}}, \text{ for } 0<x\leq\nu, \tag{2.3.24}$$

where ν is the upper truncation parameter. Figure 2-18 displays the CCDF for the full Pareto distribution – solid line – (equation (2.3.9)) and for the truncated Pareto distribution – dashed line–. One has to keep this effect in mind when investigating empirical CDF.

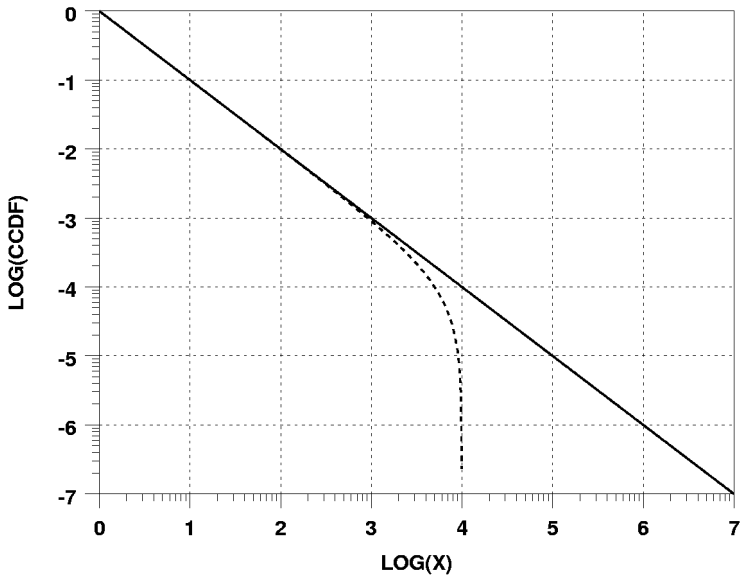


Figure 2-18: Truncated Pareto distribution
Solid line, Pareto distribution α=1 –
dashed line, truncated Pareto distribution, α=1, ν=10^4

2.3.4 Network measurements

Central to building accurate traffic models are sound measurements. “One of the biggest challenges appears to be to understand whether and how workload characteristics are changing as new applications arise and are deployed in the Internet” [CLR00].

If the network or application does not yet exist, the traffic model is based on assumptions. ‘High-level’ parameters like data volume might be extrapolated from similar networks, while structure of the traffic might be derived from application protocol standards.

If possible, measurements should be used to obtain accurate numbers. But measurements represent only a snapshot view of the traffic in a particular network. Combining the measurement results with sound forecast studies should be used to describe the development of the traffic.

Traffic measurements can be done in several ways. Measurements as basis for traffic modeling are typically passive measurements. Passive measurements record the traffic directly of a link. This reflects the actual traffic, and the traffic

in the network is not altered by this technique. Active measurements, on the contrary, inject traffic in the network, and measure the feedback from the network. Active measurements are typically used for performance analysis. A very simple example is using the ping command[13] to measure the round-trip time in a network.

In traffic modeling we are mainly interested in passive measurements. Passive measurements can be conducted at several places. Inside the network, the measurements can be performed at some link, at a router, at a gateway, or at a proxy, etc. Alternatively, one can measure at the edge of the network. Possible places include the client (e.g., Web browser) or the server (e.g., Web server).

The advantage of measuring inside the network is the possibility of easily capturing the traffic of all the users using the network. This provides a broad statistical basis, but it can lead to a very large amount of trace data. The result of the measurement is a packet trace or trace of events. If one is interested in the traffic of individual sources, as might be the case for source traffic modeling, the source objects need to be reconstructed from the individual packet trace.

Placing the measurement point in the client or the server provides the opportunity to get detailed insight in the data generation process. For example, logging the content of browser sessions on the client does not only directly show the length of a file download, but can also provide the download termination cause. Possible causes could be 'normal' termination (i.e. end of file), or user termination (e.g., download took too long), etc. Such information is of great value to understand the relation between QoS in the network and the traffic demand. But it is a high effort to place monitoring units on all relevant clients or servers. This might limit the amount of available data significantly. The result of client-server measurements can be application object traces.

Many options exist to obtain traffic data; for instance, Web servers keep log files of page requests; or modified Web clients can be used to obtain log data. Many nodes also have event counters, e.g., routers count the number of IP packets in up- and downlink direction. Specific packet capturing tools can also be used for packet traces on node interfaces. For example, **TCPDUMP** [TCPDUMP] is a tool that allows capturing all IP packets on the network.

In all cases the measurement point needs to be well chosen. Highly desirable is a measurement point at which most traffic in the network passes by and which allows access to all relevant information with respect to the traffic type.

Some aspects that need to be considered for doing traffic measurements are:

- Access to data
- Privacy and confidentiality
- Measuring the data
- Data storage and post-processing

[13] The ping command sends out a sequence of IP packets. Each IP packet is echoed by the destination. This allows for example to measure round trip times and packet drop probabilities.

Access to data

Access to data is not only a technical issue; it is also about how to get access to the network and the relevant data. In the case of measurements inside the network, the network operator must be involved in the measurements. Network data is sensitive data, which is not easily accessible. A trust relationship between the network operator and the measurement group is required for network-placed measurements. This relationship also includes the possibilities and means to access node locations.

Another kind of relationship is required in the case of measurements placed at the edge of the network. That is, the measurements are done inside the client. For this type of measurements, the users must be willing to run a modified client. In any case, a high demand on privacy and confidentiality needs to be fulfilled to allow access to data.

Privacy and confidentiality

Privacy is something that concerns the end user. It comprises the anonymity of the user and the privacy of the content. National laws regulate access and processing of private data. Privacy can be achieved if the captured traces are post-processed immediately with tools like `TCPurify` [TCPURIFY] or similar concepts [PP03]. Such tools remove the application content from captured packets and replace IP addresses or other user identification with a random number. The network operator is furthermore concerned with confidentiality of the results, because the measurements can provide insight into economical aspects of the operator. This needs to be addressed by a proper choice of the presented results. A careful selection of the results can hide any critical information about economical aspects.

Measuring the data

Measuring the data can be split in two subtasks: tapping the network and capturing the packets. Depending on the network type, tapping is intrusive or non-intrusive. For instance, in the case of optical networks (Fiber Distributed Data Interface (FDDI) or Asynchronous Transfer Mode (ATM) over Fiber), an optical splitter is needed. This is intrusive and requires greater care to not interrupt the network traffic; better are passive solutions. In the case of an Ethernet-based infrastructure, a passive solution is possible. A switch or hub which is equipped with a monitoring port and which can be put into promiscuous mode is sufficient. Capturing is the process of taking a blind copy of the packets from the network interface, adding a time stamp and writing it in a specific format on a storage device. Capturing the packets off the network requires specific tracing software and often also specific hardware. The combination of software and hardware needs to be fast enough to read all data from the interface and write it quickly enough on the disk. For network interfaces beyond the speed of 100 Mbit Ethernet, this might already be a critical issue. The Waikato Applied Network Dynamics (WAND) research group at Waikato University, New Zealand [WAND] develops special hardware (DAG cards) which can capture packets from up-to OC-48 links, with high accuracy. For slower networks, a solution of a standard network card together with

TCPDUMP might be adequate. The software (or hardware) should also provide means to filter the data. The filtering is based on content in the packet header and can be used to separate relevant from irrelevant traffic.

Central for accurate trace files is the issue of time stamping. All packets are time stamped when they are captured from the network. The time series is used to derive the stochastic process describing the traffic. But the hardware used for data capturing limits the resolution of the time stamps. Furthermore, often clocks do not run very precisely and are prone to having an offset, skew and drift [Pax98].

Data storage and post-processing

Data storage and post-processing can become a difficult task if the measurements lead to a large amount of data. Typically this is the case. For instance, measuring data for 1 day on a 30% loaded 100 Mbit/s LAN interface, leads already to 324 Mbyte of data. A large measurement project might easily lead to terabytes of data. This puts again high requirements on the post-processing hardware. The data must be easily accessible and the post-processing should be supported by data warehouse systems [MBB00].

2.3.5 Internet measurements and modeling overview

2.3.5.1 Measurement projects

Since Internet traffic has been identified as being extremely complex, and the need for measurements is accepted widely, a large number of measurement studies exist which describe Internet traffic at various levels. In the following we give a short overview of some long-term projects and some measurement efforts reported in research papers. This is a vastly growing field and therefore this overview can by no means be regarded as complete.[14]

Cooperative Association for Internet Data Analysis (CAIDA)

The Cooperative Association for Internet Data Analysis (CAIDA) is a collaborative undertaking of organizations in the commercial, government, and research sectors. CAIDA's aim is to promote greater cooperation in the engineering and maintenance of a robust, scalable global Internet infrastructure. CAIDA provides a neutral framework to support cooperative technical endeavors.

CAIDA has published a large number of papers on Internet traffic statistics. However, many of them are related to performance, topology, and infrastructural measurements. From traffic modeling perspective the results usually do not go beyond key statistics and do not include detailed source models.

TCP flow characterization is investigated in [CBP95] at network level. [BC02] presents in particular results, based on measurements, of flow characterization with respect to the length of the flows in time. We perform a similar study in

[14] The following description cites the corresponding Web-pages.

chapter 6, albeit we go beyond their approach with respect to modeling. [BHGC04] provides a study of the flow disparity in the Internet, which is the phenomenon of mice (very short) and elephant (very long) flows. We consider this as well in chapter 6. [MC00] comprises a large backbone measurement study including a survey on trends in application usage.

Internet address: http://www.caida.org

Measurements are not publicly available at their site.

NLANR measurement and network analysis group

The goal of the NLANR Measurement and Network Analysis Group (NLANR/MNA) is to characterize the behavior of high performance connection (HPC) networks. They maintain a Network Analysis Infrastructure (NAI). The NAI includes a collection of measurement data and multiple analyses, tools and methods. Since 1995, the National Laboratory for Applied Network Research (NLANR) has been collecting IP packet header traces to support research in understanding the systemic nature of the Internet.

Especially the large numbers of measurement traces are of interest for comparison studies. There are published papers as well. However, they typically include only key statistics, but no modeling. The publications in particular focus on how to perform passive measurements. For instance [MBB00] provides an overview of the NAI of NLANR at that time.

Internet address: http://moat.nlanr.net/

A large repository of Internet traffic traces is available.

The Internet traffic archive

The Internet Traffic Archive is a moderated repository to support widespread access to traces of Internet network traffic. The traces can be used to study network dynamics, usage characteristics, and growth patterns, analyzing traces and modeling traffic, as well as providing the input for trace-driven simulations.

This includes famous traces used in many Internet traffic analysis studies. For instance traces from UC Berkeley which have been used in [PF94] which is a landmark paper highlighting the importance to focus on models beyond Poisson traffic, as well as [Pax94] in which models for TCP connections have been derived.

Internet address: http://ita.ee.lbl.gov/index.html

A large repository of Internet traffic traces is available.

WAND

WAND is a research group at the University of Waikato Computer Science Department. The group is involved in a range of computer network projects mostly focused around network measurement. They develop their own

measurement cards for extreme high-performance network switches. They cooperate with CAIDA and NLANR as well. Some publications on key statistics are available, for instance [Joy00], in which gaming traffic for the Internet is investigated.

Internet address: http://wand.cs.waikato.ac.nz

Measurements are not publicly available.

Cost-257

Cost-257 has been a European research project with the goal of improving the design of broadband multi-service switching systems and network architecture. Part of this project in the context of performance evaluation has also been to model traffic demand. This also includes some publications on proposed traffic models for wireless networks.

Internet address: http://www-info3.informatik.uni-wuerzburg.de/cost/

Measurements are not publicly available.

2.3.5.2 Invariants of the Internet

The measurement and modeling effort has lead to a number of identified invariants. However, those invariants are often still subject to discussion. For example, recently [Dow01] questioned many previous papers identifying heavy-tailedness. He investigated again many old traces which had been available and often did not find the same significance on the results as the original author. In particular he claims that file sizes are actually lognormal distributed, which he clearly separates from power-tailed Pareto distribution. In many cases it depends on the particular measured network and time period. However, we list in Table 2-5 some of the often stated traffic invariants.

2.3.5.3 Models for wireless networks

We will briefly provide some examples of wireless network traffic models proposed in the literature.

[TR101112] provides standardized test environments for the design of 2.5G and 3G networks. Among other aspects (such as radio) this also includes packet traffic models, and parameterization for various usage scenarios. The packet model uses Pareto distributions to describe packet (object) size distributions. The model is very simple.

[VC02] describes high-level requirements and taxonomy for application characterizations. It focuses specifically on UMTS and multimedia applications.

[BLA00] presents a possible traffic mixture for wireless Internet as well as basic characteristics of the used applications. The focus is on established applications like HTTP, Email, FTP, Telnet, etc., while new specific applications like WAP and MMS are totally absent. Their traffic models and parameterization rely in part on measurements performed at the dial-up router. As new 2.5G cellular networks provide similar throughput to old analog and

Integrated Services Digital Network (ISDN) modems, the authors consider these results as comparable. Taking this approach seems promising, as no wireless measurement from commercial networks had been available by this time.

Invariant	Protocol Level	Description
Traffic growth	IP	Internet Traffic continues to grow exponentially
Pareto Rule	Several	10% of the files accessed account for 90% of the server requests and 90% of the bytes transferred
		10% of domains account for >75% of usage of a server or 25% of the servers account for 85% of the traffic
		1% of packet flows account for about 80% of bytes transferred
Session Arrival	Session	Poisson
Session Duration	Session	Pareto (heavy-tailed)
Session Size	Session	Pareto (heavy-tailed)
Diurnal patterns	Several	Usage follows often the day profile of humans – with one or two busy hours
WAN traffic	IP	Self-similar process (fractal, multifractal)
LAN traffic	IP	Self-similar process (fractal)
Flow length	TCP	Heavy-tailed
Packet IAT[15]	IP	Heavy-tailed
Web Page reading time	Application	Heavy-tailed
HTTP flow length	HTTP	Lognormal (body) and Pareto (tail) distributed
FTP transfers	FTP	Pareto (tail) distributed
Connection Size	Several	Lognormal distributed
Connection Duration	Several	Lognormal distributed
File Request popularity	Application	Zipf distributed
Packet size	IP	Bi-modal distributed
TCP popularity	IP	TCP packets account for most of the packets on the internet

Table 2-5: Invariants of Internet traffic [FP01] [Pit99] [AW96] [Wil01]

[KMM00] shows comprehensive results on GPRS performance simulation. The traffic models are also extrapolated models from fixed networks as the authors have done the analysis before GPRS networks were deployed. In order to adapt for mobile network usage, the models resemble truncated versions of measurement and modeling results from [WPT98] and [Pax94]. The traffic mixture focuses on standard Web applications and misses WAP and MMS applications.

[RLGPC+99] proposes a source traffic model for Web traffic, based on HTTP traces taken from wireline measurements. Session arrival times are exponentially distributed, pages within a session are log-normal distributed and page sizes are assumed to be Pareto distributed. A weakness could be that the measurements are from a LAN network, which has no wireless representative characteristics.

[15] IAT – Inter-arrival Time

[VHS04] This report is based on the same measurement as this thesis. The authors present a comprehensive source traffic model for WAP. They use a heuristic approach to separate idle and reading times in a WAP session. They also provide detailed parameters and distributions for the source traffic model.

[SFB01] uses a WAP model in their GPRS simulator GPRSSim. The hierarchical model comprises user behavior and application behavior. The model in particular focuses on modeling the packet size distribution. It is based on measurements in a wireline LAN environment.

Our literature survey has shown that few wireless traffic models, based on true cellular traffic measurements, exist. The majority of traffic models are based on extrapolation from wireline measurements. This is the only feasible method as long as no commercial mobile networks have been available to study. However this is changing recently.

2.4 Summary

In this chapter we have provided background information about cellular wireless networks, the Internet and the teletraffic theory. Several generations of cellular networks, allowing ubiquitous access on IP based data services, exist. Our particular chosen network in this thesis, the 2.5th generation network GPRS, is only one of several similar technologies, albeit a very wide deployed one. Therefore we assume that the results we derive for the GPRS will be also useful for many other similar cellular networks. The Internet protocols IP and TCP/UDP allow a very flexible deployment of services. We explained that the protocol headers allow a differentiation of the traffic according to the service type. This will be very helpful in the design of our measurement setup. Some very frequently used Internet services (Web, Email, FTP, RADIUS, etc.) have also been briefly presented. We will encounter them also in our GPRS measurements. We finished the chapter with an overview of similar measurement and modeling studies available in the literature and in Internet repositories. Albeit a number of studies extrapolate from wireline measurements, it shows that there exist no sufficient results for explicit wireless traffic modeling. This missing information shall be supplied by this thesis.

3 GPRS

Our measurements and results originate from the 2.5G cellular wireless data access system GPRS. The details of this system are explained in this chapter. Section 3.1 explains that the packet switching principle is used throughout the GPRS network architecture. Details of the architecture and protocols are explained in section 3.2 and section 3.3. The GPRS data bearer transmission speeds are listed in section 3.4. Section 3.5 introduces the Gi and Access Point Name (APN) concept, which is of large importance for our measurement setup. Section 3.6 introduces the GPRS session and mobility management, which is of relevance for our mobility investigation section. All steps involved in data transmission in GPRS are summarized in Section 3.7. Section 3.8 introduces in detail the novel GPRS applications WAP and MMS. And, section 3.9 summarizes this chapter.

3.1 Packet switching in GSM

GPRS can be seen as an upgrade to GSM. It has been standardized in GSM phase 2+ and today it is standardized in the 3GPP standardization body [3GPP]. GPRS has been smoothly integrated in GSM. It is embedded in the same physical channels (TDMA) structure and reuses many parts of the established infrastructure. Some of the nodes, that are already deployed in current GSM systems, can be shared between GPRS and GSM. Only two new node types – Serving GPRS Support Node (SGSN) and Gateway GPRS Support Node (GGSN) – have been newly introduced. Furthermore, GPRS can be implemented in the existing GSM systems using the same cell structure.

GPRS provides a packet-switched bearer in a GSM network. In particular, it provides a natural IP routing service. IP packets are routed from the mobile station through the GPRS network into the Internet, whereas no link layer encapsulation like Point-to-Point Protocol (PPP) is needed. The packet-switching principle is used throughout the whole GPRS network. IP is deployed in the GPRS backbone, connecting the special GPRS nodes. Also the resources on the radio interface are assigned to the mobile stations only temporary on a per packet basis. In contrast to ordinary GSM, where one time slot is assigned to one user for the whole call duration, in GPRS the radio resources are only assigned for the duration of one or a few IP packets.

3.2 Network architecture

Figure 3-1 depicts all nodes and interfaces defined for GPRS and GSM. Solid lines indicate user data interfaces; dashed lines indicated signaling interfaces. The most relevant nodes involved in data transmission in GPRS are the Base Station Subsystem (BSS) consisting of the BTS and BSC, as well as the new

GPRS specific nodes GGSN and SGSN. A detailed description of all nodes and interfaces can be found in [GSM3.60].

Gateway GPRS Support Node

The GGSN acts towards the external Packet Data Network (e.g., IP) as the router serving all IP addresses of the mobile stations in the GPRS network. The GGSN is connected to the external IP networks by the Gi interface. Typically the GGSN is connected to the operator's IP based backbone through which all servers of the operator (e.g., Web server, Email server, WAP Gateway, RADIUS server) as well as the external 'open' Internet can be accessed. In particular the GGSN communicates with the RADIUS server using the Dynamic Host Configuration Protocol (DHCP) for obtaining the IP address for newly attached mobile stations. By providing the interface to external IP networks, the GGSN also handles IP security and firewall functions and provides charging and session management support.

Serving GPRS Support Node

The SGSN is the interface node between the GPRS backbone and the radio network nodes. It directly communicates with the BSC using the Base Station Subsystem GPRS Protocol (BSSGP) for signaling. It serves all GPRS subscribers that are physically located within a geographical SGSN service area. Its task is ciphering and authentication, session management, mobility management and logical link management. It also provides a connection to the databases in the MSC and is involved in charging functions.

Base Station Subsystem

The BSS consists of two sub nodes which have been evolved from GSM to support GPRS. The BSC hosts all relevant protocols for the communication over the radio interface. Its function is to set up, supervise and disconnect packet-switched calls. This includes cell reselection, cell configuration and channel assignment. The Base Transceiver Station BTS is only a relay station regarding the radio interface protocols. It performs the modulation on the carrier frequencies and demodulation of the signals.

Mobile Station

The mobile station (MS) comprises the Mobile Terminal (MT) and the Terminal Equipment (TE). The mobile terminal connects to the BTS over the radio interface (Um-interface). The GPRS protocols for the transmission plane and signaling are terminated in the MT. In particular these are the Radio Link Control (RLC) and the Medium Access Control (MAC) protocol. The TE contains the client applications. This could be an external laptop that is connected to the mobile station via serial cable or Bluetooth, or it could be build into the same device and be represented by, e.g., the WAP browser. As it is common in the literature, we use the term mobile station (MS) meaning either only the mobile terminal or the combination of the mobile terminal and the terminal equipment if the distinction is not relevant.

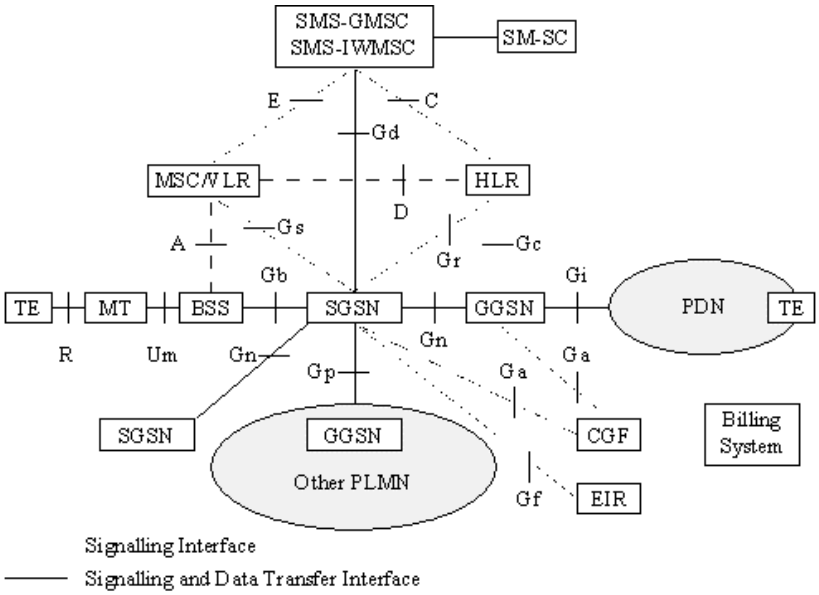


Figure 3-1: GPRS nodes and interfaces – source [GSM3.60]

3.3 Protocol stack

Figure 3-2 depicts the protocol stack of the transmission plane in GPRS. The GPRS Tunnel Protocol (GTP) is used to tunnel user data packets through the IP based GPRS backbone between GPRS Support Nodes [GSM9.60].

The Sub-Network Dependent Convergence Protocol (SNDCP) carries the network layer protocol data units (IP) in a transparent way [GSM4.65]. The main task of the SNDCP is data compression (e.g., V.42bis) and header compression (e.g., TCP/IP header compression), for improving channel efficiency.

The Logical Link Layer (LLC) protocol operates across the Gb and the Um interface, providing a logical link between the MS and the responsible SGSN [GSM4.64]. The LLC protocol functions comprise ciphering, flow and sequence control.

The RLC/MAC protocol layer provides services for the transfer of LLC PDUs using a shared medium between multiple mobile stations and the network [GSM3.64]. The MAC protocol coordinates the access on the up and downlink time slots. The RLC protocol allows the retransmission of radio blocks if they are corrupted during transmission.

The physical link layer (RF) provides physical channels to the RLC/MAC layer. This includes forward error correction coding, interleaving, monitoring of radio link signal quality, power control procedures and transmission and receiving of modulated wave forms.

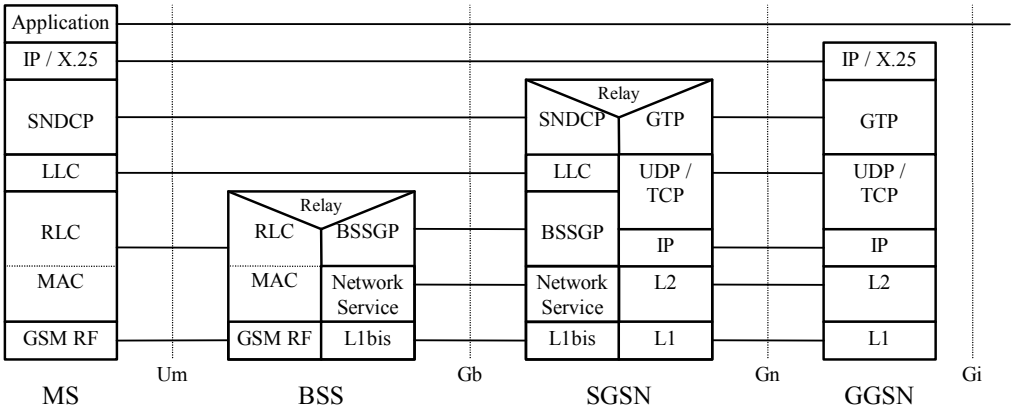


Figure 3-2: GPRS protocol stack [GSM3.60]

3.4 Data bearer speeds

GPRS introduces four new coding schemes on the radio link (cf. Table 3-1). This increase the theoretical rate on one time slot to 20 Kbit/s.[16] Furthermore, the MAC protocol allows the assigning of up to 8 time slots to one mobile station, which increases the theoretical maximum rate to 160 Kbit/s. In current deployments, which mainly offer coding scheme 1 and 2 and up to 4 time slots per mobile station, this maximum is reduced to 48 Kbit/s. In practical situations even lower values are common, as several mobile stations share the same time slots. A multitude of factors (e.g., mobile station class, radio conditions, cell load, network parameters, traffic characteristics) influence the practically reached throughput [KMM00]. But the operator naturally optimizes those values to maximize the throughput, and higher values can soon be expected.

Scheme	RLC data block without RLC header	Resulting LLC throughput rate
	(octets)	(Kbit/s)
CS-1	20	8
CS-2	30	12
CS-3	36	14.4
CS-4	50	20

Table 3-1: RLC block and LLC data rates

3.5 Gi interface and Access Point Name

The Gi interface (cf. Figure 3-1) defines the physical connection between the GGSN and an external packet data network [TS29.061]. An APN uniquely identifies a logical interface to external packet data network over the Gi interface.

Figure 3-3 depicts the case in which an Ethernet is used as physical Gi interface. The IP packets from the MS via the GGSN are switched to a router on the GPRS backbone.

[16] This calculation excludes overhead from radio protocols and specifies the theoretical available rate to IP.

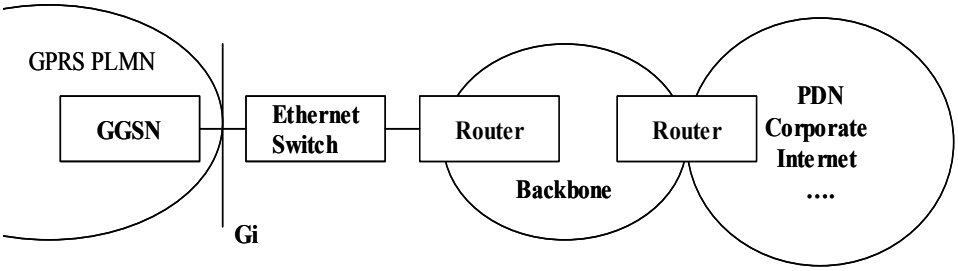


Figure 3-3: GPRS Gi interface

The logical interfaces, provided by the APN, allow separating traffic to different ISP/intranets. When the mobile station establishes a Pack Data Protocol (PDP) context[17] it specifies an APN that it wishes to connect to. In conjunction with the APN, the corresponding servers for address allocation, the authentication, the protocol configuration and the used protocols, and the security features for communication with the server and network are specified. The traffic for different APNs is separated on the Gi interface, for example, by means of Virtual Private Network (VPN) or IPsec tunnels. Figure 3-4 shows the case of three APNs on one Gi interface. In that example, one APN connects to the Internet, one APN connects to the WAP gateway, and one APN is the connection to a corporate network. Each sub-network, connected via the APN, has its own IP address space, which might even overlap. The traffic on the backbone is separated, e.g., by a VPN solution. The GGSN receives the IP address for the mobile station of each sub-network from the corresponding authentication server.

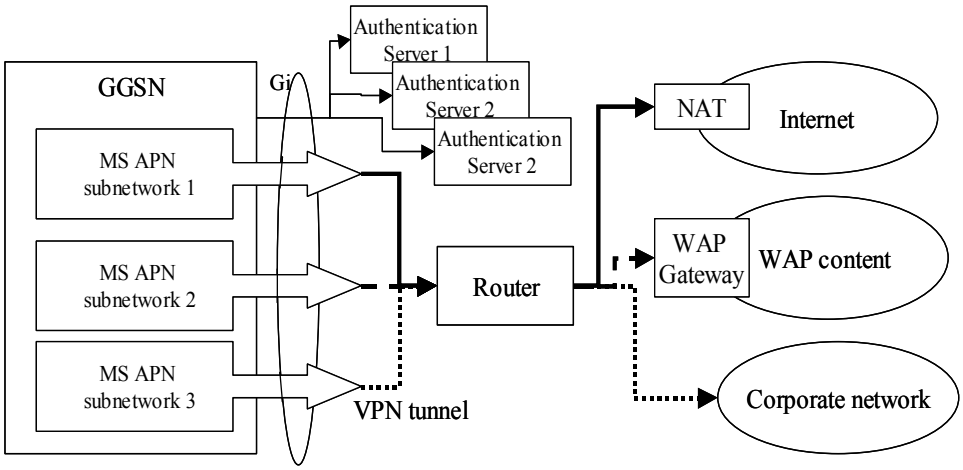


Figure 3-4: GPRS APN concept

[17] The concept of PDP contexts will be explained in section 3.6.1.

3.6 Session and mobility management

3.6.1 Session management

The mobile station can be in one of several connection states. This is handled by the session management (SM). The mobile station can be either attached or detached, and if attached it can have a PDP context activated or not (Figure 3-5). If a mobile station is not attached it is not visible to the network. No communication is possible. If the mobile station wants to use GPRS, it first attaches to the network. The attach procedure is initiated from the mobile station. When attached, the mobile station can be reached for signaling. From that point on, mobility states are also assigned to the mobile station. The mobility states will be explained in section 3.6.2. After attach, the location (routing area) of the mobile station is known. A PDP context is needed to prepare the mobile station for data communication. A PDP context activates a packet communication session within the SGSN. During the activation procedure, the mobile station either provides a static IP address or requests a temporary one from the network. It also specifies the APN by which it wants to communicate. In GPRS it is also foreseen that the mobile station can request a desired quality of service (QoS), however, this is not realized in most of today's networks.

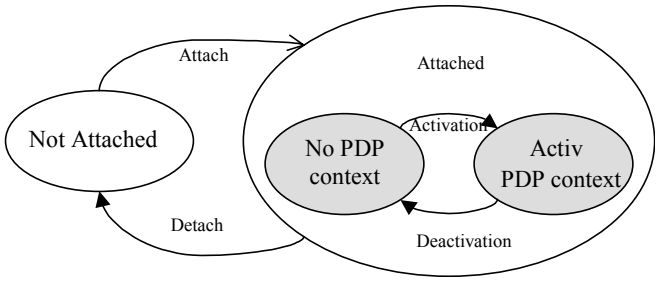


Figure 3-5: GPRS session management states

3.6.2 GPRS mobility management

The GPRS mobility management (GMM) is responsible for handling user mobility and keeping track of user locations. As GPRS is embedded in GSM, it deploys the same cell layout of the network. However, it has its own GPRS mobility management mechanism, which enables the user to move freely in the coverage area while always maintaining connectivity between the mobile station and the network. The GMM is located in the SGSN as well as peered in the mobile station, but uses functionality for location management provided by the BSC [GSM3.60].

Within GPRS two areas exist which can be individually addressed by the location management system of GPRS (Figure 3-6). A *cell* is the area which is covered by one BTS. Three cells are usually served from one site. Overlaid to this is the location structure of *routing areas* (RA) defined. One routing area represents many adjacent cells. The radii of the cells, which can vary from a few 50 meters up to 35 kilometers, and the size of the routing area, which can cover a few cells or most of the network, depends on the operator's network design and the deployed hardware.

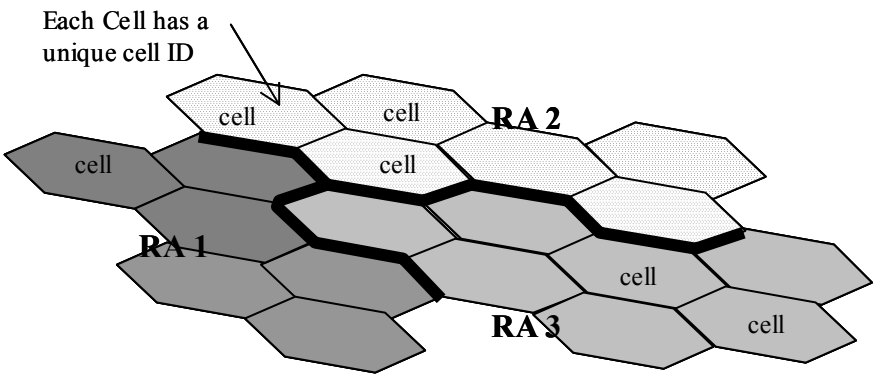


Figure 3-6: Cell and routing area IDs

The location of the mobile station in the network is expressed by either the cell Identification (ID) combined with the RA ID or solely by the RA ID. Which information is available in the network depends on the GMM state the mobile station is currently in. Three states exist: the *idle state*, in which no location information is available; the *stand-by state*, in which the location information only consists of the routing area ID; and the *ready state*, in which the location is given by the cell ID and the routing area ID.

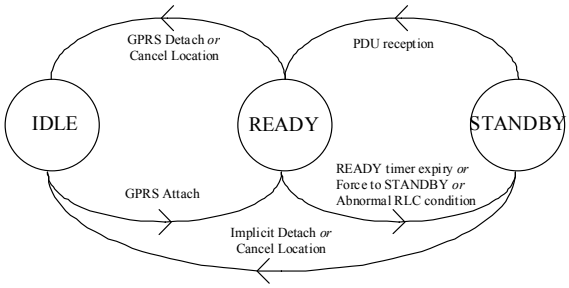


Figure 3-7: GPRS mobility management states

Figure 3-7 depicts the different mobility states of a mobile station. When the mobile station is not GPRS attached the state of the mobile station is idle. If the mobile station is attached, it can be in stand-by or ready state. As default it is in stand-by state. In this state, if packets need to be sent to the mobile station, the network sends a paging message to the mobile station in all cells of the routing area the mobile station resides in. After the mobile station responds, the exact cell location is known for further data transmission. If the mobile station itself wants to transmit data, it does so by asking for radio resources, which implicitly also announces its cell location. In both cases, at the same time when radio resources are reserved, the mobile station state is changed to ready. And even after release of the radio interface resources, the ready state is maintained for some period of time (45 seconds, operator configurable). The ready state is entered if user payload data as well as if signaling data is transmitted.

The two-layer hierarchy location information, based on cell ID and RA ID, together with the different mobility management states allows flexible and efficient mobile station localization. If the user is not transmitting data, the network is only aware of the approximate location (i.e. the routing area). But

during data transmission, the network knows the exact location for transmitting radio packets precisely to the mobile station.

Depending on the GMM state, the mobile station also reports changes in its location. In stand-by state the mobile station informs the network only if it crosses the border to a new routing area. This is called a R*outing Area Update* (RAU). Besides this, the mobile station also sends its current routing area information after a longer idle period (about once per hour, operator configurable), with so-called *periodic routing area updates*. In the ready state the mobile station informs the network with a *cell reselection* (CR) message about every cell change. As a new routing area also implies a new cell, a routing area update also implies a cell reselection.

3.7 GPRS data transmission

From the previous sections it is clear that several connection steps are involved before IP packets can be transmitted via GPRS. We depict this in Figure 3-8. Before a mobile station is allowed to transmit data, the mobile station has to attach to GPRS. This step makes the mobile station known to the network. Besides the temporarily reserved resources needed to transmit the signaling packets, no radio interface resources are reserved. The next step is a PDP context establishment. From this point on the mobile station has an IP address assigned, but still no radio interface resources are assigned. For the actual IP packet transmission over the air, as a final step, time slots are reserved on the radio interface by establishing a TBF. The TBF establishment is realized by the RLC protocol. The timeslot reservation is only kept up as long as needed to transmit a burst of IP packets. Within a PDP context several TBFs can be subsequently established for transmitting data.

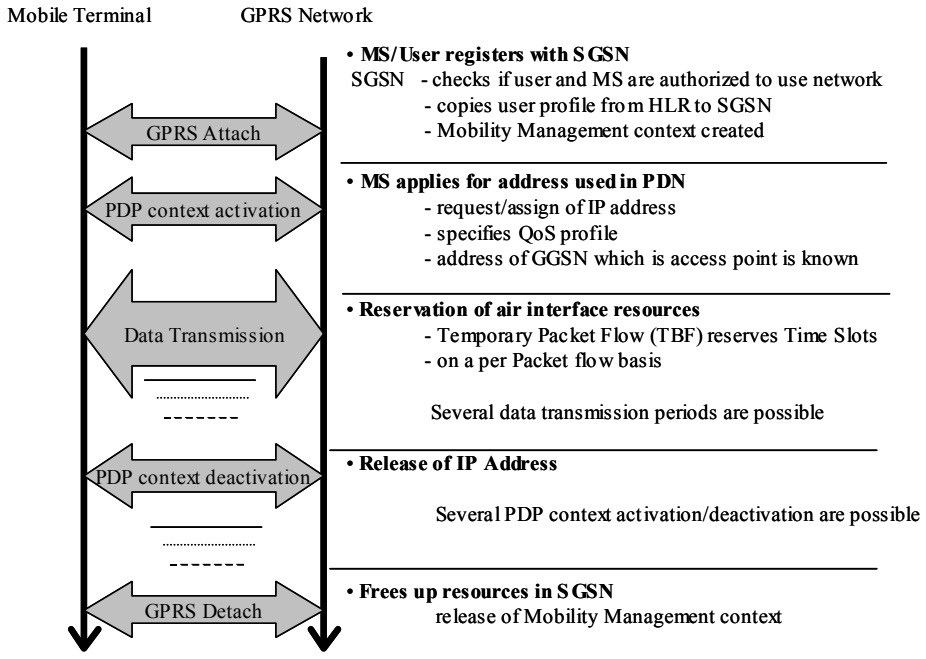


Figure 3-8: GPRS data transfer session

During an attach period the PDP context can also be closed and opened up again. Closing down the PDP context frees up IP addresses and other resources in the network, but as the mobile station is still attached, it can be easily reached again.

This hierarchical approach allows efficient resource management and also quick access if needed. GPRS attach and PDP context activation procedures are in the order of seconds, while establishing a TBF is in the order of several hundreds of milliseconds.

3.8 GPRS applications

GPRS provides a packet switched bearer for a wide range of applications (see Figure 3-9). Due to the system characteristics (especially the IP support), all applications traditionally seen in the wireline Internet are supported by GPRS. Furthermore, in particular GPRS introduces two new applications. The wireless application protocol (WAP) and the multimedia messaging service (MMS).

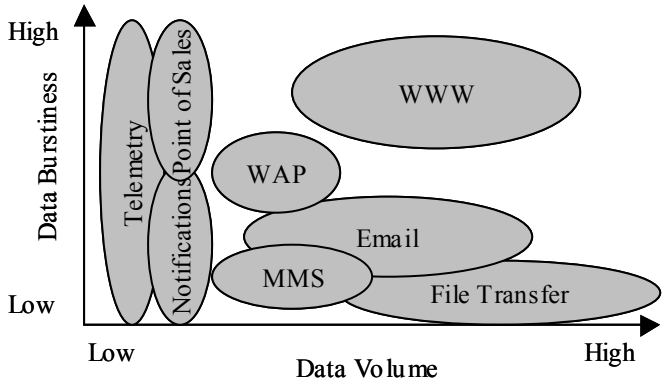


Figure 3-9: GPRS application scope

The WAP and MMS applications have been developed in particular for mobile access networks by the WAP forum (which is now the Open Mobile Alliance) [OMA] and lately 3GPP.

3.8.1 Wireless Application Protocol

One has to differentiate between WAP 1.2 and WAP 2.0. WAP 1.2 is the currently deployed version, which uses its own protocol stack. WAP 2.0 is the new version introduced at the moment, which is more tightly integrated into the established TCP/IP protocol stack.

WAP 1.2

WAP 1.2 is a whole protocol suite similar to the TCP/IP protocol suite (see Figure 3-10). It replaces many of the previously explained protocols of the TCP/IP protocol stack in favor of some protocols tailored to clients residing in mobile stations and using wireless access networks. The purpose of WAP is to facilitate specific mobile services.

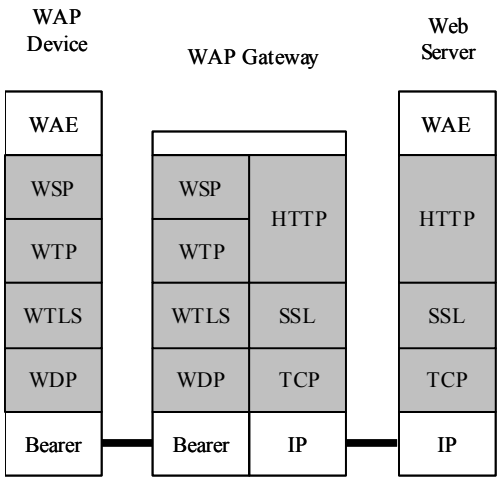


Figure 3-10: WAP protocol stack

The WAP stack is divided into six layers. They are:

- Application layer: Wireless Application Environment (WAE)
- Session layer: Wireless Session Protocol (WSP)
- Transaction layer: Wireless Transaction Protocol (WTP)
- Security layer: Wireless Transport Layer Security (WTLS)
- Transport layer: Wireless Datagram Protocol (WDP)
- Bearer networks: GPRS, Circuit Switched Data (CSD), etc.

WAP offers a similar service like Web browsing over HTTP, but for a resource restricted environments, like a mobile station and a mobile network. The client browser must operate very resource efficient in such an environment. WAP content on the server is encoded in the Wireless Markup Language (WML), which has restricted similarities with HTML but is more compact than HTML. A WAP page can also contain embedded content, which is referred to via links in the WAP page.

One important aspect of WAP is that it allows capability negotiations. Capability negotiation is used between client and server to agree on a mutually acceptable level of service, and to optimize the operation of the service provider according to the actual requirements of the service user. A list of important capabilities which can be negotiated is provided in Table 3-2 [WAP230].

The WAP protocol suite (Figure 3-10) foresees a split-architecture for communicating between the client (WAP device) and the server (Web/WAP server). It uses a WAP gateway to translate between WAP specific protocols and TCP/IP suite specific protocols. This also includes binary encoding of WAP pages and transcoding of content (e.g., minimizing embedded pictures), to reduce size.

Instead of HTTP and TCP, WAP browsing uses WSP [WAP230] and WTP [WAP224] as protocols for content request and delivery.

One can differentiate between connection-oriented and connection-less WSP-mode. In the connection-oriented mode the mentioned capability negotiation

can be used. The connection-oriented mode uses a reliable communication, allowing retransmission. This is based on WSP over WTP. In this case the so-called class-2 WTP methods are used. This is the WTP method for reliable transmission including retransmission, acknowledgement. In the case of the connectionless mode, WSP is used over WDP, which is only a datagram service, comparable to UDP. In this case the default values from Table 3-2 are used. This also implies no retransmission of corrupt data packets.

WAP 1.2 received much criticism in the established Internet protocol community. In particular they criticize that the WAP standardization reinvents established mechanisms like TCP. This was considered and therefore WAP 2.0 was proposed.

Capability Name	Default	Description
Extended Methods	None	This capability is used to agree on the set of extended methods (beyond those defined in HTTP/1.1), which are supported both by the client and the server peer, and may be used subsequently during the session.
Maximum Outstanding Method Requests	1	The client and server use this capability to agree on the maximum number of method invocations, which can be active at the same time.
Maximum Outstanding Push Requests	1	The client and server use this capability to agree on the maximum number of confirmed push invocations, which can be active at time.
Client SDU Size	1400 octets	The client and server use this capability to agree on the size of the largest transaction service data unit, which may be sent to the client during the session.
Server SDU Size	1400 octets	The client and server use this capability to agree on the size of the largest transaction service data unit, which may be sent to the server during the session.
Client Message Size	1400 octets	The client and server use this capability to agree on the size of the largest message, which may be sent to the client during the session. One message may consist of multiple transaction service data units.
Server Message Size	1400 octets	The client and server use this capability to agree on the size of the largest message, which may be sent to the server during the session. One message may consist of multiple transaction service data units.

Table 3-2: WAP 1.2 negotiable connection capability parameters[18]

WAP 2.0

An important feature of the WAP 2.0 is the extended integration into Internet protocols. WAP 2.0 supports both protocol stacks. In addition to continued support for the WAP 1.2 stack, WAP 2.0 adopts International Engineering Task Force (IETF) specifications. On the transport layer, WAP 2.0 features TCP/IP for those networks capable of transporting data over IP. On the session layer, WAP 2.0 adopts HTTP/1.1 as a protocol. WAP 2.0 adds further security features, including the adoption of the transport layer security (TLS) protocol, to provide improved end-to-end security and integration with the wireline Internet security to enable secure use of mobile commerce, mobile banking applications, and service offerings. Furthermore, WAP 2.0 uses XHTML mobile profile (XHTMLMP) for content encoding. XHTML mobile profile is based on the modularity framework of the eXtensible Hypertext Markup Language (XHTML).

[18] Abbreviations are listed in the appendix.

WAP recommends special wireless options for TCP and HTTP 1.1. These are named Wireless Profiled HTTP (WP-TCP) and Wireless Profiled TCP (WP-HTTP).

Wireless Profiled HTTP [WAP229]

WP-HTTP specification is a profile of HTTP for the wireless environment and is fully interoperable with HTTP/1.1. The basic model of interaction between the WAP Device and WAP Proxy/WAP Server is the HTTP request/response transaction. WP-HTTP supports message body compression of responses and the establishment of secure tunnels.

Wireless Profiled TCP [WAP225]

The WP-TCP is a profile for TCP. It tunes TCP for wireless environments and is fully interoperable with standard TCP implementations in the Internet. Based on work by the IETF PILC group and other research for high delay, low bandwidth networks, a number of standard track RFCs for TCP exit. Table 3-3 lists these recommendations as they are referred to in the WP-TCP.

Items	Qualifier	Support level
Large window size based on BDP		SHOULD
Window Scale Option [RFC1323]	Window size >= 64KB	MUST
	Window size < 64Kbyte	SHOULD
Timestamps Option [RFC1323] for RTTM	Window size >= 64Kbyte	SHOULD
	Window size < 64Kbyte	MAY
Large Initial Window (cwnd<=2) [RFC2581]		MUST
Large Initial Window (cwnd>2) [RFC2414]		MAY
Selective Acknowledgement Option (SACK) [RFC2018]		MUST
Path MTU Discovery [RFC1191]		SHOULD
MTU larger than default IP MTU	Path MTU Discovery NOT Supported	MAY
Explicit Congestion Notification (ECN) [RFC2481]		MAY

Table 3-3: WP-TCP options [WAP225][19]

3.8.2 Multimedia Messaging Service

MMS is an evolution of the text messaging service SMS in GSM. MMS allows the sending of multimedia rich content, in a similar way like SMS, among mobile stations. The overall MMS architecture is described in [WAP205] [WAP206] and [TS23.140].

MMS utilizes the WAP and HTTP/TCP protocol stack with a split-architecture (Figure 3-11). The specific MMS communication takes place between the WAP client and the MMS proxy/relay host.

[19] Abbreviations area listed in the appendix.

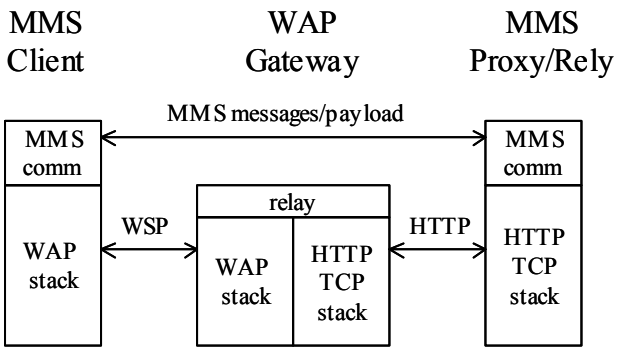


Figure 3-11: MMS protocol stack

MMS messages are created on the mobile stations, and sent to the MMS proxy/relay host. For this communication the WSP push message is used to transfer the MMS message to the WAP gateway, and subsequently HTTP post is used to transfer the message to the MMS proxy/relay.

The MMS proxy/relay notifies the recipient of the MMS message that an MMS message is pending for delivery. The MMS client can then request the MMS message from the MMS proxy/relay host for delivery. For this communication WSP between the client and the gateway and HTTP towards the server is used.

The MMS message content can be encoded either in WML or in Synchronized Multimedia Integration Language (SMIL). Today's terminals have limitations in the maximum size of the MMS message. Depending on the mobile station they are often in the order of 10 Kbyte to few hundred Kbyte.

3.9 Summary

This chapter explained important aspects of the GPRS network. It covered the network architecture, nodes, protocols and specific GPRS applications. We explained in detail steps in GPRS data transmission. It shows the influence of the deployed technique on the achievable performance and hence on the usage experience of the user. Important concepts we will frequently refer to in our analysis are the PDP context as well as GPRS mobility management. Our measurement setup bases in particular on the Gi interface and the logical APN concept. The novel applications WAP and MMS have been explained in greater detail, as we will show extensive results for them.

4 Measurement Setup and Traces

This chapter discusses our measurement approach in GPRS, which is one central contribution of this dissertation. In Section 4.1 we present our measurement setup. The measurement setup comprises the definition of the measurement points and the infrastructure for post-processing the captured data. Section 4.2 will provide an overview of the set of raw traces we obtained from our measurements. In section 4.3 we summarize this chapter.

4.1 GPRS measurement setup

In this section we present the overall measurement architecture. The specific Gi and SGSN measurements are explained in section 4.1.1 and 4.1.2. In section 4.1.3, we introduce in detail our toolset. We have incorporated existing tools and developed a large set of tools that allows us to transform the raw measurements into enhanced application level data. In particular we have developed a tool that allows us to correlate the traces from the Gi and the SGSN measurement point.

We now define our measurement setup, which is flexible enough to provides traces relevant for our studies in this dissertation and beyond. The objectives of the measurement setup are:

(a) Provide access to a large statistical data set comprising a representative user base;
(b) Treat privacy and confidentiality issues appropriately;
(c) Be non-intrusive to network operations;
(d) Provide IP packet-level traces for stochastic-process investigation;
(e) Provide flow/application level traces for application investigation and source traffic modeling;
(f) Provide GPRS specific context information for mobility and session management investigation;
(g) Allow correlation between point (d), (e) and (f)

Figure 4-1 depicts our measurement setup. It shows the network, together with the signaling and transmission plane protocols. The signaling plane is the protocol stack that is involved in out-band signaling by the GPRS mobility management (GMM) and session management (SM). The transmission plane comprises all protocols relevant for user IP packet transmission. By placing the measurement points inside the GPRS network on central nodes it is possible to fulfill all of the above listed objectives.

The two measurement points ‘GMM event measurement’ and ‘Gi – IP traffic measurement’ are pointed out with arrows. The Gi – IP traffic measurements are conducted on the Gi interface, as will be explained in detail in section 4.1.1. This relates to the IP protocol layer as marked by the gray shaded box in the

transmission plane. The GMM event measurements take place in the SGSN, and will be explained in detail in section 4.1.2. In the signaling plane this corresponds to the GMM/SM protocol layer, as is marked by the gray shaded box. As the interface and nodes are central for all user communication in GPRS we fulfill objective (a). We only use passive measurement methods to meet objective (c). Objective (b) will be satisfied as part of our post-procession procedure, and will be explained in section 4.1.3. Objective (d) and (e) can be fulfilled with the GI – IP traffic measurements, because they provide access on the user's IP packets. The GMM and SM functionality is terminated in the SGSN, which allows fulfillment of (f). And the parallel measurement of GMM events and Gi traffic is the prerequisite to accomplish objective (g). The measurements and the post-processing will be explained in detail next.

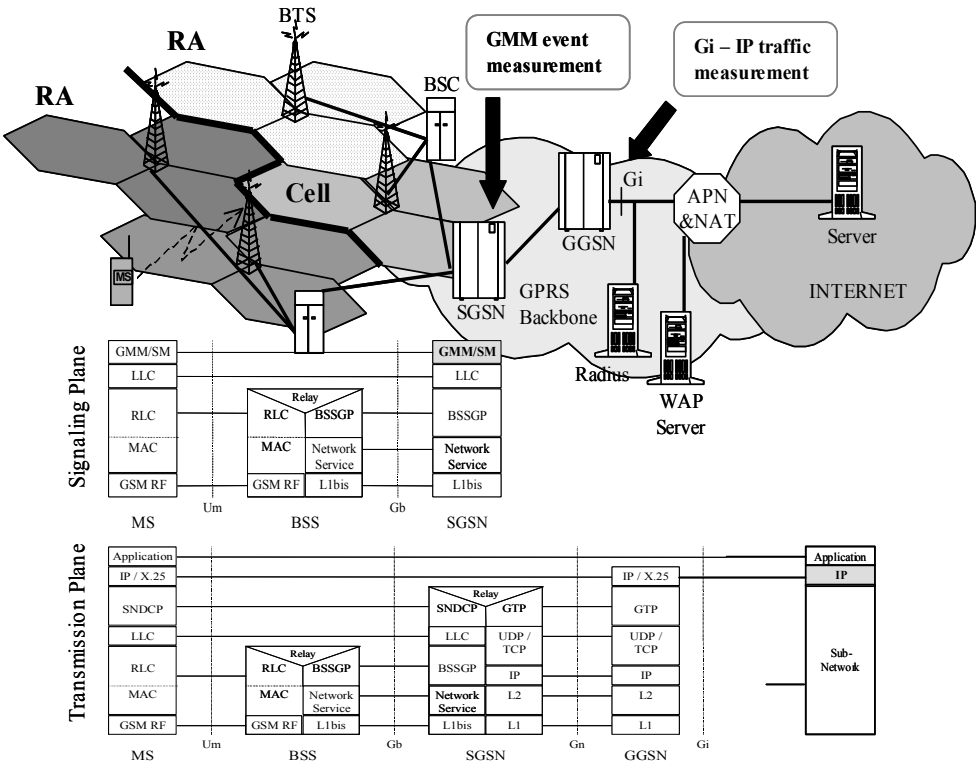


Figure 4-1: GPRS network and protocol stack, measurement setup

4.1.1 Gi measurements

In our measurements we deployed two different Gi measurement setups depending on the network topology of the operator. In all cases we have chosen Ethernet switches as measurement connection points, as this setup allows non-intrusive measurements. In the first setup we measured directly behind the Gi interface and before the NAT/APN box.[20] In the second setup we measured behind the NAT/APN box. That is, we measured at the point where the VPN is released into the target network. While in the first case we captured all APNs including traffic to the RADIUS server at once; in the latter case we had to measure various APNs and the RADIUS server separately at the same time.

[20] The NAT/APN box consists in fact of a number of nodes, connecting the various APNs (see chapter 3.5).

However, in both deployments we captured only a part of the total traffic in the network. Either we captured only the Gi interface at one of many GGSNs in the network, or we captured only some of many APNs in the network. As the GGSN are equally distributed, geographically and load-wise, we assume to capture a statistically representative subset of the total traffic in the network.[21]

We place a laptop, equipped with an Ethernet card, at the Gi interface and connect to the monitoring port of the Ethernet switch on the Gi interface. This allows capturing the packets off the network without disturbing the actual traffic flow. The laptop is equipped with a 30 Gbyte internal and 30 Gbyte removable hard disk. The laptop runs the Linux operating system and `TCPDUMP` is used as capturing software.

`TCPDUMP` takes a copy of the packets from the network interface and writes the first part of the packet, together with a time stamp, in a specific data file on the hard disk. Figure 4-2 depicts the packet part that we capture of the subnetwork (Ethernet). As explained in section 2.2, the first part of the packets is often enough to identify important application details. The traces contain the first 200 bytes of the link layer packet. That comprises the Link Layer Header (LLH), the IP header (IP-H), the Transport Layer Header (TCP-H) and parts of the Application Layer Header (Appl-H). Due to option fields in the IP-H and TCP-H as well as due to application dependent varying Appl-H length, a fixed packet capture length sometimes cuts of important parts of the Appl-H. However, in most cases the captured part of the packet contains all relevant information.

The arrows in Figure 4-2 mark where in the packet the information is stored to identify a service. That is, the IP header contains the address of the server and the transport layer protocol number. The transport layer header contains the port number, which identifies the application, and in the application header application specific information is stored.

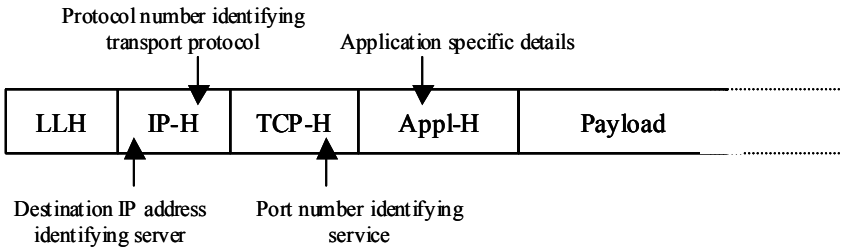


Figure 4-2: Captured packet header

The laptop is placed at the capturing point for several weeks. Regularly, the trace files are picked up and stored on a central server for post-processing. The central servers for post-processing are two Linux machines, interlinked with 1 Gbit/s Ethernet, and with almost 1 Terabyte of hard disk capacity.

[21] This assumption is based on the fact, that, for instance, the GGSN are all equally loaded with traffic and cover similar regions of the country.

4.1.2 GMM event measurements

The GMM/SM events we are interested in correspond to BSSGP protocol messages on the Gb interface. As one SGSN has many Gb interface, we have chosen to conduct the measurements directly inside the SGSN. That is, we do not actually capture BSSGP protocol messages, but we capture the events they trigger in the SGSN. Taking this approach we are not only bound to events triggered by the BSSGP messages, but we also capture events representing various internal events. For example if a timer in the SGSN triggers a GMM/SM state change, which is otherwise not represented by a BSSGP message, we are able to capture this.

With the support of Ericsson node-development engineers, we developed a proprietary tool for the SGSN that allows capturing all GMM/SM message correspondent events. (The gray-shaded protocol box indicates the logical protocol level at which we capture the events.) The capture tool logs in a proprietary raw format each GMM/SM event in the SGSN, together with a time stamp, an identifier for the user, and the location information of the involved terminal. The location information comprises the cell ID and the routing area ID. The SGSN knows this, as this information is available in each BSSGP message header.

The raw log files are regularly collected from the SGSN via remote operation and maintenance (O&M) connections and stored on the same central server for post-processing as used for the Gi measurements.

4.1.3 Post-processing tools

After the IP packets and GMM events have been captured, stored and collected at the central data server, a number of post-processing tools is used to extract the relevant data. Figure 4-3 depicts the processing chain from data capturing to specific statistical results, including most of our developed tools for this process. The left hand depicts the processing in the GPRS Network, on site; the right hand side depicts the processing which takes place after storing the data on the central data server.

Two different tracks are defined for capturing and post-processing Gi trace files and GMM log files. The top part in Figure 4-3 shows the Gi measurement track based on the freely available **TCPDUMP**. The raw data traces are directly anonymized for privacy reasons. The lower part shows the capturing of GMM events based on the proprietary event tracer. The GMM log file does not allow back tracking of the users identity, as only temporary user identities are stored.

Due to the anonymization of the Gi traces, even after matching both traces (as depicted by the dashed line), no subscriber identification is possible.

The most important tools, partly developed for the dissertation, partly based on freeware, and partly based on Ericsson proprietary software, are presented in the following. Section 4.1.3.1 introduces tools specific to the GI trace; section 4.1.3.2 lists the tools specific to the GMM log files; and section 4.1.3.3 provides an overview on general purpose tools used for many different results.

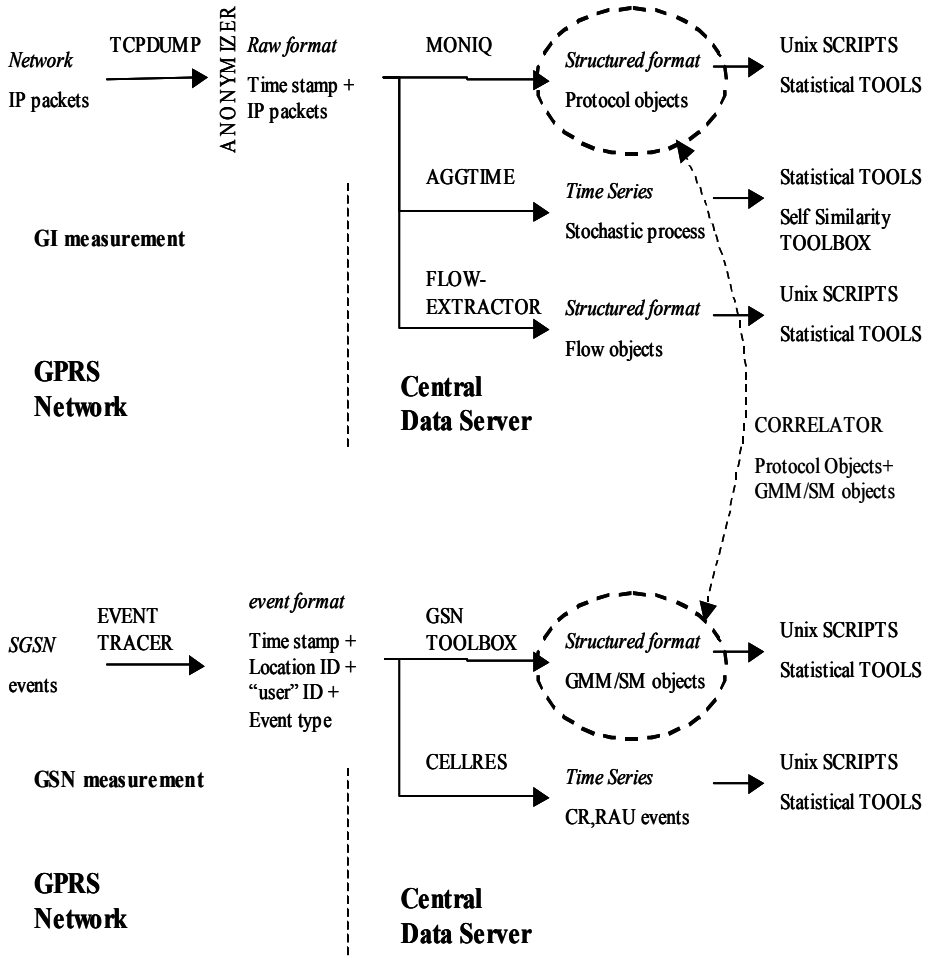


Figure 4-3: Post-processing tool chain

4.1.3.1 Gi capturing and post-processing

TCPDUMP

The `TCPDUMP` tool is an open source tool [TCPDUMP], available for many operating systems for capturing packets from a local network card.

The inputs are link layer packets (e.g., Ethernet frames) from the network card and the output is a truncated version of such packets. The storage format is a raw trace file, which sequentially stores all captured packets prefixed only by a time stamp. `TCPDUMP` allows filtering the incoming packets and captures only selected packets. This is an important feature, because it allows us to select individual APNs in our measurements, and limits the total data volume captures. `TCPDUMP` filtering is based on [MJ93]. `TCPDUMP` is not changing the packets while capturing them.

If the packets arrive too quickly for the processing speed of the card or the computer, `TCPDUMP` might drop packets. We logged these overflow events and had a typical dropping rate of 0-20 packets out of 1 million packets.[22]

[22] In the majority of measurements we had 0 packet drops reported.

TCPDUMP allows access on the raw packet trace and a condensed output. Figure 4-4 lists a short excerpt from the condensed output from one of our traces.

Timestamp	Source IP address – Port	Destination IP address – Port	Protocol	Length	Extra Info
1056974528.354548	145.7.38.46.2001	10.0.92.56.2000	udp	30	
1056974528.356839	145.7.38.46.2001	10.0.97.153.2000	udp	30	
1056974528.362961	10.65.99.156.8502	192.168.251.150.9201	udp	5	
1056974528.365254	10.65.100.74.8502	192.168.251.150.9201	udp	5	
1056974528.374013	24.209.8.110.1214	10.7.7.154.1488	tcp	1368	(DF)
1056974528.379167	199.245.28.196.80	10.65.100.5.4108	tcp	1380	
1056974528.380600	192.168.251.150.9201	10.65.93.164.8502	udp	350	
1056974528.383604	10.65.94.113.50017	192.168.251.150.9201	udp	5	
1056974528.387260	10.7.7.154.1488	24.209.8.110.1214	tcp	0	(DF)

Figure 4-4: TCPDUMP trace

Anonymizer

After capturing the packets, the **TCPDUMP** raw packet trace is anonymized. Several tools exist for this; for example **TCPurify** [TCPURIFY] and **TCPDPRIV** [TCPDPRIV], [PP03]. However, the main tool we used is an anonymization processor build into the **Moniq** tool (the **Moniq** tool will be explained below). The anonymization process randomizes the IP address as well as sensitive information in the captured part of the application header. The randomization is performed in a way that it is not possible to track back true identities of subscribers or application level information. However, the randomization allows the identification of unique users. This means that the new – randomized – identity remains the same throughout the tracing process. This allows performing long term statistical analysis of user behavior. The outcome of the anonymization process is again in the **TCPDUMP** trace file format.

Moniq

The **Moniq** tool is an Ericsson proprietary commercial tool[23] for traffic trace analysis at the application level. It extracts (application) protocol relevant information from the **TCPDUMP** trace files. The outcome of **Moniq** is a set of log files with condensed traffic information as explained below. The original packet content is not needed afterwards. The advantage of this approach is that the information stored is semantically more meaningful than the raw trace file and further that the storage space needed is only a fraction of the original trace file. However, details on the packet process are lost with this approach. Two primary log files are generated,[24] the *PDP session log* file and the *transaction log file*. A PDP session log file has one entry for each completed PDP context captured in the Gi trace. The information comprises, among others, the start time of the PDP context, the duration, number of transmitted bytes (up/downlink), the anonymized user ID, which applications have been used, number of transactions per applications, etc. Figure 4-5 depicts a short extract from the PDP session log file, showing some of the possible fields.

[23] Developed at Ericsson Hungary (ETH) and Ericsson business unit global services (BUGS).

[24] Moniq is capable of many more statistics and log files, which are not listed here.

Note, that the required PDP contexts start and stop information can be also extracted from the Gi traces because of the captured RADIUS packets. As can be seen from Figure 4-1, the RADIUS server is behind the Gi interface, which allows us to capture the RADIUS messages together with the user IP packets. As a PDP context activation also includes a RADIUS IP address assignment, and the deactivation includes a revoke of the address, it is possible to mark the start and the end of the PDP context within the Gi trace.

A transaction log file has one entry for each ‘transaction’ in the Gi trace. A transaction is defined based on the application data it conveys. In the case of TCP based applications (for instance HTTP, SMTP, POP3, IMAP, etc), one transaction corresponds one-to-one to a single TCP flow. A TCP flow, in this case called a transaction, starts with a TCP syn packet and ends with a TCP fin packet.

In the case of WAP and MMS, which are based on UDP, the transaction definition is based on the corresponding application objects. For example for WAP traffic, the transaction entry corresponds to fetching a WAP object, which is a ‘Get/Reply’ message exchange (see section 2.2.3.5). Hence, one WAP transaction comprises all UDP packets involved in this ‘Get/Reply’ transaction. In the case of MMS, one transaction comprises sending or receiving one MMS message. The definition of UDP transactions for other applications are based on heuristics.

Each transaction entry comprises, among others, the start time of the transaction, the PDP context it belongs to, the duration, the number of transmitted bytes, the port number, the identified application, the anonymized user id, etc. Figure 4-6 shows a short extract from the transaction log file, with some of the possible fields.

Timestamp	apn	IP addr	pdpid	duration	transaction	packet in	out	byte in	out
1056974861.624778	Web.	10.0.110.242	373	63.138898	3	31	28	2357	1603
1056974707.620499	Web.	10.0.110.205	216	218.438876	12	14	25	7835	1688
1056974917.840860	live.	10.65.100.177	417	9.423002	4	4	7	1801	1286
1056974773.891541	live.	10.65.86.188	289	153.515509	3	2	5	179	1112
1056974928.406265	live.	10.65.58.123	428	3.141010	3	2	5	151	244
1056974876.862154	live.	10.65.103.6	390	55.918557	9	12	17	7075	2069
1056974619.658486	Web.	10.0.110.191	127	313.166427	4	14	16	2797	917

Figure 4-5: PDP session log file

Timestamp	pdpid	app	IP address in	out	port in	out	protocol	duration	packet in	out	byte in	out
1056974811.770418	125	WAP_CO	10.65.101.28	192.168.251.150	50016	9201	17	8.914598	2	2	454	115
1056974819.651280	294	WAP_CO	10.65.95.30	192.168.251.150	50016	9201	17	1.087062	1	2	171	133
1056974815.060752	323	POP3	10.64.122.69	213.156.0.240	51717	110	6	5.709442	11	11	914	624
1056974819.483232	315	WAP_CO	10.65.102.186	192.168.251.150	8502	9201	17	1.315646	1	2	385	125
1056974806.838920	205	WAP_CO	10.65.101.79	192.168.251.150	8502	9201	17	13.964053	3	2	375	130
1056974819.639232	157	WAP_CO	10.65.101.255	192.168.251.150	8502	9201	17	1.284814	1	2	838	119
1056974820.291422	295	WAP_CO	10.65.100.152	192.168.251.150	8502	9201	17	0.860043	1	2	372	126
1056974811.438186	325	MMS_CO	10.65.98.219	192.168.251.150	8502	9201	17	9.799472	7	8	326	6558
1056974819.473530	313	WAP_CO	10.65.97.114	192.168.251.150	8502	9201	17	1.807010	1	2	928	139

Figure 4-6: Transaction log file

Timestamp	userid	event	cell-id
1042048342.160721	121	MM state change -> standby_reachable	17-1-5021
1042048568.052631	121	Suspend	17-1-5021
1042048644.841099	121	RA update to resume, old RA, local TLLI,intra-BSC	17-1-5023
1042048644.842447	121	Authentication and Ciphering	17-1-5023
1042048693.037314	121	Cell update, intra-BSC	17-1-33061
1042048726.653562	121	Cell update, intra-BSC	17-1-5022
1042048770.652684	121	MM state change -> standby_reachable	17-1-5022
1042049071.509531	121	Suspend	17-1-5022
1042049725.601350	121	RA update to resume, old RA, local TLLI, intra-BSC	17-1-5023
1042049725.602669	121	Authentication and Ciphering	17-1-5023

Figure 4-7: GSN log file

AGGTIME

We developed the **AGGTIME** tool for stochastic process analysis. The **AGGTIME** tool is a script taking the **TCPDUMP** raw trace as input and converts the information into a time series of some metric. The time series is a sequence of bins spanning over equidistant time periods, each filled with the metric for this time period; the metric is for example 'number of packets', 'number of bytes'.

Taking the IP packet trace as input, it is possible to generate the number of aggregated bytes or absolute number of packets transferred, in e.g., 100 milliseconds. The output of this tool is used in the analysis of the packet arrival process and byte arrival process with respect to its self-similarity features (chapter 8).

FLOW-EXTRACTOR

The **FLOW-EXTRACTOR** works similar to the **Moniq** tool but is specifically designed for WAP 2.0 traffic. We developed this tool for this thesis because **Moniq** could not handle WAP 2.0 at the time of the analysis.[25] We use this tool in our analysis in chapter 6 to extrapolate from WAP 1.2 transactions to WAP 2.0 transactions. The **FLOW-EXTRACTOR** identifies WAP 1.2 'Get/Reply' transactions in the **TCPDUMP** trace and extrapolates them to WAP 2.0 transactions. The extrapolation method will be explained, in detail, in chapter 6.

4.1.3.2 GMM event capturing and post-processing

EVENT-TRACER

The **EVENT-TRACER** is an Ericsson proprietary tool that was developed to trace GPRS events in SGSN nodes. We modified the tool to allow long term tracing periods.[26] The **EVENT-TRACER** runs as part of the operating system of the SGSN node and is notified in the case of particular GMM/SM events. The **EVENT-TRACER** writes one entry for each event to a trace file. Each entry comprises the time stamp of the occurrence, the mobile station ID, the type of the event and the location. The mobile station ID is unique only to the GSN. That is, the mobile station ID is unique in the trace file but has no direct relation to the user ID in, e.g., the **Moniq** log file. In case of PDP context activation events, the entry also contains the assigned IP address.

Since the **EVENT-TRACER** was running on a commercial node, great care had to be taken to not disturb the operation of the node. For this reason the **EVENT-TRACER** had lower priority than system functions and was limited in its resources. The trace duration was limited to 3 days. However, it was not possible to evaluate the number of missed events due to the lower prioritization of the capturing software. But, due to the general high capacity of the node and the limited load in the network, we expect that the impact from lower prioritization was low.

---

[25] More specifically, Moniq cannot extrapolate to WAP 2.0 traffic.

[26] The implementation and the modification were done by Jan Scheurich.

Figure 4-7 shows a sample raw output from the event tracer.

GSN toolbox

The **GSN toolbox** is a set of scripts we developed to post-process the output of the **EVENT-TRACER** specifically for the purpose of our investigation.[27] It has some similarities with the **Moniq** tool. It takes as input the trace file of GMM/SM events and converts this to a structured log file, representing logical GMM and SM objects. This are for example 'GPRS attach' session, or 'routing area updates'. Three log files are generated, the GPRS attach log file, the PDP context log file, and the ready log file.

Each GPRS attach-detach period, identified by the corresponding events, is listed as one entry in the GPRS attach log file. The entry comprises information on the start time, duration, number of enclosed PDP contexts, number of cell reselections and routing area updates, etc. The PDP context log file comprises one entry for each complete PDP context, comprising the start time, duration, IP address, number of ready states, number of cell reselections and ready states, etc. This entry is based on the PDP context activation, deactivation information from the GSN trace. And the ready log file comprises one entry for each ready state, again with similar correlated information. We do not exploit all the information stored in the log files, but in particular use the cell reselection and routing area update information for our general analysis in chapter 7.

CellRes

CellRes is a specific script we developed that extracts the mobility-relevant events from the raw GSN event trace file and provides a time series of individual cell reselection and routing area update events. It allows specific per user or per PDP context filtering. This time series is used for the cell reselection and routing area inter-arrival time investigations in chapter 7.

CORRELATOR

In our specific investigation on how mobility and application usage is correlated, in chapter 7.6, we need to combine the separately stored information on used application and mobility.

The **CORRELATOR** tool, which we developed for this purpose, allows the correlation of log files from the **Moniq** tool, which shows the used application, with log files from the GSN toolbox, which indicate the mobility. Most difficult in this context is that both capturing machines are not time synchronized. The clocks have an unknown offset as well different drifts. Furthermore, the user ID is not the same in both traces and the only identifier which can be used in both trace files, the IP address, is not unique assigned to one user, as it can be reassigned periodically. We developed a matching algorithm, based on the PDP context length together with the sequence of occurring IP addresses to match corresponding data entries. Figure 4-8 depicts two traces showing PDP

[27] The GSN toolbox including the CellRes and CORRELATOR concept for mobility analysis was developed as part of this thesis, and was implemented by Daniel Spelmezan.

contexts in which the order of Gi trace file entries matches the order of GMM log file entries. This can be used to relate the userid to the pdpid. The **CORRELATOR** tool applies this method to match the Gi traces with the GMM logs.

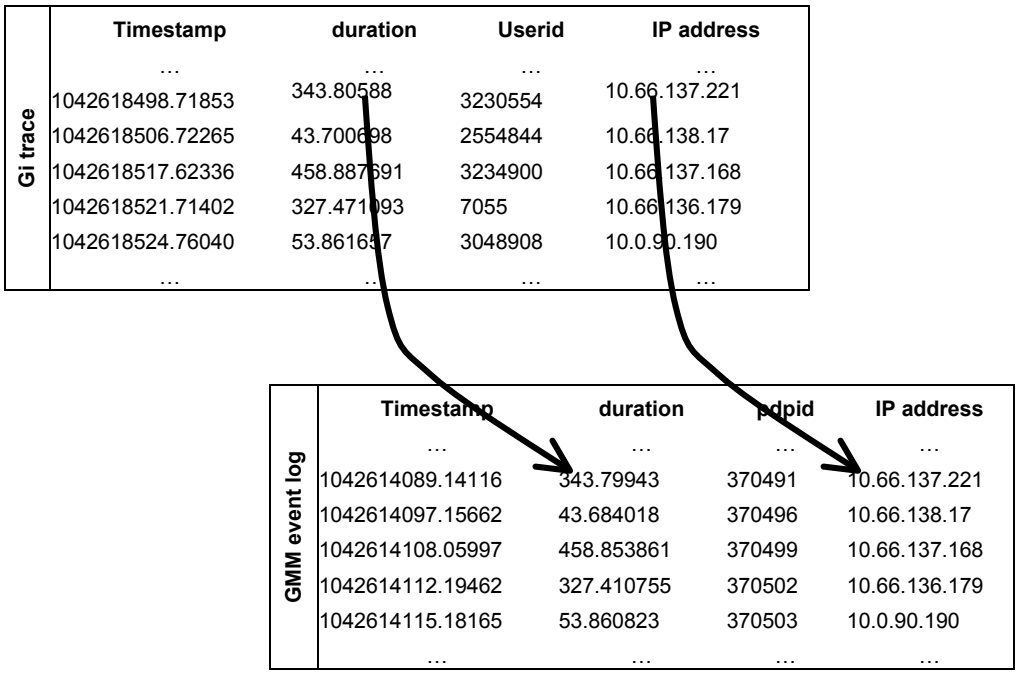


Figure 4-8: Gi trace matching GMM event log

4.1.3.3 General post-processing tools

Unix scripts toolbox

This comprises a great number of small UNIX scripts, based on **gawk**, **perl**, **bash**, etc., we developed for post-processing of the log files by the above mentioned tools. Often specific filtering and correlating of the log files are needed to derive a result.

Statistical tools

To support our work of statistical inferring we used several general-purpose statistical tools. In particular, a wide number of statistical methods (e.g., for distribution fitting, goodness of fit test, Hurst parameter estimation, etc.) are already implemented and ready to use with those tools. If possible we verified the implemented algorithms before applying in our analysis.

The three main tools we used are **Matlab** [MATLAB], a powerful general-purpose mathematic tool; **Xmgrace** [XMGRACE], a tool for applying simple statistical methods and visualizing thereof, especially for *very* large data sets; and the **DATAPLOT** [DATAPLOT] tool, which provides many methods for exploratory data analysis and data fitting techniques.

Self-similarity tools

The self-similarity toolbox is a synonym for two tools we used to investigate the packet time series on self-similarity features. These tools implement several Hurst parameter estimators. The term self-similarity has been defined in section 2.3.3.2 and we will introduce the different Hurst parameter estimator methods in section 8.1. We used the `SELFIS` tool [SELFIS] [KF02], which implements many of the well known Hurst parameter estimators; these are the R/S method, the variance method, the absolute moment method, the ratio of variance of residuals, the periodogram method, the Whittle estimator, and the wavelet (Abry-Veitch) method. The tool has its limitations, as it does not apply all methods in the most sensible way. As we will outline in section 8.1 the methods need special care when applied. Therefore we used the `SELFIS` tool only for deriving some intermediate statistics and manually applied the reminding part of the methods. The other tool we used is the Log scale Diagram (LD) estimate-code [`LDCODE`]. This tool was written by Darrel Veitch, one of the inventors of the Abry-Veitch method. The Abry-Veitch method is based on wavelets for estimating the degree of self-similarity of a stochastic process [AV98].

AESG tool

The `AESG` tool is used to estimate the heavy-tailedness of empirical distributions. It implements the scaling method by [CT99] and can be found at [AEST]. Heavy-tailedness describes a property of distributions in which the extreme tail slowly decays. We explain the scaling method in section 6.5.2.3 after defining heavy-tailedness more in detail.

PH-fit

The `PH-fit` tool is a special implementation of the expectation-maximization (EM-) algorithm to fit phase type (PH-) distributions to empirical distributions. The implemented method was provided by Rachid El Abdouni Khayari and is explained in [KSH03]. The properties of the PH-distributions together with the EM-fitting algorithm will be explained in section 6.5.2.2. We apply this method in our in-depth investigation of application flow length distributions (chapter 6) and cell reselection inter-arrival time distributions (chapter 7).

4.2 Measurement traces from commercial GPRS networks

The measurements for this dissertation have been conducted as part of a joint project activity of Vodafone and Ericsson. We used traces from three different European GPRS networks for the investigations in the following chapters.

For confidentiality reasons we cannot state the number of total subscribers and the absolute data volume in the networks. However, both, subscribers and absolute data volume, are significant quantities.

In all three networks we had the opportunity to measure at one of several Gi interfaces. That is, our traces reflect only a subset of the total data volume in

the respective network. As the Gi interfaces are typically distributed according to load and or geographical reasons, we are confident that we have measured a representative data set.

4.2.1 GI measurements

Table 4-1 lists the Gi trace files we used for the results in chapter 5 to chapter 7.

The letters A, B and C in the names correspond to the three different countries the traces are taken in.

Trace files GI_A8 and GI_A18 cover only WAP APNs. APNs for general Internet access (Web related) are not included.

Trace files GI_B7, GI_B8, GI_B10 cover WAP and Web related APNs.

Trace file GI_C7 covers WAP and Web related APNs.

Name	From	To	Hours	Days	Packets	Bytes
GI_A18	18/7/2003 18:15:18	25/8/2003 9:37:52	903.37	37.64	125.40M	55480.53M
GI_A8	19/8/2002 10:00:28	26/9/2002 8:34:03	910.55	37.94	22.02M	6894.68M
GI_B10	30/6/2003 14:01:31	11/7/2003 19:25:05	269.39	11.23	145.38M	52991.53M
GI_B8	29/4/2003 10:45:39	22/5/2003 17:27:10	558.69	23.28	209.63M	69134.54M
GI_B7	9/1/2003 12:03:04	6/2/2003 11:54:53	671.86	28.00	174.02M	53036.78M
GI_B4	27/09/2002 12:19:13	17/11/2002 17:35:02	1230.25	51.00	124.12M	43269.42M
GI_C7	2/9/2003 19:08:11	4/9/2003 22:12:15	51.06	2.128	190.75M	54383.54M

Table 4-1: Gi measurement data sets

For the specific self-similarity investigation in chapter 8 we have measured a larger data set, including more APNs in network A and network B. They are listed in Table 4-2. The additional APNs are related to corporate access points. The snap length of those traces was only 40 bytes, covering only time stamp, transport protocol type (TCP or UDP) and length of packet.

Name	From	To	Hours	Days	Packets	Bytes	Info
GI_A18b	2/8/2003 00:00:00	8/8/2003 23:59:59	168	7	21.23 M	10686.98 M	20% Corporate Traffic
GI_B10b	1/7/2003 00:00:00	11/7/2003 19:25:05	164	7	132.14 M	48768.32 M	46% Corporate Traffic

Table 4-2: Gi extra measurement data sets

GI_B10b is a subset of GI_B10 and GI_A18b is an enhanced subset of GI_A18 where in addition GPRS traffic from corporate APNs was captured.

4.2.2 GMM event measurements

Table 4-3 lists the GMM trace we used for the investigation in chapter 7. The GMM trace in fact consists of several smaller trace files, each covering 1 or 3 days. As we measured in commercial SGSN nodes, we had to give the

measurement lower priority and fewer resources. For the investigation we considered all traces, covering the time as indicated in the table. The column 'Events' lists the total number of events captured. This included many events related to GMM and SM. We need all these events to correctly construct the GPRS attach, PDP context and ready periods. Out of all the events, we extracted all mobility events indicating any kind of cell reselections (including routing area up-date and implicit cell changes.) The column 'Mobile events' lists the number of these mobility events. We have used the long traces over 3 days (cf. section 4.1.2) for our specific mobility investigation results presented in chapter 7, to limit the bias towards shorter time periods.

Info	From	To	Hours	Days	Events	Mobility events	Info
GMM_B5.6	08/01/2003 11:22:08	03/02/2003 14:15:04	840.03	35.01	50,84 M	6,44 M	non-continuous

Table 4-3: GMM measurement data sets

4.3 Summary

We presented in this chapter the measurement setup and the post-processing tools we used in our investigation.

The measurement setup is unique in that it allows capturing traces at two distinct places: at the Gi interface, which allows to capture all IP packets; and inside the SGSN, which allows to capture all GPRS session and mobility events.

Several investigations require specific tools, therefore the post-processing tool-set comprises a larger amount of freeware tools and proprietary developed tools.

The measurement traces are obtained from several commercial Vodafone GPRS networks throughout Europe, over the time frame autumn 2002 to autumn 2003.

5 GPRS Usage

This chapter provides an overview of the application usage in GPRS. It is our first step in understanding how GPRS is used. It sheds light on the traffic mixture that should be applied for traffic modeling. As we pointed out in section 2.3, most current traffic studies on wireless data networks extrapolate from wireline Internet measurements. This led to a traffic mixture with mainly Web, Email and some FTP traffic, while new mobile network applications like WAP and MMS are totally missing.

In particular, we discuss in this chapter several high-level traffic aspects. First, in section 5.1, we show the diurnal profiles for GPRS attach/detach, PDP context activations and application data volume. In section 5.2 we categorize GPRS subscribers into 4 classes, depending on the application they use and discuss in detail, which transport protocols and which applications are used. We will see that the high WAP and MMS usage breaks the dominance of TCP as transport protocol. We will pick up this circumstance later when we discuss the length of application flows in chapter 6. In section 5.3 we discuss the duration of PDP contexts. In that section we also investigate how PDP contexts are utilized and by which application. Those results are very useful when we investigate mobility in GPRS in section 7. We finish this chapter in section 5.4 with conclusions.

5.1 Diurnal usage profile

The diurnal profile is important when dimensioning the network capacity. In particular, it is necessary to know the busy hour load, as this determines the maximum capacity needed in the network. We show some representative results for GPRS.

The usage of GPRS follows a typical diurnal profile, which is known already from voice services in GSM as well as from wireline Internet data usage, with lower intensity at night and higher during the day.

The GMM event measurements can be used to show the diurnal profile for signaling traffic. Figure 5-1 depicts the amount of GPRS attach/detach events per time unit and the amount of PDP context activation/deactivations per time unit. The figure displays the number of events averaged over 10 minute intervals. The x-axis indicates the time of the day. The y-axis indicates the relative frequency of events. The frequency is relative to the total number of events, of a particular type. Further, based on the Gi measurements, Figure 5-2 depicts the diurnal profile of the data volume for a number of applications. The figure depicts the relative data volume per application over the day averaged over 10-minute intervals. In both diagrams a clear night/day profile is visible,

which can be accounted to the fact that we perform our measurements always in one single country in which people follow a certain day pattern.[28]

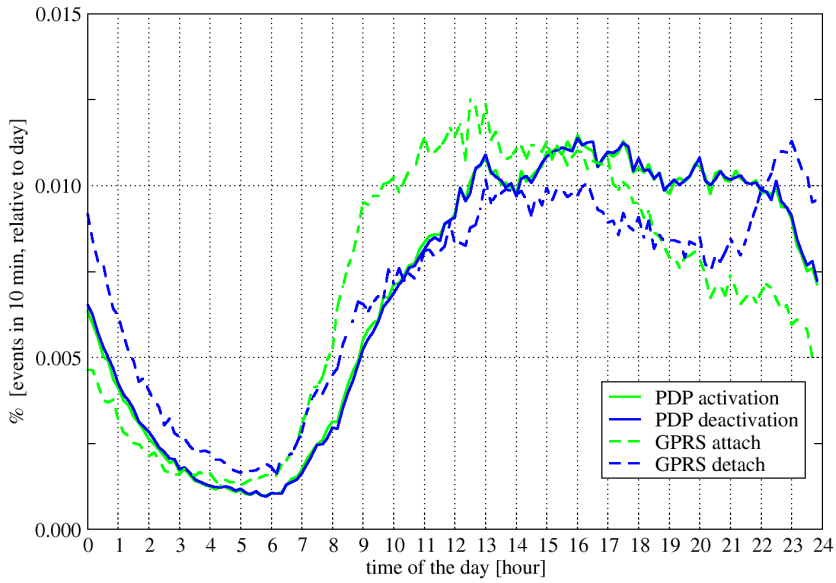


Figure 5-1: Diurnal GPRS attach and PDP context profile trace GMM5_6

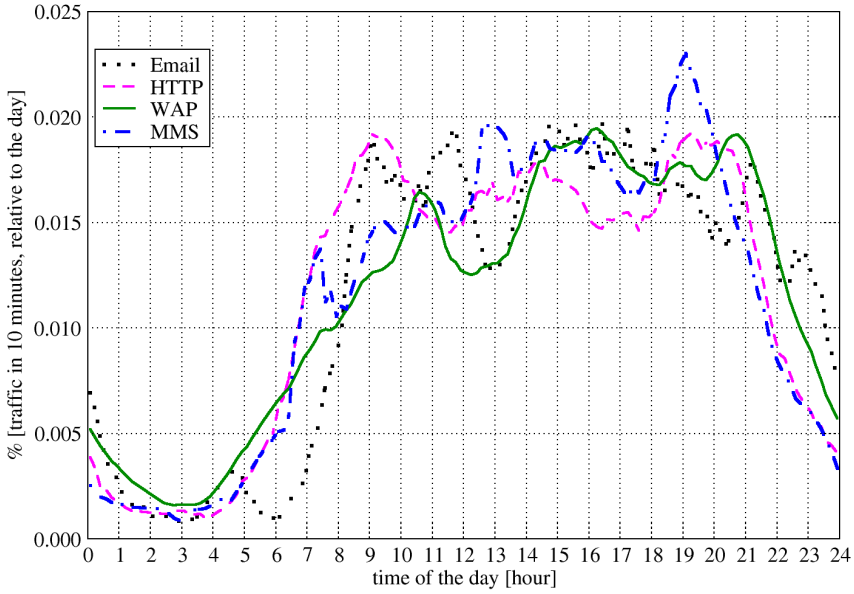


Figure 5-2: Diurnal application usage profile trace GI_B10

We can in particular observe that more attach events occur in the morning, while detach events occur more in the evening hours. These specific spikes in the GPRS attach/detach profile can be explained by the fact that many mobile stations issue a GPRS attach when being powered on and they are probably turned off in the evening hours, leading to a detach. But we do not see such a

[28] On the contrary, on long haul backbone links, the daily profile is not so pronounced, as they carry traffic by several countries from different time zones.

pronounced activity for the PDP context activations/deactivations over the day. The activation rate and deactivation rate is about the same per time unit for the whole daytime. The reason for this might be the short duration of the PDP contexts, as we will see in section 5.3.1.

Therefore, dimensioning the attach rate should be based on the busy peak between 10:00 and 12:00 and the detach rate should be based on the peak at 23:00. The PDP context activation deactivation rate is quite stable and could be dimensioned around 13:00 to 18:00.

Based on the Gi measurements the diurnal profile for application usage can be modeled. Interestingly, the diurnal profile of the data volume per application indicates a specific usage profile of different applications. Email usage starts at around 7.00-8.00 in the morning and remains quite constantly used over the day with a dip at lunch time between 12.30 and 13.30. WAP, HTTP and MMS start all about one hour earlier. HTTP has three peaks (busy hours). One in the early morning (8:00-9:00), one around lunchtime and one in the evening. WAP usage constantly rises until the afternoon, and is also used in the evening. This might indicate private usage, for example by students after school. MMS rises similar like WAP and has its peak in the evening.

Consequently, each application has its own diurnal profile to follow. That is, modeling the busy hour of the aggregated traffic in a GPRS network requires selecting a traffic mixture depending on the time of the day, and this might further depend on the network investigated as shown next. Our results provide only an indication, the exact absolute figures for dimensioning need to be derived by measurements.

5.2 Used applications in GPRS

We show next that the novel applications in GPRS (namely WAP and MMS) contribute significantly to the application and protocol mixture. That is, as we will also see in later chapters, all application and traffic statistics of GPRS are strongly influenced by these two applications.

5.2.1 Subscriber categories

In [KVWS03] we introduced a categorization of subscribers according to the applications they use. We group the subscribers of GPRS in one of four categories (Table 5-1), depending on which applications they used during the investigated tracing period. The categorization was based on assumptions we made about available features in the accessing mobile stations. Users in the *WAP user* category use only the WAP application. This category is motivated by the fact that all GPRS mobile stations have a built-in WAP browser nowadays. Many mobile stations provide even WAP, MMS, and Email clients. The latter combination is considered in the next category, which we call *GPRS user*, and it comprises users which use applications that are typically built into GPRS mobile stations. On the far end of the usage patterns are users of the category *Internet users*. These users use applications that are typically only accessible with the help of more powerful terminals, such as laptops. Users who used only Email also belong to this category. The last category of

advanced users comprises users who have been accessing all listed applications.

In order to achieve the categorization, we uniquely separate users based on their subscriber identity. For each subscriber we mark the type of applications used, considering a measurement period of 1 month.

Subscriber category	Applications					
	WAP	MMS	Email	Web	FTP	Other
WAP user	X					
GPRS user	X	X	X			
Internet user			X	X	X	X
Advanced user	X	X	X	X	X	X

Table 5-1: GPRS subscriber categories [KVWS03]

We show in Figure 5-3 four different results on subscriber categorizations, depending on the year of measurements and the network. The Pie charts (a) and (b) show the mixture for 2 different countries for about the same time period in 2002. The usage profiles differ significantly. We believe this is related to the very different marketing campaigns in the two countries at the time of the measurement. At the time of GI_A18, GPRS was advertised already as a consumer service,[29] supported by a large number of reasonably priced GPRS mobile stations. This is reflected in the high number of WAP users. On the contrary, at the time of GI_B4, GPRS was primarily advertised for corporate users, and supported by specific contracts for this. The usage in that network comprises many Internet and advanced users, which indicates that those users have been accessing GPRS mainly with their laptops.

At the end of 2002, both countries started the 'Vodafone live!' service, which is a service highlighting especially WAP and MMS usage and targeting consumer users. Pie chart (c) depicts the result for the same country as GI_B4, but after the introduction of 'Vodafone live!'. It is basically very similar to GI_A18, and demonstrates the influence of the marketing campaign. Chart (d) is from another country taken at about the same time as GI_B8, showing a similar categorization. We see this kind of subscriber distribution in later measurements in many countries.

We can assume that for the time present, a subscriber distribution like in chart (a), (c), and (d) will be predominant in most networks, as most GPRS operators focus on WAP and MMS services. However, these results also show the influence of external factors on the application usage. By changing the marketing campaign, the operator can significantly change the usage pattern.

[29] In this section we use the shortcut e.g., "GI_A18" to identify the trace as well as the network *and* the time period at which the trace (e.g., GI_A18) was taken. It is clear from the context what we refer to.

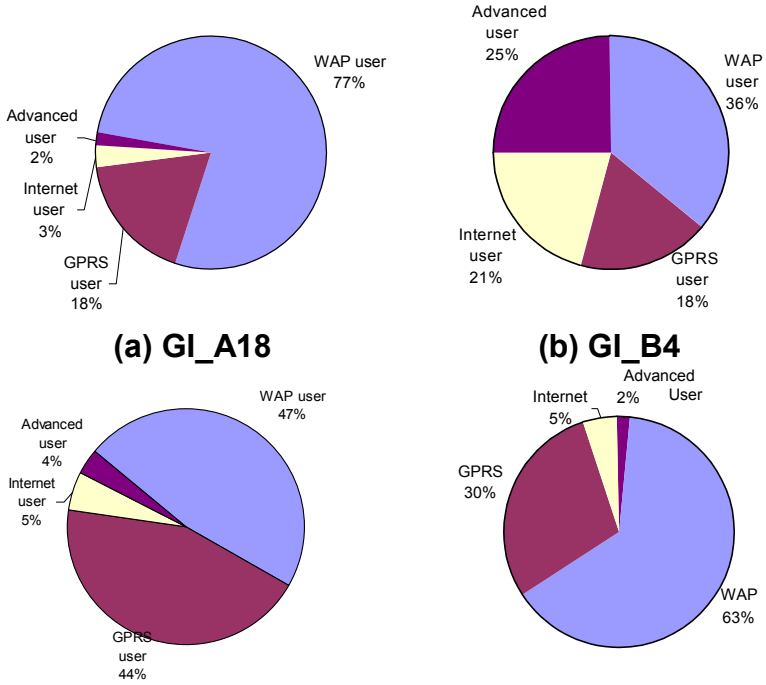


Figure 5-3: Subscriber profiling

Interesting to note is that in most cases the user uses only one application over the whole measurement period. Table 5-2 shows the fraction of users using one or more applications. The results are for the same periods as above results. The table lists the fraction of users that used 1 or more applications (up to 5 applications) in the measurement period. For each subscriber the number of different applications used is counted. The applications indicated might have been used separated in time. As one can see, 71.6% or 85.5% of the users use only one application type in about 4 weeks. This is not to be confused with the intensity of application usage. Although the accesses only one application type he or she might use it very frequently.

Number of Applications used	Subscriber Penetration GI-B8	Subscriber Penetration GI-C7
1	85.52%	71.61%
2	12.03%	27.27%
3	1.92%	0.94%
4	0.50%	0.17%
5	0.03%	0.01%

Table 5-2: Application usage by subscriber

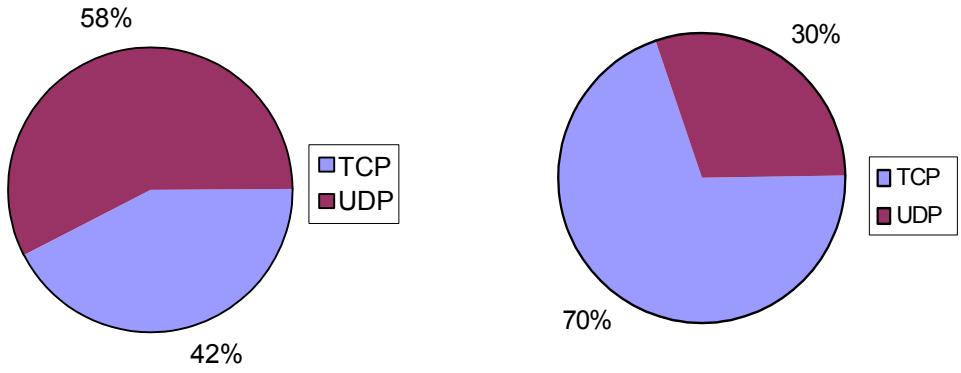
5.2.2 Protocol numbers

Results on the applied transport protocols are useful for modeling and dimensioning the network. Because, one fundamental difference between UDP and TCP is that TCP deploys a congestion avoidance mechanism, and UDP does not. Consequently, TCP reacts on the traffic situation in the network and adopts accordingly, while UDP does not. Hence, a high fraction of UDP traffic

makes congestion situations more critical, and furthermore might render active queue management useless [PJS00]. This should be considered when modeling the traffic mixture of the network.

Figure 5-4 depicts the relation of TCP and UDP traffic for two measured networks. The pie charts depict the fraction per transport protocol (TCP, UDP) relative to the total traffic volume.[30] The traffic in up and downlink direction is accumulated. We can see that in both countries the UDP fraction is very high compared to common figures for the wireline Internet. In the wireline Internet usually 90% of the traffic volume is assigned to TCP [TMW97], while we observe 30% or even almost 60% of UDP traffic. Figure 5-4 (a) shows the protocol mixture for a network, which exhibits a high WAP usage. Figure 5-4 (b), shows for about the same time results for a different network in which Web and Email is used to a higher degree. This points already to a reason behind the difference. We will show this in the next section.

In most measurements we observed a very similar UDP/TCP mixture as depicted in Figure 5-4. More recent measurements from late 2003, even exhibit an increase of UDP traffic. Therefore we can conclude that in current GPRS networks the amount of UDP traffic is significantly higher than in the current wireline Internet.



(a) GI-C7 **(b) GI-B10**

Figure 5-4: Transport protocol volume

5.2.3 Major applications

As the subscriber categorization and the high fraction of UDP traffic has already indicated, the application mixture in GPRS must be different to that in the wireline Internet.

As explained in section 2.2.3, the port number indicates the application generating the traffic. Therefore, we derive the application based on the TCP and UDP port numbers (on the server side).[31] We additionally check the used

[30] In some measurements IPsec was dominantly present. However as IPsec hides the transport protocol packets we could not consider these measurements. Therefore, we show only results for networks in which IPsec is not dominant.

[31] The server side port number refers to either the source port, if sent from the server, or the destination port, if sent to the server. The server side is identified by the IP address; if the IP address is outside of the IP address space for mobile terminals it must belong to the server.

applications with the help of the **Moniq** tool. This tool is not only relying on the port number but investigates the actual protocol on top of TCP and UDP and is therefore capable to countercheck the truly used application. This is in particular important for MMS, as MMS deploys WAP over UDP as transport mechanism and therefore cannot be identified by solely reading out the UDP port number.

For the major applications WAP, Web, Email, FTP,[32] and MMS we investigate their typical data volume share and the subscriber penetration. Table 5-3 shows results for three different networks. The data volume column gives the total fraction of up and downlink traffic for the particular application. The penetration column lists the fraction of subscribers that run the corresponding application at least once in the investigated time period.

The results stem from about the same time period of the year in three European networks. The measurements in network GI_A18 are different, as we measured only the WAP related APNs. Other APNs which, for instance, mainly carried Web and Email traffic were not accessible during the measurement period.

The high penetration numbers in the results show that most users have tried out WAP and MMS. But even so, only 4% of subscribers in one network and about 7% in the other network have used Web, Web contributes a considerable amount of the total traffic share. The reason for this is that Web generates much higher data volume per session than WAP does. In section 5.3.3 we show the average amount of data generated by the different applications.

In all networks, by taking measurements over a time span of one year, we could see an increase in the MMS traffic share.

Application	GI_A18		GI_B8		GI_C7	
	Volume Share	Penetration	Volume Share	Penetration	Volume Share	Penetration
WAP	47.5%	99.0%	40.5%	94.5%	36.6%	94.8%
MMS	39.9%	66.1%	2.0%	44.0%	6.3%	29.5%
Web	2.6%	0.4%	35.8%	7.1%	26.7%	3.8%
Email	0.4%	0.2%	8.0%	7.2%	7.8%	3.3%
FTP	0.1%	0.0%	0.7%	0.6%	0.5%	0.2%
Other	9.4%	7.8%	13.0%	34.2%	22.1%	10.5%

Table 5-3: Application volume and penetration in GPRS

Besides the dominant applications shown in Table 5-3, many more applications are used. Based just on the server port numbers the ‘extra’ applications can be grouped into ‘streaming’, ‘peer-to-peer’, ‘instant messaging’, ‘file transfer’, and ‘other’. In some networks as much as about 9% of peer-to-peer traffic was encountered. Also, streaming and file transfer like applications accumulated about 3%-4% of the total traffic volume in some measurement periods.

[32] We included FTP, not because it is a major GPRS application, but because it is commonly addressed to in wireline Internet measurement reports.

However those values are not sustained values over several measurement periods. But the high numbers indicate that GPRS can and will be also used for other, more traditional wireline applications. As the pricing scheme develops towards lower volume tariffs, we will probably see more of the applications with higher data intensity.

Currently, when building a traffic model for GPRS, the major applications considered should be WAP and Web. Furthermore, Email and MMS should be considered as they also contribute higher data volume percentages. Soon, MMS will probably belong to the dominant applications in GPRS. On the other hand, FTP-like applications are currently not much encountered.

5.3 PDP context usage

Application usage occurs inside of PDP contexts, and, as we will show in chapter 7, mobility can be tracked inside of PDP contexts. As we will need this understanding, we investigate the PDP context usage in this section.

5.3.1 PDP context duration

The PDP context duration is measured as the time between the first and the last RADIUS message, measured on the Gi interface. The RADIUS message results from a PDP context activation/deactivation in the SGSN/GGSN.

Figure 5-5 depicts the empirical CCDF for the duration of PDP contexts for one selected network measurement. Other network measurement results are similar.

In Table 5-4 we summarize the statistics for the same measurement set. As can be observed, the mean value is quite high, compared to the median value. We investigated several measurements, and the median has always been comparable to around 50 seconds, while the mean value was largely biased by the maximum values in the trace. We found no single maximum value for all traces, but the maximum value was heavily influenced by the measurement duration.

Important to note is that many PDP contexts are very short. 90 percent of the PDP contexts are about 11 minutes or less. Note that the sharp drop at the end in Figure 5-5 (approximately at $6x10^5$ seconds or about 7 days) does not result from the truncation of the measurement period, as this is much longer. The truncation might be due to settings in the network. Note, also the small dip around $7x10^4$-$8x10^4$ seconds or about one day – which is probably due to phones being switched off at the end of the day. The very short durations below 10 seconds might be due to automatic connection setups or broken application establishment attempts.

In the previous section we have shown that WAP, Web and MMS are the dominant applications in GPRS. Therefore we can assume that those applications are primarily responsible for the specific PDP context duration distribution. To countercheck this we investigated the correlation between the length of the PDP context and the used application types within the particular PDP context.

Statistic	[seconds]	Statistic	[seconds]
Mean	5635.76	**10%tille**	5.90201
Variance	2.29569e+09	**50%tille (median):**	47.2063
Std	47913.4	**90%tille**	690.24
Min	1.178436	**98%tille**	35399.7
Max	946148.53	**99%tille**	168267

Table 5-4: PDP context duration statistics trace GI_B10

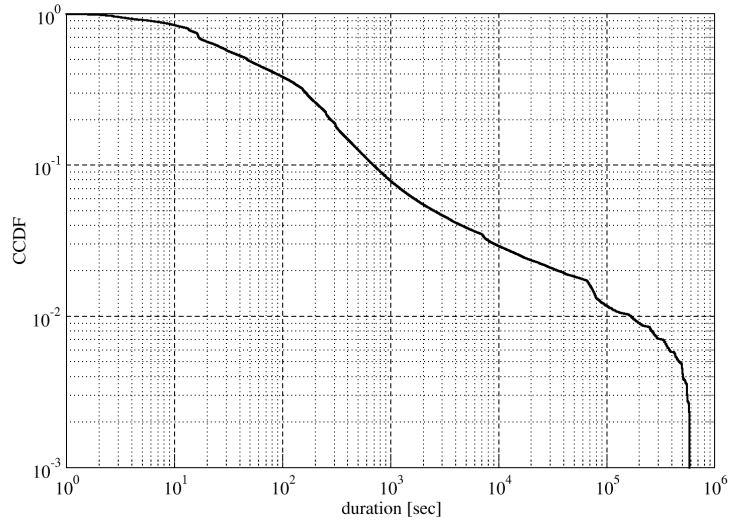


Figure 5-5: CCDF – PDP context duration trace GI_B10

For this we grouped the PDP contexts into groups of a certain length, and denoted them as PDP context length groups. For each PDP context length group we investigated which applications have been used. Figure 5-6 (a) depicts[33] this matrix between application and PDP context length normalized on PDP context length group. That is, we show for each PDP context group (x-axis) the distribution over different applications. Figure 5-6 (b) depicts the same scenario normalized on application type. That is we show for each application (x-axis), in which PDP context length group they are used.

Again we see the dominance of WAP traffic. Short PDP contexts are solely utilized by the WAP application. In the midrange-length of around 30 minutes to several hours, HTTP and Email start to play a stronger role. Surprisingly, very long PDP contexts are mainly utilized by WAP applications. However, depicting the same data with a per application point of view, Figure 5-6 (b) shows that WAP is still mainly used within short PDP contexts (48% of WAP flows are in PDP contexts shorter than 1 minute). The reason for this will be discussed in the next section.

[33] We specify in the figure only the upper boundary of the PDP context length group. That is a value of "<upper", specifies the group "lower≤x<upper", while "lower" is equivalent to "upper" of the previous smaller group.

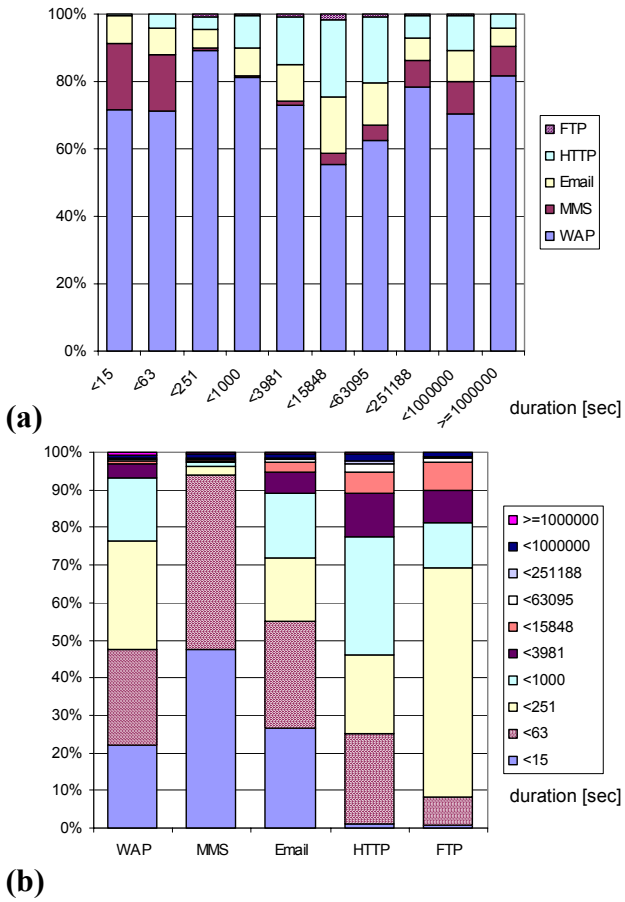


Figure 5-6: Application type versus PDP context duration

5.3.2 PDP context utilization

In this section we investigate how PDP contexts are utilized. By utilized we mean how much of the total PDP context time is used for data transfer, and where within the total PDP context duration the data transfer took place.

In Figure 5-7 we depict a density graph relating the length of all data transfer periods within a PDP context, to the total length of the PDP context, both in seconds. In the density graph, darker areas indicate more PDP contexts of a particular 'transfer length to PDP context length'-ratio. PDP contexts placed along the diagonal straight line, starting from the origin, would be 100% utilized. Right and down from the diagonal line represents less utilized PDP contexts. As indicated by the very dark area close to the diagonal line, the transfer period spans typically almost the whole PDP context period, especially for short PDP contexts. But the transfer period duration does not grow with the PDP context duration beyond 30-60 minutes. The graph shows to the right (more than 60 minutes) no PDP contexts with long transfer periods. In particular this result indicates that PDP contexts are used for the duration of the data transfer but to a smaller extent are 'always online'[34] connected to the network. That might be

[34] By "always online" we paraphrase the user behavior in which he or she stays connected to the network but is not intentionally transferring data. However, applications can constantly check, e.g., the server for Emails.

surprising as GPRS was designed in such a way to support ‘always online’ usage.[35]

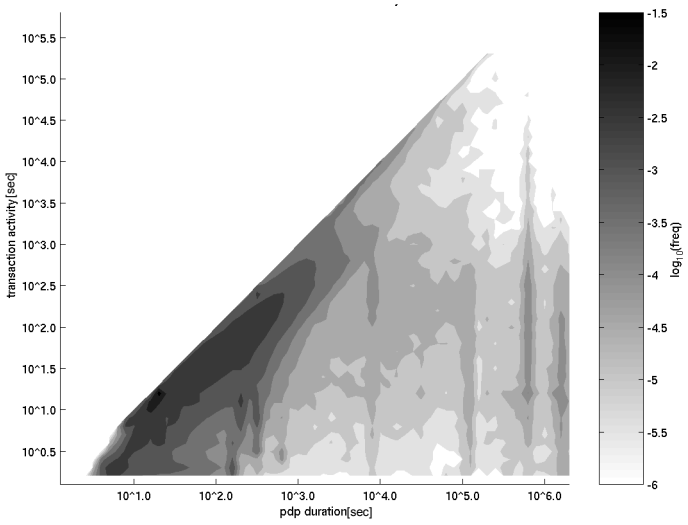


Figure 5-7: Activity phase duration vs. PDP context duration

Further refinement of this utilization investigation is to examine when the user is active during the total PDP context duration. In order to investigate the periods of utilization of the PDP context, we split the PDP context into equidistant sub-periods and count if data transfers took place within such sub-periods.

Figure 5-8 depicts the result of this investigation. On the x-axis, the sub-period of the PDP context is indicated by percentage portion. That is, depending on the PPD context length the sub-periods are longer. On the y-axis the already introduced PDP context length groups are indicated. That is, each PDP context falls into one of those groups. On the z-axis the amount of data transfers (normalized over the total length of the PDP context) that fall within a certain sub-period are indicated.

Figure 5-8 shows that most PDP contexts have an increased utilization phase at the beginning of the context. The only exception are very short PDP contexts, in which the protocol has to ramp up first. The peak at the beginning of the longer PDP contexts comes naturally as the PDP context activation is triggered by the need to transmit data. But this high, utilized phase drops quickly. Over the rest of the context duration, the activity phases are smoothly distributed. At the end of the context another peak indicates another – final – activity phase. We investigated this further and found that the reason for this lies in the WAP application usage of PDP contexts. This is because the WAP client on many terminals maintains a WSP session for the whole PDP context duration, and is terminated at the end when the PDP context is also terminated. This always results in a final transaction. Therefore we can state that in the PDP context group beyond 1 hour, the main utilization is centralized in the beginning of the PDP context.

[35] This might be influenced by a multitude of factors including terminal and network settings as well as tariff schemes.

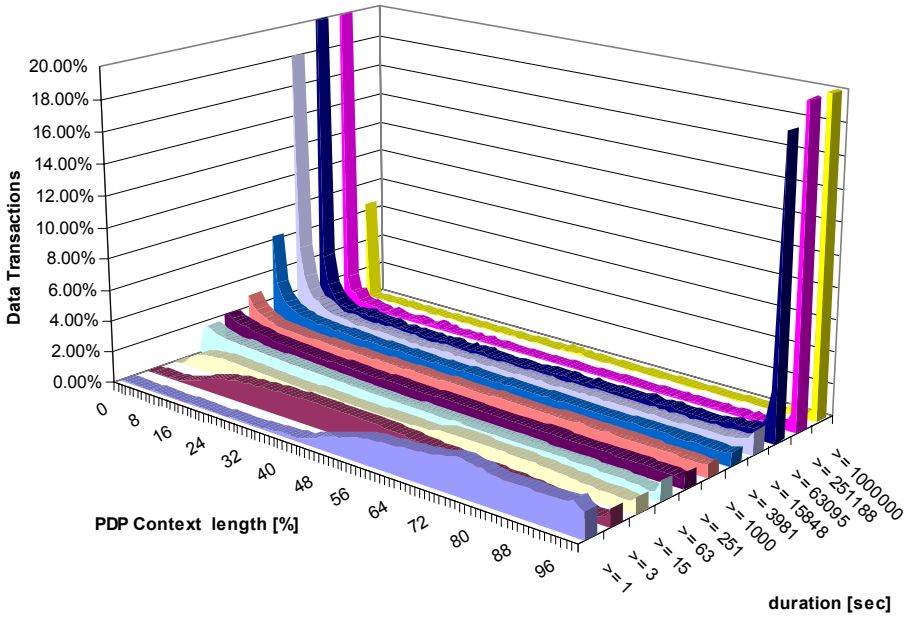


Figure 5-8: Data transfer periods in PDP contexts

Coming back to the previous section in which we were surprised by the frequent correlation between long PDP context durations and WAP usage, we can state that this is accompanied with a low utilization. And we can assume that the high utilization of the long PDP contexts by WAP is mainly due to non-terminated PDP contexts after usage through the build-in WAP-client in the mobile station.

Therefore, in modeling a PDP context, we have to consider the type of application. MMS and WAP applications are mainly embraced by short PDP contexts. Medium and long PDP contexts (around 1 hour) are used by Email and HTTP applications. Very long PDP contexts are mainly utilized at their beginning.

5.3.3 Application usage characteristics

In this section we finish our high-level view on GPRS by providing a very brief overview of the average duration and byte transfer per application session. An application session is a subpart of a PDP context. We define an application session as the continuous usage of one application by one user.

The start of the application session is marked by the time of the first application packet. The end of the application session is based on a heuristic. If the idle time between application packets is larger than 1 hour, the application session is considered to be finished. The time between the first and the last packet is the duration of the application session, and all packets in this session contribute to the data volume transferred by the application. We show the separate results for the up and downlink direction.

The exact numbers vary with the investigated network and time period. However, we see that in most networks the data volume and duration per application session stay in a certain range. For example, the average

downloaded data volume per WAP session lies in the range of 5 to 30 Kbyte. MMS sessions lie in a similar range. The average data volume per Web sessions lies in the range of 100 to 500 Kbyte.

Application	GI-A18			GI-B8			GI-C7		
		Data volume			Data volume			Data volume	
	Duration [minutes]	Up [bytes]	Down [bytes]	Duration [minutes]	Up [bytes]	Down [bytes]	Duration [minutes]	Up [bytes]	Down [bytes]
WAP	1.32	1976	6981	3.59	3891	24185	3.18	5214	19899
MMS	0.39	6878	6480	0.34	7791	2993	0.44	12205	2839
Web	18.9	30408	100141	9.15	28861	170540	16.24	126055	322551
Email	2.09	7576	29891	4.9	13420	47602	6.56	26090	65801
FTP	10.8	151158	597464	7.04	52334	166285	35.61	75353	238994
Other	8.43	16066	40136	20.45	12571	39194	11.43	45011	68550

Table 5-5: Average application usage

As expected, the results show that WAP applications are typically short. The amount of uplink and downlink data volume is much less than for Web applications. In chapter 6 we investigate this result in more detail along with the length of transport protocol flows.

5.4 Conclusion

Based on extensive measurements in GPRS, we showed a few representative statistics on GPRS usage, which can be used to select the appropriated traffic mixture in further analyses. However, we are aware that the traffic mixture is not a fixed result; many aspects influence this. Therefore the traffic mixture must be updated constantly with new measurements and extrapolated into the future, based on forecasts.

Specifically, we have shown that a traffic mixture for GPRS networks consists of WAP, Web and increasingly of MMS traffic. The new applications WAP and MMS change the protocol mixture towards a higher fraction of UDP traffic. This must be considered in traffic engineering of GPRS networks, especially as UDP (WAP/MMS) currently has no congestion control. In particular, we will pick up the dominance of WAP traffic later when we discuss the length of application flows in chapter 6. Though the protocol mixture might change towards TCP[36] soon, the other statistics presented here might not change quickly. We have shown in our subscriber categorization that the majority of users accesses only one application (WAP) and few use several applications. As the categorization heavily depends on the network investigated, our categorization can be seen only as a first step. Deeper insight might be gained at this level with additional measurements and advanced data mining methods like clustering. This is left for further research.

We also investigated how PDP contexts are utilized and by which application. Those results are very useful when we investigate mobility in GPRS in section 7. We have shown that the majority of PDP contexts, which form the equivalent

[36] Reason for this could be the introduction of WAP 2.0.

to dial-in sessions in fixed networks, are extremely short. The length of PDP contexts strongly depends on the used application, and is typically only used for a short initial time period. Especially, long PDP contexts are not utilized by application data. Therefore, modeling PDP contexts should be done by modeling application usage. That is, by modeling the application usage, the duration of the PDP context is a natural result of all transmission periods. However, independently, long lasting PDP contexts without application usage, can be of concern for an operator as they also bind resources. If this is of interest the PDP context needs to be modeled separately.

6 Application Flow Lengths

In this chapter we study the length of application flows in GPRS. An example would be an HTTP application flow which is identified as a TCP connection carrying HTTP messages. Investigating such flows are means to look at application objects at transport protocol level, without dissecting the actual application objects. For example, wireline Internet measurements repeatedly show heavy-tailed file length distributions. As the TCP flow length in many cases directly corresponds to the file length or a sum of file length, we are interested if TCP flows in GPRS as well have a heavy-tailed distribution. In section 6.1 we motivate the usefulness of the flow length results. Section 6.2 provides a definition of flows used in our investigation. The flow statistics per application and transport protocol are presented in section 6.3. Next, in section 6.4, we discuss the orientation of flows. That is, we show in which direction the data packets and in which direction the acknowledgment packets are transmitted. It is important to differentiate this, since the direction of the data packets is determining the TCP behavior. Section 6.5 provides an overview of the fitting techniques we apply in subsequent sections. In section 6.6 we show results on the lengths of flows. We will first study it along the line of the traditional view on the tail of length distribution. In this context we also investigate the so-called 'mice and elephant' metaphor for GPRS. Next, we provide results on the body of the length distribution, and in particular focus on the flow length counted in packets, as this is critical for the TCP performance. Section 6.7 concludes the flow length investigation.

Important to point out, we investigate WAP 1.2 flows, and as WAP 2.0 will replace WAP 1.2, we also investigate a future scenario in which WAP 2.0 replaces the current WAP 1.2.

The results in this chapter are based on measurement set GI_B10. In particular the results have been counterchecked with Gi_B7 and Gi_C7, which confirmed the conclusions drawn.

6.1 Motivation

When talking about the heavy-tailedness of file lengths, one is only concerned with the tail of the distribution. We listed in section 2.3.3.2 why knowing heavy-tailedness is important when modeling traffic. While the heavy-tailedness has important implications when it comes to network performance, the distribution body, which is often neglected in investigations of this kind, can have an impact on the performance of the network as well. In particular, TCP can be heavily affected by the flows belonging to the body of the distribution. Therefore we investigate the flow length counted in bytes, with a focus on the (heavy-) tail, and additionally we investigate the flow length counted in packets, with a focus on the body of the distribution. Having pointed out already why the tail is

important, we elaborate next more on the impact of the body of the flow distribution.

One use of results on flow length is to understand the typical scenarios in which Internet protocols are utilized. Based upon this understanding, research can be focused on appropriate protocol optimizations that improve the protocol performance in such typical scenarios. As we will show, the majority of flows in GPRS is very short. This is especially of interest in the context of TCP flows and their performance. Many of the advanced loss recovery schemes (being) standardized for TCP are only triggered if the TCP sender has a certain amount of packets outstanding or if the TCP sender is not application-limited, i.e., the application is still generating more data to transmit. Examples of such loss recovery schemes include fast retransmit [RFC2581], limited transmit [RFC3042], and SACK-based loss recovery [RFC3517] [RFC2018]. However, if the flows are too short to trigger these advanced loss recovery schemes, the TCP sender instead has to rely on its retransmission timer for loss recovery. Due to the relatively high Round-Trip Timer (RTT) values often found in wide-area wireless networks, inaccuracies of the current Retransmit TimeOut value (RTO) calculation [RFC2988] as outlined in [EL04], and the conservative minimum RTO of 1 second [RFC2988], this dependency on the retransmission timer can degrade the performance perceived by the end user. Hence, in many cases, a loss will have to be recovered through a costly TCP timeout. The shorter the flows, the less likely it is that TCP can recover a packet loss using its advanced loss recovery schemes. Especially critical for the TCP performance is a certain minimum length of 7 packets [AA02]. If a flow consists of fewer packets, there is potentially a risk that it solely relies on its retransmission timer for loss recovery.

Though short flows are not in general new, as they have been already reported in wireline Internet measurements [CBP95], we show that the percentage of short flows is even higher in GPRS and in particular absolutely dominant for the WAP application. While from a network-engineering point of view it is important to consider the impact of heavy tails, the body is important to consider for the perceived performance by the end user. We believe the end user might be more concerned in how the majority of his WAP transfers perform than how some seldom occurring long file transfers perform. Concretely, if TCP is not reacting appropriately on packet loss in short flows, whereas about 80-99% of the flows in the most used application (WAP) are below the critical length, this could well be negatively visible to the end user.

6.2 Flow definitions

In this section we define the flows for our investigation. Based on our tools **FLOW-EXTRACTOR** and **Moniq** we are able to identify different applications and their objects.

A flow is in the most general sense a sequence of packets that are related by the application object they carry. For instance, a TCP connection carrying one HTTP Get/Reply transaction represents a flow according to our definition. A flow can bi-directionally send packets; therefore it can be described by the number of bytes and the number of packets transferred in either direction. As

we see one potential usage of our results in modeling TCP flows for performance investigations and protocol optimization, our interest lies in the length of the flows in the direction of the object data transfer. Therefore, we define the length of flows as the length in the direction of the object data transfer. We assume that the opposite direction is mainly constituted by protocol acknowledgements, which are only reactive to the data side. In section 6.4, we show the direction for the majority of flows for different applications, which further motivates this approach.

TCP flow

A *TCP flow* corresponds to a TCP connection (i.e. the quadruple IP address and TCP port for source and destination). The start of the flow goes with the TCP SYN segment and the TCP FIN segment marks the end of the flow.

We define the length of the flow as the number of packets or bytes transmitted in the direction of the application object.

We use a heuristic to choose the transmission direction of the application object. Although TCP is bi-directional, a connection is typically used predominantly either for uploading or downloading data. In such a case the TCP connection carries data packets in one direction and acknowledgments in the reverse direction. Acknowledgement segments typically have only a size of 40 bytes (IP/TCP header), whereas the segments with payload are larger than the 40 bytes. Therefore, we measure the length of a TCP flow in both directions and define the length of the TCP flow as the number of bytes transferred in the direction which carried more bytes during the flow. Using this heuristic, we most likely choose the direction of the TCP flow in which the data is flowing.

HTTP Flow

An *HTTP flow* is defined by a TCP flow carrying HTTP application data.

All TCP flows which use the HTTP ports 80 and port 8080 on the server side[37] *and* carry HTTP Get/Reply or HTTP Post/Reply messages[38] are considered to be HTTP flows.

In HTTP 1.0, one HTTP flow corresponds to one Get/Reply or Post/Reply message. In HTTP 1.1, with pipelining, several Get/Reply or Post/Reply messages can be carried in one HTTP flow. As we are interested in the flow length that TCP has to cope with, it is not necessary to differentiate these cases.

We measure the length of the HTTP flow in terms of transmitted data bytes or packets. Again we use the same heuristic as for TCP flows to select the direction of the segments with the payload.

---

[37] The server side port number refers to either the source port, if send from the server, or the destination port, if sent to the server. The server side is identified by the IP address, if the IP address outside of the mobile terminals IP address space it must belong to the server.

[38] In correspondence with section 2.2.3.1, we call HTTP request messages GET and response messages REPLY. A POST message is a message sent with user data from the client to the server.

We use **Moniq** to investigate HTTP flows. In the transaction log file each TCP connection has a separate entry, which also includes the type of application used.

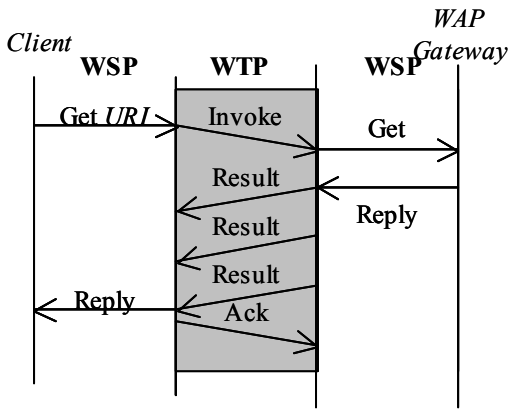


Figure 6-1: WAP 1.2 flow

UDP flow

A *UDP flow* is defined by UDP packets exchanged in one UDP 'connection' (i.e. the quadruple IP address and UDP port for source and destination; cf. section 2.2.3). But, since UDP is not connection oriented we need to use a heuristic in some cases to separate UDP flows. In the case of WAP over UDP, the exact flow length is defined by the WAP flow length, see below. In the case of other *unknown* applications, we separate the flows based on the following heuristic: if the time between packets is larger than 60 seconds, a new flow starts. [CBP95] and [ZBPS02] propose various threshold values to separate flows, ranging from a few milliseconds, to 1 hour. We found that 60 seconds is appropriated for our measurements.

WAP 1.2 flow

A *WAP 1.2 flow* is defined by a UDP connection that carries one WAP object, e.g., a WML page or an embedded image.

Figure 6-1 depicts the packet flow diagram of one WAP 1.2 object. This is performed by a WSP Get/Reply or WSP Post/Reply transaction. WAP uses WTP class 2 methods instead of TCP. In case one of the messages (e.g., Reply) is larger than the Maximum Transfer Unit (MTU) of the connection, the message is segmented and transmitted via several WTP result packets (as indicated in Figure 6-1). Each WTP packet corresponds to an IP packet in our measurement trace. No pipelining, i.e., the technique whereby multiple application-layer objects may be transferred on a single transport-layer connection, is defined for WAP 1.2. Hence, each *WAP 1.2 flow* consists of exactly one WSP Get/Reply or WSP Post/Reply object download (depicted by the gray-shaded area).

Again, we are interested in the flow length in terms of transmitted data bytes or packets. Therefore, we chose the flow direction with more bytes transferred to

determine the WAP 1.2 flow length. We use the **`Moniq`** tool to derive the WAP 1.2 flow statistics.

With the exception of section 6.3, we exclude WAP 1.2 flows that carry MMS messages from the considered set of WAP 1.2 flows. WAP flows with MMS messages are considered separately as *MMS flows*.

WAP 2.0 flow

We are interested in the length of *WAP 2.0 flows* but have only WAP 1.2 flows in our traces. Therefore we extend our WAP 1.2 flow definition to extrapolate to WAP 2.0 flows.

Our extrapolation is based on the assumption that the WAP content structure (WML pages and embedded objects) will be similar for WAP 2.0. The only difference is that in WAP 2.0, WSP is replaced by WP-HTTP and WTP/UDP is replaced by WP-TCP.

We assume that one WAP 1.2 WSP Get/Reply transaction will correspond to a HTTP Get/Reply transaction if WAP 2.0 is used. Furthermore, we assume that in case WTP segmented a WSP Reply message, also TCP would segment the HTTP Reply message into the same number of IP packets.

The WAP 2.0 standard does not explicitly require pipelining for WP-HTTP1.1, but we assume that WP-HTTP in WAP 2.0 will deploy pipelining, i.e. the request and transfer of several WAP message transfers can be carried out using one single TCP connection. We assume that this will be a standard implementation case; otherwise each message transaction would open up a separate TCP connection.

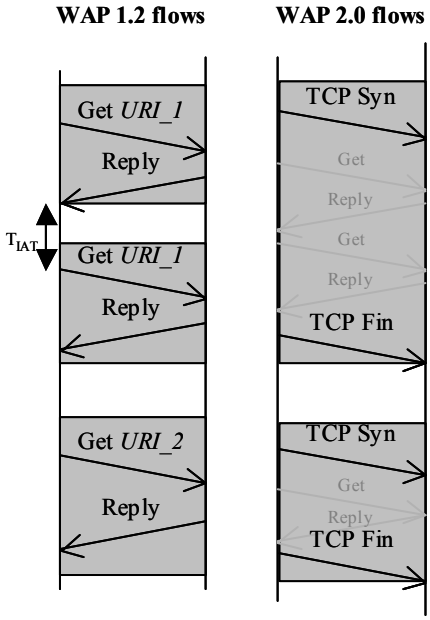


Figure 6-2: WAP 2.0 flows

Based on this assumption we define a WAP 2.0 flow in our measurements as a sequence of WAP 1.2 flows carrying one or several WAP objects (e.g., a WML

page or an embedded image) to or from *one server*. A new WAP 2.0 flow starts if either an object from a different server is requested or if the inter-arrival time between the last acknowledgement of the previous WAP 1.2 flow and the next Get of the succeeding WAP 1.2 flow is beyond a – heuristically selected – time-out value T_{IAT}. We denote those flows WAP 2.0 flows (T_{IAT} sec) (Figure 6-2, right, shaded areas). The WAP 2.0 flow length is the sum of all Reply messages.

We give an example in Figure 6-2. On the left-hand side three WAP 1.2 flows, to two different servers (URI_1 and URI_2) are depicted. We assume that the first two WAP 1.2 flows are closer together than T_{IAT}, and hence both are combined to one WAP 2.0 flow based on our heuristic. The next WAP 1.2 flow is directed to a different server (URI_2) and hence translates into another WAP 2.0 flow.

HTTP 1.1 [RFC2616] does not recommend an explicit TCP persistence value, i.e., the value after which a server closes a TCP connection to a host, for HTTP pipelining, which we could use for T_{IAT}.

Therefore we have chosen 3 different values for T_{IAT}: 3 seconds, 15 seconds and 60 seconds. The first value of 3 seconds is based on the average inter-arrival time (IAT) between subsequent Get messages. In our measurements and the authors in [VHS04] found that the average time between subsequent Get messages for embedded objects of the same WAP page are around 2 seconds. Therefore we have chosen 3 seconds to combine WAP 1.2 flows belonging to the same WAP page into one WAP 2.0 flow. The value of 15 seconds stems from the default value for persistent TCP connections in current Apache Web servers [APACHE]. We assume that this would be also the default value for WAP 2.0 servers if pipelining is used. The value of 60 seconds is taken from [NGBS+97], which is the original work advocating pipelining to improve HTTP performance.

The proposed extrapolation method cannot be handled with the **Moniq** tool. We therefore developed the **FLOW-EXTRACTOR** tool, which works directly on the IP traces to derive the extrapolated WAP 2.0 statistics.

MMS flow

An *MMS flow* is a WAP 1.2 flow in which case the object is an MMS message. The content type of MMS is indicated in the WSP message. **Moniq** uses this information to mark the flows accordingly in the log files.

The length of the MMS flow is defined by the length of the MMS message.

6.3 Application and protocol statistics

In this section we investigate the mixture of type of application flows.

We calculate the fraction of flows by taking the total sum of all flows and separate this into flows according to the different transport protocols and applications.

The transport protocol statistics for the captured GPRS traffic looks quite different compared to corresponding statistics from the wireline Internet. This could be already assumed based on the application statistics in section 5.3.3. In our measurement trace, UDP contributes to 30% of the transferred bytes. Based on our flow definition we have about 80-90% of UDP flows in the same measurement trace, which is considerably higher than what is commonly reported from measurements carried out in the wireline Internet. For comparison, [TMW97] reports 95% of all bytes, 85%-90% of all packets, and 70%-75% of all flows as belonging to TCP. Figure 6-3 depicts the distribution of transport protocol and application protocol on all flows in our GPRS trace. WAP currently contributes most of the flows, which accounts mainly for the large number of UDP flows.[39]

When WAP 2.0 is introduced, the UDP flows carrying WAP will turn into TCP flows. In particular, using our definition of a WAP 2.0 flow, the fraction of WAP flows would be reduced, since one WAP 2.0 flow may contain several WAP 1.2 flows, as follows. The fraction of WAP flows in GPRS (cf. Figure 6-4) would change to 61% (T_{IAT} =3 sec), 52% (T_{IAT} =15 sec) and 47% (T_{IAT} =60 sec), and, accordingly, the fraction of HTTP flows is increased to 24%, 30% and 33% respectively.[40] Hence, even applying the WAP 2.0 flow definition, WAP still remains the dominant flow type. Note, that we focus only on the absolute number of flows. If we focus on the data volume per application type, nothing would change by extrapolating to WAP 2.0.

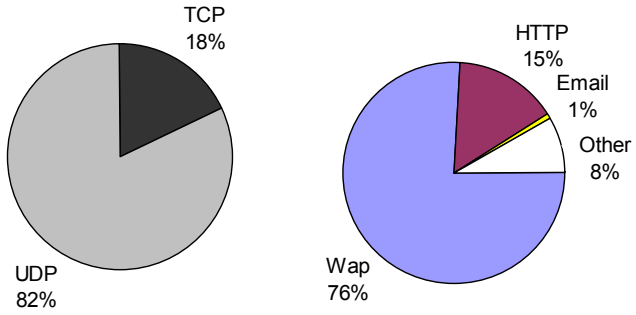


Figure 6-3: Application flows in GPRS trace Gi_B10

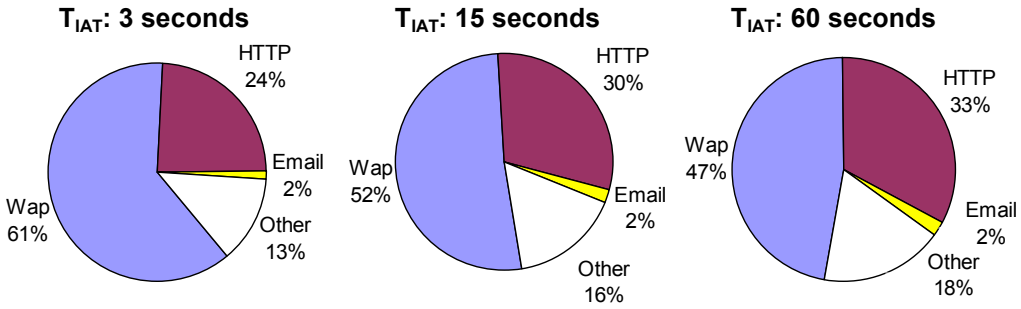


Figure 6-4: Application flows for extrapolated WAP 2.0 flows trace Gi_B10

[39] WAP flows in figure 6 include also MMS flows (as they are carried in WAP).

[40] Note, the numbers are truncated to the next lower integer value.

6.4 Flow direction

In this section we motivate further our *flow length* definition from section 6.2. Many investigations on flow lengths, based on wireline Internet measurements, do not differentiate the direction of the flow. But, as we are interested in the description of TCP flows which can be used for TCP performance analysis or TCP modeling, we need to know the length of the TCP data transfer. Therefore, we defined the length of flows as being the length from the data sending side.

Although, we know that in our measurement setup the server will always be outside the mobile network (beyond the Gi interface) and the clients will be inside the mobile network, we cannot determine, based on this, in which direction the application data is transmitted. We show in this section that the direction depends on the application type and particular usage case.

We illustrate our results with the help of density plots. Figure 6-5 to Figure 6-7 depict the two-variant distribution of flows, for the two parameters 'flow-length in uplink direction' and 'flow length in downlink direction'. The gray-shaded areas show which specific up/downlink byte combination most often occurs for flows. Darker shaded areas indicate those combinations that appear most often. The fraction of flows falling into this region is measured in log scale (see right bar in the figures). The line of crosses, starting in the origin, splits the flows into up- and downlink oriented flows. Flows on the right-hand side of the line download more data than they up load, and on the left-hand side the situation is the contrary.

The results show that it is important to select the direction of the data flows, instead of just using the flow length in e.g., the downlink direction, because each application has its specific up/downlink pattern.

We can read from Figure 6-5 that HTTP flows are mainly download orientated. More of the dark-shaded areas are on the right-hand side, and the upper right corner of Figure 6-5 indicates more longer transfers in the download direction. But at the same time we also see a large number of flows with more uploaded bytes than downloaded bytes. This most likely represents flows which upload data from the client to the server. This could be HTML forms, file uploads, Email upload through webmailer, etc.

The result is more differentiated for WAP flows. We consider WAP flows excluding MMS flows. MMS flows will be considered separately. As we can see in Figure 6-6, WAP flows are mainly download-oriented from the server to the client. Nevertheless, there are few flows which have more data transferred in the uplink than in the downlink direction. The upper boundary for the uplink flows is around 1000-2000 bytes. These are probably WML forms (WSP Post messages) from the server to the client.

Finally, MMS flows are very distinct. Figure 6-7 depicts the density plot for MMS flows. They either send 'only' data in the uplink or 'only' in the downlink direction. This reflects the two cases MMS message sending and MMS message receiving.

Taking all results on HTTP, WAP and MMS together, it is clear that it is important to differentiate on the flow direction. For example, if we always considered only the downlink direction, we would sometimes see statistics on very short flows, because the actual data transfer was in the opposite direction. But these are not the statistics we are interested in.

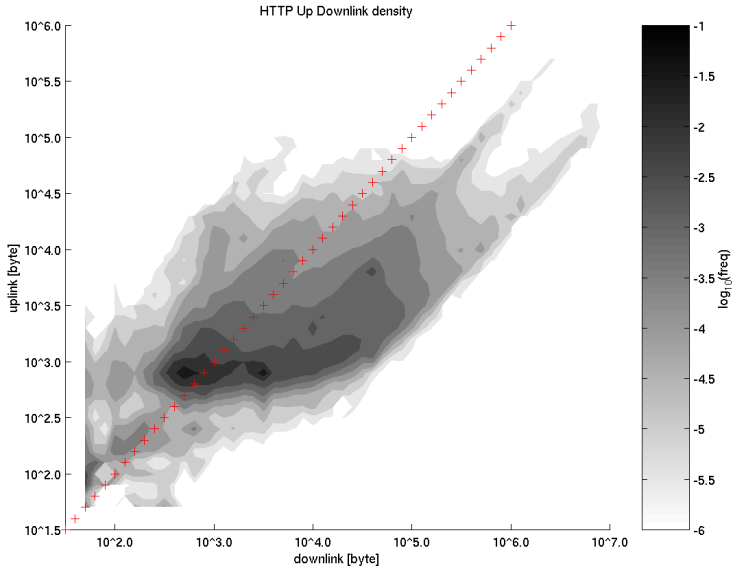


Figure 6-5: HTTP Up-downlink flow length

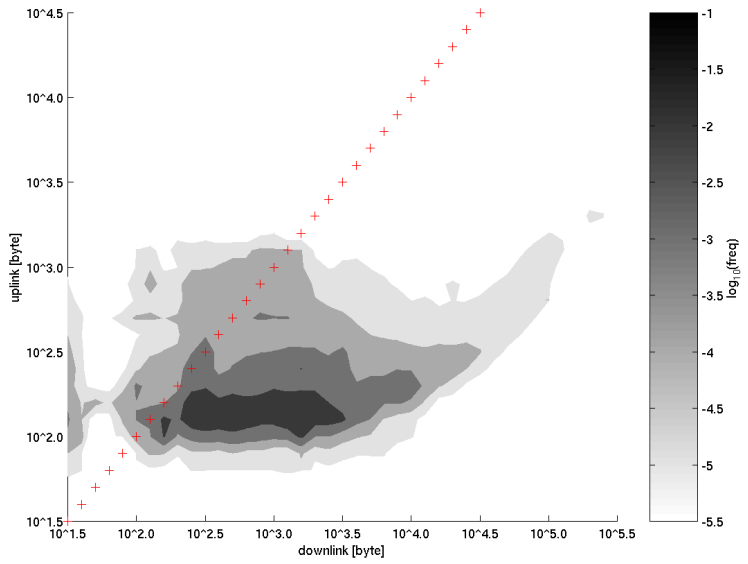


Figure 6-6: WAP up-downlink flow length

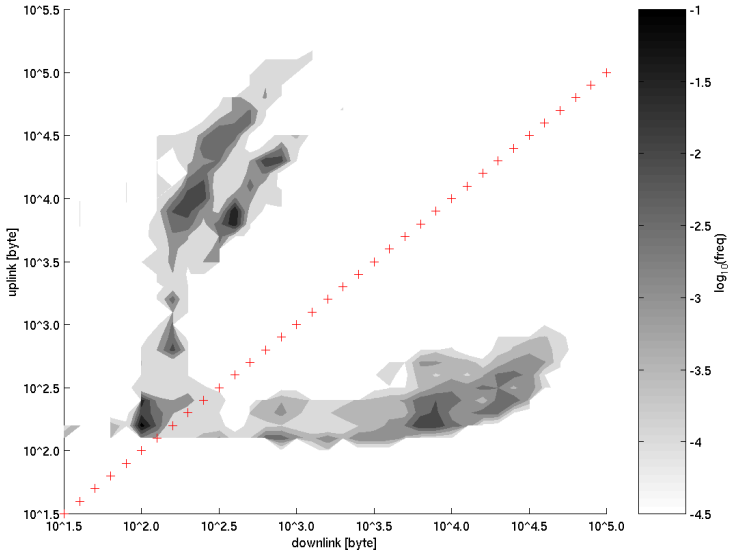


Figure 6-7: MMS up-downlink flow length

6.5 Fitting of distributions to empirical data

As needed in this chapter and also in later chapters, we briefly introduce the technique of distribution fitting. We continue with actual results in section 6.6.

Fitting an analytical distribution to empirical data consists of three main steps. First, we need to assure the *appropriateness* of the sampled data. The data needs to be independent (random), stationary and not showing trends. The next step would be to assume a distribution best describing the random variable from which the samples are and to *estimate the parameters* of this distribution. And in the fourth step the *goodness of the fit* is tested. In case the distribution is not a good fit, we start the process over at step two. We explain each of these four steps below.

For all explanations in the following $X=(x_1,..,x_N)$ is the set of data samples of size N.

6.5.1 Appropriateness tests for data sets

6.5.1.1 Independence

Independence is determined by the autocorrelation function (ACF) for different values of the lag. Lag 0 always has to be 1. But from lag 1 onwards the autocorrelation function is decaying to zero. In the case of independent samples, lag 1 should already be very small.

With the autocorrelation defined to be:

$$ACF(l) = \frac{E[(x_n - \mu)(x_{n+l} - \mu)]}{\sigma^2}, \qquad (6.5.1)$$

where μ is the sample mean and σ^2 is the sample variance.

A mathematical notion indicating independence is [PF94]:

$$\left|ACF(l)\right| \le \frac{\Phi^{-1}(1-\frac{\alpha}{2})}{\sqrt{N}} \quad \text{for } l \ge 1, \tag{6.5.2}$$

where Φ^{-1} is the percent point function of the standard normal distribution and α is the level of significance to assume independence. However, in practical circumstances often a visual inspection is done. Especially very large data sets often do not fulfill equation (6.5.2).

6.5.1.2 Stationarity

Stationarity implies a stable sample mean, without shifts and trends. A full definition of strict and weak stationarity is given in section 2.3.3.2. Strict stationarity is often difficult to prove. We will only consider stationarity of the first moment (mean). Stationarity is determined by visual inspection of the plots of the cumulative sample mean. The cumulative sample mean $\overline{X}_{cum}(s)$ is defined as:

$$\overline{X}_{cum}(s) = \frac{1}{s}\sum_{j=1}^{s} x_j \ . \tag{6.5.3}$$

For stationarity the cumulative sample mean should converge to a stable value with increased sample size.

Another appropriate visual inspection method is the moving sample mean $\overline{X}^{l}_{mov}(s)$ with a sliding window of size l, over which the average is taken. The definition of the moving sample mean is:

$$\overline{X}^{l}_{mov}(k) = \frac{1}{l}\sum_{j=0}^{l-1} x_{k+j} \ . \tag{6.5.4}$$

Plotting the moving sample mean can help to detect short term and long-term trends.

6.5.1.3 Periodicity

Another property for which the sample can be checked is periodicity. The data samples should also not show periodicity. Randomness exhibits white noise, that is, all frequencies are equally present in the ‘signal’. We can check the periodicity with the help of the discrete Fourier transformation, which is defined as:

$$DFT(k) = \sum_{j=0}^{N-1} x_j \left(e^{\frac{-2\pi i}{N}} \right)^{kj} \tag{6.5.5}$$

In case of periodicity strong spikes are visible in the DFT. Therefore, we use visual inspection of the DFT to check for periodicity in the time series.

6.5.2 Estimating distribution parameters

Several methods can be used to estimate the parameters of a distribution. The most often applied method is the maximum likelihood estimation (MLE), which we introduce below. Another parameter estimation method exists for the specific class of phase type distributions. For this class we introduce the specific expectation maximization (EM) Algorithm, below. If the distribution is assumed to be composed of several distributions and the tail distribution is heavy-tailed the scaling method can be used to estimate the parameter of the heavy-tailed distribution. We will introduce this method as well below.

6.5.2.1 Maximum Likelihood Estimation of distribution parameters

Loosely speaking, MLE optimizes the probability that a particular data set of data yields from a chosen probability model.

The method works as follows:

Let $f(x; p_1,..., p_q)$ be the assumed density function and $F(x; p_1,..., p_q)$ the corresponding distribution function; with $P = (p_1,.., p_q)$ the set of q parameters to be estimated.

The likelihood function L is defined as

$$L = \prod_{j=1}^{N} f(x_j; p_1,..., p_q) . \tag{6.5.6}$$

For computational ease, often the log-likelihood function

$$\log L = \sum_{j=1}^{N} \log f(x_j; p_1,..., p_q) \tag{6.5.7}$$

is used.

The maximum likelihood estimator $\hat{P}$ of the parameter vector is calculated by maximizing the log-likelihood function.

Maximizing the log-likelihood function is done by solving the set of partial differential equations:

$$\frac{\partial \log L}{\partial p_k} = \sum_{j=1}^{N} \frac{\partial f / \partial p_k}{f(x_j; p_1,..., p_q)}, \quad k = 1,..., q \tag{6.5.8}$$

The maximum likelihood estimator $\hat{P}$ sets all these partial derivatives to zero. That is, solving

$$\frac{\partial \log L}{\partial \hat{p}_k} = 0, \quad k = 1,..., q \tag{6.5.9}$$

yields the MLE.

The MLE function for $\hat{P}$ needs to be specified for each distribution function that one wishes to investigate.

6.5.2.2 Phase Type approximation of distributions

Phase type distributions were introduced in section 2.3.3.1 as being specifically appropriate to approximate general distributions arbitrarily close and at the same time to be analytically tractable for the purpose of traffic engineering. In the following we will focus on the hyper-exponential distribution as this class is often used to approximate empirical distributions from measurements. Several methods exist to derive the parameters of a hyper-exponential distribution. For instance two methods are described in [FW98] and in [ANO96]. Another non-parametric method especially suited to fit hyper-exponential distributions to heavy-tailed distributions is proposed in [KSH03]. We introduce this method now. This method is based on the EM-algorithm and uses only the empirical data. The method is iterative with a complexity of $O(NI)$ for each iteration with N the number of measurements and I the number of phases. Iteratively in each step the parameters c_i and λ_i are chosen to maximize a quality function describing how much better a newly chosen set of c_i and λ_i is. This can be described in terms of the following algorithm.

EM-algorithm for phase type distributions [KSH03]:

Let I be the number of phases for the hyper-exponential distribution, and $x_1,\ldots,x_N$ the N independent observations. Select initial values for c_i and λ_i (i=1,…,I). Set ε to be the precision, when the iterative algorithm can be stopped.

Let

$$p(x_n \mid \lambda_i) = \lambda_i e^{-\lambda_i x_n} \tag{6.5.10}$$

$$p(x_n \mid (c_i, \lambda_i)) = \sum_{i=1}^{I} c_i p(x_n \mid \lambda_i) \tag{6.5.11}$$

Compute in each iteration:

$$p(i \mid x_n, \lambda_i) = \frac{c_i p(x_n \mid \lambda_i)}{p(x_n \mid (c_i, \lambda_i))} \tag{6.5.12}$$

$$c_i' = \frac{1}{N} \sum_{n=1}^{N} p(i \mid x_n, \lambda_i) \tag{6.5.13}$$

$$\lambda_i' = \frac{\sum_{n=1}^{N} p(i \mid x_n, \lambda_i)}{\sum_{n=1}^{N} p(i \mid x_n, \lambda_i) x_n} \tag{6.5.14}$$

The iteration stops when the $|c_i' - c_i| < \varepsilon$ and $|\lambda_i' - \lambda_i| < \varepsilon$. Before starting the next iteration set c_i to c_i' and λ_i to λ_i'. The hyper-exponential distribution is fully

specified by the final set of c_i and λ_i values (cf. equation (2.3.6)). The **PH-fit** tool uses the described algorithm.

6.5.2.3 Scaling method

Typically an empirical data set does not stem from a single analytical distribution. In the case of Internet measurements often the heavy-tailedness only starts at a certain point x_h. That is, the distribution is composed of two or more distributions and only the tail is described by a heavy-tailed distribution. Using the Hill estimator, one has to assume the starting point x_h which is often difficult. Instead, the scaling method automatically detects the most appropriate region of heavy-tailedness and estimates α (the slope) for this region.[41] It exploits the fact that the shape of the tail of a heavy-tailed distribution determines the scaling properties of the dataset when aggregated. Taking a data set $X=\{x_1,..,x_N\}$, with $X^{(m)}(k)$ the aggregated data set at scaling level m (cf. equation (2.3.11)). The sum of variables with heavy-tailed distributions (and parameter α) converge again to a so-called stable distribution with the same α [CT99]. Therefore the method aggregates the data set and estimates for each aggregation level the parameter α. To find the optimum region, it compares the different α estimates at the different aggregation levels and chooses the one which is most consistently present over all aggregation levels.

The scaling method has been implemented in the **AEST** tool [AEST], which we use in the following analysis. The output of the **AEST** tool shows the different aggregation levels as well as the identified regions with the optimum estimated parameter α. For instance, in Figure 6-9, each line represents the CCDF of one aggregation level of the data set. The left-most down depicted curve represents the original data set. Moving up to the right shows higher aggregation sets. Dark dots on the lines indicate those regions that have been identified 'to belong' to a heavy-tailed distribution. For an exact heavy-tailed distribution, all lines are fairly parallel and most of their tails would be marked as heavy-tailed.

6.5.3 Goodness of fit test

Having the parameters estimated, we are interested in how well the chosen distribution assembles the empirical data set. Three main methods exist for this. The most general is the chi-square method. It can be applied for any distribution, either discrete or continuous, for which the cumulative distribution function can be computed. A disadvantage is that it requires the setting of a parameter (bin size) which influences the results, but for which no general rule can be given. Two other closely related methods that are more robust are the Kolmogorov-Smirnov test (KS test) and a refinement thereof, the Anderson-Darling test. Both are only defined for continuous data. The KS test is distribution free in the sense that the critical values do not depend on the specific distribution being tested. The Anderson-Darling test makes use of the specific distribution in calculating critical values. As the Anderson-Darling test needs to be specified for each distribution, and not all distributions are available in the literature, we use the KS test.

[41] The Hill estimator is more accurate for pure Pareto distributions. But the scaling method is good for first identifications of heavy-tailed distributions, and when it is not clear how the distribution is composed.

Definition: KS test

The KS test considers the difference between the empirical CDF and the assumedly 'true' distribution function.

We assume that the sample data results from the distribution F with the density function $f(x; p_1,..., p_q)$, having parameters p_1 to p_q.

$F_e(x)$ is the empirical CDF as defined in equation (2.3.2).

The difference measure used by the KS test is:

$$\begin{aligned} D &= \sup_x \left| F_e(x) - F(x) \right| \\ &= \max\left(\sup_x \left(F_e(x) - F(x)\right), \sup_x \left(F(x) - F_e(x)\right)\right) \end{aligned} \tag{6.5.15}$$

D is called the KS statistic. For each true distribution a critical value can be specified to either reject or accept an assumption that the data set is from the chosen distribution. The critical value must be defined for each distribution and depends on the significance level α. The significance level specifies the reliability of the result.

We point out that any goodness-of-fit test in fact does not prove that two distributions are the same, but rather they show if two distributions are different.[42] Therefore the hypothesis H_0: 'The data is from a specific distribution' is rejected if the KS statistic is above the critical value. The significance level α specifies how likely it is that we reject H_0 even if it is true. With a high significance level, the test rather rejects H_0 than accepts it, even if it *is* from the assumed distribution, while with a lower significance level the test rather assumes that H_0 cannot be rejected, even if the distributions are different. In practice, often a significance level of 5% is used.

However, empirical data rarely stems from a single assumed theoretical distribution; therefore D is seldom below the critical value for real measurement data. In practice the KS-statistic is used to compare different 'goodness-of-fits', that is, which distribution fits the empirical data best or more precisely which distribution can be rejected by the lowest KS statistic.

Additionally, the fit should be visually inspected by plotting the empirical distribution function together with the analytical distribution function in a CCDF plot or a quantile-quantile plot.

6.6 Application flow length statistics for GPRS

In this section we will investigate the length of flows. First we focus on the distribution tail of the flow length, measured in bytes. In section 6.6.1 we present the empirical flow length distribution. In section 6.6.2 we investigate the heavy-tailedness of the flow length distribution. In section 6.6.3 we verify the appropriateness of the data set, and derive the analytical distributions thereof in

[42] In practice it can never be proven that a sample data set comes from a particular distribution, as the sample data set is always of limited size.

section 6.6.4. Section 6.6.5 discusses the 'mice and elephant' phenomenon. After this, we will focus on the body of the empirical CDF in section 6.6.6 and evaluate the length of flows in terms of packets; in particular with respect to its impact on TCP performance.

6.6.1 Flow lengths in bytes

Figure 6-8 shows the empirical CDF of the flow length for the 6 introduced flow types. We show a log-linear plot to provide a first overview of the distributions. As can be seen, the CDFs for the new GPRS applications WAP and MMS are very irregular. Only the HTTP and WAP 2.0 distributions appear smooth. The irregular jumps in the MMS and WAP distributions probably stem from default message sizes in the MMS clients[43] and special WAP portal pages.[44] We can already observe that especially WAP flows are currently in general quite small.

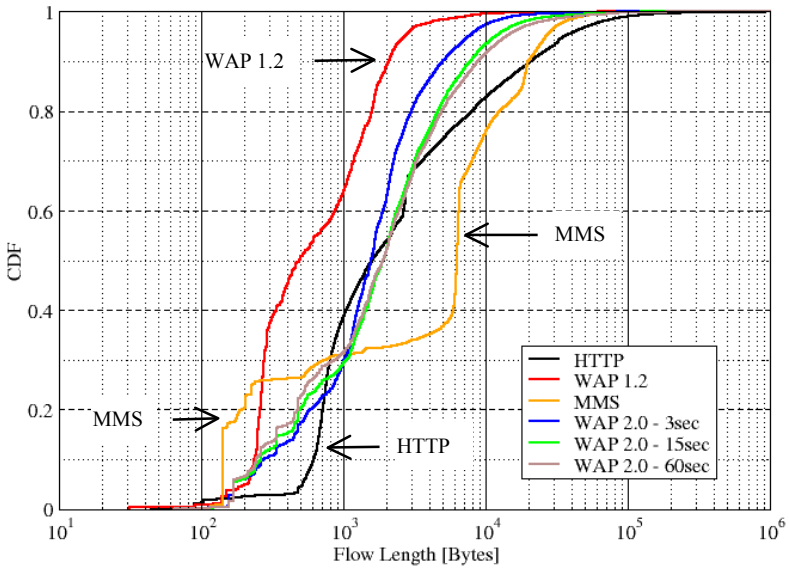


Figure 6-8: Flow length in bytes

In the following sections we focus only on HTTP traffic and the WAP 1.2 and WAP 2.0 (15 sec) traffic scenarios[45] excluding MMS, since the distribution for MMS did not allow a conclusive distribution related investigation due to its heavy irregularity.

In accordance with wireline measurements we first assume heavy-tailedness also for GPRS flows. Therefore, we will first focus on this. We introduced heavy tails in section 2.3.3.2 and the 'scaling method' by [CT99] to investigate the tail of the flow distributions in section 6.5.2.3. We show results on this, next.

[43] For example, most GPRS terminals have a built-in camera. All of them currently have approximately the same pixel resolution. Sending MMS with embedded pictures taken with the terminal might result in jumps in the MMS flow length distribution as depicted in Figure 6-8.

[44] Many operators configure the WAP clients of their subscribers to start with a specific operator portal page.

[45] The CDFs for WAP 2.0 (3 sec) and WAP 2.0 (60 sec) are very similar.

6.6.2 Heavy-tailedness estimation of the data sets

Figure 6-9 contains the output of the **AEST** tool for HTTP flows, Figure 6-10 shows it for WAP 1.2, and Figure 6-11 shows it for WAP 2.0 (15 sec) flows.

In the case of HTTP traffic, a heavy tail could be identified for the 'far end' of the tail. That is, the flow length of HTTP traffic appears to be asymptotically heavy-tailed. The estimator for α is 1.158, which would indicate strong heavy-tailedness. However, the graph is not absolutely conclusive on heavy-tailedness. Large parts of the distribution do not follow a heavy-tailed distribution. However, the assumption of heavy-tailedness would be in line with HTTP measurement in wireline Internet. We shall investigate its possible underlying distribution in section 6.6.4

Figure 6-10 shows the output from the **AEST** tool for WAP 1.2 traffic. Again some regions are indicated to follow a heavy-tailed distribution, with α=1.223. However, the regions are irregular, not close to the tail and very short. [CT99] recommends not assuming heavy-tailedness for this type of results. Therefore we assume no heavy-tailedness for WAP traffic and reject the proposed α. Again, we shall investigate the distribution more in detail in section 6.6.4.

The same is true for WAP 2.0 (15 sec). The indicated parts are further to the top of the curves and short. Also the distribution is very curved at the tail. [CT99] proposes that this kind of curves might belong to 'lighter' heavy-tailed distributions like lognormal or Weibull. The proposed α=1.223.

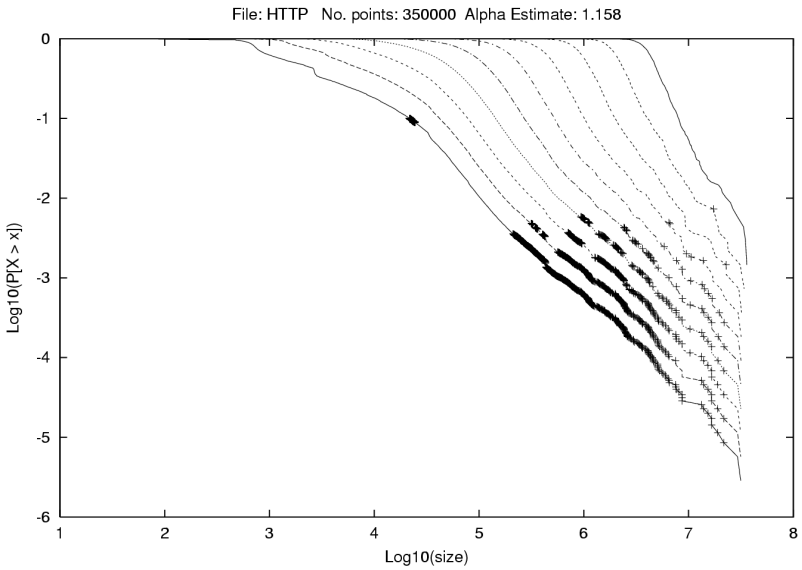


Figure 6-9: Heavy tail scaling regions for HTTP flow lengths

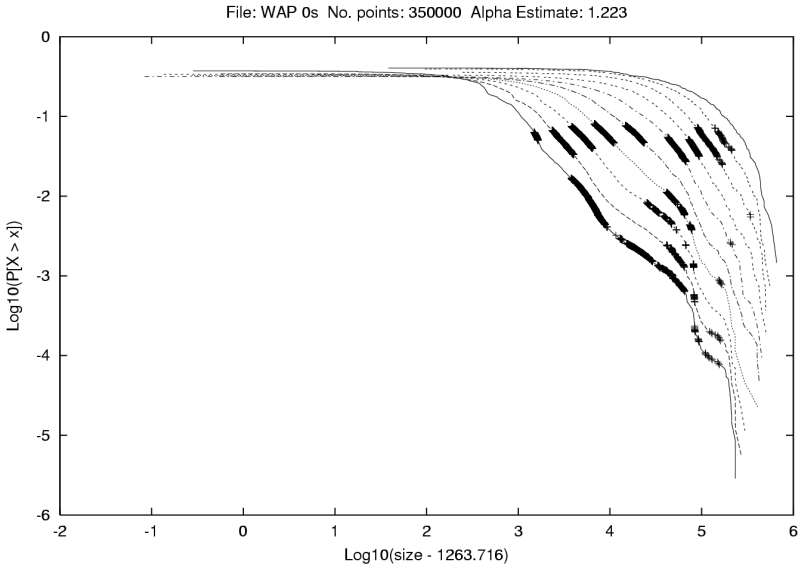


Figure 6-10: Heavy tail scaling regions for WAP 1.2 flow lengths

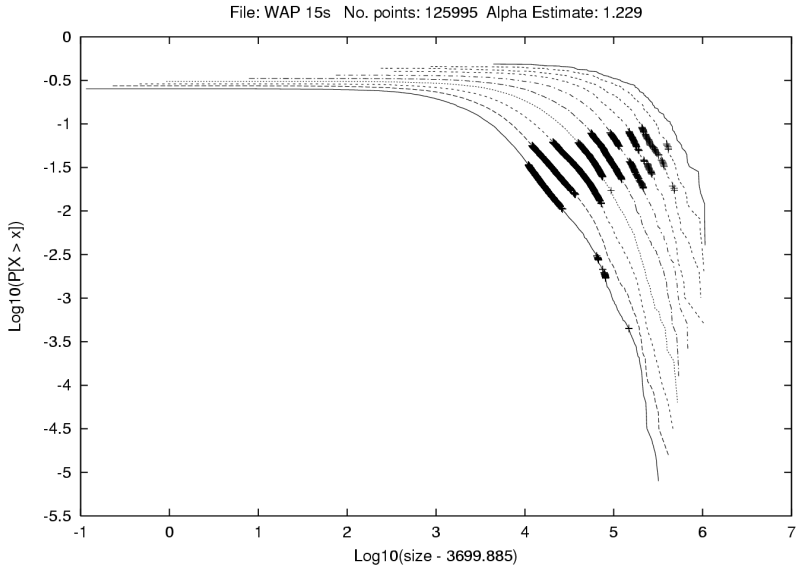


Figure 6-11: Heavy tail scaling regions for WAP 2.0 flow lengths

6.6.3 Data set validation

Having investigated in general the heavy-tailedness, we will derive the flow length distribution based on the MLE method. First, in this section we validate the data sets. We select a part of the data set, covering some hours of the day in the busy period and test this data set as to its appropriateness for modeling (cf. section 6.5.1). Figure 6-12 displays the result for a HTTP flow length sample trace. We do not show it here, but we tested in the same way the WAP flows, which provided similar results. In all but plot (b) and (c), the x-axis depicts the index of the sample in the time series.

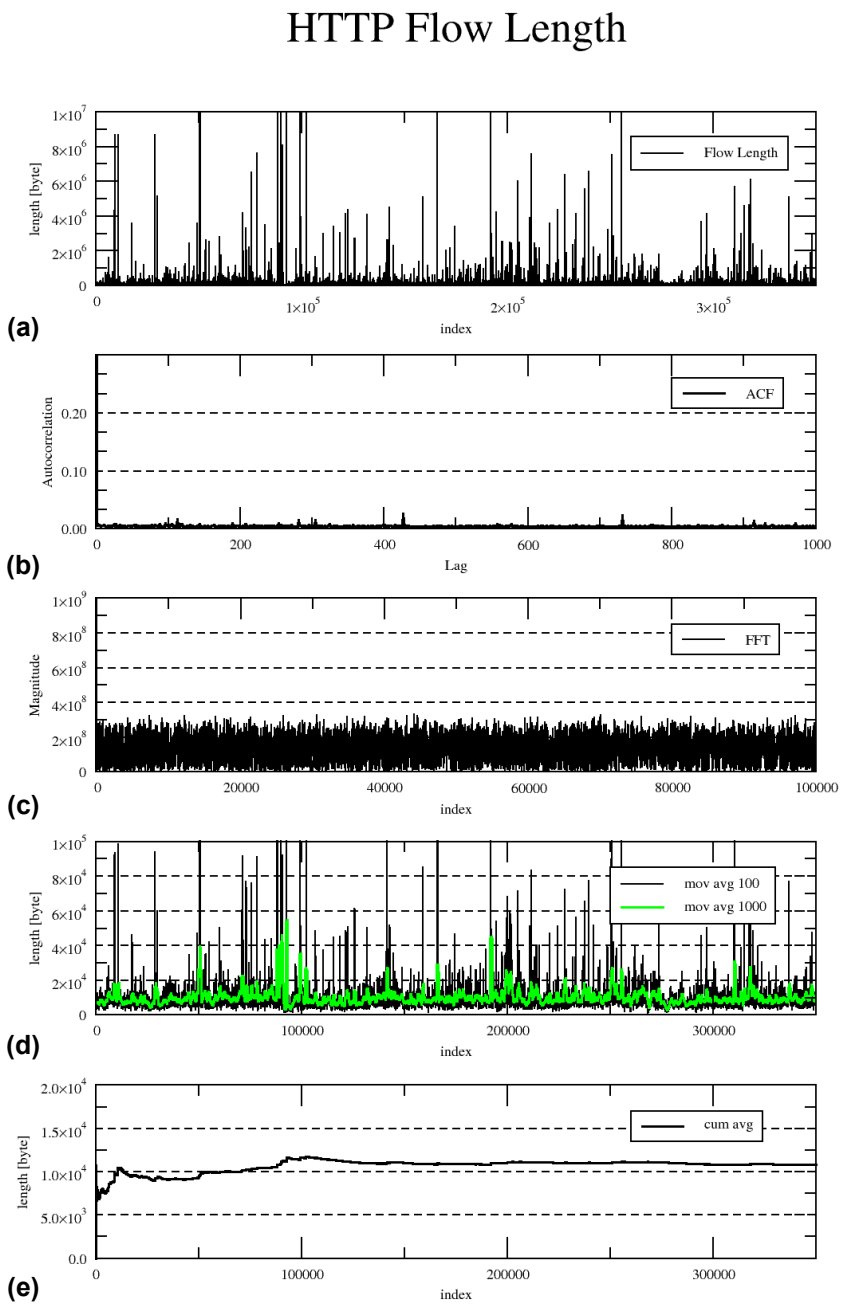


Figure 6-12: Data set validation for HTTP flow length

Figure 6-12 (a) provides a visualization of the time series of the HTTP flow length. It shows a high variance of the data values. Plot (b) shows the ACF for lag 0 to 1000. The ACF (1) drops immediately to below 0.01, and stays at this level. This indicates highly uncorrelated values and is below the 5% significance level for this data set. Plot (c) shows the results from the Discrete Fourier Transformation (DFT). No distinct dominating frequency parts are visible. Plot (d) depicts the moving average for a window length l=100 and l=1000. At this level still quite some fluctuation is visible, which could be due to the assumed heavy-tailedness. However, no clear singled-out trends or level shifts are visible, which is the important conclusion. Finally plot (e) depicts the

cumulative mean over the whole data set. After a short transient period in the beginning, the mean quickly stabilizes and shows no visual trends.

Based on the visual inspection, we can conclude that independence and stationarity are a valid assumption. Therefore all of the data sets are appropriate for testing on the underlying distribution.

6.6.4 Fitting of flow length data sets

In this section we investigate which analytical distribution best fits the empirical CDF for the flow length of HTTP, WAP 1.2 and WAP 2.0. A single analytical distribution is better tractable in further analysis. Therefore we use the MLE method to derive the parameters of a single distribution. But often the empirical distribution does not belong to a single distribution class. Therefore, additionally, we use the EM algorithm introduced in section 6.5.2.2. This method allows a close fitting on the empirical distribution, while still yielding a tractable description. In line with the parsimonious requirement in section 2.3.3 we only consider phase-type distributions with 2 or 4 phases.

The **AEST** tool indicated that the heavy-tailedness exists only asymptotically for the tail part. Therefore, we assume a different distribution for the body and for the tail of the HTTP distribution. We split up the HTTP flow distribution in a body and a tail at the length of 10^4 bytes.[46] This modeling approach is also proposed for wireline HTTP flows in [BC98]. However, we assume the distribution of the WAP 1.2 and WAP 2.0 flows can be described each by a single analytical distribution.

Table 6-1 summarizes the results from testing different distributions. We show for the different data sets the mean, median and coefficient of variation (CV) statistics, for the empirical as well as the fitted distributions.

We used the MLE and KS test methods implemented in the **DATAPLOT** tool [DATAPLOT] for deriving the analytical distribution. We tested each empirical distribution against the normal, exponential, gamma, logistic, extreme value, Weibull, lognormal, and Pareto distribution (cf. Appendix B). In all cases some distribution from the class of heavy-tailed distributions (Pareto, lognormal, Weibull) scored best. However, none passed the KS test. That is, the KS-statistics is never below the critical value defined for the KS test. Therefore we list the top three ranked distributions in the MLE section in Table 6-1. In order to check the appropriateness of the Pareto distribution we have included it for the case when the HTTP tail starts at 10^4 bytes and at 10^6 bytes.

We also show the results for the phase-type distribution for each data set in the EM section in Table 6-1. In case of HTTP we show the EM results only for the total empirical distribution, and not for body and tail separately.

Table C-1 in Appendix C lists the corresponding parameters for the fitted distributions.

[46] We tried out several values. We present results for a split at 10^4 and 10^6 bytes.

Flow length		Distribution	Mean	Median	CV	KS-statistic
HTTP body						
Empirical			2067.647	1133	0.991764	
MLE	**1st**	lognormal	2108.478	1364.436	1.178128	0.093395
	2nd	Gamma	2067.647	NA	0.88706	0.123352
	3rd	Weibull	2078.149	581.0078	0.891748	0.124916
HTTP tail						
Empirical			40489.89	14109	8.86301	
MLE	**1st**	lognormal	36825.57	11881.79	2.933571	0.052849
	2nd	Weibull	33067.29	11218.63	1.578251	0.067133
	3rd	Gamma	40489.89	NA	1.401365	0.137995
	4th	Pareto	73051.7	12818.46	NA	0.14798
(tail starts at 10^6 byte)		Pareto	390141	123855.3	NA	0.0473
HTTP Total						
Empirical			10821.22	1757	14.20517	
EM		PH 2-phase	9473.915	1912.137	2.439723	
		PH 4-phase	10821.22	1865.381	8.261301	
WAP 1.2						
Empirical			1263.716	854	2.599148	
MLE	**1st**	Lognormal	1200.212	748.5217	1.253407	0.090788
	2nd	Gamma	1263.716	444.4164	0.951797	0.104647
	3rd	Exponential	1327.716	NA	0.982386	0.111165
EM		PH 2-phase	1263.716	769.9679	2.320188	
		PH 4-phase	1263.716	764.1593	2.838391	
WAP 2.0						
Empirical			3699.885	1888	2.260354	
MLE	**1st**	Lognormal	3651.368	1739.848	1.845107	0.066968
	2nd	Weibull	3574.891	1182.613	1.24251	0.078783
	3rd	Gamma	3699.885	NA	1.010861	0.123792
EM		PH 2-phase	3699.885	1856.392	1.900573	
		PH 4-phase	3699.885	1843.729	2.513669	

Table 6-1: GPRS flow length – distribution fitting results

If we assume 10^4 bytes as splitting point of the HTTP flow length distribution, the body and the tail is well fitted by a lognormal distribution. If the split point is moved to 10^6 bytes a Pareto distribution is the best choice for the tail. Additionally, α the shape parameter of the Pareto distribution, is in both considered cases close to the resulting α from the **`AEST`** tool. Figure 6-13 depicts the corresponding empirical and analytical curves. As can be seen the good lognormal KS test results stem from the upper part of the tail between 10^4 and 10^6 bytes, while the Pareto distribution has a parallel slope at the extreme tail part. This suggests indeed that a Pareto distribution is a good fit for the tail. Therefore we suggest that HTTP flows for GPRS can be best modeled with a lognormal distribution for the body and a Pareto distribution for the tail. This is

also in accordance with suggestions for wireline HTTP traffic as presented in [BCT98].

The results for the phase-type distribution for the entire empirical distribution for HTTP are also listed in Table 6-1. The mean, median and CV statistics, as well as Figure 6-14, suggests that 2 phases are not enough, but 4 phases seem to match well for the considered part of the distribution.

In the case of WAP 1.2 again the lognormal distribution is the best-fitting distribution (cf. Figure 6-15). The shape parameter of 0.97 indicates 'very little' heavy-tailedness. Also note that the next two best-fitting distributions gamma and exponential are not heavy-tailed. The KS statistic for the exponential distribution was also only 0.1111647 and therefore close to the best KS-statistic of the lognormal distribution. As the **AEST** tool also suggested no heavy-tailedness, we assume no heavy-tailedness for WAP 1.2 flows. This corresponds also to the fact that the results for the phase type distributions in Figure 6-16 suggest a very good fit for 2 phases and 4 phases.

The fitting results for WAP 2.0 (15 sec) also indicate a lognormal distribution, though with a larger shape parameter, indicating 'stronger' heavy-tailedness. Additionally, the Weibull distribution has a shape parameter less than one indicating heavy-tailedness. However, the Pareto distribution did not score for the first 4 ranks. As Figure 6-17 also shows a very good fit for the lognormal distribution, we assume that WAP 2.0 should be modeled with a lognormal distribution. Figure 6-18 depicts the results for the 2 and 4 phase type distributions. Again 2 and 4 phases appear to yield a good approximation, allowing also a practical approximation via phase type distributions.

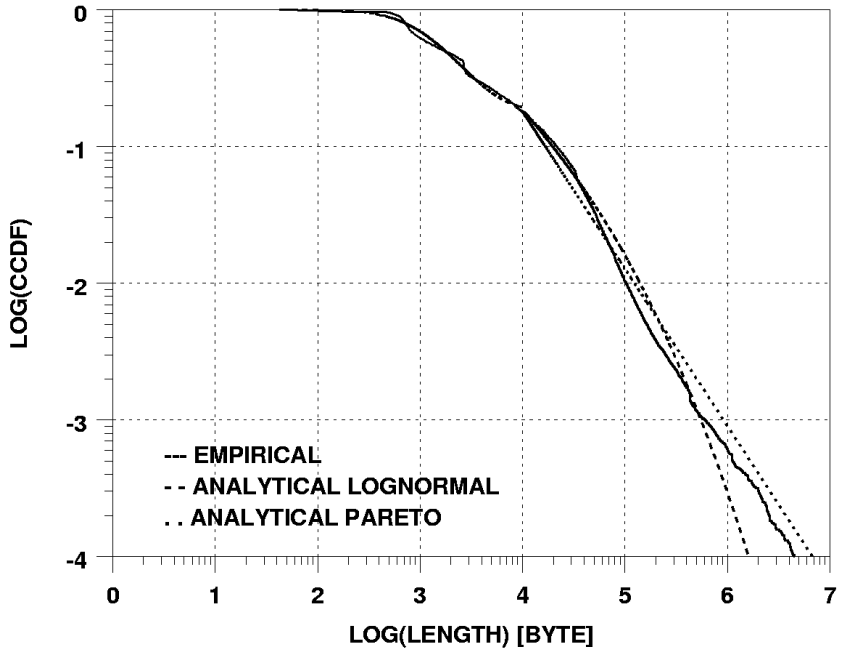


Figure 6-13: HTTP flows – lognormal and Pareto (partial) fitting on distribution

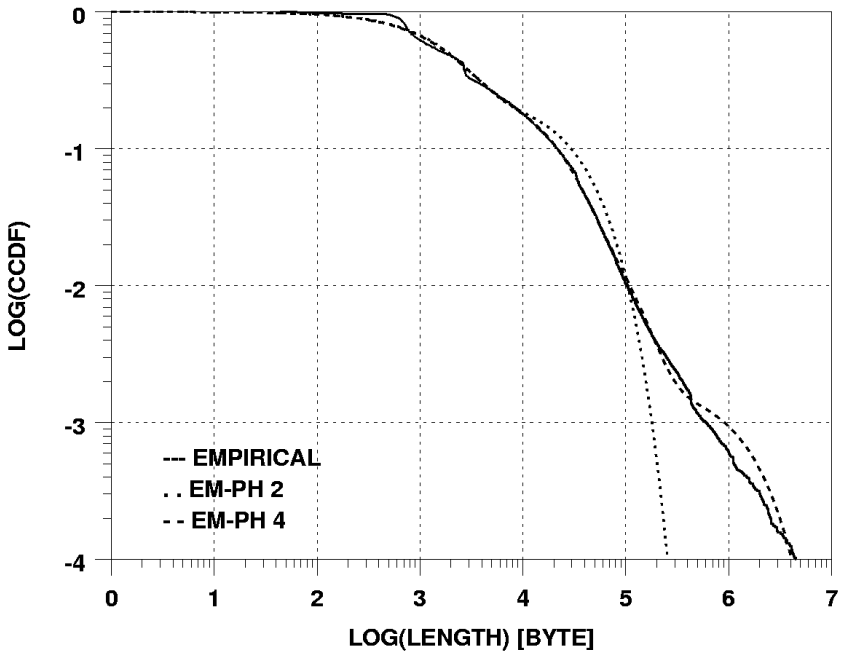


Figure 6-14: HTTP flows – EM - PH fitting on distribution

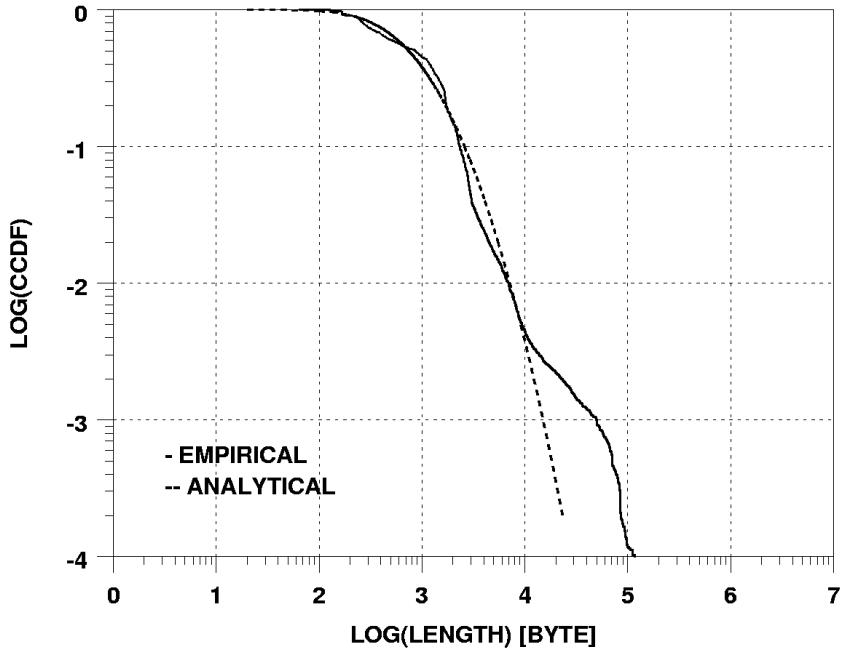


Figure 6-15: WAP 1.2 flows – lognormal fitting on distribution

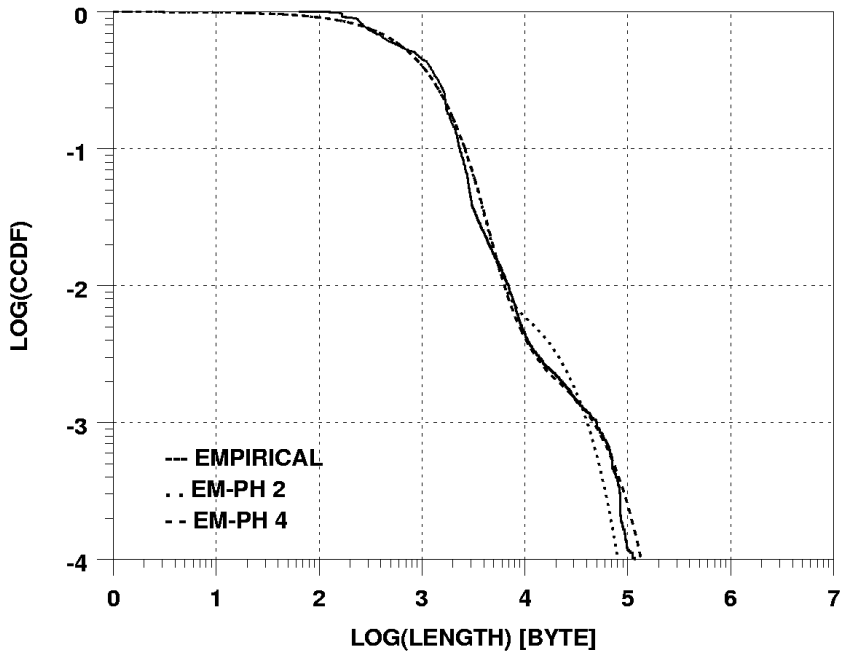


Figure 6-16: WAP 1.2 flows – EM - PH fitting on distribution

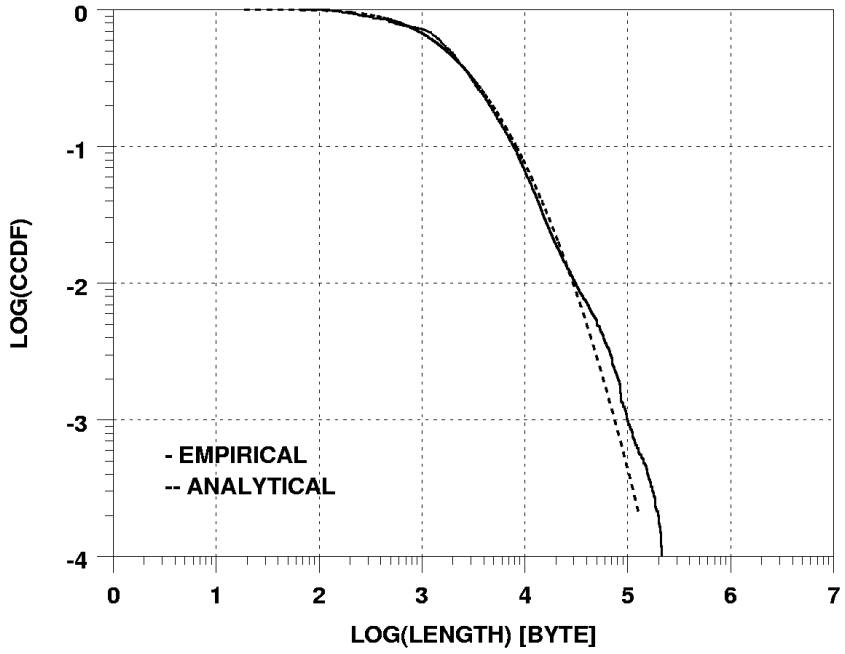


Figure 6-17: WAP 2.0 (15sec) flows – lognormal fitting on distribution

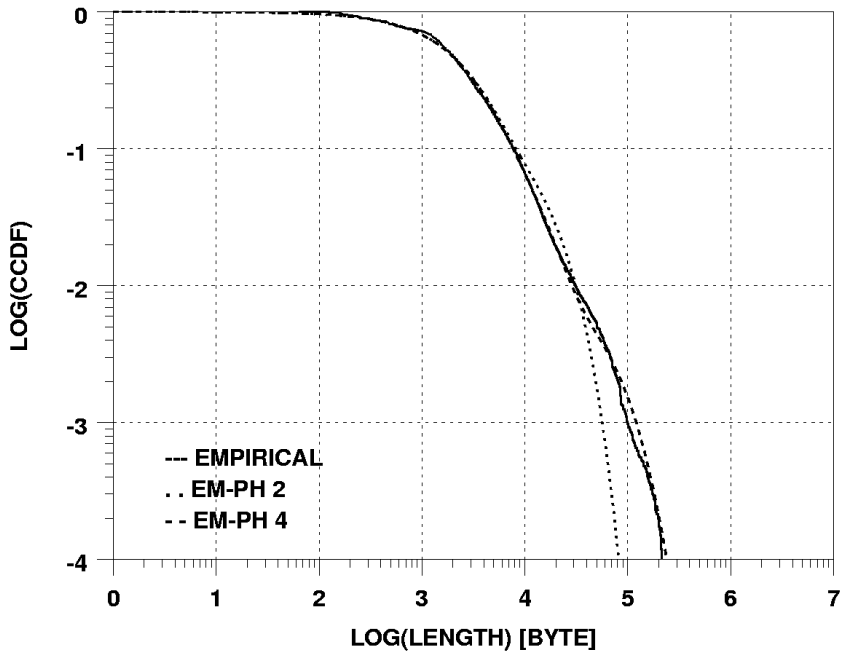


Figure 6-18: WAP 2.0 (15sec) flows – EM - PH fitting on distribution

6.6.5 Volume and flow disparity

The heavy-tailedness of flows in the wireline Internet is manifested by a phenomenon frequently referred to as 'mice and elephants'. This phenomenon describes the observation that traffic consists of a few large flows (elephants) contributing most of the traffic and many small contributors (mice). This is also called size disparity [BHG+04]. Examples of disparity exist for long lasting flows [BC02] and for bitrate or burst elephants [LH03]. Understanding the disparity is important for planning, routing and engineering [BHG+04]. For example, the presence of this phenomenon allows the control of the total volume via control of a few (large) contributors. Also data reduction can be achieved by handling short flows in a special way (e.g., frequent but short flows get shorter ID-bit strings assigned). In the context of flow modeling, knowing this phenomenon allows to model a large fraction of the flows by modeling few long lasting flows. And in particular if TCP is modeled as long lasting flows, it can be more easily modeled, being in an equilibrium state. Disparity is another viewpoint on heavy-tailed distributions, and might reveal more about the heavy-tailedness of the flows in GPRS, therefore we investigate disparity as well.

We investigate this phenomenon by inspecting empirical CDFs for flow length combined with the aggregated number of packets and data volume thereof. Figure 6-19 shows such a combined plot. The curve 'Flows' depicts the CDF for the flow length in bytes. In parallel to this, the curves 'to Packets' and 'to Bytes' show the fraction of the total amount of packets and bytes these flows carry accumulated. The vertical arrow in Figure 6-19 shows, for example, that 80% of the TCP flows (which themselves are shorter than 7.3 Kbyte) carry only 10% of all TCP bytes and about 33% of all TCP packets.

The mice and elephant phenomenon can be noticed in GPRS when considering only TCP flows. We can read from Figure 6-19 that 2% of the TCP flows are longer than 73.2 Kbyte but account for 56% of the total number of TCP bytes. These are the elephants, while the majority of flows (mice) are even shorter than the average of 11.3 Kbyte.

In the case of HTTP flows, the results look quite similar (Figure 6-20). The TCP and HTTP curves have very similar shapes. Two percent of all HTTP flows carry 41% of all HTTP-bytes. These 2% of HTTP flows are longer than 89.6 Kbytes. This is in line with our finding in the previous section. That is, the heavy-tailedness of the HTTP flows introduces the mice elephant phenomenon. Therefore, in order to optimize the majority of traffic volume, TCP and HTTP flows can be modeled and treated similar to flows in the wireline Internet (see beginning of this section). But we point to section 6.6.6 in which we will discuss the possible impact on performance by the very short flows constituting the body of the distribution.

If we consider only WAP 1.2 and WAP 2.0 (15 sec) flows,[47] flow disparity appears not to be present anymore (Figure 6-21 and Figure 6-22). The top 2% of the WAP flows, which are longer than 4.2 Kbyte for WAP 1.2 and longer than 18.7 Kbyte for WAP 2.0, carry only 23% and 24% of all WAP bytes,

[47] The scenarios of 3 seconds and 60 seconds are very similar.

respectively. That is, a large fraction of the data volume resulting from WAP can be described by short flows. Therefore, if the handling of the majority of WAP data *volume* is to be optimized, short flows must be considered.

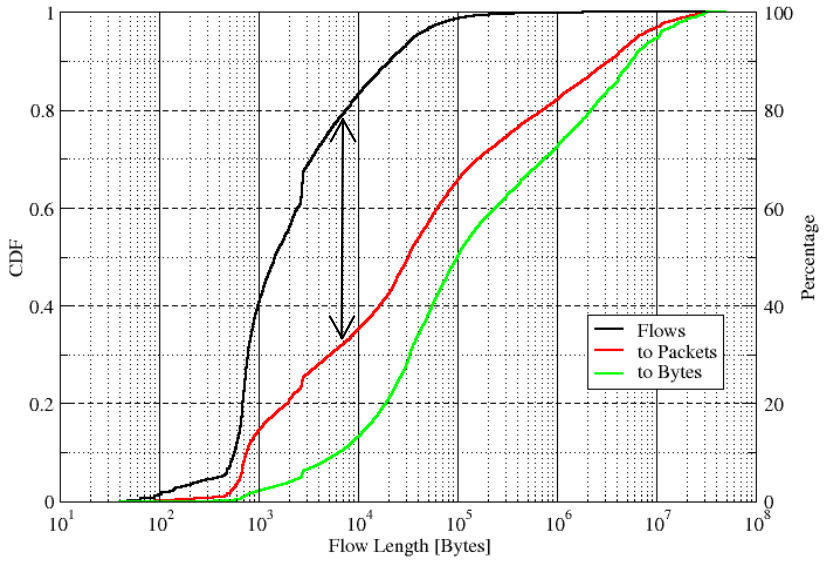


Figure 6-19: TCP flow byte and packet cumulative percentage

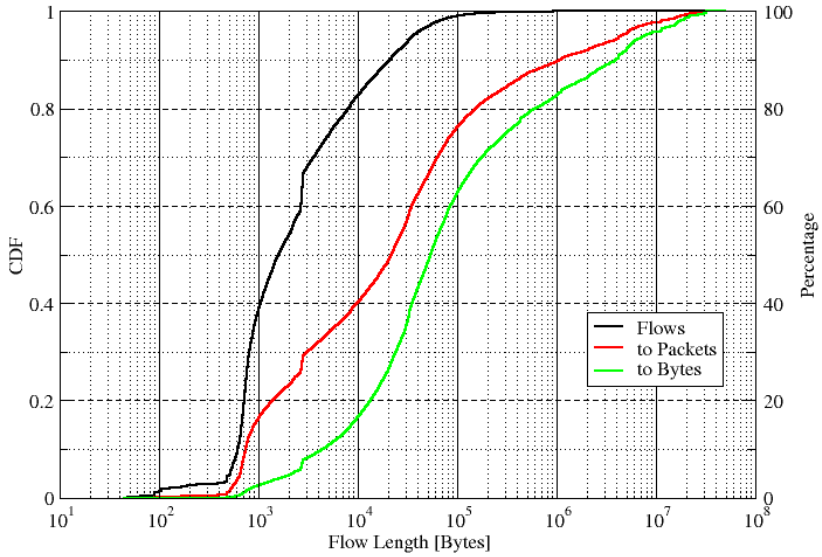


Figure 6-20: HTTP flow byte and packet cumulative percentage

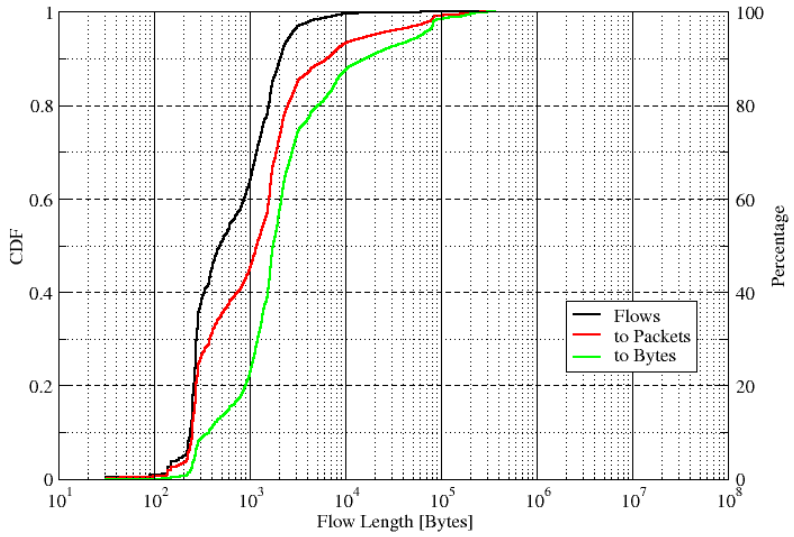


Figure 6-21: WAP 1.2 flow byte and packet cumulative percentage

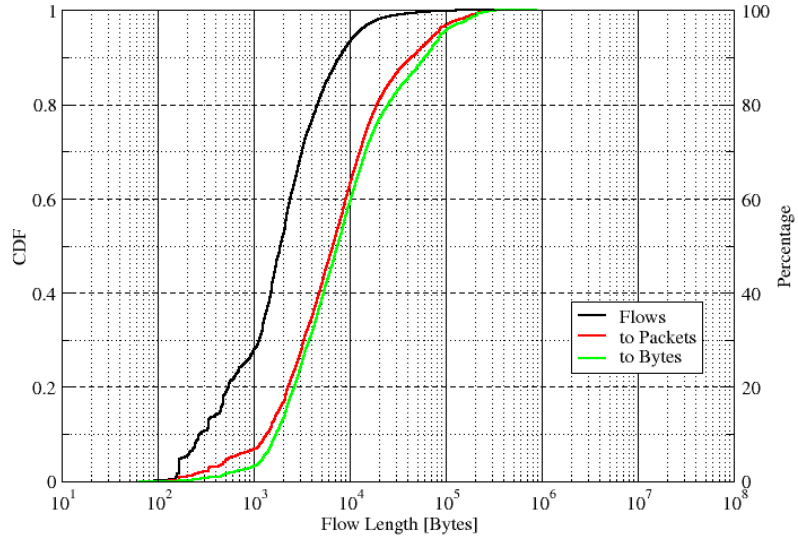


Figure 6-22: WAP 2.0 (15 sec) byte and packet cumulative percentage

6.6.6 The body of the flow length

The results from the previous sections can be used to model the flow length for the purpose of traffic engineering. In this section we take a different approach – we focus on the body of the distribution which is of interest for the impact on the TCP performance through packet loss, as will be elaborated below.

In this section we investigate the flow length of HTTP, WAP 1.2, MMS and also of the extrapolated case for WAP 2.0. Our attention is focused in particular on very short flows. In [AA02] the authors argue that 7 packets (and for some TCP flavors even less) is a very critical value for TCP flows. The shorter the flows, the less likely it is that TCP can recover a packet loss using its advanced loss

recovery schemes. In many cases, TCP will instead be forced into a costly timeout for error recovery, decreasing the end-to-end performance of the flow. In particular, flows below the critical level definitely have to rely on its retransmission timer for loss recovery [AA02].

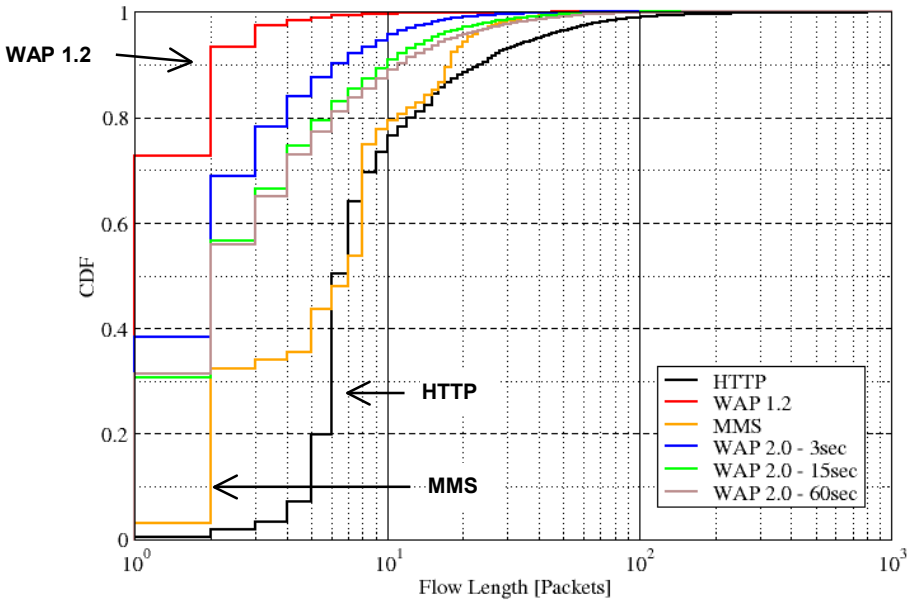


Figure 6-23: Flow length in packets

In terms of packets, most of the HTTP flows and almost all WAP flows are shorter than the critical number of 6-7 packets. HTTP flows consist on average only of 18.65 packets, and 64% of all HTTP flows have 7 or fewer packets. WAP 1.2 flows even have an average of only 1.47 packets, and about 99% of all WAP 1.2 flows consist of 7 or less packets. Figure 6-23 depicts the empirical results including the extrapolated flow lengths for WAP 2.0. The average number of packets for WAP 2.0 flows is 3.1 packets, 4.5 packets and 5.2 packets for a T_{IAT}-value of 3 seconds, 15 seconds and 60 seconds respectively.

We extracted from the empirical CDFs the percentage of flows below the critical value proposed in [AA02] and list them in Table 6-2. The table must be read like a CDF; for example for MMS, 53.67% of all flows consist of 7 or less packets.

Flow length [packets]	HTTP	WAP 1.2	MMS	WAP 2.0		
				3 sec	15 sec	60 sec
≤ 4	7.07%	98.36%	35.56%	84.06%	74.68%	72.86%
≤ 5	19.79%	98.86%	43.73%	87.49%	79.33%	77.32%
≤ 6	50.33%	99.29%	47.86%	90.22%	83.02%	81.01%
≤ 7	64.09%	99.48%	53.67%	92.04%	85.48%	83.62%
≤ 8	69.64%	99.60%	74.87%	93.36%	87.38%	85.50%

Table 6-2: GPRS flow length below critical value for TCP

An important finding is that WAP 1.2 flows are essentially all in the critical area. And WAP 2.0 flows are below the critical value in 83% of the cases, even with

an optimistic assumption of having all 'Get' messages going to the same server being pipelined, and having a TCP persistence timer of 60 seconds. That is, even WAP 2.0 flows would depend largely on timeouts for loss recovery.

6.7 Conclusion

Based on a specific method to measure application flows, we have focused in our investigation separately on the body and on the tail of flow length distributions.

Our main finding is that the majority of flows in GPRS, which can be accounted to WAP flows, is very short, and is showing only weak signs of heavy-tailedness. We found that although WAP 1.2 flows are best modeled by a lognormal distribution, an exponential distribution also comes surprisingly close to the empirical distribution. The length of WAP 2.0 flows is best modeled by a lognormal distribution.

Since we did not find strong indications for heavy-tailedness of WAP flows, PH-type distributions appear to be a good candidate for modeling WAP traffic. This is especially of interest as PH-type distributions are much better to handle in analytical studies. Therefore we provided PH-type distributions for the investigated application flows. In particular our results suggest that a parsimonious PH-type model with 2 or 4 phases is feasible for WAP flows.

HTTP flows, on the other hand, have strong signs of heavy-tailedness also in GPRS. We found the same model as suggested by [BC98] in that HTTP flows are best split into a body, modeled by a lognormal distribution, and a tail modeled by a Pareto distribution. As consequence, phase type distributions with up to 4 phases provide only an approximation of the distribution in the observed data range.

Another important result is the dominance of short flows, which we derived by focusing on the body of the distribution. Due to the – already pointed out – lack of heavy-tailedness, the mass of WAP flows, in terms of flow length in bytes and packets, as well as in terms of the total traffic volume, is carried by short flows. This is very critical as WAP 2.0 in the future will be carried over TCP and this protocol is very performance sensitive for short flows. In [AA02] the authors show that TCP flows falling short of a critical length of 6-8 packets can be degraded seriously in performance, as these flows need to recover from packet loss always by a costly packet time-out.

In our study we showed that even in conservative assumptions, about 80% of the WAP flows carry less than the critical number of packets. To counteract potential performance degradation we therefore recommend, that research on TCP especially needs to consider short flows. Existing TCP options should be validated for their appropriateness for a traffic mixture with short flows. For example, none of the WP-TCP [WAP225] options recommended for WAP 2.0 take into consideration the unique shortness of WAP flows. As WAP[48] will

[48] This should be read as WAP including WAP 2.0 and MMS.

probably stay the dominant application in GPRS-like networks, adapting WP-TCP is essential.

Furthermore, we recommend that TCP models for wireless networks need to capture the body of the distribution as well as the tail. This is contradictory to common practice in which many TCP studies investigate TCP only in equilibrium state. That is, TCP is only modeled when the TCP slow start is over and the file length tends to infinity [KT04].

7 User Mobility

This chapter discusses results on user mobility in GPRS. In section 7.1 we motivate our approach. Our specific concept of perceived mobility is introduced in section 7.2. The derived metrics to measure mobility are introduced in section 7.3. The limitations of our approach are discussed in section 7.4. Using the proposed approach and metrics we are able to evaluate the general level of mobility of the GPRS users. The results of this analysis are presented in section 7.5. Our investigation of the correlation between user mobility and application usage is presented in section 7.6. This shows in practice the limitations of our current measurement approach on mobility. As an important part of a mobility model, we present in section 7.7 the derived analytical distribution for the cell reselection inter-arrival times. Section 7.7 concludes the investigation on user mobility.

The results in this chapter are based on the traces GMM_5.6 and Gi_B7.

7.1 Motivation

Mobility is a central aspect of cellular networks and, hence, needs to be taken into consideration in the modeling of traffic in such networks. Cellular networks are built in a way to allow seamless roaming. But the impact of mobility can often not be totally hidden. For instance, cell reselections[49] often introduce noticeable delays or even packet loss, which depends on the type of cell reselection. Handling mobility also introduces signaling between the mobile station, the BSC and the SGSN. Therefore, modeling the user mobility allows the consideration of those aspects in dimensioning and performance analysis of cellular networks.

User mobility can be modeled in terms of geographical movements, that is the actual trajectories of the users are modeled, or in terms of cell-related movement. In the latter case only cell reselections from one cell to another cell are modeled.

Many mobility models used today are based to a large extent on *assumptions* about user behavior. For instance in [ZD97] an extensive model is developed, enhancing prior work on mobility models for mobile networks. Based on geometric considerations of the user movement pattern through cells, an analytical model describing *cell handover inter-arrival times* is derived. Problematic is that this model does not consider any measurements, and the assumptions on user movement are purely based on intuition.

[49] Cell handover and cell reselection are basically the same. As cell reselection is the GPRS terminology, we will use cell reselection.

A different approach is taken for instance by the authors of [SHSK01] and [HSSK01]. They measured the exact movement of 4 different classes of vehicles with mounted Global Positioning System (GPS) receivers and derived from this the cell handover inter-arrival times. They mapped the exact loci of the vehicles on a fictitious cell layout and acquired by using this method the cell locations and cell handovers. In other research the authors measured the mobility events in a mobile network [TB02] [TP03]. The authors used the mobility event information to central network nodes to derive the cell handover and channel holding times.

However, using this direct mobility information is generally not much considered in previous research; in particular how the geographical mobility is perceived by the network. The latter, which we call *perceived mobility*, is related to the mobility events (e.g., cell reselections) that occur in the network. The generation of such events is dependent on the actual geographical movement of the mobile station, the particular network cell layout, *and* also the data traffic load per user. This viewpoint on *perceived mobility* leads to the specific cell reselection inter-arrival time and the mobility activity profile over time and space in a particular network, considering additionally to which extent the user combines mobility with data transmission.

In the following sections we discuss user mobility in GPRS taking our specific approach of perceived mobility into consideration.

7.2 Network perceived mobility

If mobile stations move through a cellular network they cause cell reselection, that is, the network and the mobile station agree on a new cell that is serving the mobile station, and the mobile station is switching to this new cell. But many movements of the mobile station do not result in a change of the cell. Therefore, we need to differentiate between the true geographical movement of the mobile station and the movement as the network notices it by means of GMM messages triggered through cell reselections and routing area updates. We call the former case ‘geographical mobility’ and the latter case ‘perceived mobility’.

The *geographical mobility* is based on the actual geographical movement of the user. Figure 7-1 depicts the path of a mobile station, which is partly in ready state, through an area crossing several cells and routing areas. In order to measure the geographical mobility, exact location tracking of the mobile stations is required. Measurement of exact geographical locations can be done via GPS. All mobile stations need to be equipped with a GPS receiver for exact geographical measurements. Our measurement setup did not allow us to perform this kind of measurement. Instead, we focused on the perceived mobility. This is a fundamental different approach which we take in contrast to the papers mentioned in the section 7.1.

The *perceived mobility* is based on the cell reselections as noticed by the network. In section 3.6 we explained that the network maintains cell and routing area location information of the mobile station. Depending on the GMM state (cf., section 3.6), the network receives either only the routing area ID or also

the cell ID. A new location is either explicitly announced with the help of cell reselections (CR) and routing area updates (RAU) messages, or the new location is implicitly made available by e.g., session management events like PDP context activation, which also carry the cell ID. That is, if the mobile station moves through the cell structure of the network, the geographical movement is mapped onto a series of cell IDs and routing area IDs (Figure 7-1). But, this mapping is not a one-to-one mapping, since different geographical movements can result in the same time series of cell IDs and routing area IDs.

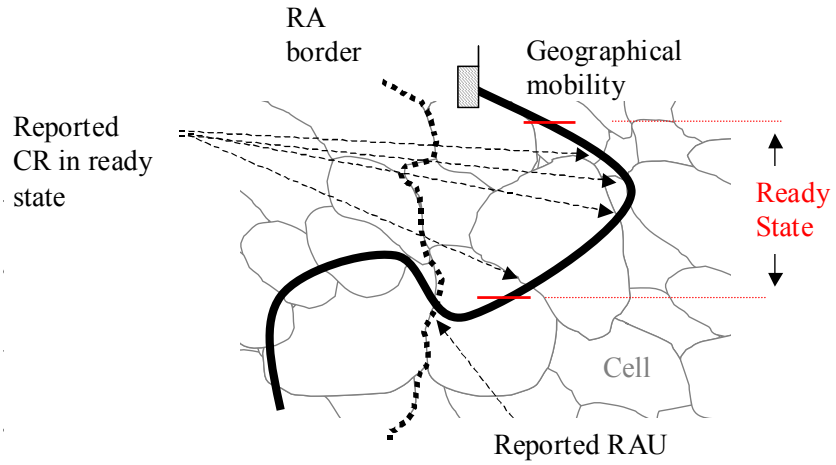


Figure 7-1: Geographical mobility

7.3 Metrics

In the following mobility investigation we focus on the perceived mobility as outlined in section 7.2. The metrics for determining the perceived mobility are based on counting GMM events during relevant time periods.

We do not only consider explicit cell reselection and routing area update events. Rather, we interpret all GMM events as measured by the GMM measurement setup described in section 4.1.2. Each time an event for the same user contains a different cell ID, we count this as a CR; and each time the event contains a new RA ID, we count this as a routing area update (RAU). Note that a new RA also implies a CR, as routing area boundaries are always on cell boundaries.

We consider three SM-/GMM- time periods for which we investigate the mobility metrics:

- the duration that a mobile station is GPRS attached;
- the duration that a mobile station has an active PDP context;
- the duration that the mobile station is in ready state.

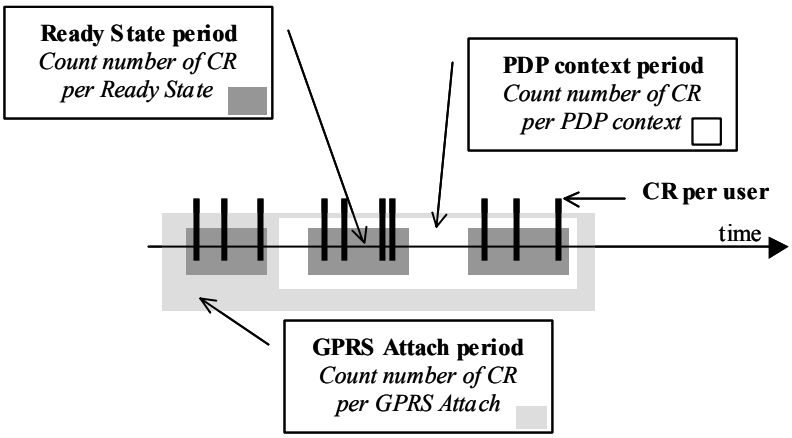


Figure 7-2: GPRS periods considered in mobility investigation

Figure 7-2 depicts how the different time periods relate. Within each time period the number of CR or RAU are counted. The number of CR within PDP contexts also includes all CR within ready state periods and in-between. The number of CR within a GPRS attach period also includes all CR within PDP contexts inside the attach period and in-between PDP contexts. The same applies for RAU. Note that ready state periods also appear outside of PDP contexts, if signaling messages are transmitted.

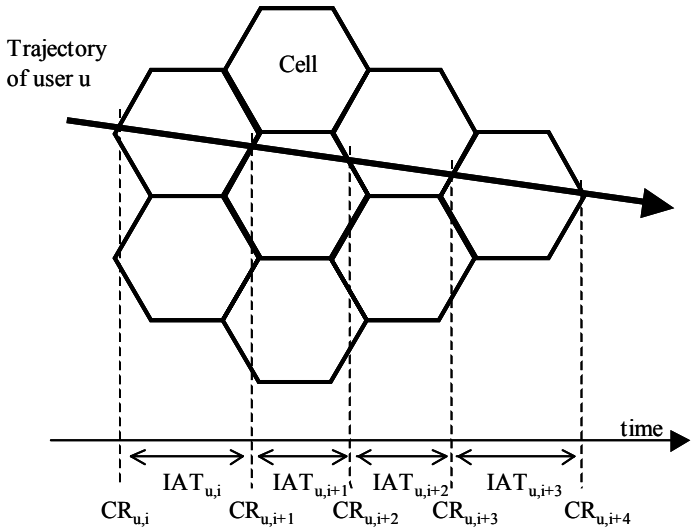


Figure 7-3: Mobility metrics

Figure 7-3 depicts the resulting time series of cell reselections $CR_{u,i}$ from the movement of user u. $CR_{u,j}$ denotes the time stamps of the CRs. The time between consecutive cell reselections i and i+1 for user u is denoted by $IAT_{u,i}$. An analog definition applies for $RAU_{u,j}$.

Derived from these basic metrics we investigate the number of *unique cell reselected* (UCR) and the number of *unique routing area updated* (URA). UCR and URA are related to a specified time period. A UCR is the ID of a target cell[50] of a CR that has not previously been visited in the defined time period.

[50] The target cell is the cell at which the mobile station arrives after a cell reselection.

We define the number of UCR (#UCR) as the number of unique cell IDs visited within the given timer period. That is, each cell ID is only considered once in the time period. We also count the starting cell, that is if #CR=1, than #UCR=2. Correspondingly, the number of URA (#URA) is the number of unique routing area IDs visited within the time period.

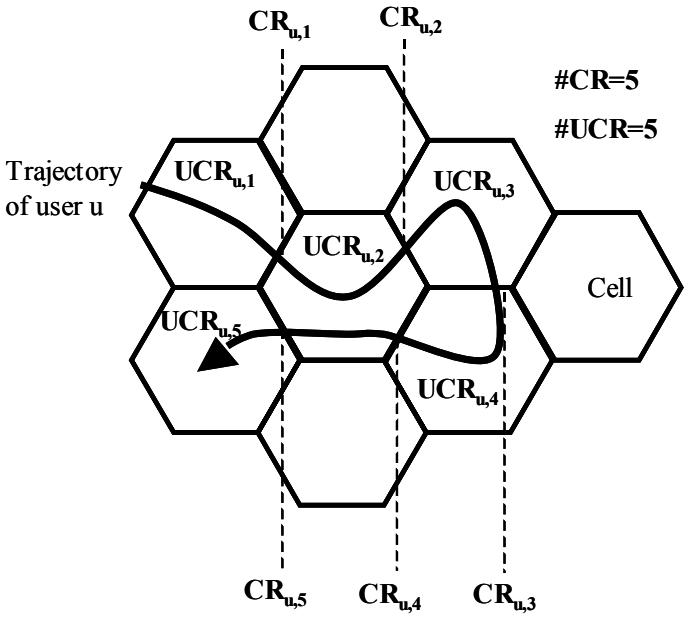


Figure 7-4: Unique cell reselections concept

Figure 7-4 depicts an example to illustrate the UCR definition. User u moves through the indicated cells, visiting the middle cell twice. In total we measure $CR_{u,1}$ to $CR_{u,5}$ that is #CR=5. But $CR_{u,4}$ does not result in a UCR, because the cell was visited previously. Therefore we measure $UCR_{u,1}$ to $UCR_{u,5}$ which corresponds to #UCR=5, because we visited 5 unique cells.

The #UCR and #URA can be used to derive the spatial mobility. By spatial mobility we mean the area in which a mobile station is moving. We assume that each cell covers a different geographical area.[51] Therefore, a higher number of UCR or URA indicates large-area mobility of the user. In particular the UCR and URA can be used to filter out circular movements and pathological artifacts in which the mobile station selects two cells alternately. The latter can happen if a mobile station is close to a cell boundary and the signal strength varies due to shadowing. In this case the mobile station might alternatively select one of two or more cells. Considering only CR in our analysis, would lead to an indication of high mobility, where as the UCR metric removes the 'noise'.

7.4 Limitations of the measurement approach

If the user moves through the cells of the network, the geographical movement is mapped onto a series of GMM events. But not all CRs by mobile stations are visible to the network. As explained in section 3.6.2, the mobile station sends out a GMM CR messages only if it is in ready state. That implies we only

[51] This might not be the case if one cell is an overlay cell to another cell.

measure user mobility if the user actually crosses the cell boundary and if the user is also in ready state at that moment. If the mobile station is moving to a new cell, but is not in ready state, it is not informing the network about this cell reselection.

The limits of this approach can be derived by investigating the dependency between cell size and user mobility on the number of reported cell reselections.

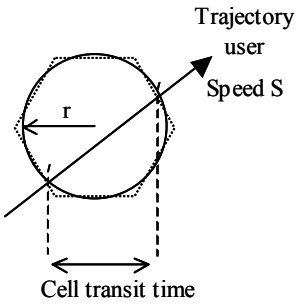


Figure 7-5: Cell transit time

Figure 7-5 depicts an idealistic cell with a radius r and the trajectory of a user through the cell with speed S. The cell transit time, that is the time it takes for the user to move through the cell, is in the case of a straight linear movement through the origin r/S. But if the user is not moving linearly, the cell transit time could differ. It would be shorter, if the egress point is closer to the entry point, or longer if the users move in a curved fashion through the cell.

Based on the assumption of the linear movement through the cell, Table 7-1 lists approximate cell transit times for a specific cell radius and user speed. The speeds correspond roughly to walking, and driving in a city, rural area or on a highway as well as traveling by train. The values indicate that the mobile has to move quite fast in order to have low cell transit times. However, in [HSSK01] it was shown that many vehicles in fact do not move very fast.

Cell radius		250 m	750 m	2100 m	6000 m
Speed	3 km/h	300 s	900 s	2520 s	7200 s
	25 km/h	36 s	108 s	302 s	864 s
	50 km/h	18 s	54 s	151 s	432 s
	100 km/h	9 s	27 s	76 s	216 s
	180 km/h	5 s	15 s	42 s	120 s

Table 7-1: Theoretical cell transit times

The mapping from geographical mobility to perceived mobility is more precise if the cells are small. The maximum radius of a GSM cell is about 35 kilometers. But typical for today's networks are much smaller cell sizes. Especially in urban areas, the cell sizes go down to a few hundred meters.

Based on cell layout plans for the network in which we did the investigation on mobility, we derived the approximate cell radii for all cells. We assumed a circular cell shape for all cells.

In Table 7-2 some percentile values for cell radii in the measured network are presented. 50% of the cells are smaller than 750 meters and 99% of the cells are smaller than 6000 meters. This statistic indicates that the mapping from the geographical mobility to cell reselections is in general not very coarse.

Cell radius	CDF Percentile
250 m	10 %
750 m	50 %
2100 m	90 %
6000 m	99 %

Table 7-2: GPRS network cell size statistics

Another aspect limiting the approach is the length of the time periods in determining the resolution of the mobility events. Short time periods can report only few mobility events, while longer time periods potentially report more events. We note from section 4.2.2 that the GMM measurements are limited in their length, which strongly influences the maximum length of the time periods.

For our specific GMM measurements, we list in Table 7-3 key statistics of the length of the three time periods which were mainly considered: GPRS attach, PDP context and ready state. Comparing the statistics on the length of the PDP context period with those in Table 5-4 reveals that the PDP length statistics obtained from the traces for the mobility investigation is truncated in the tail. This is due to the shorter GMM measurement period of only 3 days. We can expect that this truncation also influences the CR inter-arrival time distributions. On the other hand, it is not possible to compare the length distribution for the GPRS attach and ready state periods, as this could not be measured on the Gi interface, and is therefore only available for GMM measurements. But the same truncation must be true for the attach period statistics, as they can be assumed to be generally longer than PDP contexts. On the other hand ready states are so short, that truncation might not have a strong impact.[52]

Statistic	Attach [sec]	PDP context [sec]	Ready State [sec]	Ready State in PDP context [sec]
Mean	17306.9	471.039	68.1866	145.279
Variance	1.01E+09	1.52E+07	115145	370294
Std	31738.1	3899.41	339.331	608.518
10%til	27.945822	4.732089	43.996479	44.007298
50%til	2274.601631	47.545436	45.517130	72.676934
80%til	29627.20994	282.581747	54.821472	182.430265
90%til	52308.20704	570.220584	83.211161	293.714629
98%til	126703.8431	2651.820899	238.126723	698.364539
99%til	157713.1787	6475.109385	371.601203	951.207560

Table 7-3: Length statistics for mobile investigation periods trace GMM_5_6

[52] In fact we experimented with varying measurement length and did not see significant impact on the distribution of the ready state length.

Therefore, the results presented in the following sections are valid only under the constraint of the length statistics presented here. However, we assume that the conclusions drawn from the mobility investigation are still useful, as we only truncate time periods greater than 3 days.

Another aspect needs to be considered from section 5.3, which states that the used application is correlated with the length of the PDP context. That is, as the length of the PDP context is correlated with the application, we must also believe that the mobility metric is correlated with the application.

The argumentation in this section outlined that we are limited in deriving the true geographical mobility. Considering the perceived mobility definition from the previous section, we see that the metric *number of CR per time period* T, with $T \in$ {GPRS attach, PDP context, ready state} depends on

- the cell layout;
- the speed (and geographical movement) of the user;
- the length of the considered time period T;
- and the length depends on the type of application used

Despite this limitations, we have still a novel measurement of the perceived mobility at hand. This is still important, because the perceived mobility reflects the events, due to mobility, that can have an impact on the data transmission and the load on network. On the other hand geographical mobility that is not mapped onto perceived mobility, e.g., if a mobile station changes to a new cell while in stand-by state, does not cause interruptions in data transmission and additional packet delay or signaling load. Therefore, by modeling the perceived mobility we are able to understand the impact of user mobility on the network in terms of signaling load or the impact on QoS in terms of packet delay and throughput.

7.5 GPRS user mobility report

In this section we investigate two high-level aspects of user mobility. In section 7.5.1 we evaluate the general mobility of the mobile stations. This reveals to what degree users are actually mobile while using GPRS. In section 7.5.2 one approach on relating the mobility events to the actual spatial movement is presented. The results allow us to judge whether the perceived mobility stems actually from geographical mobility or is due to some GMM artifact.

7.5.1 General mobility

As a first step we are interested to which degree perceived user mobility is visible. Table 7-4 and Table 7-5 show the number of cell reselections (#CR), the number of routing area updates (#RAU), unique cells reselected (#UCR) and unique routing areas updated (#URA) within the three considered time periods: GPRS attach, PDP context and ready state. For example 79.13% of all PDP contexts report 0 CR events during its activation time and only 1.79% report more than 7 CR.

We list only the #UCR and #URA with the precondition of #CR=1 or #RA=1. For instance $Pr(\#UCR=5|\#CR>1)=2.94\%$ for PDP contexts. Therefore the #UCR

starts at 2, because in each CR, 2 cells are involved. On the other hand, #URA starts at 1 in our measurements. This is due to events that trigger an RAU with the old RA (e.g., periodic routing area updates (PRAU)). We filter these out in the analysis in section 7.7.

Period	GPRS Attach		PDP Context		Ready State	
Events	#CR	#UCR	#CR	#UCR	#CR	#UCR
0	43.00%		79.13%		80.72%	
1	11.66%		9.02%		10.59%	
2	9.17%	35.72%	4.79%	68.13%	4.53%	74.60%
3	5.27%	17.56%	2.07%	15.21%	1.67%	14.05%
4	4.37%	10.80%	1.42%	5.68%	0.93%	4.50%
5	3.09%	7.32%	0.80%	2.94%	0.49%	2.14%
6	2.73%	5.20%	0.60%	1.78%	0.32%	1.28%
7	2.04%	3.85%	0.38%	1.20%	0.19%	0.87%
>7	18.67%	19.56%	1.79%	5.06%	0.56%	2.56%

Table 7-4: Cell reselection statistics

Period	GPRS Attach		PDP Context		Ready State	
Events	#RAU	#URA	#RAU	#URA	#RAU	#URA
0	85.98%		98.14%		98.24%	
1	3.54%	25.28%	1.01%	54.42%	1.30%	74.08%
2	2.71%	44.45%	0.38%	36.12%	0.31%	24.45%
3	1.32%	16.18%	0.16%	5.93%	0.10%	1.28%
4	1.32%	8.28%	0.09%	2.49%	0.03%	0.16%
5	0.65%	3.62%	0.05%	0.69%	0.01%	0.03%
6	0.77%	1.47%	0.04%	0.24%	0.01%	0.00%
7	0.41%	0.48%	0.02%	0.07%	0.00%	0.00%
>7	3.28%	0.25%	0.11%	0.06%	0.00%	0.00%

Table 7-5: Routing area update statistics

Based on the results in the tables above, the time periods can be broadly grouped into stationary periods (gray-shaded cells), these are periods in which no mobility event at all is observed, and mobile periods in which cell reselections are observed. In the case of PDP contexts and ready states, about 20% of the PDP context and ready states belong to the mobile group as they contain some cell reselection. But *of those* PDP contexts and ready states, about 50% contain only one cell reselection. The overall perceived user mobility appears to be low. However, these results seem to be in line with results from mobility measurements for GSM voice.[53] That is, the general user mobility in GPRS is similar to the user mobility in GSM.

One main factor influencing the perceived mobility might be the length of the considered time period. Considering for instance that 90% of the PDP contexts are shorter than 570 seconds, the mobile station's speed must be in the range

[53] This is based on information by the operator and not confirmed by measurements.

of the values indicated by the gray-shaded cells in Table 7-1 to yield 1 or more CR.

We can conclude, when modeling the impact on performance through user mobility, that 80% of all PDP contexts and ready states are not affected.

As attach periods are much longer than PDP contexts, they naturally capture more cell reselections. But most of those cell reselections are outside of the PDP context and hence have no impact on data transmission performance.

Furthermore, when it comes to routing area updates, typically in less than 2% of the PDP contexts and ready states, a routing update must be considered as having an impact on performance. This is certainly related to the fact that routing areas cover a wider geographical area.

7.5.2 Spatial mobility

In the previous section we showed that the perceived user mobility is in general low. This needs to be considered when modeling the impact on performance. We now separate further the perceived mobility into GMM events resulting from geographical (spatial) mobility and GMM events without 'true' mobility triggering this.

CRs do not only occur due to geographical movements. A mobile station is usually in the position to receive the signal by several base stations. Therefore the mobile station might also change into neighboring cells if they provide better radio conditions. For instance, shadowing and multipath fading might considerably change the radio conditions, which might be due to minimal movements of the mobile station itself or of movements of objects in the surroundings. In such a case the mobile station might reselect its cell without intended geographical movement of the user.[54]

Depicting the number of UCRs ($\#\mathrm{UCR}$) can indicate the difference between the CRs due to true user mobility and those that can be ascribed to (local) radio condition changes. Figure 7-6 and Figure 7-7 depict the relation between the absolute number of cell reselections within PDP contexts and ready states, and the unique number of cells visited. They show density plots, displaying the frequency of $\#\mathrm{UCR}$ over $\#\mathrm{CR}$. The shaded area specifies the intensity, whereas darker shading means higher frequency of a particular combination. The frequency bar is in $\log(10)$-scaling. The dots represent particular combinations, which occurred very seldom and hence would not be covered by the density graph itself.

As one can see for the PDP context and the ready state period, two main areas in which the two variables $\#\mathrm{UCR}$ and $\#\mathrm{CR}$ lie exist (marked with A1 and A2). In area A1 the number of uniquely visited cells is almost as high as the number of total cell reselections. That is, they are almost linear proportional. In area A2 the cell reselections lead to a quite low number of uniquely visited cells. That is, a high number of CRs does not relate to a high number of UCRs.

[54] The MS and BTS try to avoid frequent reselection by using a hysteresis function when selecting a new cell.

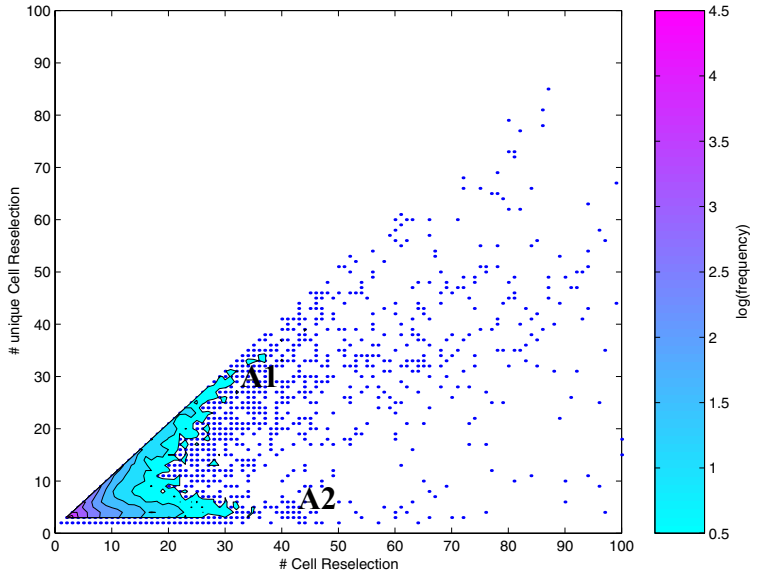


Figure 7-6: Cell reselection versus unique cell reselection – PDP context

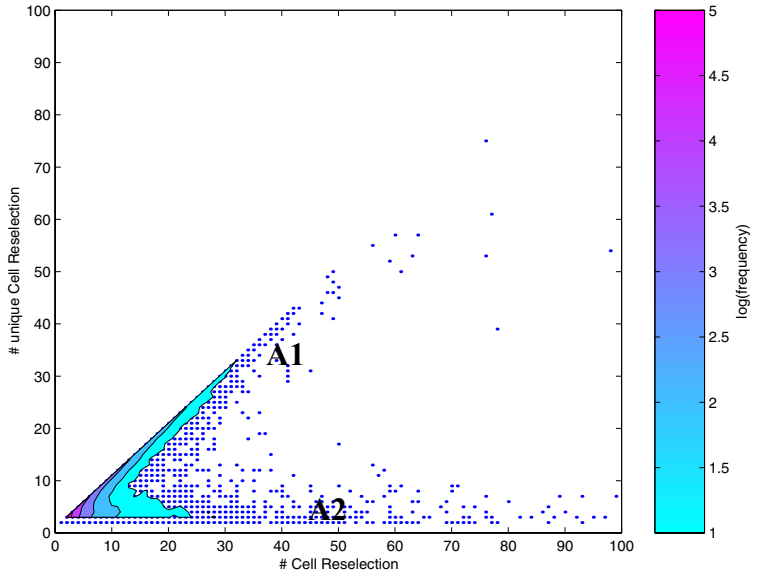


Figure 7-7: Cell reselection versus unique cell reselection – Ready state

These two areas might reflect two different mobility groups. In the group in which the high number of cell reselections leads to a high number of uniquely visited cells (A1), the user might be truly geographically mobile, as he traverses typically many cells. The other group with a low number of uniquely visited cells (A2), might indicate users residing close to a cell border and experiencing the described local effect.

This grouping of the users can be used to separate ‘true’ mobile users from ‘artifact’ mobile users. Though both cell reselections have the same impact if one considers performance impact through outage times, the users might behave differently with respect to service usage. Assuming that an environment in which the user is mobile influences his choice of application, the first group having a high number of uniquely visited cells would be of higher importance to be considered. For this reason we will use the number of uniquely visited cells

as mobility degree indicator in the next section, where we present the combination of application usage and user mobility.

7.6 Application usage and mobility correlation

We are interested in which application might be most affected by user mobility. Therefore we investigate the correlation between user mobility and the choice of applications. This information is useful to form the right traffic mixture for dimensioning, depending on the assumed user mobility. Furthermore, applications used during phases when the user is mobile should be more robust against packet loss and delay than stationary used applications need to be.

We use the CORRELATOR tool to derive correlated information on mobility and application usage.

As we have shown in section 7.5.2, the number of uniquely visited cells per PDP context is a good indicator for true geographical user mobility. The correlation between applications used within a PDP context and the number of UCR therein is depicted in Figure 7-8. Again, one has to be aware that the length of the application session influences how many cell reselections can be observed.

Figure 7-8 depicts the fraction of PDP context with the specific number of UCRs that carry data from a specific application. That is, the figure depicts a histogram normalized[55] per application. For example, most PDP contexts (92.9%) which carry MMS application data have no CR (#UCR<2). Very few have two unique cells visited. On the other hand, PDP contexts which carry HTTP application data visit in 58% more than one unique cell. And 26.7% of all WAP flows are within a PDP context with two unique cells visited.

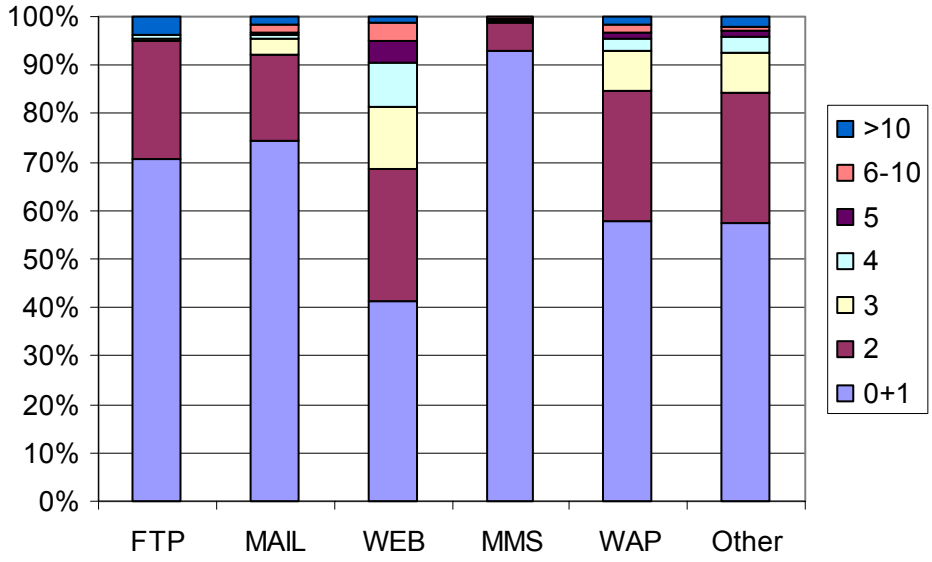


Figure 7-8: User mobility and application usage trace GMM_5.6 and GI_B7

[55] The bars per application sum up to 1.

The results depicted in Figure 7-8 allow a limited conclusion on the correlation between mobility within a PDP context and the used applications in the same PDP context. Especially the already indicated positive correlation between the length of the PDP context and the application used (see section 5.3) and the length of the PDP context and the #UCR (see section 7.4) limit the approach.

However, even if no clear correlation can be concluded, the results allow pointing out possible impacts from mobility on certain applications. According to the results in Figure 7-8, MMS applications are not affected in more than 90% of the PDP contexts. The same applies for FTP application data, which is seldom correlated with UCRs. On the other hand, Web and WAP application data transmissions are exposed to some mobility.

7.7 Modeling of cell reselection IAT

In this section we derive a distribution for the cell reselection inter-arrival times (CR-IAT). The cell reselection inter-arrival time is the time between consecutive cell reselections for one mobile station. In section 7.5.1 we have shown that 20% of PDP contexts and ready state periods contain one or more CRs. A description of the cell reselection inter-arrival times is needed to specify a mobility model for this group of 'mobile' PDP contexts.

We used the MLE, EM parameter estimation and KS test methods implemented in the **`DATAPLOT`** and the **`PH-fit`** tool, for deriving appropriate distributions for the CR-IAT presented in the previous section.

7.7.1 Empirical cell reselection inter-arrival time distribution

To ensure stationarity we have chosen one particular day (weekday – 24th of January 2003) from which we investigate the CR-IAT for the three periods: GPRS attach, PDP context and ready state.

We consider only inter-arrival times between CRs within the same time period per user. That is, for instance, we do not consider the inter-arrival times between the last CR in one PDP context and the first CR in the next PDP context, but only within the same PDP context.

Figure 7-9 depicts the empirical CCDF (log-log scaled) of cell reselection inter-arrival times. We show the curves for weekdays (WD) and for weekend days (WE) for all three periods. In all curves we have removed periodic routing area updates. Periodic routing area updates take place every hour, but do not directly relate to the mobility of the user (cf. section 3.6.2).

The curves appear to be similar, which we can expect when considering the relation between the periods (cf. Figure 7-2). We can recognize 3 regions for all 6 curves. The first slope is in the time span up to 45 seconds. That corresponds to the default time-out value for ready state periods. The next slope represents an almost straight line in the log-log scaled plots. This indicates a heavy-tailed distribution for this part of the curve. The final region is characterized by a sharp drop. This probably indicates that the distribution is truncated.

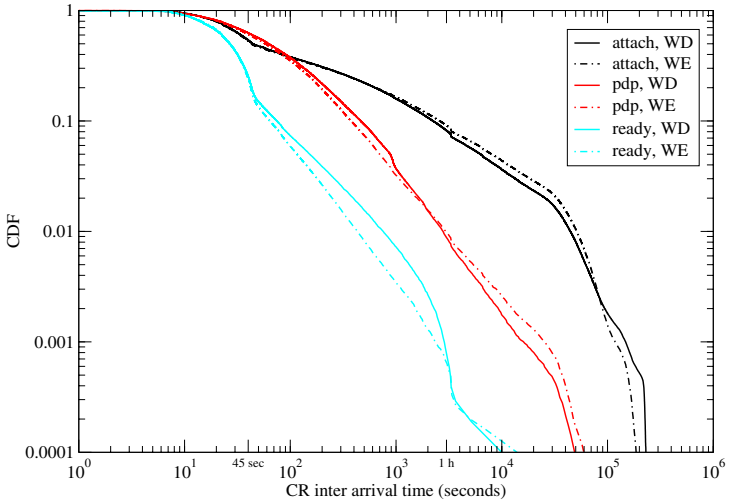


Figure 7-9: CR inter-arrival times – *PRAU* removed

The indication of a heavy-tailed distribution is in line with the results in [TP03], where the authors have shown that the cell reselection inter-arrival times for the university WLAN network is Pareto distributed. Therefore, we investigate the type of underlying distribution.

7.7.2 Data set validation

As a first step we investigate again the time series of the CR-IAT on its appropriateness for the estimation methods (cf. section 6.5.1). We performed the tests for all 8 data sets, but present here only the result from the CR-IAT for ready periods inside of a PDP context. Figure 7-10 depicts the results. In all but plot (b) and (c), the x-axis depicts the index of the sample in the time series. The tests for the 7 other data sets yielded similar results.

Figure 7-10 (a) provides a visualization of the time series of the CR-IAT. It already shows that high variance of the data values. Plot (b) shows the ACF for lag 0 to 100. The ACF shows quite low correlation around a value of 0.1. However, it failed the threshold, indicating independence, stated by equation (6.5.2). For our data set the threshold is in the order of 0.01. That is, we cannot clearly claim independence of the data values. However, the ACF seems to be reasonably low. Plot (c) shows the results of the DFT. Some low frequency shares are stronger visible, but in total no distinct dominating frequency parts are visible. Plot (d) depicts the moving average for a window length $l=100$ and $l=1000$. At this level still quite some fluctuation is visible. However, no clear singled-out trends or level shifts are visible. Finally plot (e) depicts the cumulative mean over the whole data set. After a short transit period in the beginning, the mean quickly stabilizes, and shows no visual trends.

Based on the visual inspection, we can conclude that independence and stationarity is a valid assumption for our investigation.

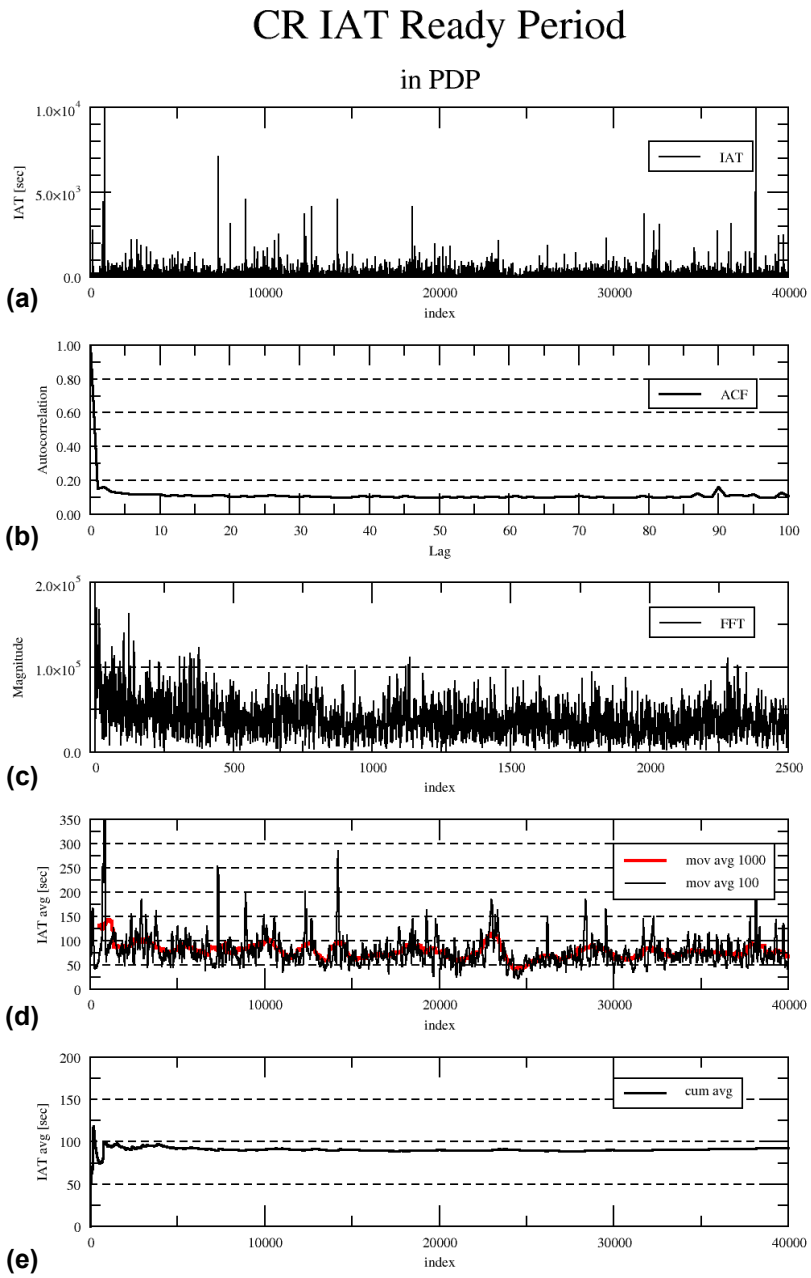


Figure 7-10: Data set validation – CR-IAT

7.7.3 Fitting of GPRS cell reselection data sets

As indicated in section 7.7.1, the distributions have different regions. In particular the distributions seem to be different below 45 seconds from the tail. Therefore we investigate additionally to the full data set a censored data set. In the censored data set all values below 45 seconds have been removed and all remaining values are shifted by -45 seconds. We do this to check the distribution of the tail of the empirical distribution. Furthermore, we investigate the ready period separately inside of a PDP context and outside of a PDP context. The ready periods are much longer inside of a PDP context, on the other hand it can be assumed the user is more mobile outside of an active data session (that is, outside of a PDP context). Therefore we assume a difference in the two distributions. This results in total in 8 different cases we investigate.

Table 7-6 summarizes the results for the 4 time periods. We present results for the tail of the empirical distribution (for values larger than 45 seconds) as well as for the total empirical distribution. Not shown here, but, the body of the empirical distribution is in all cases well fitted by an exponential or even a uniform distribution.[56] We show the mean, median and coefficient of variation (CV) statistics for the empirical distribution, the MLE distributions and the phase-type distributions. The results for the phase-type distributions are shown for the censored case above 45 seconds and for the complete empirical distribution.

Again we tested each empirical distribution against the normal, exponential, gamma, logistic, extreme value, Weibull, lognormal, and Pareto distribution. None of the distributions passed the KS test, as the KS statistic was always above the critical value. Therefore we show again the ranking of the distribution to choose the best fitting distribution among the listed ones. The lognormal, Weibull and the Pareto distribution always scored best. The exponential and normal distributions are often on the last position in the ranking. This is in accordance with other measurement-based results in [SHSK01] and [HSSK01] but contradictory to the, (still very often assumed) non-heavy-tailed distributions (e.g., exponential) for cell reselection inter-arrival times. For instance, the model in [ZD97] yields a gamma distribution, which is not heavy-tailed. Table C-2 in Appendix C lists the corresponding parameters for the fitted distributions. Figure 7-11 to Figure 7-18 depict the empirical curves together with the fitted distributions. For comparison reasons we have always included the lognormal and the Pareto distribution as well.

The IAT in the attach period cannot be well fitted with a single analytical distribution as listed in Appendix B. We assume that the attach period is truncated by considering only traces up to a length of 3 days. On the other hand, phase-type distributions of 4 phases match well the empirical distribution. Therefore no clear conclusion on the model can be given. However, as the attach period consists of the PDP contexts and Ready states, the IAT for the attach period can be modeled on the results for those periods.

The IAT in the PDP context can be well fitted with a lognormal distribution. Again, phase-type distributions of 4 phases match the empirical distribution very well.

The IAT in the ready state outside of PDP context can be very well fitted by Weibull or a Pareto distribution. In particular, the Pareto distribution shows a slope parallel to the empirical distribution tail (Figure 7-15). On the other hand, the tail of the IAT inside of PDP contexts cannot so well be fitted (Figure 7-17). However, phase-type distributions with 2 and 4 phases are good fits for both cases (cf. Figure 7-16 and Figure 7-18).

Note that in all cases the Weibull distribution had a shape parameter, clearly less than 1, indicating heavy-tailedness. That is, in total we can conclude that the IAT could be in all time periods also modeled by a heavy-tailed distribution.

[56] From the range of possible distributions, exponential and normal distribution scored similar good.

IAT		Distribution	Mean	Median	CV	KS-statistic
GPRS attach period						
Empirical	**>45 sec**		1814.317	278.393	3.109068	
MLE	**1st**	Lognormal	2805.582	263.6612	10.59377	0.02328
	2nd	Weibull	1167.563	462.8204	1.788304	0.047816
	3rd	Pareto	NA	85.50305	NA	0.088751
EM	**>45 sec**	PH 2-phase	1814.317	265.5841	2.145108	
	>45 sec	PH 4-phase	1814.317	284.2201	3.103335	
Empirical	**Total**		1033.994	59.19107	4.142358	
EM	**Total**	PH 2-phase	1033.994	76.58045	2.488403	
	Total	PH 4-phase	1033.994	68.54235	4.117064	
PDP context period						
Empirical	**>45 sec**		352.5137	84.931	4.774799	
MLE	**1st**	lognormal	341.2144	81.70586	4.054636	0.028258
	2nd	Weibull	286.5269	95.12877	1.788304	0.069495
	3rd	Pareto	NA	65.37869	NA	0.078469
EM	**>45 sec**	PH 2-phase	352.5137	96.4998	2.661588	
	>45 sec	PH 4-phase	352.5137	84.62244	4.847992	
Empirical	**Total**		246.7615	62.10886	5.326581	
EM	**Total**	PH 2-phase	246.7615	72.28768	2.828591	
	Total	PH 4-phase	246.7615	66.26319	5.48346	
Ready state period inside PDP context						
Empirical	**>45 sec**		102.6753	45.8665	3.305251	
MLE	**1st**	Weibull	98.23159	33.22646	1.385748	0.039162
	2nd	lognormal	128.6271	39.62178	3.088517	0.051096
	3rd	Pareto	318.405	57.60945	NA	0.071403
EM	**>45 sec**	PH 2-phase	102.6753	48.52507	2.01092	
	>45 sec	PH 4-phase	102.6753	45.65351	2.938037	
Empirical	**Total**		78.42429	39.45071	2.987085	
EM	**Total**	PH 2-phase	78.4243	42.31888	2.001227	
	Total	PH 4-phase	78.4243	41.62922	3.303074	
Ready state period outside PDP context						
Empirical	**>45sec**		147.3766	31.23615	8.260864	
MLE	**1st**	lognormal	169.9391	26.30533	6.382389	0.044308
	2nd	Pareto	205.008	56.32831	NA	0.047036
	3rd	Weibull	113.3968	36.39974	1.972437	0.05594
EM	**>45 sec**	PH 2-phase	147.3766	35.14794	2.787481	
	>45 sec	PH 4-phase	147.3766	31.62278	6.888168	
Empirical	**Total**		78.42429	39.45071	2.987085	
EM	**Total**	PH 2-phase	50.34963	22.97879	3.406847	
	Total	PH 4-phase	50.34963	22.66601	9.672481	

Table 7-6: Fitting results for cell reselection IAT.

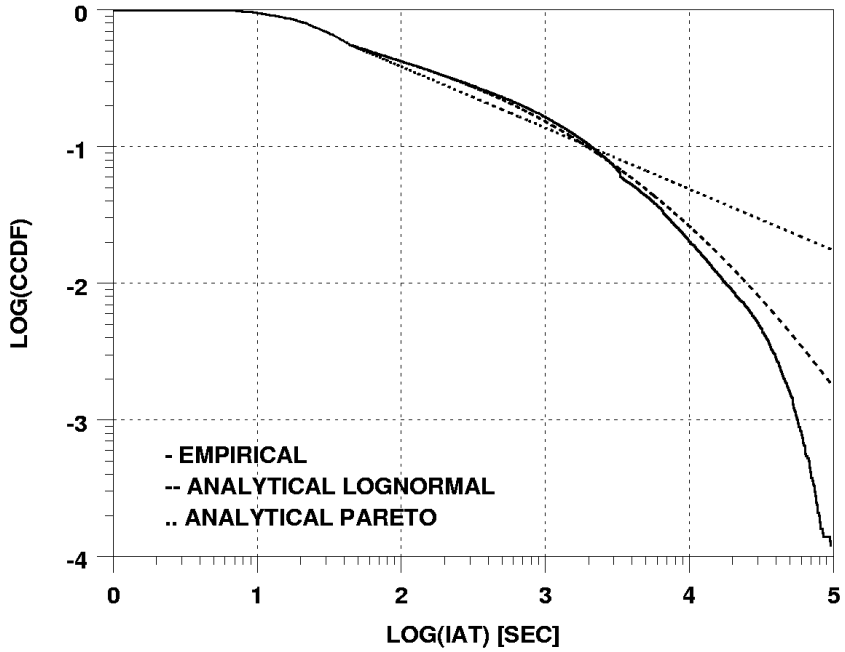


Figure 7-11: GPRS attach – CR-IAT distribution – lognormal, Pareto tail (>45sec)

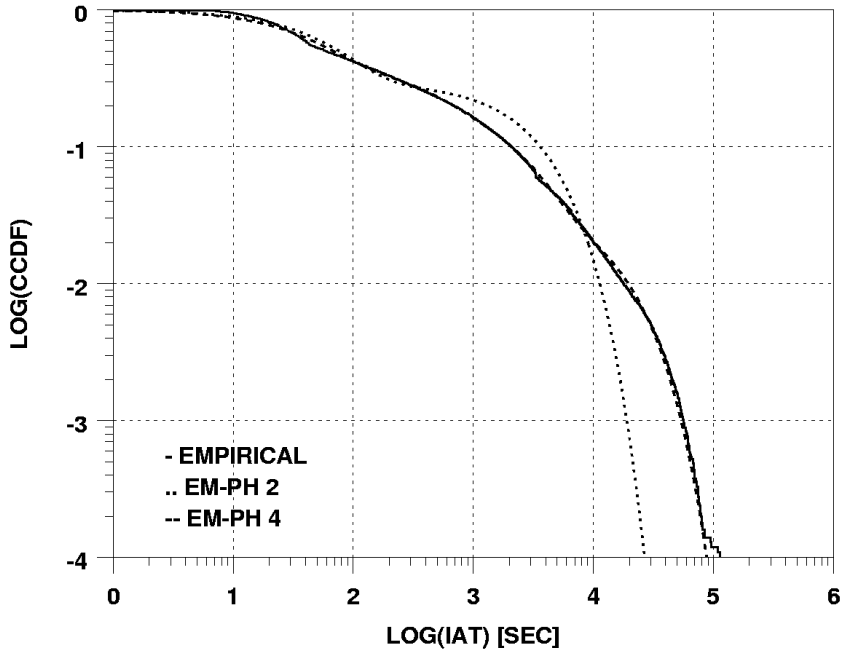


Figure 7-12: GPRS attach – EM PH fitted on CR-IAT distribution

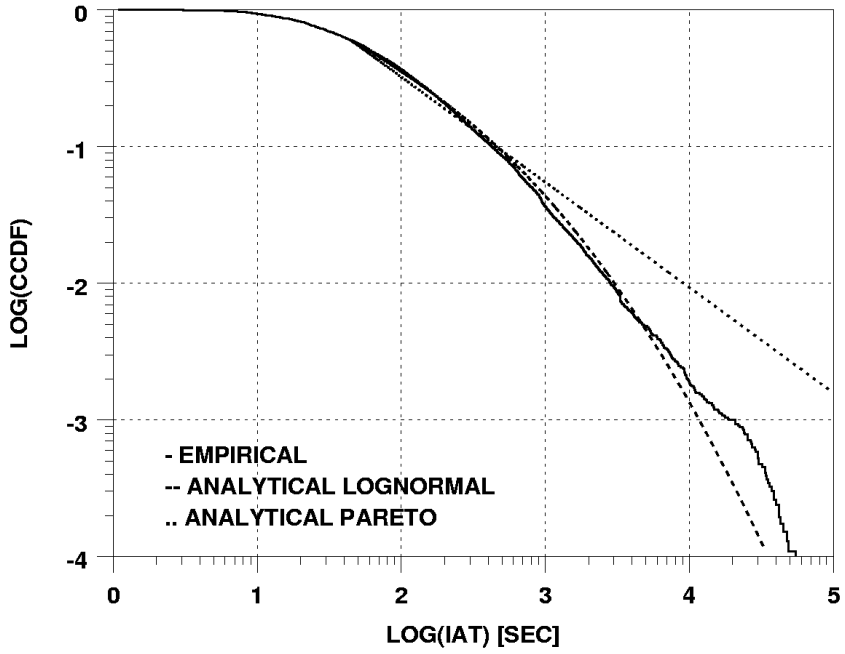


Figure 7-13: PDP context – CR-IAT distribution – lognormal, Pareto tail (>45sec)

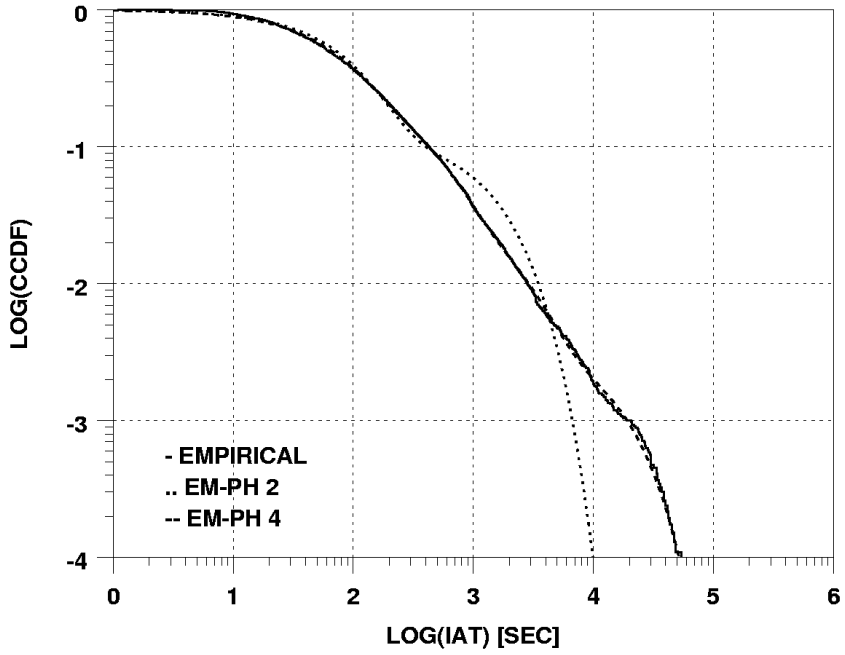


Figure 7-14: PDP context – EM PH fitted on CR-IAT distribution

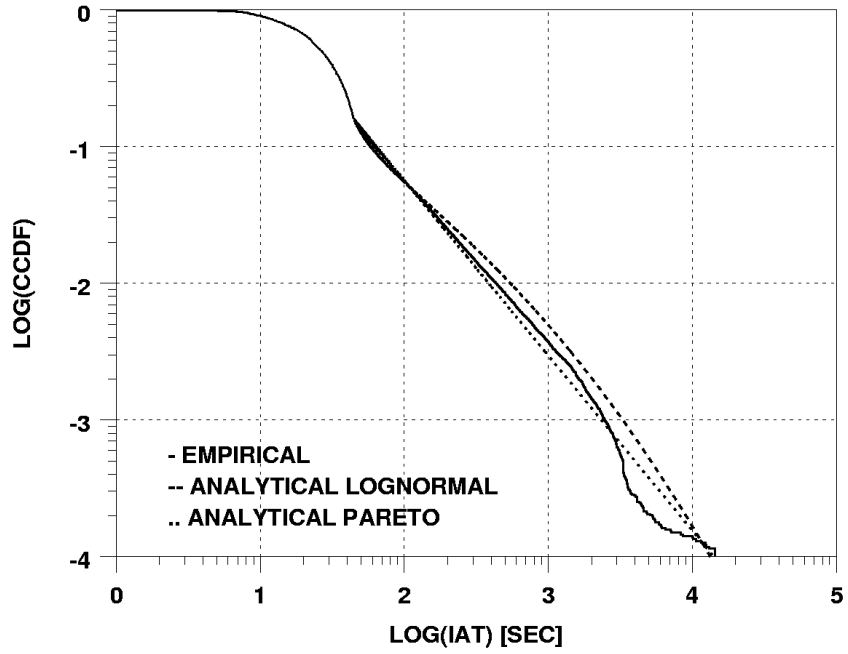


Figure 7-15: Ready state out PDP context – CR-IAT distribution – lognormal, Pareto tail (>45sec)

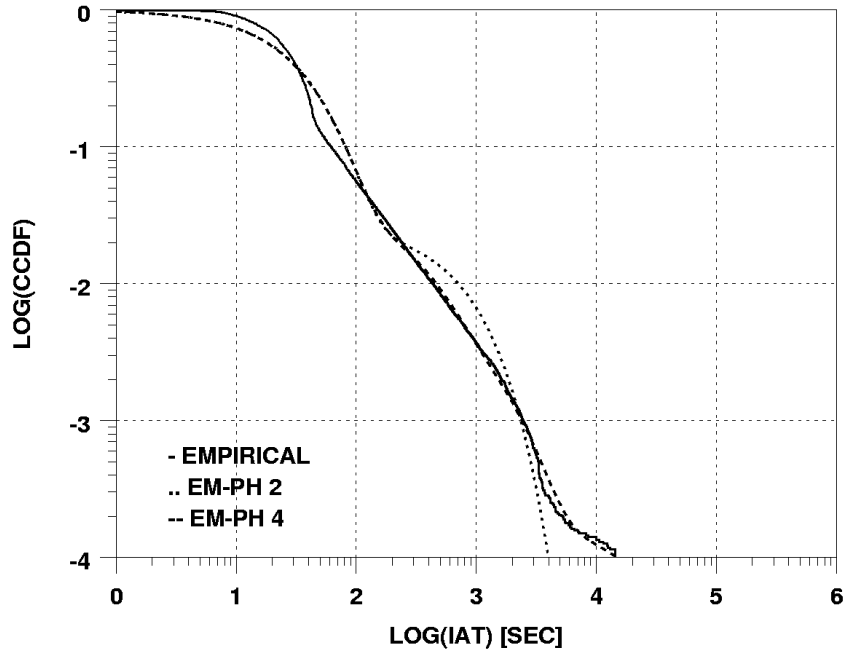


Figure 7-16: Ready state out PDP context – EM PH fitted on CR-IAT distribution

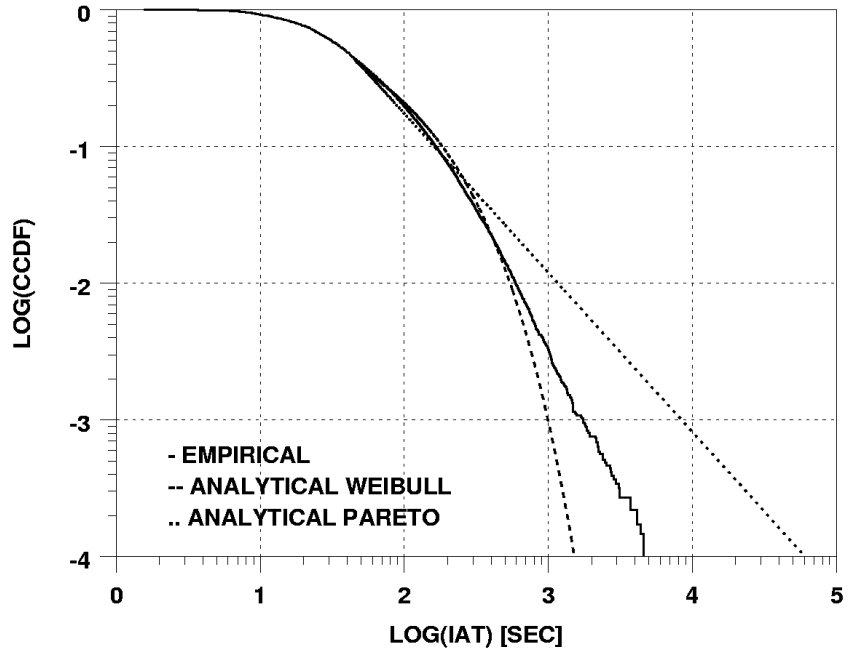


Figure 7-17: Ready state in PDP context – CR-IAT distribution – Weibull, Pareto tail (>45sec)

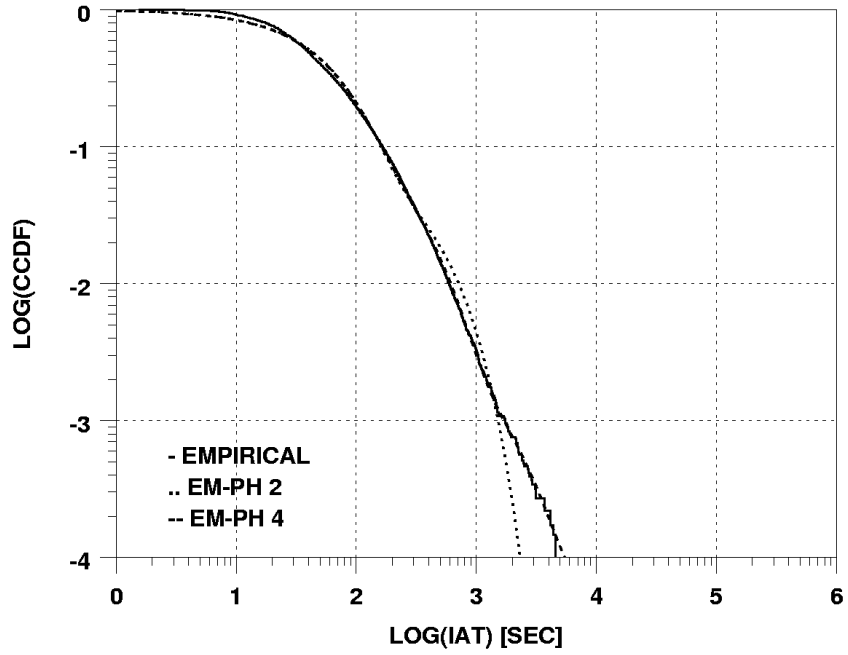


Figure 7-18: Ready state in PDP context – EM PH fitted on CR-IAT distribution

7.8 Conclusion

In this chapter we investigated user mobility as an important part of cellular networks. Our goal has been to describe user mobility with specific focus on its potential impact on performance.

Our data was derived from the GMM events in the GPRS network. Though this provides a unique view on the perceived mobility and, hence, allows modeling the impact on the network performance perceived by users, we also clearly lined out its limitation. This is in particular the strong correlation between the length of the considered periods and the visibility of the mobility events.

An important result is that impact from mobility on the transmission performance must be expected only in 2% of PDP contexts for the case of routing area updates (RAU) and in 20% for the case of cell reselections (CR). The same holds true for ready state periods. That is, the large majority of data transmission periods is not directly affected by mobility.

Assuming mobility, we provided results on the correlation between user mobility and application usage. Though these results are biased by the correlation between the length of the considered time period and the number of CR, we showed that long FTP sessions basically have no CRs, while long HTTP sessions encounter many CR. In addition, MMS, for which we showed long flow lengths in the previous chapter, does not encounter any cell reselection in more than 90% of the sessions, while WAP does. These are indications that usage of applications is indeed correlated with the mobility of the user. However, it requires more fine-grained mobility measurements to overcome the bias in the study. A combination of information on the true geographical mobility (for instance based on GPS measurements), together with the presented perceived mobility, seems to be a promising approach. Our setup, with the ability to correlate IP traces and mobility event information, could be also used in this case.

On the condition of mobility of the user, we derived the distributions describing the times between cell reselections. Heavy-tailed distributions fit well in all three considered time periods (GPRS attach, PDP context, ready state). In particular, inter-cell reselection times within ready states are well matched by a Pareto distribution. These results confirm findings in [TP03], in which the authors also found heavy-tailed distributions to be the best fit. Consequently, exponential inter-cell reselection time models, as proposed elsewhere [ZD97], fall short of a good model for mobility in GPRS networks. On the other hand we found a good fit for the PH-type distribution with 4 phases. They are not heavy-tailed, but match the *observed* empirical distribution quite well. These are good alternatives for analytical tractable models.

8 Traffic Self-similarity

In this chapter, we investigate the packet arrival process and the data volume arrival process from GPRS traffic on statistical self-similarity; in particular we focus on the WAP and the Web traffic.

Self-similar traffic has serious performance implications, as already mentioned in section 2.3.3.2. Consequently, it is important to understand the self-similar nature of the traffic in a network in order to apply the right methods for performance investigations and network dimensioning.

Self-similarity is regarded as an invariant of data network traffic as many studies have found evidence for self-similarity in various network types (e.g., [PF94], [LWTW94] and [WTE96]). However, GPRS traffic and, in general, cellular access network traffic has not yet been investigated on its self-similarity property. As we have seen, GPRS is not just a new access technology, it also introduces novel applications such as WAP and MMS, and provides Internet access in a nomadic or mobile user environment. This yields a special traffic composition, different from wireline Internet traffic. We have shown that up to 60% of the traffic volume in GPRS networks is UDP traffic (cf. chapter 5). This is in sharp contrast to the usual 80% of TCP traffic in the fixed Internet. Additionally, we have shown that WAP and MMS file sizes are in general much shorter than Web and FTP file sizes (cf. chapter 6). In particular we have shown that WAP flows, which contribute considerable to the total traffic mixture, is weakly heavy-tailed. How this influences the aggregated GPRS traffic is still an open question.

In section 8.1 we present the commonly used methods to test self-similarity based on estimating the Hurst value H. In particular we introduce the Abry-Veitch method, which is the most robust Hurst estimation method. In section 8.2 we present the results of the Hurst parameter estimation methods to assess the degree of self-similarity in GPRS for two classes of application traffic. Section 8.3 concludes the investigation of the self-similarity property of the GPRS traffic.

The results in this chapter are based on traces Gi_A18b and Gi_B10b.

8.1 Tests on self-similarity

Self-similarity represents the phenomenon that a stochastic process displays structural similarities across a wide range of scales. Loosely speaking, self-similar traffic looks the same in a wide range of (or at all) time scales. However, in the context of traffic traces we typically mean similarity in the distributional sense. Self-similar processes are highly bursty and keep this property over many aggregated time scales.

The degree of self-similarity is expressed by the Hurst value H; large values indicate stronger self-similarity (cf. section 2.3.3.2). Various methods for estimating the Hurst parameter H exist [Pop01]. These estimation methods can be grouped into time-domain based and frequency-domain based methods. For instance, the aggregated variance method, the R/S plot method, and the absolute moment method are time-domain based methods, while the periodogram method and the Abry-Veitch method both belong to the frequency-domain method. We explain in the following (section 8.1.1) the Abry-Veitch method in detail, as we apply this method to derive our results. Additionally, we briefly scetch the aggregated variance-, the R/S-, the absolute moments-, the variance of residual-, and the periodogram-method in section 8.1.2 to section 8.1.6. In section 8.1.7 we describe the tools we use to derive the Hurst parameter.

8.1.1 Abry-Veitch method

This method is based on the multi-resolution analysis (MRA) and the discrete wavelet transformation [AV98]. It is the most comprehensive and robust method for determining the scaling behavior of traffic traces. Its strength follows from the fact that the multi-resolution analysis itself has a structural affinity to the scaling process under study. That is, multi-resolution analysis itself exploits scaling, but transfers the complex scaling process to a much simpler wavelet domain, in which short range dependent (SRD) statistics can be applied to infer answers on the scaling of the process. As our results are mainly based on the Abry-Veitch method, we provide a more detailed overview of the wavelet method and the underlying principles of the wavelet transformation.

The wavelet transformation is in principle similar to the Fourier transformation in that it transforms a signal to another domain with specific basis functions [Gra95]. In the case of the Fourier transformation the basis function consists of sine and cosine functions. In the case of the Wavelet transformation the basis function is defined by a wavelet ψ. The big difference is that wavelets are localized in time and frequency, whereas Fourier transformed signals are stretched out in either time or frequency. With the help of the Fourier transformation the signal can either be only localized in time (time domain), but in this case no information about the frequency components is available; or, the signal can be localized in frequency (frequency domain), in which case no information about the time is available. In contrast wavelet transformed signals allow to specify the signal at the same time in the time and frequency domain (which is then denoted as scale). The wavelet-transformed signal provides a high time resolution and low frequency resolution for high frequency components, and a low time resolution and high frequency resolution for lower frequencies parts of the signal.

Definition: Wavelet

A wavelet is a function $\psi(t), t \in \mathbf{R}$ such that $\int_{-\infty}^{\infty} \psi(t)dt = 0$.

The wavelet is limited in its expansion in time and frequency, i.e. it is limited or decreases very fast in the time and frequency domain.

An important property is the number of vanishing moments. A wavelet is said to have N vanishing moments if

$$\int_{-\infty}^{\infty} t^k \psi(t) dt = 0 , \quad k= 0,1,\ldots N\text{-}1 \tag{8.1.1}$$

Important wavelets are the Haar wavelet, Daubechies wavelets and the Spline wavelet.

Definition: discrete wavelet transformation (coefficients)

The discrete wavelet transformation of a stochastic process $\{X(t), t \in \mathbf{R}\}$ is

$$d_{j,k} = \int_{-\infty}^{\infty} X(t) \psi_{j,k}(t) dt, \quad j,k \in \mathrm{Z} \tag{8.1.2}$$

$d_{j,k}$ are called the coefficients. Where $\psi_{j,k}(t)$ are 'dilations' by a factor 2^j and 'translations' by k units of $\psi(t)$. The factor 2^j is called scale and j is called octave.

The wavelet coefficients encode information differential of the process $X(t)$ between adjacent scales centered about scale 2^j and the time instant $2^j k$.

With the help of the multiresolution analysis, the wavelet coefficients can be computed extremely fast with a computational cost of only $\mathrm{O}(n)$ for n coefficients.

The coefficients have some special properties, which make the wavelet transformation so suitable for analyzing self-similar traffic: The wavelet coefficients of self-similar and long-range dependent (LRD) processes themselves exhibit self-similarity and LRD. Thus, this allows to study the scale invariance (of the self-similar and LRD process) in the wavelet domain. In particular, wavelet coefficients of self-similar and long-range dependent processes share the same fundamental properties: (i) stationarity at fixed scale (ii) short range statistical dependence and (iii) reproduction in the wavelet domain of the power-laws defining the scale invariance phenomena. In particular (ii) is the strength of this method, as it allows to deal with the wavelet coefficients by simpler SRD statistics, while still showing the LRD properties of the transformed process. Therefore, by using the wavelet method, we can leave the domain of LRD statistical matters and use established SRD statistical tools. Furthermore wavelet coefficients $d_{j,k}$ are the same for $X(t)$ and for $X(t)+P(t)$, where P is a polynomial of degree N-1 when ψ has N vanishing moments. Even if P is not exactly a polynomial function but can be closely approximated, the effect will be small [AV98]. Another nice property, which will not be further followed up on, is that the wavelet method not only has those properties for self-similar processes but also for $1/f$ noise, fractal process, multifractal process and multiplicative cascades [AV98].

Definition: Wavelet analysis

The coefficient $|d_{j,k}|^2$ measures the amount of energy in a signal X around the time $t_0=2^j k$ and about the frequency $f_0=2^{-j}\lambda_0$, where λ_0 is a reference frequency which depends on the wavelet ψ.

Using the fact that the time average

$$E_j = \frac{1}{N_j}\sum_k \left|d_{j,k}\right|^2, \tag{8.1.3}$$

with N_j the 'number of coefficients at octave j', is the amount of energy at octave j.

The wavelet analysis is based on the fundamental relationship:

$$\log_2 \mathrm{E}[E_j] = \log_2\left(\frac{1}{N_j}\sum_k \left|d_{j,k}\right|^2\right) = (1-2H)j + C \tag{8.1.4}$$

with

$$C \sim \frac{-1}{N_j \ln 2} \tag{8.1.5}$$

The Hurst parameter H can be estimated by a linear regression through $y_j = \log_2 \mathrm{E}\left[E_j\right]$ in the range $j=[j_1,j_2]$. The detection of scaling is performed by identification of region(s) of linear alignment of (j,y_j), and the determination of their lower and upper cutoff octaves j_1 and j_2

Important to note is that the wavelet analysis introduced above is, despite its name, only specified for continuous signals. If applied to discrete signals (as it is the case for packet time series from measurements) special care needs to be taken. [AVT00] provides a technique by which the discrete time series needs to be pre-processed to make it suitable for the wavelet analysis.

We use the tool **`LDcode`** [LDCODE] which provides a log scale diagram of (j,y_j), including confidence intervals and an estimator of the lower cutoff j_1. The **`LDcode`** tool automatically applies the technique proposed in [AVT00] for discrete time series.

Based on the log scale diagram it is possible to infer the type of scaling (self-similar, long-range, multiscaling, monoscaling, etc) and to estimate the corresponding scaling parameter H. We discuss various log scale-plots in section 8.2.3 using our data traces.

In the case of discrete time series, we note the following correspondence: if the initial resolution of the time series is t_0 (bin size), the time resolution at each scaling level j is $t_j=2^j t_0$. This follows automatically from the definition of the wavelet coefficients.

8.1.2 Aggregated variance method

The aggregated variance method is based on the slowly decaying variance property as stated in equation (2.3.21) [ENW96]. The method can indicate long-range dependency. The slope β of the straight line in a log-log plot, depicting the sample variance over the block size of each aggregation, is used to roughly estimating H. H is then given by $H = 1 - \beta/2$.

8.1.3 R/S plot method

This method is based on empirical observations by Hurst [PKC96]. It estimates H based on the R/S statistic, and then indicates (asymptotically) second-order self-similarity. H is (roughly) estimated through the slope of a linear line in a log-log plot, depicting the R/S statistics over the number of points in the aggregated series.

For a given set of observations $X=\{X_n,\ \mathrm{n}=1,2,3,\ldots\}$ with sample mean $\overline{X}(n)$, sample variance $S^2(n)$ and range $R(n)$, the rescaled adjusted range R/S statistic is given by

$$\frac{R(n)}{S(n)} = \frac{\max(0,\Delta_1,\Delta_2,...,\Delta_n) - \min(0,\Delta_1,\Delta_2,...,\Delta_n)}{S(n)} \tag{8.1.6}$$

where

$$\Delta_k = \sum_{i=1}^{k} X_i - k\overline{X}, \text{ for } k=1,2,..n. \tag{8.1.7}$$

It was empirically shown that

$$E\left[\frac{R(n)}{S(n)}\right] \sim cn^H, \tag{8.1.8}$$

where $n \to \infty$ and c being a positive constant.

8.1.4 Absolute moments method

This method is related to the aggregated variance method, but computed for the first moment [TT97]. The slope β of the straight line in a log-log plot, depicting the first moment of the aggregated block over the block size, provides an estimator for H, by $H = 1+\beta$.

8.1.5 Variance of residual method

This method uses the property that the variance of residuals (per aggregation level m) is proportional to m^{2H} [TT98]. For this, the time series is divided into blocks of size m, and within each block, the partial sums of the series are calculated:

$$Y(t) = \sum_{i=1}^{t} X_i, \tag{8.1.9}$$

and a least-square line (a+bt) is fitted to the partial sums. The sample variance of the residuals is computed by,

$$s_r^m = \frac{1}{m}\sum_{t=1}^{m}(Y(t) - a - bt)^2 \qquad (8.1.10)$$

This variance of residuals is computed for each block, and the median (or average) is computed over the blocks. The slop of a least-squares line, fitted to the logarithm of the median (or average) variance of residuals versus the logarithm of the aggregation level m, provides the estimate of H.

8.1.6 Periodogram method

This method is based on the 'power-spectrum singularity at 0'-property as stated in equation (2.3.22) [ENW96]. H can be estimated by the slope of the spectral density as it approaches 0. The slope is approximately $(1\text{-}2H)$. With $I_N(w)$ being the estimated spectral density of a stochastic process *X(t)*, plotting log $(I_N(w))$ versus $\log(w)$ and fitting a straight line to the curve for low frequencies yields H.

The periodogram can be estimated by a Fourier series operation over a time period N:

$$I_N(w) = \frac{1}{2\pi N}\left|\sum_{k=1}^{N} X_k e^{jkw}\right|^2 \qquad (8.1.11)$$

8.1.7 Testing approach and tools

We show results of our traces based on all mentioned methods. In particular, we will use the Abry-Veitch method to derive the scaling nature of the process. Though the Abry-Veitch method is the most comprehensive and robust method and therefore would be sufficient, we apply all methods in order to obtain comparable results to other papers in which self-similarity for wireline Internet measurements has been detected.

We use the **SELFIS** [SELFIS] tool for applying all mentioned methods, with exception of the Abry-Veitch method. For the Abry-Veitch method we use the **LDcode** from the authors of the wavelet method. In particular we derive the scaling nature of our investigated processes with the help of the Abry-Veitch method.

The **SELFIS** tool allows applying all of the Hurst estimation methods explained above. A shortcoming of the tool is that it takes no special care, as it is necessary for the methods and explained below, when estimating the Hurst value based on the linear line fitting.

All introduced methods result in some intermediate statistics, based on which the Hurst value is derived. For instance, in the case of the variance method, these are the aggregated variance values for each aggregation level; or, in the case of the Abry-Veitch method, the intermediate statistics used are the

wavelet coefficients. Based on those values, linear regression is used to fit a straight line to derive the Hurst value.

What all methods require is that typically the linear regression should *not* consider all of the values from the intermediate statistic. In the case of the R/S method, the variance method and the absolute moment method, it is recommended not to consider the results of the first few aggregations levels and neither the last few aggregation levels. The reason for this is that these values are not very reliable because either the aggregation level is too low (sampling too few points per block) or it is too high (sampling all points in just a few blocks). In the case of the periodogram method, it is recommended only to use approximately the first 10 percent of the results, close to the frequency 0. This is justified by the asymptotic LRD property close to the frequency 0. For instance, Figure 8-1 depicts the intermediate statistics $I_N(w)$ (depicted as dots) for one of our data sets. **SELFIS** applies linear regression through all points; this is depicted by the light line. Note how the many points in the body on the right influence the slope. The dark thick line is the linear regression we use, based on 10 percent of the points *towards* 0 (left side). The difference in the slope and hence the estimated Hurst parameter is clearly visible.

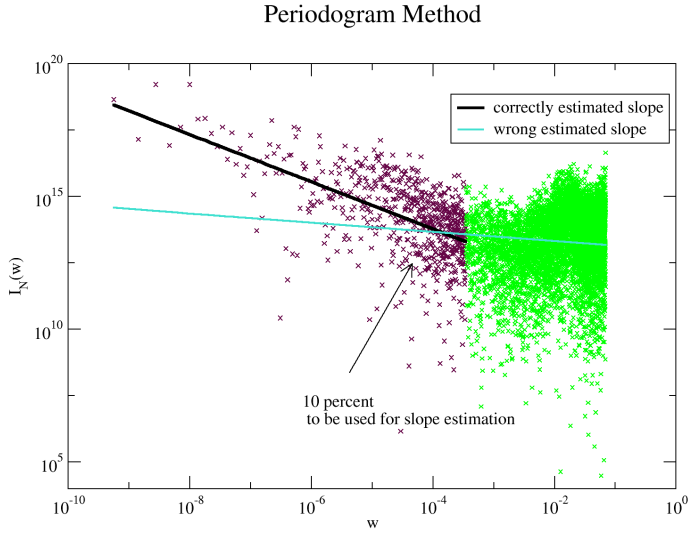


Figure 8-1: Linear fit for the periodogram method

As outlined above, the linear regression in the Abry-Veitch method should also be done only over the range $[j_1,j_2]$. The **LDcode**, which we use for the Abry-Veitch test, suggests an optimal starting point for j, based on the χ^2-goodness-of-fit test. And in the case of assumed LRD traffic, the regression line is fitted from this starting point to the largest available octave in the data. Furthermore, the **LDcode** provides a visual result of the scaling behavior allowing to judge the type of scaling. We will discuss the scaling behavior together with the results in section 8.2.3.

To overcome the problem with the **SELFIS** tool we only used it to derive the intermediate statistics, on which we applied our own linear regression, taking into consideration the constraints just noted. Therefore, we crop the intermediate statistical results to all but the first 2 and the last 2 aggregation

levels. In all cases applying the manual regression, the fit is very different than the **SELFIS** tool provides directly. The differences in the values of H between manually applied linear regression and the final results of **SELFIS** are sometimes quite large. Without manual adjustment the H values vary greatly from method to method for one trace, and in particular give quite different results for the Abry-Veitch method. On the other hand, the manual application yields very similar results. This stresses the importance of the recommended adjustments.

8.2 Analysis of stochastic processes

In this section we use the introduced Hurst estimation methods to test the GPRS traffic on its self-similarity property. In section 8.2.1 we present the specific data set that is analyzed. In section 8.2.2 we validate the data set on its appropriateness for the estimation methods. And in section 8.2.3 we present the results from the Hurst estimation methods.

8.2.1 Investigated stochastic processes

In our analysis we investigate the packet arrival process (PAP) and the data volume process (DVP). We generate the PAP from the original trace as a discrete time process by counting the number of packets and the DVP as the total number of bytes, both within a time interval (bin) of 100 ms. We present results from the trace GI_A18b and GI_B10b.

We are interested in the scaling behavior of the aggregated traffic of WAP traffic and of Web traffic. For this purpose we look at three 'sup-'sampled traces. First, we investigate the total aggregated traffic (up and downlink traffic), which we measured on the Gi interface. Next, we have split up the traffic into Web-oriented traffic and WAP-oriented traffic according to the APN that carried the data (see section 3.5 on APNs). We can do this separation because the APNs are typically assigned to different types of traffic by the operator. We checked our measurements for the used applications and found that indeed most of the traffic on one APN consists of Web-like applications, including HTTP, FTP, Email, etc. On the other APN we see mostly WAP-like traffic, comprising WAP and MMS. On rare occasions we see Web traffic on the WAP APN and vice versa. We denote the aggregated traffic sample as *AGG*-, the separated WAP-APN traffic as *WAP*- and the separated Web-APN traffic as *Web*-trace in the following analysis.

The GI_B10b trace is according to this rule split up into 70% traffic belonging to the Web-oriented class and 30% traffic belonging to WAP-oriented traffic class. The other trace from GI_A18b is split up into 25% Web-oriented traffic and 75% WAP-oriented traffic.

We investigated several busy hour periods from the GI_A18b and GI_B10b and other traces. All of the results were similar. We will discuss in the following only the results from one day in two networks.

8.2.2 Data set validation

Since all estimation methods (with exception of the Abry-Veitch method) require stationarity and often are very sensitive to underlying trends in the traffic process [KFR02], we investigated the chosen busy-hour periods with the same tests on trends and periodicity we have introduced in section 6.5.1.

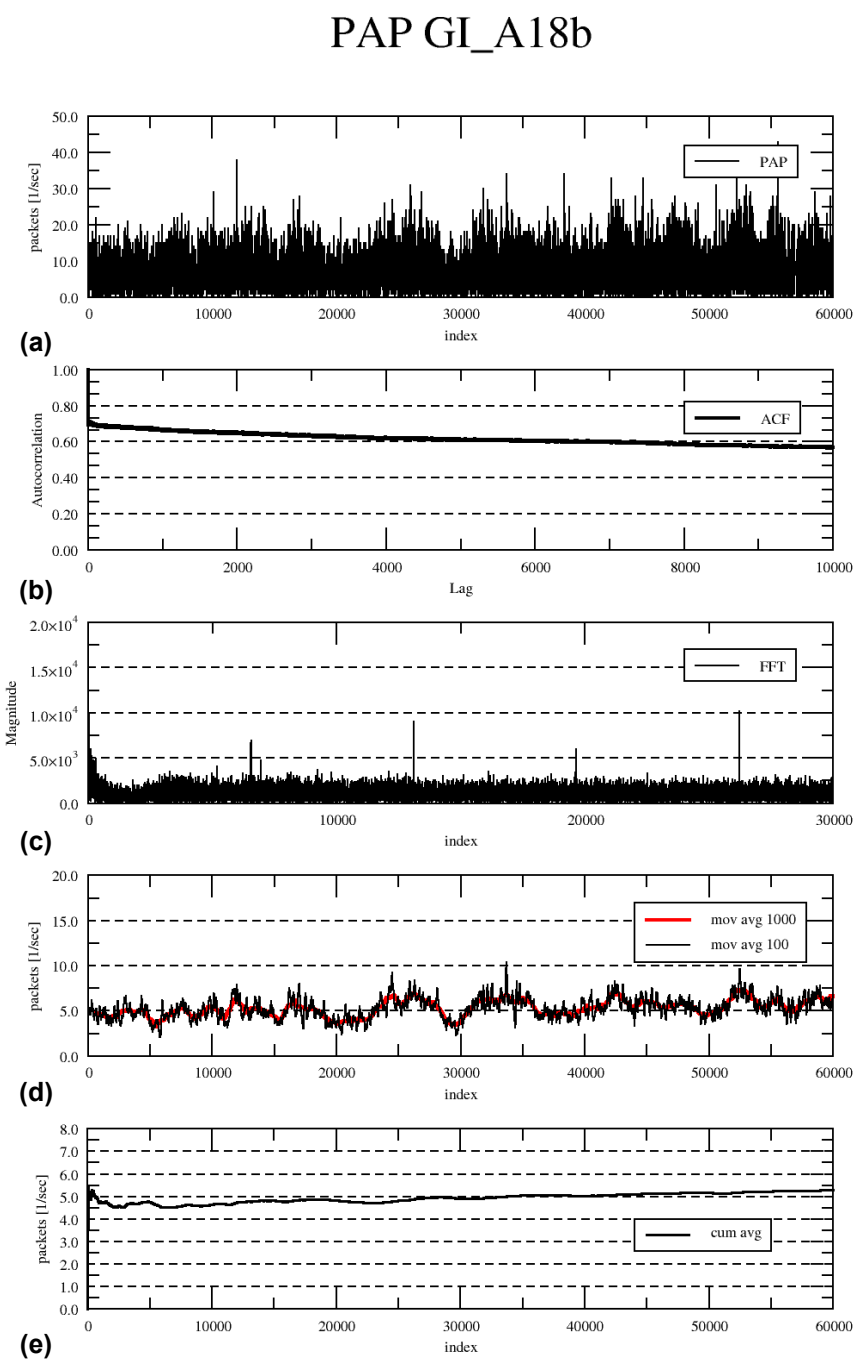


Figure 8-2: Data set validation for packet arrival process – aggregated traffic

Figure 8-2 depicts the test results. What can be noticed is that the autocorrelation function (b) does not drop close to 0 at lag 1 but instead stays high at about 0.6, up to lag 10000. This is already a first indication that we deal with LRD traffic, as the plot indicates a hyperbolically decaying ACF property,

described in equation (2.3.20). Also, the discrete Fourier transformation reveals a stronger upward trend for small frequencies (left hand side of (c)). At the same time it shows some peaks at distinct frequencies. This indicates some periodic generation process, which we could not filter out. However, as the peaks are not very strong we assume little influence. It should be also no problem, since we rely primarily on the Abry-Veitch method when estimating the Hurst parameter, and that method is robust with respect to periodicity [AFTV00]. The moving mean (d) and cumulative mean plot (c) also show some periodicity and a slight upward trend. However, both are small, and again the Abry-Veitch method is robust, up to polynomial trends of the order of the vanishing moments of the used wavelet. In our test we use the default proposed Daubechies wavelet with 3 vanishing moments. We also tried higher numbers of vanishing moments, which did not change the results. The test results indicate appropriateness of the data.

8.2.3 Verifying self-similarity of GPRS traffic processes

We have derived the Hurst parameter and the detailed scaling behavior for all 12 data sets,[57] based on the Abry-Veitch method. Furthermore, we obtained Hurst parameter estimations based on the aggregated variance method, the R/S method, the variance of residual method, the absolute moment method, and the priodogram method. In most cases the results have been very similar, and all results indicated some form of self-similarity or LRD.

PAP	GI_B10b			GI_A18b		
AV-Method	**H**	**Conf.**	**Scaling**	**H**	**Conf.**	**Scaling**
Agg	0.86	[0.76,0.95]	Figure 8-10	0.90	[0.81,0.98]	Figure 8-7
Web	0.83	[0.74,0.92]	Figure 8-10	1.02	[0.93,1.11]	Figure 8-7
WAP	1.06	[0.96,1.14]	Figure 8-9	0.89	[0.79,0.97]	Figure 8-7

Table 8-1: A-V method – Hurst estimation for packet arrival process

Table 8-1 shows the results for the estimated Hurst values using the Abry-Veitch method for the packet arrival process. The column labeled ‘H’ contains the estimated Hurst values and the column marked ‘conf.’ the confidence values for the estimated Hurst values. The column labeled ‘Scaling’ refers to plots describing the type of scaling. We discuss this below. Figure 8-3 and Figure 8-4 illustrate the Hurst values obtained from the different methods, for comparison. All values are very similar with Hurst values of about 0.8 and higher. This strongly suggests long-range dependency for the PAP for both measured networks.

In some cases the Hurst value is above 1, actually precluding LRD. However, the confidence intervals derived by the Abry-Veitch method include also values below 1, which still suggests LRD. Furthermore, at the end of this section we discuss the log scale diagram plots. Inspection thereof suggests in such cases still asymptotic self-similarity.

[57] We have two data sets, we investigate the PAP and DVP for each, furthermore we split up the trace in AGG, Web, and WAP traffic; this results in 12 data sets total to be investigated.

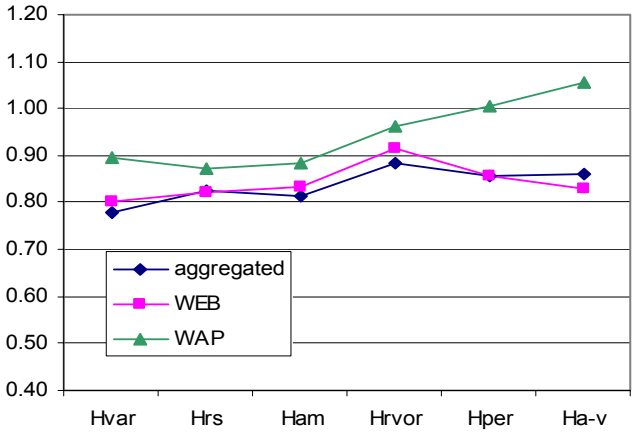


Figure 8-3: All Hurst values for PAP and GI_B10b

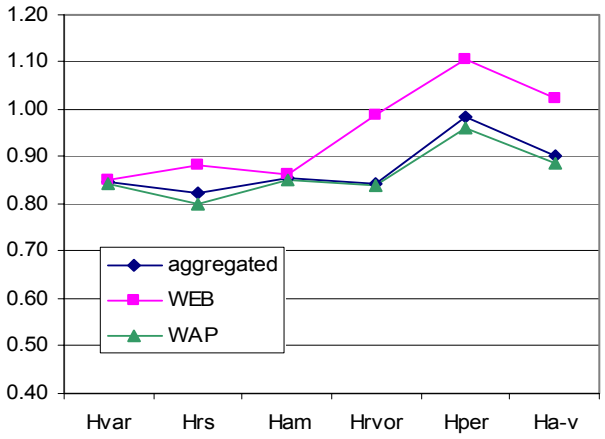


Figure 8-4: All Hurst values for PAP and Gi_A18

Hvar' stands for variance method, 'Hrs' for R/S method, 'Ham' for absolute moment method, 'Hrvor' for variance of residuals, 'Hper' for periodogram method and 'Ha-v' for Abry-Veitch method.

Table 8-2 shows the Hurst values for the data volume process. Again all results show LRD. For comparison we show in Figure 8-5 and Figure 8-6 the Hurst parameters estimated by the other methods.

DVP	GI_B10b			GI_A18b		
AV-Method	**H**	**Conf.**	**Scaling**	**H**	**Conf.**	**Scaling**
Agg	0.69	[0.65,0.72]	Figure 8-8	0.82	[0.73,0.90]	Figure 8-7
Web	0.68	[0.64,0.71]	Figure 8-8	1.07	[0.98,1.15]	Figure 8-9
WAP	0.92	[0.88,0.96]	Figure 8-9	0.81	[0.72,0.89]	Figure 8-7

Table 8-2: A-V method – Hurst estimation for data volume process

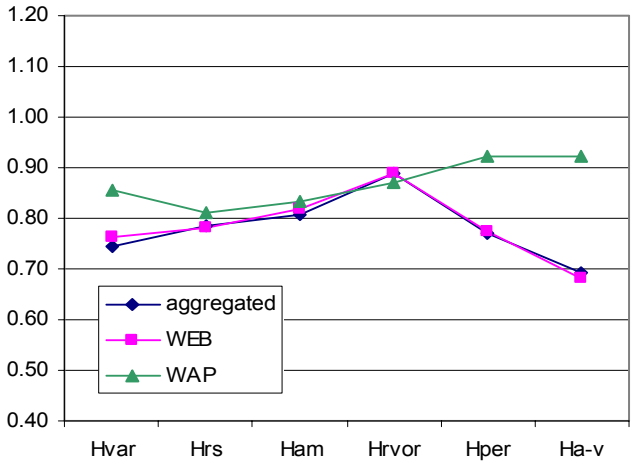


Figure 8-5: All Hurst values for DVP and GI_B10b

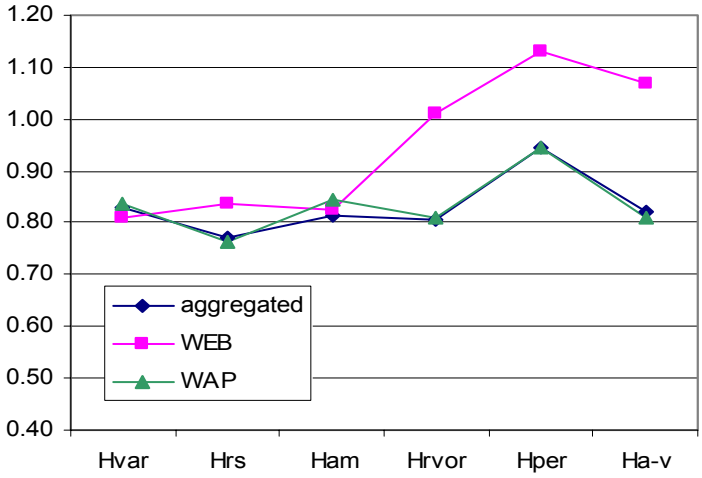


Figure 8-6: All Hurst values for DVP and GI_A18b

'Hvar' stands for variance method, 'Hrs' for R/S method, 'Ham' for absolute moment method, 'Hrvor' for variance of residuals, 'Hper' for periodogram method and 'Ha-v' for Abry-Veitch method

We already mentioned that all estimation methods, except for the Abry-Veitch method, assume an LRD model beforehand. That is, the Hurst estimation value can only be regarded as correct if the assumption of an LRD process holds. In contrast, the Abry-Veitch test is not based on such assumptions. It shows the scaling of the process for all time scales in the resulting log scale diagram. The true nature of the process (e.g., self-similarity, long-range dependency, multiscaling) is determined by interpreting the results in that diagram [AFTV00].

In our investigation of the AGG, Web, and WAP traffic we have encountered four basic log scale diagram types, depicted in Figure 8-7 to Figure 8-10. In Table 8-1 and Table 8-2 we list in the 3rd column for each process the plot that comes closest, respectively. The individual plots looked very similar to the exemplary plots, but with different scales on the y-axes, therefore we show only the principle type. The figures depict the log scale diagram; that is, $y_j = \log_2 \mathrm{E}[E_j]$ over the octave j (cf. section 8.1.1)

Figure 8-7 and Figure 8-9 both show a typical plot for LRD traffic. The second-order scaling starts from a certain octave on and continues until the largest available scale in the trace. For small scales we do not see second-order scaling behavior. We have found this scaling behavior for all WAP processes.

Figure 8-8 also exhibits at least LRD scaling, but actually has two scaling regions. One region from approximately 1 to 8 and one from 8 to the maximum octave. This is called bi-scaling. We have found this in the case of DVP for Web traffic in GI_A18b. Figure 8-10 depicts the case where the process has second-order scaling over all scales. This indicates strictly second-order self-similarity. We see this in the case of PAP for Web traffic also in GI_A18b.

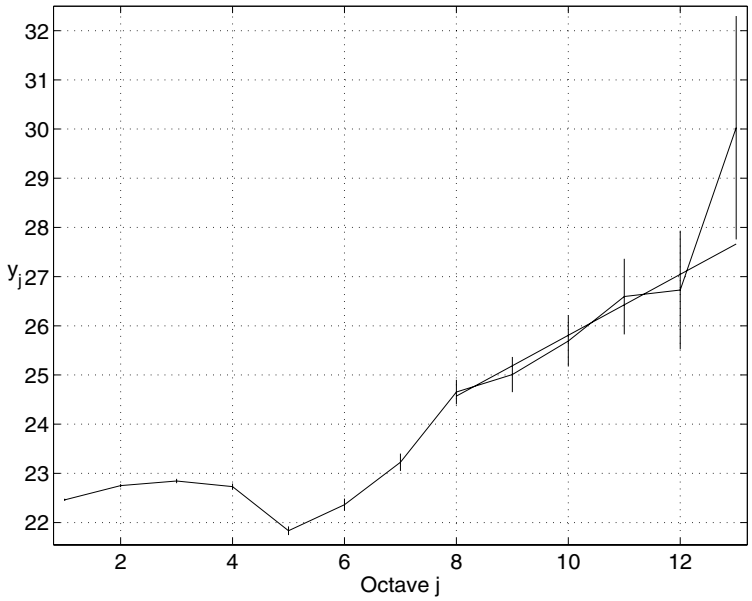


Figure 8-7:. **Typical plot for processes showing long-range dependency. Below a certain scale (j=5) no regular linear scaling exists. The linear part is divided into two scaling regions at j=8.**

We point out some interesting observations (cf. Table 8-1 and Table 8-2). First of all, the Hurst value of the aggregated traffic is always very close to the Hurst value of the majority of the traffic. This is in agreement with [PKC96]. In the case of GI_A18b the major part of the traffic is WAP traffic, in the case of GI_B10b it is Web traffic. More in detail, even the whole scaling behavior, as depicted by the log scale diagram, is very similar between the majority traffic and the aggregated traffic. This implies that by knowing the scaling of the majority traffic one obtains the scaling of the aggregated traffic as well. Second, the minor traffic has always slightly higher Hurst values, with changing roles of WAP and Web in the cases of GI_A18b and GI_B10b. We do not have an explanation for this. One reason might be that even if we have applied the estimation methods on separated traces per APN, it might not be possible to truly separate them, since they have both been traveling together through the GPRS network, thereby most likely affecting each other. This requires more research.

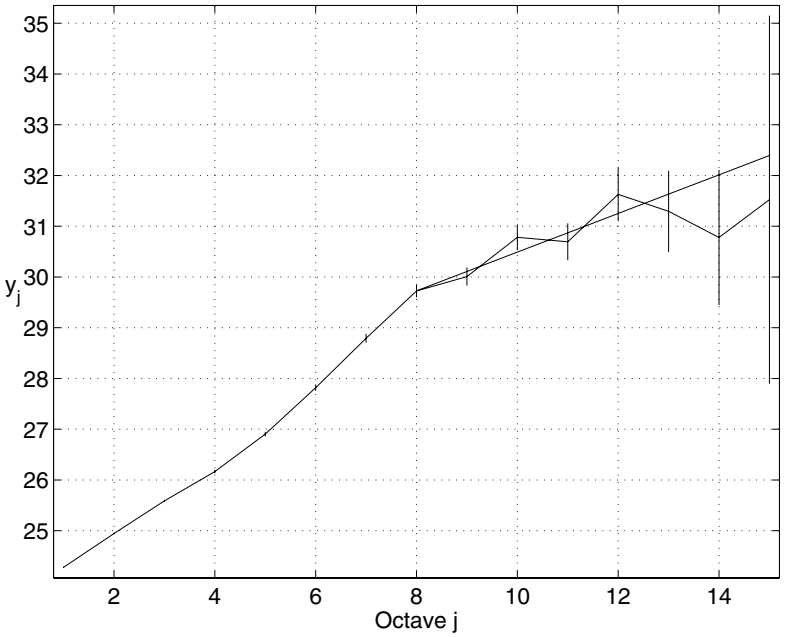


Figure 8-8: Typical plot for processes showing bi-scaling.
The second scaling region starts at j=8. It also implies long-range dependence.

An interesting question arises whether it is actually possible to truly identify the Hurst value for each type of traffic separately. In [PKC96] the authors have shown that non-self-similar UDP traffic is affected by self-similar TCP traffic. But this effect is only strong if the self-similar traffic is the major traffic. In our case we observe high Hurst values even when WAP (that is UDP) traffic is dominant. This suggests that WAP traffic itself is strongly long-range dependent.

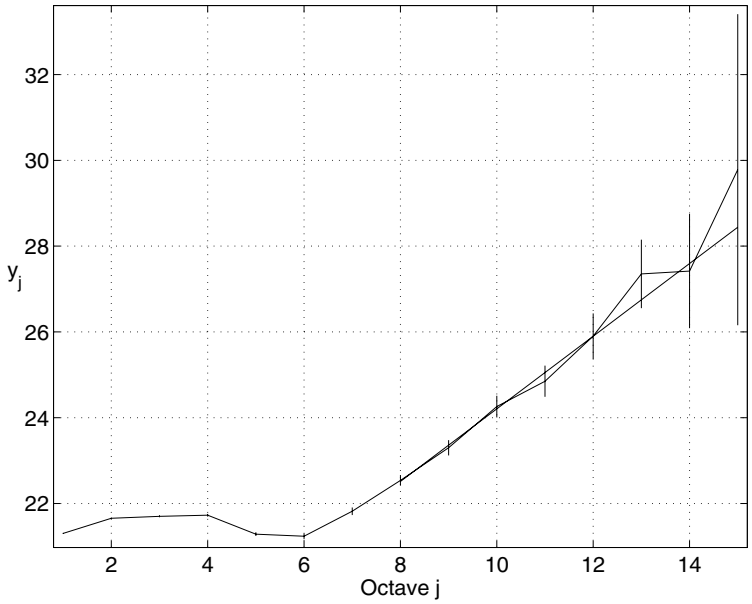


Figure 8-9: Typical plot for processes showing long-range dependency.
Below a certain scale no regular linear scaling exists.

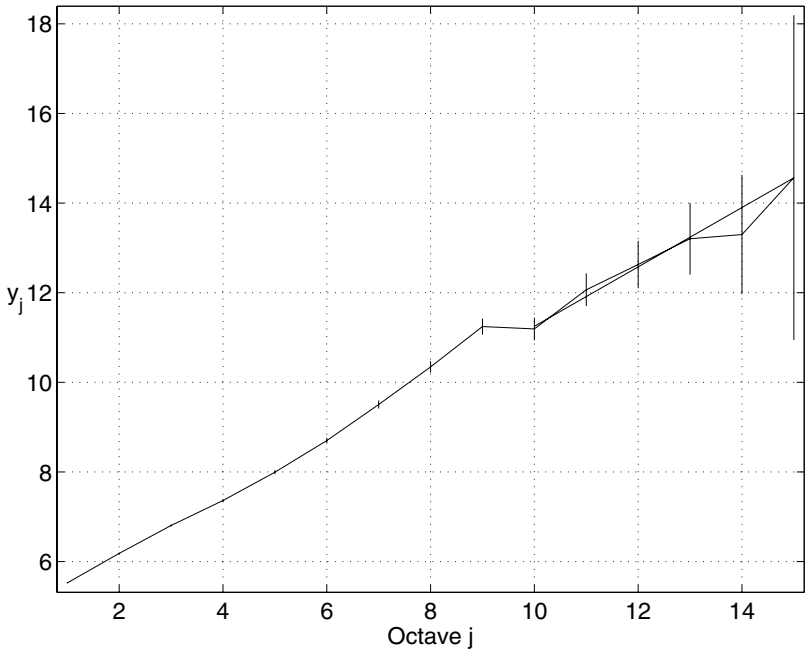


Figure 8-10: Linear scaling over the whole range.
Although there is a step at j=9, the slope on both sides is almost the same. This indicates second-order self-similarity.

We provided references in section 2.3.3.2 that the reason for LRD can be found in the heavy-tailed statistics of files or sessions, and the reason for small scaling behavior can be found in the TCP interaction. Taking this reasoning for GPRS traffic, we see that WAP traffic has a very different scaling behavior for small scales compared to Web traffic. The reason for the different small scaling behavior can be assumed to be the very different transport mechanism of TCP versus UDP, as used for WAP. As we explained in section 8.1.1, the octave j in the log scale diagram also expresses the timescale of the aggregated process. Based on this insight it is possible to determine over which timescales scaling occurs. Our stochastic processes are realized with an initial bin size of 100 ms, which leads to corresponding time values of t_j=0.1, 0.2, 0.4, 0.8, 1.6, 3.2, … seconds for scaling j=1, 2, 3, 4, 5, … , respectively. For all processes in which we observed a scaling like the one depicted in Figure 8-7 and Figure 8-9, the knee-point at which the linear scaling starts is at about j=5 or j=6, i.e., respectively 1.6 seconds to 3.2 seconds. In [VHS04] the authors show that the average download time per WAP page, including embedded objects, is in the order of 1.5-3 seconds. Hence, this timescale marks the demarcation line between the scaling due to the WAP transport layer protocol and due to the user behavior. Therefore the different small scale scaling between Web and WAP might indeed be due to the different transport layer protocols. However this requires further research.

On the other hand regarding the large time scales, we have shown in section 6.6.4 that WAP 1.2 flows, as such, are not heavy-tailed. This in itself would require a different explanation for the LRD we see for WAP traffic. On the other hand, our extrapolation study on WAP 2.0 showed that taking several WAP 1.2 flows together within one user session is weak heavy-tailed (lognormal). Therefore, we may assume that the LRD in WAP can be accounted to the

heavy-tailed flow length distribution of WAP flows per aggregated user session. Further investigation of the reason for self-similarity is open research.

8.3 Conclusion

Based on the robust Hurst estimation method by Abry-Veitch [AV98], we have shown that GPRS traffic in general as well as individual WAP traffic and Web traffic have strong signs of self-similarity. We showed that this is rigorously confirmed by many established Hurst parameter estimation methods. Though this might come as no surprise, we have proven that one has to deal with the same type of self-similar traffic when planning and dimensioning wireless networks as for wireline networks. We had stated in our introduction on self-similarity that this property requires great care when dimensioning buffer and link sizes. The heavy-tailedness of buffer length might otherwise lead to high packet delay and/or packet loss. In wireline networks the problem of self-similarity is frequently met by a simple engineering rule: over-dimensioning of bandwidth [Fow99] [Rob01]. How this can be met in wireless networks with the generally more constrained network resources is still an unsolved problem. Though our measurements are done at network aggregation level, [LWTW94] has shown that the self-similarity property remains even when the traffic is split up. Therefore, we also have to deal with self-similar traffic at radio access level and at cell level.

Our findings on self-similarity are interesting in the light of our earlier conclusion on flow length distributions for WAP 1.2 traffic. In chapter 6 we stated that the WAP 1.2 flow length distribution is not heavy-tailed. But we could show in this chapter that WAP traffic is nevertheless self-similar. On the other hand, frequently, the heavy-tailedness of flows is brought up as reason for self-similarity. Besides being influenced by the other Web traffic we assume that this can be explained by the ‘weak’ heavy-tailedness of aggregated WAP flows within one WAP session. The WAP 2.0 flows we introduced in section 6.2 can be regarded as one WAP session. However, the reason behind the self-similarity of WAP traffic is for further study.

Furthermore, we showed that the small-time self-similar scaling is missing for WAP. We indicated that this might be due to the fact that WAP is using UDP and not TCP. This is in line with the generally assumed reason for small time scaling: the transport protocol. However, the exact small-time scaling of GPRS is for further study.

9 Conclusion and Outlook

In this dissertation, the measurement and modeling of GPRS user traffic has been addressed. Accurate and up-to-date traffic models are the basis for solid network design, planning and dimensioning. However, almost all of the currently proposed models for traffic in wireless networks are based on extrapolation from wireline network measurements. Therefore the goal of this dissertation was to enhance the knowledge about mobile network usage and to model mobile user traffic and mobility in commercial mobile networks. This chapter gives an overview of the major achievements of this dissertation and points out directions for further work.

9.1 Conclusion and results

Based on measurements from three commercial networks, over a time span of one year, we focused on four traffic aspects: (1) we studied the general application and session usage of GPRS; (2) we modeled WAP and MMS application flows; (3) we modeled the mobility of GPRS users; and, (4) we assessed the self-similarity property of GPRS traffic. We summarize each of these aspects in the remainder of this section.

We designed a measurement environment, which allowed us to capture application usage data, based on IP traces, and at the same time to log GPRS network internal events (cf. chapter 4). A unique feature of this setup is the possibility to correlate the two trace types to achieve a deeper understanding of the mobile usage of GPRS.

In chapter 5, we derived the application and protocol traffic mixture for the investigated GPRS networks. We showed that WAP is the dominant application. It is not only dominant in the number of users using it, but also in the number of application flows, and it contributes a high fraction (30%-60%) of the total byte and packet volume. Networks with a high number of business users carry significant amounts of Web and Email traffic as well. This specific traffic mixture needs to be considered when modeling GPRS, because, especially the dominance of WAP leads to particular traffic characteristics, as we laid out in chapter 5 and subsequent chapters.

In chapter 6 we investigated the flow length of several applications. We found that in particular WAP flows are extremely short in terms of bytes and packets (WAP 2.0: 90% of all flows have less than 7 packets). This is, for instance, critical for TCP, especially in the case of packet loss.[58] TCP is commonly forced into a time-out in such a situation, which leads to performance degradations for the flows in terms of higher delay and lower throughput. The list of wireless TCP options which are currently promoted in the WAP 2.0 standard appear

[58] In the future, WAP 2.0 will deploy TCP. We extrapolated from current WAP 1.2 measurements to the WAP 2.0 usage case.

insufficient to handle this; therefore we recommend improving current TCP for such short-lived flows. Furthermore, the short flows might lead to an increased overhead in handling resources in GPRS. This should be considered in dimensioning.

Also in chapter 6 we modeled the flow lengths with various closed form analytical functions. We showed that in particular the length of the WAP flows is not heavy-tailed, and can well be modeled by lognormal[59] and parsimonious hyper-exponential distributions.

As a consequence of the non-heavy-tailedness, we also showed that the 'mice and elephant' phenomenon does not hold for GPRS; that is, the majority of traffic in GPRS is (especially for WAP) carried in short flows (mice) and not long flows (elephants). This should be considered when modeling TCP and optimizing resource handling in GPRS, because, in TCP models TCP is often assumed to be in an equilibrium state, which corresponds to the TCP behavior for very long flows. This approach appears adequate, as long as these very long flows constitute the majority of the network traffic volume, which, however, is not the case for GPRS. The same concern holds for common resource handling approaches, which are based on the assumption that the majority of traffic is carried in few long flows. These resource-handling methods need also be adopted for GPRS. For such investigations, WAP traffic, which constitutes the majority of GPRS traffic, should be modeled by extremely short, light-tailed flows.

We concluded in chapter 6 that the performance of short flows is particularly sensitive to packet losses. Therefore, we subsequently focused on two more aspects, which potentially lead to packet loss: mobility and self-similar traffic. Understanding thereof can be useful in properly dimensioning the network such that undesired packet loss is avoided.

In chapter 7, we focused on the network perceived mobility, based on cell reselections, in GPRS, which is a unique approach to model user mobility. We exploited the same information the network uses to track user mobility. We found in general very little indications of mobility. Especially the routing area updates happen only in 2% of all data transmissions. On the other hand, cell reselections happen often. We measured cell reselections in 20% of data transmission periods.

We also showed a weak correlation between the type of application used and the level of mobility.[60] Trends are visible in which FTP is used in a stationary fashion. Web browsing experiences many cell reselections. WAP is used in a mobile environment, and MMS shows little correlation with cell reselections.

Furthermore, we showed that the cell inter-arrival times should be modeled with heavy-tailed distributions such as Pareto or lognormal. Additionally, we

[59] A lognormal distribution "lies in-between" short-tailed distributions such as exponential and heavy-tailed distributions like Pareto. It depends on the shape parameter to which it should be considered.

[60] However, we must note that this result is heavily biased by the correlation between the PDP context length and the chosen application, which we also showed in our study.

presented results for hyper-exponential distributions, which are in many cases a good approximation.

In chapter 8 we showed, as a first study of its kind, that GPRS traffic is long-range dependent (self-similar). We observed this property for different traffic mixtures in GPRS as well as for separated Web dominant traffic and WAP dominant traffic. A consequence of self-similar traffic is that the resulting queue lengths are heavy-tailed. This requires careful dimensioning to achieve the right balance between packet loss due to buffer overflow and high delay due to long queues. Especially considering the problems of short flows with packet loss, this might be a serious issue.

After highlighting the main results of this thesis, we would like to note that the measurements partly revealed diverse results, depending on the network, the country, the time of measurements, the marketing, and pricing, etc. We did not have the capabilities to follow-up all these influencing factors. Certainly, as also previous studies have shown, these factors must be considered when designing the traffic mixture for a particular scenario. We saw, for example, that the subscriber categorization changed considerably, when the operator launched a new marketing campaign or changed pricing schemes. The main consequence of this is that one has to look for invariants in the vast amount of statistics. The second consequence is that constant measuring is required to update the knowledge and to monitor such developments.

However, comparing several networks in different countries over a longer measurement period, we have also found some trends in the results, which indicate that our results have a wider applicability. In particular, we assume that the trend towards WAP with short flows will continue for a longer time. As long as the terminals have constrained processing capacities, there will be several orders of magnitude between the flow length for laptops and those of mobile terminals. Eventually, this might fade out, and the flow length distributions might be very similar, but until then we consider this as being an invariant for WAP. On the other hand, low mobility could only be shown based on measurements from one network. In view of market studies from Asia, which state that ‘I-mode’ is used heavily while commuting to and from work, we can envision a higher degree of mobility in the future in Europe as well. Finally, we have encountered self-similarity in different networks and traffic mixtures. We can expect that this result will hold for future wireless networks as well, in particular if the fraction of heavy-tailed TCP flows increases. Therefore, we also consider the self-similarity property for GPRS traffic as an invariant.

9.2 Directions for future work

In the previous section, we discussed the main findings of this dissertation. They lead to a number of open research issues, which we recommend for further research.

We showed the dominance of short (WAP) flows in GPRS. In particular, we mentioned that the performance of short TCP-based flows is negatively influenced in case of packet losses. This has also been discussed in [AA02]. Furthermore, with the mobility and self-similarity investigation we showed two possible sources for packet loss. We recommend to develop corresponding

packet loss models for GPRS, and to evaluate their impact on the different flow types encountered in GPRS.

In chapter 7, on user mobility, we pointed out a number of limitations of our measurement approach. In particular the shortcomings of the approach, due to the current mobile-state-constraint mobility information, are visible when correlating user traffic information and mobility information. One possible remedy to this are more fine grained measurements. Combining our specific perceived mobility measurements with detailed positioning information would lead to more complete, less limited, results. One possibility to obtain detailed positioning information is to use terminals equipped with a global positioning system (GPS) receiver ([HSSK01]). We recommend such a follow-up study on mobility in GPRS.

Furthermore, it might be interesting to investigate the reasons behind self-similarity (scaling) in GPRS. In the literature (e.g., [PKC96]), small time scaling is attributed to protocol behavior, while large time scaling is attributed to user behavior. However, we showed that the small time scaling for Web and WAP is different in GPRS, and attributed this to the different transport protocols. In the same literature (e.g., [PKC96]) the reason for large time scaling is attributed to the heavy-tailedness of the aggregated flow lengths. However, we found that WAP traffic, the majority of GPRS traffic, does not consist of heavy-tailed flows. In both cases the reasons behind the scaling are interesting study subjects.

Many more aspects of mobile network traffic have not been investigated or modeled, yet. As we lined out in the motivation section, sound traffic models are a basis for sound traffic engineering. We still see the need to derive detailed (source) traffic models for wireless networks for current applications like WAP, MMS, Email, FTP and Web as well as for upcoming applications, like gaming, peer-to-peer file sharing, and machine-to-machine communication. We have to leave this for future research as well.

Appendix A: Wireless Data Networks

Name	Basic Technology	Services	Service features	Information
D-AMPS (IS-136)	FDMA/ TDMA	Speech		
GSM	FDMA/ TDMA	Speech, SMS, fax CSD	Speech: 13.8 Kbit/s Data: 9.6 Kbit/s	Global roaming Text messaging (SMS)
PDC	FDMA/ TDMA	Speech CSD	Data: 9.6 Kbit/s	
CdmaONE (IS95a)	CDMA	Speech CSD	Speech: 13 Kbit/s Data: up to 14.4 Kbit/s	
IS95b	CDMA	Speech CSD	Data: up to 64 Kbit/s	
GPRS	Integrated in GSM	PSD, MMS	Data: up to 160 Kbit/s QoS provisioning	Always online IP-based Internet access
HSCSD	Integrated in GSM	CSD	Data: up to 57.6 Kbit/s	
EGPRS /EDGE	FDMA/ TDMA Evolution to GSM and GPRS	enhanced radio link rates Speech uses AMR	Speech: 13.8 Kbit/s Data: up to 437 Kbit/s	Always online IP-based Internet access
UMTS	WCDMA	Speech uses AMR SMS, MMS, fax, etc CSD, PSD	Speech: 13.8 Kbit/s Data: Up to 2 Mbit/s (384 Kbit/s mobile)	Always online IP-based Internet access
HSDPA	Integrated in UMTS	Enhanced radio link rates CSD, PSD	Data: more than 2 Mbit/s in the order of 10 Mbit/s–20 Mbit/s	
CDMA2000	WCDMA	Speech CSD PSD	Data: 1x: 144 Kbit/s 1xRTT: 307 Kbit/s 1xEVDO: 2.4 Mbit/s	Always online Internet access
P-PDC	Integrated in PDC	PSD	Data: up to 28.8 Kbit/s	Always online IP-based Internet access
Bluetooth	TDMA	CSD, RS-232 PSD, TCP/IP Service profiles for many services (incl. speech)	Data: up to 1 Mbit/s	WLAN type No handover support
IEEE 802.11b	CSMA/CA	PSD	IP over Ethernet Data: up to 11 Mbit/s	WLAN Layer 2 handover support Always online Internet access
IEEE 802.11a	CSMA/CA	PSD	IP over Ethernet Data: up to 54 Mbit/s	WLAN Layer 2 handover support Always online Internet access

Table A-1: Characteristics of wireless data networks[61]

[61] Abbreviations are listed in the appendix.

Appendix B: Analytical Distributions

In this appendix we list some key formulas for some distribution functions. We use the notation as used by **DATAPLOT** [DATAPLOT] and **REGRESS+** [McL99].

I. Normal distribution

The normal distribution is also called the Gaussian distribution.

Parameters for $\mathrm{nor}(x;\mu,\sigma)$:

Scale: σ

Location: μ

The general form of the normal probability density function (PDF) is:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)} \tag{B.1}$$

and the cumulative distribution function (CDF) has the form:

$$F(x) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{1}{2}\left(\frac{t-\mu}{\sigma}\right)^2} dt \tag{B.2}$$

The characteristics are:

Mean = μ (B.3)

Variance = σ^2 (B.4)

Median = μ (B.5)

II. Exponential distribution

The exponential distribution is sometimes also called negative exponential distribution.

Parameters for $\exp(x;\mu,\beta)$:

Scale: β

Location: μ

The general form of the exponential probability density function is:

$$f(x) = \frac{1}{\beta} e^{-(\frac{x-\mu}{\beta})}, \text{ for } x \geq \mu \quad \text{(B.6)}$$

and the cumulative distribution function has the form:

$$F(x) = 1 - e^{-(\frac{x-\mu}{\beta})}, \text{ for } x \geq \mu \quad \text{(B.7)}$$

The characteristics are:

Mean = $\mu+\beta$ (B.8)

Variance = β^2 (B.9)

Median = $\mu+\beta \ln(2)$ (B.10)

III. Weibull distribution

Parameters for $\text{wei}(x;\gamma,\mu,\alpha)$:

Shape: γ

Scale: α

Location: μ

For the minimum order statistic, the general form of the Weibull probability density function is:

$$f(x) = \frac{\gamma}{\alpha}\left(\frac{x-\mu}{\alpha}\right)^{\gamma-1} e^{-\left(\frac{x-\mu}{\alpha}\right)^{\lambda}}, \text{ for } x \geq \mu \quad \text{(B.11)}$$

and the cumulative distribution function has the form:

$$F(x) = 1 - e^{-\left(\frac{x-\alpha}{\beta}\right)^{\lambda}}, \text{ for } x \geq \mu \quad \text{(B.12)}$$

The characteristics are:

Mean = $\mu + \alpha\Gamma\left(\frac{\gamma+1}{\gamma}\right)$ (B.13)

Variance = $\alpha^2\left[\Gamma\left(\frac{\gamma+2}{\gamma}\right) - \Gamma^2\left(\frac{\gamma+1}{\gamma}\right)\right]$ (B.14)

Median = $\mu + \alpha \sqrt[\lambda]{\log(2)}$ (B.15)

IV. Gamma distribution

Parameters for $\mathrm{gam}(x;\gamma,\mu,\beta)$:

Shape: γ

Scale: β

Location: μ

The general form of the gamma probability density function is:

$$f(x) = \frac{\left(\frac{x-\mu}{\beta}\right)^{\gamma-1} e^{-\left(\frac{x-\mu}{\beta}\right)}}{\beta\,\Gamma(\gamma)} \text{ for } x \geq \mu \tag{B.16}$$

and the cumulative distribution function has the form:

$$F(x) = \frac{\Gamma\left(\gamma, \frac{x-\mu}{\beta}\right)}{\Gamma(\gamma)}, \text{ for } x \geq 0 \tag{B.17}$$

and $\Gamma(a,b)$ is the incomplete gamma function.

The characteristics are:

Mean = $\mu + \beta\gamma$ (B.18)

Variance = $\beta^2\gamma$ (B.19)

Median = no simple closed form

Note:

If λ is an integer the gamma distribution is the Erlang distribution

If $\lambda=1$ the gamma distribution is the exponential distribution

V. Pareto distribution

Parameters for $\mathrm{par}(x;\gamma,k)$:

Shape: γ

Location: k

The general form of the Pareto probability density function is:

$$f(x) = \frac{\gamma k^{\lambda}}{x^{\gamma+1}} \text{ for } x \geq k \tag{B.20}$$

and the cumulative distribution function has the form:

$$F(x) = 1 - \left(\frac{k}{x}\right)^{\gamma}, \text{ for } x \geq k \tag{B.21}$$

The characteristics are:

$$\text{Mean} = \frac{k\gamma}{(\gamma - 1)} \tag{B.22}$$

$$\text{Variance} = \frac{k^2\gamma}{(\gamma - 2)(\gamma - 1)^2} \tag{B.23}$$

$$\text{Median} = k\sqrt[\gamma]{2} \tag{B.24}$$

VI. Extreme-value distribution

The extreme value distribution is also called Gumple.

Parameters for $\mathrm{ev1}(x;\mu,\beta)$*:*

Scale: β

Location: μ

For the minimum order statistic, the general form of the extreme value probability density function is:

$$F(x) = 1 - e^{-e^{\left(\frac{x-\mu}{\beta}\right)}} \tag{B.25}$$

and the cumulative distribution function has the form:

$$f(x) = \frac{1}{\beta} e^{\frac{x-\mu}{\beta}} e^{-e^{\frac{x-\mu}{\beta}}} \tag{B.26}$$

The characteristics are:

$$\text{Mean} = \mu\text{-}0.5722\beta \tag{B.27}$$

$$\text{Variance} = \left(\frac{\pi^2}{6}\right)\beta^2 \tag{B.28}$$

VII. Logistic distribution

Parameters for $\log(x;\mu,\sigma)$*:*

Scale: σ

Location: μ

The logistic probability density function is:

$$F(x) = \frac{1}{1 + e^{-\frac{x-\mu}{\sigma}}} \tag{B.29}$$

and the cumulative distribution function has the form:

$$f(x) = \frac{\frac{e^{-(x-\mu)}}{\sigma}}{\sigma\left[1 + e^{\frac{-(x-\mu)}{\sigma}}\right]^2} \tag{B.30}$$

The characteristics are:

Mean = μ (B.31)

Variance = $\frac{1}{3}(\pi\sigma)^2$ (B.32)

Median = μ (B.33)

VIII. Lognormal distribution

A variable X is lognormal distributed if $\log(X)$ is normally distributed.

Parameters for $\lg n(x;\sigma,m,\theta)$:

Shape: σ

Scale: m

Location: θ

The general form of the lognormal probability density function is:

$$f(x) = \frac{1}{(x-\theta)\sigma\sqrt{2\pi}} e^{\frac{-\left(\ln(\frac{x-\theta}{m})\right)^2}{2\sigma^2}}, \text{ for } x \geq \theta \tag{B.34}$$

and the cumulative distribution function has the form:

$$F(x) = \Phi\left(\frac{\ln(x) - \ln(m)}{\sigma}\right) (\text{for } \theta = 0) \tag{B.35}$$

The characteristics are:

Mean = $e^{\ln(m)+\frac{\sigma^2}{2}}$ (B.36)

Variance = $\left(e^{2\ln(m)+\sigma^2}\right)\left(e^{\sigma^2} - 1\right)$ (B.37)

Median = m (B.38)

Appendix C: Results of Distribution Parameters

Flow length		Distribution	Parameter			
HTTP body						
MLE	**1st**	Lognormal	Shape	0.932979	Scale	748.5217
	2nd	Gamma	Shape	1.270849	Scale	1219.585
	3rd	Weibull	Shape	1.123434	Scale	2168.302
HTTP tail						
MLE	**1st**	Lognormal	Shape	1.504118	Scale	11881.79
	2nd	Weibull	Shape	0.655713	Scale	24436.78
	3rd	Gamma	Shape	0.509211	Scale	79515.02
	4th	Pareto	Shape	1.1586	Location	10000
(tail starts at 10^6 byte)		Pareto	Shape	1.34466	Location	100000
HTTP Total						
EM	**Total**	PH 2-phase	λ_0	5.42E-04	c_0	7.53E-01
			λ_1	3.06E-05	c_1	2.47E-01
	Total	PH 4-phase	λ_0	0.000604	c_0	0.696531
			λ_1	6.14E-05	c_1	0.260152
			λ_2	1.51E-05	c_2	0.041389
			λ_3	7.18E-07	c_3	0.001929
WAP 1.2						
MLE	**1st**	Lognormal	Shape	0.971754	Scale	748.5217
	2nd	Gamma	Shape	1.036182	Scale	1219.585
	3rd	Exponential	Scale	1263.716	Location	64
EM	**Total**	PH 2-phase	λ_0	0.000913	c_0	0.989655
			λ_1	5.77E-05	c_1	0.010345
	Total	PH 4-phase	λ_0	0.000937	c_0	0.964116
			λ_1	0.000307	c_1	0.032526
			λ_2	2.6E-05	c_2	0.003305
			λ_3	2.36E-05	c_3	5.28E-05
WAP 2.0						
MLE	**1st**	Lognormal	Shape	1.217624	Scale	1739.848
	2nd	Weibull	Shape	0.810749	Scale	3185.072
	3rd	Gamma	Shape	0.978627	Scale	3780.69
EM	**Total**	PH 2-phase	λ_0	0.000456	c_0	0.829553
			λ_1	9.07E-05	c_1	0.170447
	Total	PH 4-phase	λ_0	0.000468	c_0	0.779992
			λ_1	0.000144	c_1	0.20788
			λ_2	2.1E-05	c_2	0.01196
			λ_3	8.06E-06	c_3	0.000168

Table C-1: GPRS flow length – distribution parameters

IAT		**Distribution**	**Parameter**			
GPRS attach period						
MLE	**1st**	lognormal	Shape	2.174719	Scale	263.6612
	2nd	Weibull	Shape	0.495729	Scale	762.1619
	3rd	Pareto	Shape	0.44818	location	45
EM	**>45 sec**	PH 2-phase	λ_0	0.005642	c_0	0.615033
			λ_1	0.000226	c_1	0.384967
	>45 sec	PH 4-phase	λ_0	0.022639	c_0	0.231285
			λ_1	0.00389	c_1	0.316117
			λ_2	0.000568	c_2	0.375735
			λ_3	7.24E-05	c_3	0.076862
EM	**Total**	PH 2-phase	λ_0	0.015749	c_0	0.704083
			λ_1	0.000299	c_1	0.295917
	Total	PH 4-phase	λ_0	0.023128	c_0	0.57841
			λ_1	0.002747	c_1	0.196445
			λ_2	0.000492	c_2	0.187774
			λ_3	6.73E-05	c_3	0.037371
PDP context period						
MLE	**1st**	lognormal	Shape	1.69079	Scale	81.70586
	2nd	Weibull	Shape	0.591875	Scale	187.039
	3rd	Pareto	Shape	0.770163	location	45
EM	**>45 sec**	PH 2-phase	λ_0	0.009043	c_0	0.84367
			λ_1	0.000603	c_1	0.15633
	>45 sec	PH 4-phase	λ_0	0.015333	c_0	0.569205
			λ_1	0.00314	c_1	0.353445
			λ_2	0.000658	c_2	0.069955
			λ_3	7.66E-05	c_3	0.007396
EM	**Total**	PH 2-phase	λ_0	0.011434	c_0	0.878043
			λ_1	0.000718	c_1	0.121957
	Total	PH 4-phase	λ_0	0.014917	c_0	0.731964
			λ_1	0.002835	c_1	0.228215
			λ_2	0.000574	c_2	0.035994
			λ_3	7.03E-05	c_3	0.003827
Ready state period inside PDP context						
MLE	**1st**	Weibull	Shape	0.734035	Scale	81.01982
	2nd	lognormal	Shape	1.534626	Scale	39.62178
	3rd	Pareto	Shape	1.164591	Location	45
EM	**>45 sec**	PH 2-phase	λ_0	0.01736	c_0	0.841354
			λ_1	0.002927	c_1	0.158646
	>45 sec	PH 4-phase	λ_0	0.02647	c_0	0.519928
			λ_1	0.008007	c_1	0.424792
			λ_2	0.002258	c_2	0.053935
			λ_3	0.000221	c_3	0.001345
EM	**Total**	PH 2-phase	λ_0	0.017962	c_0	0.92254

			λ_1	0.002862	c_1	0.077461
	Total	PH 4-phase	λ_0	0.018975	c_0	0.879435
			λ_1	0.004405	c_1	0.117055
			λ_2	0.000815	c_2	0.003425
			λ_3	6.54E-05	c_3	8.51E-05
Ready state period outside PDP context						
MLE	**1st**	lognormal	Shape	1.931667	Scale	26.30533
	2nd	Pareto	Shape	1.281236	Location	45
	3rd	Weibull	Shape	0.548425	Scale	66.31413
EM	**>45 sec**	PH 2-phase	λ_0	0.024892	c_0	0.845019
			λ_1	0.001366	c_1	0.154981
	>45 sec	PH 4-phase	λ_0	0.127042	c_0	0.252352
			λ_1	0.015963	c_1	0.619681
			λ_2	0.001696	c_2	0.125726
			λ_3	6.91E-05	c_3	0.002242
EM	**Total**	PH 2-phase	λ_0	0.031314	c_0	0.972904
			λ_1	0.001405	c_1	0.027096
	Total	PH 4-phase	λ_0	0.032128	c_0	0.961354
			λ_1	0.003779	c_1	0.032172
			λ_2	0.000843	c_2	0.006287
			λ_3	4.18E-05	c_3	0.000186

Table C-2: Fitting results for cell reselection IAT

List of Figures

List of Tables

Abbreviations

3GPP	Third Generation Partnership Project
ACF	Auto Correlation Function
ADSL	Asymmetric Digital Subscriber Line
AH	Authentication Header
AMPS	Advanced Mobile Phone Service
AMR	Adaptive MultiRate (codec)
API	Access Point Interface
APN	Access Point Name
AR	AutoRegressive process
ARIMA	AutoRegressive Integrated Moving Average process
ARMA	AutoRegressive Moving Average process
ARQ	Automatic Repeat and reQuest
ATM	Automatic Teller Machine (only on page 15)
ATM	Asynchronous Transfer Mode
AuC	Authentication Center
AV-Method	Abry-Veitch Method
BDP	Bandwidth Delay Product
BSC	Base Station Controller
BSS	Base Station Subsystem
BSSGP	Base Station Subsystem GPRS Protocol
BTS	Base Transceiver Station
CCDF	Complementary Cumulative Density Function
CDF	Cumulative Density Function
CDMA	Code Division Multiple Access
CN	Core Network
CDPD	Cellular Digital Packet Data
CR	Cell Reselection
CS	Circuit Switched
CSD	Circuit Switched Data
CSMA/CA	Carrier Sense Multiple Access / Collision Avoidance

CV	Coefficient of Variation
D-AMPS	Digital AMPS
DFT	Discrete Fourier Transformation
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name Service
DVP	Data Volume Process
ECN	Explicit Congestion Notification
EDGE	Enhanced Data GSM Environment
EGPRS	Enhanced GPRS
EM	Expectation Maximization
FDD	Frequency Division Duplex
FDDI	Fiber Distributed Data Interface
FDMA	Frequency Division Multiple Access
FEC	Forward Error Correction
FHDMA	Frequency Hopping Division Multiple Access
FM	Frequency Modulation
FTP	File Transfer Protocol
GGSN	Gateway GPRS Service Node
GIF	Graphics Interchange Format
GMM	GPRS Mobility Management
GPRS	General Packet Radio Service
GPS	Global Positioning System
GSM	General System for Mobile communication
GSN	GPRS Support Node
GTP	GPRS Tunnel Protocol
HLR	Home Location Register
HPC	High Performance Connection
HSCSD	High Speed Circuit Switched Data
HSDPA	High Speed Downlink Packet Access
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
IAT	Inter-Arrival Time
ICMP	Internet Control Message Protocol
ID	IDentification
IDC	Index of Dispersion for Counts

IETF	International Engineering Task Force
IMAP	Internet Message Access Protocol
IP	Internet Protocol
IPIP	IP over IP
IPsec	IP security
ISDN	Integrated Services Digital Network
ISP	Internet Service Provider
ITU	International Telecommunication Union
JPEG	Joint Photographic Experts Group
KS	Kolmogorov-Smirnov
LAN	Local Area Network
LD	Log scale Diagram
LLC	Logical Link Control
LLH	Link Layer Header
LRD	Long-Range Dependent
MA	Moving Average
MAC	Medium Access Control
MLE	Maximum Likelihood Estimation
MMS	Multimedia Messaging Service
MP3	MPeg Layer 3 (codec)
MPEG	Moving Pictures Experts Group
MRA	Multi Resolution Analysis
MS	Mobile Station
MSC	Mobile Switching Center
MSS	Maximum Segment Size
MT	Mobile Terminal
MTU	Maximum Transfer Unit
NA	Not Available
NAT	Network Address Translation
PAP	Packet Arrival Process
PDA	Personal Digital Assistant
PDC	Personal Digital Cellular
PDCH	Packet Data CHannel
PDF	Probability Density Function
PDN	Packet Data Network

PDP	Pack Data Protocol
PDTCH	Packet Data Traffic CHannel
PDU	Packet Data Unit
PH	PHase (type)
PLMN	Public Land Mobile Network
POP3	Post Office Protocol (version 3)
P-PDC	Packet Switched PDC
PPP	Point to Point Protocol
PR	Packet Radio
PRACH	Packet Random Access CHannel
PRAU	Periodic Routing Area Update
PSD	Packet Switched Data
PSN	Packet Support Node
PSTN	Public Switched Telephone Network
QoS	Quality of Service
RA	Routing Area
RADIUS	Remote Authentication Dial-In User Service
RAN	Radio Access Network
RAU	RA Update
RF	Radio Frequency
RFC	Request For Comments
RLC	Radio Link Control
RPE_LTP	Regular Pulse Excited codec with Long Term Prediction
RTO	Retransmit TimeOut (timer)
RTTM	Round-Trip Time Measurements
RTT	Round-Trip Timer
SACK	Selective ACKnowledgement
SDMA	Spatial Division Multiple Access
SDU	Service Data Unit
sec	seconds
SGSN	Serving GPRS Support Node
SM	Session Management
SMIL	Synchronized Multimedia Integration Language
SMS	Short Message Service

SMSC Protocol	Short Message Service CentreSMTP Simple Mail Transfer
SNDCP	Sub Network Dependency Control Protocol
SR-ARQ	Selective Repeat ARQ
SRD	Short Range Dependent
SS	Self-Similar
SYN	SYNchronize
TE	Terminal Equipment
TBF	Temporary Block Flow
TCP	Transport Control Protocol
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
TES	Transform Expand Sample
TLLI	Temporary Logical Link Identifier
TLS	Transport Layer Security
TS	Time Slot
UCR	Unique Cell Reselection
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunication Systems
URA	Unique RA
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
VBR	Variable Bit Rate
VLR	Visitor Location Register
VPN	Virtual Private Network
WAE	Wireless Application Environment
WAN	Wide Area Network
WAP	Wireless Application Protocol
WAP_CL	WAP Connection Less
WAP_CO	WAP Connection Oriented
WCDMA	Wideband CDMA
WD	WeekDays
WDP	Wireless Datagram Protocol
WE	WeekEnd (days)
WEB	The HTTP accessible Internet

WLAN	Wireless LAN
WML	Wireless Markup-Language
WP-HTTP	Wireless Profiled HTTP
WP-TCP	Wireless Profiled TCP
WSP	Wireless Session Protocol
WTLS	Wireless TLS
WTP	Wireless Transaction Protocol
WWW	World Wide Web
x-DSL	variant x of DSL (Digital Subscriber Line)
xHTML	eXtensible HTML
XHTMLMP	xHTML Mobile Platform
XML	Extensible Markup Language

References

Bibliography

[AA02] U. Ayesta and K. Avrachenkov. IETF internet draft: On reducing the number of TimeOuts for short-lived TCP connections. IETF, 2002.

[AFTV00] P. Abry, P. Flandrin, M. S. Taqqu, and D. Veitch. Wavelets for the analysis, estimation, and synthesis of scaling data. In Self-Similar Network Traffic and Performance Evaluation, pp 39–84, Wiley, 2000.

[AG03] N. Azzouna and F. Guillemin. Analysis of ADSL traffic on an IP backbone link. In Proceedings of IEEE GLOBECOM, 22(1):3742–3746, 2003.

[AK99] A. Arvidsson and P. Karlsson. On traffic models for TCP/IP. In Proceedings of the 16th international teletraffic congress (ITC16), pp. 455–466, 1999.

[ANO96] S. Asmussen, O. Nerman, and M. Olsson. Fitting phase-type distribution via the EM algorithm. Scandinavian Journal of Statistics, 23:419–441, 1996.

[AV98] P. Abry and D. Veitch. Wavelet analysis of long-range-dependent traffic. IEEE Transactions on Information Theory, 44(1):2–15, 1998.

[AW96] M. F. Arlitt and C. L. Williamson. Web server workload characterization: The search for invariants (extended version). In Proceedings of the ACM SIGMETRICS, 24(1):126–137, 1996.

[BC02] N. Brownlee and K. Claffy. Understanding Internet traffic streams: Dragonflies and tortoises. IEEE Communications Magazine, 40(10):110–117, 2002.

[BCT98] M. E. Crovella, M. S. Taqqu and A. Bestavros. Heavy-Tailed Probability Distributions in the World Wide Web. In A Practical Guide To Heavy Tails, pp. 3–26, Chapman & Hall, 1998.

[BHGC04] A. Broido, Y. Hyun, R. Gao and K. Claffy. Their share: Diversity and disparity in IP traffic. In Proceedings of the Passive and Active Network Measurement: 5th International Workshop, PAM 2004, pp. 113–125, 2004.

[CB96] M. E. Crovella and A. Bestavros. Self-similarity in world wide web traffic: evidence and possible causes. In Proceedings of the ACM SIGMETRICS, pp. 160–169, 1996.

[CBP95] K. Claffy, H.-W. Braun, and G. C. Polyzos. A parameterizable

methodology for internet traffic flow profiling. IEEE Journal of Selected Areas in Communications, 13(8):1481–1494, 1995.

[CDF+00] R. Caceres, N. G. Duffield, A. Feldmann, J. Friedmann, et al. Measurement and analysis of IP network usage and behavior. IEEE Communications Magazine, 38:144–151, 2000.

[CK74] V. Cerf and R. Kahn. A protocol for packet network interconnection. IEEE Transactions on Communications, pp. 637–648, 1974.

[Cla88] D. D. Clark. The design philosophy of the DARPA internet protocols. In Proceedings of ACM SIGCOMM, pp. 106–114, 1988.

[CLR00] M. Crovella, C. Lindemann, and M. Reiser. Internet performance modeling: the state of the art at the turn of the century. Performance Evaluation, 42(2-3):91–108, 2000.

[CT99] M. Crovella and M. Taqqu. Estimating the heavy tail index from scaling properties. Methodology and Computing in Applied Probability, 1(1):55–79, 1999.

[DdHR00] H. Drees, L. de Haan, and S. Resnick. How to make a hill plot. Technical Report No. 1215, Cornell University, The Institute of Mathematical Statistics, 2000.

[Dow01] A. B. Downey. Evidence for long-tailed distributions in the Internet. In Proceedings of the ACM SIGCOMM Internet Measurement Workshop, pp. 229–241, 2001

[Eck85] A. E. Eckberg. Approximations for bursty (and smoothed) arrival queueing delays based on generalized peakedness. In Proceedings of the 11th international teletraffic congress (ITC11), pp. 331–335, 1985.

[EL04] H. Ekstroem and R. Ludwig. The peak-hopper: A new end-2-end retransmission timer for reliable unicast transport. In Proceedings of IEEE INFOCOM, 2004.

[ENNS00] A. Erramilli, O. Narayan, A. L. Neidhardt, and I. Saniee. Performance impacts of multi-scaling in wide-area TCP/IP traffic. In Proceedings of the IEEE INFOCOM (1), pp. 352–359, 2000.

[ENW96] Erramilli, A., O. Narayan, and W. Willinger. Experimental Queueing Analysis with Long-Range Dependent Traffic. IEEE Transactions Networking 4(2), pp. 209–222, 1996.

[EW94] A. Erramilli and J. L. Wang. Monitoring packet traffic level. In Proceedings of IEEE GLOBECOMM, pp. 274–280, 1994.

[FA03] G. S. Fishman and I. Adan. How heavy-tailed distributions affect simulation-generated time averages. Technical Report UNC/OR TR03-2, University of North Carolina, Chapel Hill, 2003.

[Fae02] J. Faerber. Network game traffic modelling. In Proceedings of the 1st workshop on Network and system support for games, pp. 53–57, 2002.

[FGHW99] A. Feldmann, Anna C. Gilbert, Polly Huang, and Walter Willinger. Dynamics of IP traffic: A study of the role of variability and the impact of control. In Proceedings of the SIGCOMM, pp. 301–313, 1999.

[FGMS01] M. J. Fischer, D. Gross, D. M. B. Masi and J. F. Shortle. Analyzing the Waiting Time Process in Internet Queueing Systems with the Transform Approximation Method. The 2001 Telecommunications Review, pp. 21–32, Mitretek Systems, McLean, VA, 2001.

[FGWK98] A. Feldmann, A. C. Gilbert, W. Willinger, and T. G. Kurtz. The changing nature of network traffic: scaling phenomena. ACM SIGCOMM Computer Communication Review, 28(2):5–29, 1998.

[FL93] H. J. Fowler and W. E. Leland. Local area network traffic characteristics, with implications for broadband network congestion management. IEEE Journal of Selected Areas in Communications, 7(9):1139–1149, 1993.

[FM94] V. S. Frost and B. Melamed. Traffic modeling for telecommunications networks. IEEE Communications Magazine, 32:70–81, 1994.

[FML+03] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely, and S. C. Diot. Packet-level traffic measurements from the sprint IP backbone. IEEE Network, 17:6–16, 2003.

[FMMO99] A. Furuskaer, S. Mazur, F. Mueller, and H. Olofsson. EDGE: Enhanced data rates for GSM and TDMA/136 evolution. IEEE Personal Communications Magazine, 63:56–66, 1999.

[Fow99] T. B. Fowler. A short tutorial on fractals and internet traffic. The 1999 Telecommunications Review, 10:1–15, Mitretek Systems, McLean, VA, 1999.

[FP01] S. Floyd and V. Paxson. Difficulties in simulating the internet. IEEE/ACM-Transactions on Networking, 9(4):392–403, 2001.

[FW98] A. Feldmann and W. Whitt. Fitting mixtures of exponentials to long-tail distributions to analyze network performance models. Performance Evaluation, 31(8):963–976, 1998.

[Gra95] A. Graps. An introduction to wavelets. IEEE Computational Science and Engineering, 2(2):50–61, 1995.

[GSFM02] D. Gross, J. F. Shortle, J. Fischer and D. M. B. Masi. Difficulties in simulating queues with Pareto service. Proceedings of the 2002 Winter Simulation Conference, pp. 407– 415, 2002.

[HB04] M. Heath and A. Brydon. Vodafone live! versus I-mode - lessons and prospects for the rise of global wireless services. Consultancy report, Analysys, 2004.

[HBF00] C. M. Harris, P. H. Brill and M. J. Fischer. Internet-type queues with power- tailed inter-arrival times and computational methods

for their analysis. INFORMS Journal on Computing, 12:261–271, 2000.

[HKS99] H. Hlavacs, G. Kotsis and C. Steinkellner. Traffic source modeling. Technical Report No. TR-99101, Institute of Applied Computer Science and Information Systems, University of Vienna, 1999.

[HM98] C. M. Harris and W. G. Marchal. Distribution estimation using Laplace transforms. INFORMS Journal on Computing 10 (4) 448–458, 1998.

[HSSK01] H. Hidaka, K. Saitoh, N. Shinagawa and T. Kobayashi. Teletraffic characteristics of cellular communication for different types of vehicle motion. IEICE Transactions on Communications, E84-B(3):558–565, 2001.

[Jai91] R. Jain. The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling. John Wiley & Sons, 1991.

[JMW96] D. Jagerman, B. Melamed and W. Willinger. Stochastic modeling of traffic processes. Frontiers in Queueing: Models and Applications in Science and Engineering, CRC Press, 1996.

[Joy00] S. Joyce. Traffic on the internet - report. Technical report, The University of Waikato, WAND Network Research Group, 2000.

[KA98] P. Karlsson and A. Arvidsson. On TCP/IP traffic modeling. In Proceedings of the 14th Nordic Teletraffic Seminar, pp. 156–166, 1998.

[KBB+03] T. Karagiannis, A. Broido, N. Brownlee, K. C Claffy and M. Faloutsos. File-sharing in the internet: A characterization of P2P traffic in the backbone. Technical report, University of California, Riverside, 2003.

[KBBM00] T. Kunz, T. Barry, J. Black and H. Mahoney. WAP traffic: Description and comparison to WWW traffic. In Proceedings of the 3rd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems, pp. 11–19, 2000.

[KCF+00] U. R. Krieger, J. Charzinski, J. Faerber, K. Dolzer, S. Koehler, R. Macfadyen, N. Markovitch, K. Tutschku, N. Vicari, A. Vidacs and J. T. Virtamo. An overview of activities on wireless networks in the European COST-257 project, 2000.

[KE04] R. Kalden and H. Ekstrom. Searching for mobile mice and elephants in GPRS networks. To be published in ACM SIGMOBILE, Mobile Computing and Communications Review, 2004.

[KF02] T. Karagiannis and M. Faloutsos. SELFIS: A tool for self-similarity and long-range dependence analysis [extended abstract]. 1st Workshop on Fractals and Self-Similarity in Data Mining: Issues and Approaches (in KDD), 2002.

[KFR02] T. Karagiannis, M. Faloutsos, and R. H. Riedi. Long-range dependence: Now you see it now you don't!. In Proceedings of the IEEE GLOBECOM, 21(1):2177–2181, 2002.

[KI04] R. Kalden and S. Ibrahim. Searching for self-similarity in GPRS. In Proceedings of the Passive and Active Network Measurement: 5th International Workshop, PAM 2004, pp. 83–92, 2004.

[Kle75] L. Kleinrock. Queueing Systems, Volume I: Theory. Wiley, New York, 1975.

[Kle76] L. Kleinrock. Queueing Systems Volume II: Computer Applications. John Wiley & Sons, New York, 1976.

[KMM00] R. Kalden, I. Meirick and M. Meyer. Wireless internet access based on GPRS. IEEE Personal Communications, 7(2):8–18, 2000.

[KS04] R. Kalden and B. Sanders. Cell reselection inter-arrival time investigation for GPRS. In Proceedings of the WiOpt04 Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, 2004.

[KSH03] R. E. A. Khayari, R. Sadre, and B. Haverkort. Fitting world-wide web request traces with the EM-algorithm. Performance Evaluation, 52:175–191, 2003.

[KT04] I. Khalifa and L. Trajkovic. An overview and comparison of analytical TCP models. In Proceedings of the IEEE International Symposium Circuits and Systems, vol. V, pp. 469–472, 2004.

[KVWS03] R. Kalden, T. Varga, B. Wouters, and B. Sanders. Wireless service usage and traffic characteristics in GPRS networks. In Proceedings of the 18th International Teletraffic Congress (ITC18), vol. 2, pp. 981–990, 2003.

[LH03] K. Lan and J. Heidemann. On the correlation of internet flow characteristics. Technical Report ISI-TR-574, USC Information Sciences Institute, 2003.

[LK00] R. Ludwig and R. H. Katz. The Eifel algorithm: Making TCP robust against spurious retransmissions. ACM Computer Communication Review, 30(1):30–37, 2000.

[LKJK02] R. Ludwig, A. Konrad, A. D. Joseph, and R. H. Katz. Optimizing the end-to-end performance of reliable flows over wireless links. ACM Wireless Networks Journal, 8(2/3):289–299, 2002.

[LMW94] K. K. Leung, W. A. Massey, and W. Whitt. Traffic models for wireless communication networks. In Proceedings of the INFOCOM (3), pp. 1029–1037, 1994.

[LWTW94] W. E. Leland, W. Willinger, M. S. Taqqu, and D. V. Wilson. On the self-similar nature of ethernet traffic (extended version). IEEE/ACM Transactions on Networking, 2(1):1–15, 1994.

[MBB00] A. McGregor, H.-W. Braun, and J. Brown. The NLANR network analysis infrastructure. IEEE Communications Magazine, 38(5):122–128, 2000.

[MC00] S. McCreary and K. Claffy. Trends in wide area IP traffic patterns: A view from ames internet exchange. In Proceedings of the 13th ITC Specialist Seminar on Measurement and Modeling of IP Traffic, pp. 1–11, 2000.

[McL99] M. P. McLaughlin. Regress+: A compendium of common probability distributions, 1999. http://www.causascientia.org/math_stat/Dists/Compendium.pdf

[MJ93] S. McCanne and V. Jacobson. The BSD packet filter: a new architecture for user-level packet capture. In USENIX Winter, pp. 259–269, 1993.

[Mor95] P. Morin. The impact of self-similarity on network performance analysis. Technical Report Computer Science 95.495, Carleton University, 1995.

[MSH03] M. Meyer, J. Sachs, and M. Holzke. Performance evaluation of a TCP proxy in WCDMA networks. IEEE Wireless Communications, 10(5):70–79, 2003.

[NGBS+97] H. F. Nielsen, J. Gettys, A. Baird-Smith, E. Prud'hommeaux, H. Lie, and C. Lilley. Network performance effects of HTTP 1.1, CSS1 and PNG. ACM SIGCOMM Computer Communication Review, 27(4), 1997.

[Nor94] I. Norros. A storage model with self-similar input, Queueing Systems, 16:387–396. 1994.

[Oli99] M. W. Oliphant. The mobile phone meets the Internet. IEEE Spectrum, 36:20–28, 1999.

[Ols98] M. Olsson. The EMpht-programme, Chalmers University of Technolgy and Goeteborg University, 1998. http://www.maths.lth.se/matstat/staff/asmus/pspapers.html

[Pax94] V. Paxson. Empirically derived analytic models of wide-area TCP connections. IEEE/ACM Transaction on Networking, 2(4):316–336, 1994.

[Pax97] V. Paxson. Measurements and Analysis of End-to-End Internet Dynamics. PhD thesis, University of California, 1997.

[Pax98] V. Paxson. On calibrating measurements of packet transit times. Performance Evaluation Review, 26(1):11–21, 1998.

[PF94] V. Paxson and S. Floyd. Wide-area traffic: the failure of poisson modeling. In Proceedings of the ACM conference on communications architectures, protocols and applications, pp. 257–268, 1994.

[Pit99] J. Pitkow. Summary of WWW characterizations. World Wide Web, 2(1-2):3–13, 1999.

[PKC96] K. Park, G. Kim, and M. Crovella. On the relationship between file sizes, transport protocols, and self-similar network traffic. In Proceedings of the 1996 International Conference on Network Protocols (ICNP-96), pp. 171–180, 1996.

[PKC97] K. Park, G. Kim, and M. Crovella. On the effect of traffic self-similarity on network performance. In Proceedings of SPIE International Conference on Performance and Control of Network Systems, pp. 296–310, 1997.

[Pop01] A. Popescu. Traffic self-similarity. In Proceedings of the ICT2001, June 2001.

[PP03] R. Pang and V. Paxson. A high-level programming environment for packet trace anonymization and transformation. In Proceedings of the 2003 ACM conference on Applications, technologies, architectures, and protocols for computer communications, pp. 339–351, 2003.

[PT98] K. Park and T. Tuan. Congestion control for self-similar network traffic. Technical Report CSD-TR-98014, Department of Computer Sciences, Purdue University, 1998.

[Rap96] T. S. Rappaport. Wireless Communications: Principles and Practice. Prentice-Hall PTR, 1996.

[RLGPC+99] A. Reyes-Lecuona, E. Gonzlez-Parada, E. Casilari, J. C. Casasola and A. Diaz-Estrella. A page-oriented WWW traffic model for wireless system simulations. In Proceedings of 16th international teletraffic congress (ITC16), pp. 1271–1280, 1999.

[Rob01] J. Roberts. Traffic theory and the internet. IEEE Communications Magazine, 39(1):94–99, January 2001.

[RV00] M. Roughan and D. Veitch. A review of TCP modeling with reference to dimensioning TCP networks. Technical Report 2000-01, EMULab, University of Melbourne, Australia, 2000.

[SFB01] P. Stuckmann, H. Finck and T. Bahls. A WAP traffic model and its appliance for the performance analysis of WAP over GPRS. In proceedings of the IEEE International Conference on Third Generation Wireless and Beyond (3Gwireless 2001), 2001.

[SHSK01] K. Saitoh, H. Hidaka, N. Shinagawa and T. Kobayashi. Vehicle motion in large and small cities and teletraffic characterization in cellular communication systems. In IEICE Transactions on Communications, E84-B, pp. 805–813, 2001.

[SM00] P. Stuckmann and F. Mueller. GPRS radio network capacity considering coexisting circuit switched traffic sources. In European Conference on Wireless Technology, pp. 66–69, 2000.

[ST99] Z. Sahinoglu and S. Tekinay. On multimedia networks: self-similar traffic and network performance. IEEE Communications Magazine, 37(1):48–52, 1999.

[Ste92] W. R. Stevens. Advanced Programming in the Unix Environment. Addison-Wesley, 1992.

[TB02] D. Tang and M. Baker. Analysis of a metropolitan-area wireless network. Wireless Networking, 8(2/3):107–120, 2002.

[TGSL01] P. Tran-Gia, D. Staehle, and K. Leibnitz. Source traffic modeling

of wireless applications. AEU - International Journal of Electronics and Communications, 55(1):27–36, 2001.

[TMW97] K. Thompson, G. Miller, and R. Wilder. Wide-area internet traffic patterns and characteristics (extended version). IEEE Network, 11(6):10–23, 1997.

[TP03] S. Thajchayapong and J. M. Peha. Mobility patterns in microcellular wireless networks. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC) 4(1):1963–1968, 2003.

[TT97] M. Taqqu and V. Teverosky. Is network traffic self-similar or multifractal?. Fractals, 5(1), 63–73, 1997.

[TT98] M. S. Taqqu and V. Teverovsky. Estimating long-range dependence in finite and infinite variance series. In A Practical Guide to Heavy Tails: Statistical Techniques for Analyzing Heavy-Tailed Distributions, Birkhäuser, 1998.

[TWS97] M. S. Taqqu, W. Willinger and R. Sherman. Proof of a fundamental result in self-similar traffic modeling. ACM/SIGCOMM Computer Communications Review, 27(2):5–23, 1997.

[VC02] F. J. Velez and L. M. Correia. Mobile broadband services: classification, characterization, and deployment scenarios. IEEE Communications Magazine, 40(4):142–150, 2002.

[VHS04] T. Varga, B. Haverkamp and B. Sanders. Analysis and modeling of WAP traffic in GPRS networks. To be published at 16th ITC specialist seminar, 2004.

[VLLX02] J. D. Vriendt, P. Laine, C. Lerouge and X. Xu. Mobile network evolution: a revolution on the move. IEEE Communications Magazine, 40(4):104–111, 2002.

[Wil01] C. Williamson. Internet Traffic Measurement. Technical Report. University of Calgary, 2001.

[WP98] W. Willinger and V. Paxson. Where mathematics meets the internet. Notices of the American Mathematical Society, 45(8):961–970, 1998.

[WPRT01] W. Willinger, V. Paxson, R. Reidi, and M. Taqqu. Long-Range Dependence and Data Network Traffic. Long-range Dependence: Theory and Applications, Birkhauser, 2001.

[WPT98] W. Willinger, V. Paxson, and M. S. Taqqu. Self-similarity and heavy tails: Structural modeling of network traffic. A Practical Guide to Heavy Tails: Statistical Techniques and Applications, Birkhauser, 1998.

[WTE96] W. Willinger, M. S. Taqqu and A. Erramilli. A bibliographical guide to self-similar traffic and performance modeling for modern high-speed networks. Stochastic Networks: Theory and Applications, pp. 339–366, 1996.

[WTSW95] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson. Self-similarity through high-variability: statistical analysis of ethernet LAN traffic at the source level. In Proceedings of conference on Applications, technologies, architectures, and protocols for computer communication, pp. 100–113. ACM Press, 1995.

[ZAB99] M. Zeng, A. Annamalai, and V. K. Bhargava. Recent advances in cellular wireless communications. IEEE Communications Magazine, 37:128–138, 1999.

[ZD97] M. M. Zonoozi and P. Dassanayake. User mobility modeling and characterization of mobility patterns. IEEE Journal on selected areas in communications, 15(7):1239–1252, September 1997.

[ZBPS02] Y. Zhang, L. Breslau, V. Paxson, and S. Shenker. On the characteristics and origins of Internet flow rates. In proceedings of conference on Applications, technologies, architectures, and protocols for computer communications, pp 309–322, 2002.

Standards

[RFC2018] M. Mathis, J. Mahdavi, S. Floyd, A. Romanow. RFC 2018 TCP Selective Acknowledgement Options. October 1996.

[RFC0768] J. Postel. RFC 0768 User Datagram Protocol. Aug-28-1980.

[RFC0791] J. Postel. RFC 0791 Internet Protocol. Sep-01-1981. (Updated by RFC1349)

[RFC0959] J. Postel, J.K. Reynolds. RFC 0959 File Transfer Protocol. (Updated by RFC2228, RFC2640, RFC2773) Oct-01-1985.

[RFC0959] J. Postel, J.K. Reynolds. RFC 0959 File Transfer Protocol. Oct-01-1985. (Updated by RFC2228, RFC2640, RFC2773)

[RFC1191] J.C. Mogul, S.E. Deering. RFC 1191 Path MTU discovery. Nov-01-1990.

[RFC1323] V. Jacobson, R. Braden, D. Borman. RFC 1323 TCP Extensions for High Performance. May 1992.

[RFC1700] J. Reynolds, J. Postel. RFC 1700 Assigned Numbers. October 1994 (Obsoleted by RFC3232)

[RFC1939] J. Myers, M. Rose. RFC 1939 Post Office Protocol - Version 3. (Updated by RFC1957, RFC2449) May 1996.

[RFC2401] S. Kent, R. Atkinson. RFC 2401 Security Architecture for the Internet Protocol. November 1998.

[RFC2414] M. Allman, S. Floyd, C. Partridge. RFC 2414 Increasing TCP's Initial Window. September 1998.

[RFC2481] K. Ramakrishnan, S. Floyd. RFC 2481 A Proposal to add Explicit Congestion Notification (ECN) to IP. January 1999.

[RFC2581] M. Allman, V. Paxson, W. Stevens. RFC 2581 TCP Congestion

Control. April 1999.

[RFC2616] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee. RFC 2616 Hypertext Transfer Protocol - HTTP/1.1. June 1999.

[RFC2821] J. Klensin. RFC 2821 Simple Mail Transfer Protocol. April 2001.

[RFC2865] C. Rigney, S. Willens, A. Rubens, W. Simpson. RFC 2865 Remote Authentication Dial In User Service (RADIUS). June 2000.

[RFC2866] C. Rigney. RFC 2866 RADIUS Accounting. June 2000.

[RFC2988] V. Paxson, M. Allman. RFC 2988 Computing TCP's Retransmission Timer. November 2000.

[RFC3042] M. Allman, H. Balakrishnan, S. Floyd. RFC 3042 Enhancing TCP's Loss Recovery Using Limited Transmit. January 2001.

[RFC3155] S. Dawkins, G. Montenegro, M. Kojo, V. Magret, N. Vaidya. RFC 3155 End-to-end Performance Implications of Links with Errors. August 2001.

[RFC3232] J. Reynolds. RFC 3232 - Assigned Numbers: RFC 1700 is Replaced by an On-line Database. January 2002.

[RFC3481] H. Inamura, Ed., G. Montenegro, Ed., R. Ludwig, A. Gurtov, F. Khafizov. RFC 3481 TCP over Second (2.5G) and Third (3G) Generation Wireless Networks. February 2003.

[RFC3501] M. Crispin. RFC 3501 INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1. March 2003.

[RFC3517] E. Blanton, M. Allman, K. Fall, L. Wang. RFC 3517 A Conservative Selective Acknowledgment (SACK)-based Loss Recovery Algorithm for TCP. April 2003.

[TR101112] ETSI, TR 101 112 Universal Mobile Telecommunications System (UMTS); Selection procedures for the choice of radio transmission technologies of the UMTS V3.2.0 (UMTS 30.03 version 3.2.0) (1998-04).

[TS22.105] 3GPP, TS 22.105 Technical Specification Group Services and System Aspects Service aspects; Services and Service Capabilities (3G TS 22.105 version 3.7.0) (1999-12).

[TS23.140] 3GPP, TS 23.140 Technical Specification Group Terminals; Multimedia Messaging Service (MMS); Functional description; Stage 2 (V3.1.0 Release 1999) (2002-06).

[TS29.061] 3GPP, TS 29.061 Technical Specification Group Core Network; Interworking between the Public Land Mobile Network (PLMN) supporting packet based Services and Packet Data Networks (PDN) (V3.14.0 Release 1999) (2003-12).

[GSM0260] ETSI, GSM 02.60 Technical Specification Digital cellular telecommunications system (Phase 2+); General Packet Radio Service (GPRS); Service description, Stage 1 (GSM 02.60

version 8.1.0 Release 1999) (1999-07).

[GSM03.60] ETSI, GSM 03.60 Digital cellular telecommunications system (Phase 2+); General Packet Radio Service (GPRS); Service description; Stage 2 (GSM 03.60 version 7.4.0 Release 1998) (2000-03).

[GSM03.64] ETSI, GSM 03.64 Digital cellular telecommunications system (Phase 2+); General Packet Radio Service (GPRS); Overall description of the GPRS radio interface; Stage 2 (GSM 03.64 version 8.5.0 Release 1999) (2000-07).

[GSM04.60] ETSI, GSM 04.60 Digital cellular telecommunications system (Phase 2+); General Packet Radio Service (GPRS); Mobile Station (MS) - Base Station System (BSS) interface; Radio Link Control/ Medium Access Control (RLC/MAC) protocol (GSM 04.60 version 8.5.0 Release 1999) (2000-07).

[GSM04.64] ETSI, GSM 04.64 Digital cellular telecommunications system (Phase 2+); General Packet Radio Service (GPRS); Mobile Station - Serving GPRS Support Node (MS-SGSN) Logical Link Control (LLC) layer specification (GSM 04.64 version 8.3.0 Release 1999) (2000-2).

[GSM04.65] ETSI, GSM 04.65 Digital cellular telecommunications system (Phase 2+); General Packet Radio Service (GPRS); Mobile Station (MS) - Serving GPRS Support Node (SGSN); Subnetwork Dependent Convergence Protocol (SNDCP) (GSM 04.65 version 8.0.0 Release 1999) (2000-2).

[GSM05.02] ETSI, GSM 05.02 Digital cellular telecommunications system (Phase 2+); Multiplexing and multiple access on the radio path (GSM 05.02 version 8.5.0 Release 1999) (2000-07).

[GSM05.03] ETSI, GSM 05.03 Digital cellular telecommunications system (Phase 2+); Channel coding (GSM 05.03 version 8.5.0 Release 1999) (2000-07).

[GSM09.60] ETSI, GSM 09.60 Digital cellular telecommunications system (Phase 2+); General Packet Radio Service (GPRS); GPRS Tunnelling Protocol (GTP) across the Gn and Gp Interface; (GSM 09.60 version 7.1.0 Release 1998) (1999-09).

[WAP205] WAP Forum, WAP-205-MMSArchOverview-20010425-a WAP MMS Architecture Overview, Version 25-April-2001.

[WAP206] WAP Forum, WAP-206-MMSCTR-20020115-a WAP MMS Client Transactions, Version 15-Jan-2002.

[WAP224] WAP Forum, WAP-224-WTP-20010710-a Wireless Transaction Protocol, Version 10-Jul-2001.

[WAP225] WAP Forum, WAP-225-TCP-20010331-a Wireless Profiled TCP, Version 31-March-2001.

[WAP229] WAP Forum, WAP-229-HTTP-20010329-a Wireless Profiled HTTP Version 29-Mar-2001.

[WAP230] WAP Forum, WAP WSP WAP-230-WSP Wireless Session Protocol Specification, Version 5-July-2001.

Links and Tools

[3GPP] 3GPP - 3rd Generation Partnership Project
http://www.3gpp.org

[AEST] AEST - The Scaling Estimator Tool
http://www.cs.bu.edu/faculty/crovella/aest.html

[APACHE] Apache HTTP Server
http://httpd.apache.org

[DATAPLOT] DATAPLOT - Tool for Exploratory Data Analysis
http://www.itl.nist.gov/div898/software/dataplot/homepage.htm

[ISC] ISC - Internet Systems Consortium
http://www.isc.org

[LDCODE] LDcode - Log scale Diagram Utility for Self-Similar Traffic Traces
http://www.cubinlab.ee.mu.oz.au/~darryl/secondorder_code.html

[MATLAB] MATLAB - Powerful Mathematic Utility
http://www.mathworks.com

[NS2] NS-2 Simulator
http://www.isi.edu/nsnam/ns/index.html

[OMA] OMA - Open Mobile Alliance http://www.openmobilealliance.org/

[OPNET] OPNET Simulator, OPNET Technologies Inc.
http://www.opnet.com

[SELFIS] SELFIS - Tool to Explore Self-Similar Traces
http://www.cs.ucr.edu/~tkarag/SELFIS/SELFIS.html

[TCPDPRIV] TCPDPRIV - Trace Sanitizer
http://ita.ee.lbl.gov/html/contrib/tcpdpriv.html

[TCPDUMP] TCPDUMP - Packet Tracer
http://www.tcpdump.org

[TCPURIFY] TCPurify - Trace Sanitizer
http://masaka.cs.ohiou.edu/~eblanton/tcpurify/

[XMGRACE] XmGrace - Tool for Statistical Evaluation of Large Datasets
http://plasma-gate.weizmann.ac.il/Grace/