A simple model for the IP packet service time in UMTS networks

Marc C. Necker

Institute of Communication Networks and Computer Engineering (IKR)
University of Stuttgart, Pfaffenwaldring 47, D-70569 Stuttgart, Germany
email: necker@ikr.uni-stuttgart.de

Abstract This paper presents a simple model for the service time of IP packets in UMTS networks. The model reflects both the delay characteristics for an individual IP packet, as well as the correlation of the delay of successive IP packets. Our goal is not to obtain a quantitatively exact model. Instead, we provide a simple and easy to implement model, which qualitatively reflects the statistical properties of a UMTS radio link. Such a model is of great interest for network level simulations, where cross-layer interactions are the main focus. It provides a simple means to investigate the behavior of higher-layer protocols in a UMTS network without the need to implement a detailed model of the UMTS link layer. We verify our model by comparing the statistical properties of the IP packet service time to those obtained using a detailed UMTS model. Additionally, the results of our TCP simulations show a very good match between the simple model and the detailed UMTS link layer model.
Keywords: UMTS, WCDMA, ARQ, reordering delay

1 Introduction

It is well known that cellular communication systems of the 2nd and 3rd generation feature a relatively high packet delay and a large delay jitter if they are used for packet switched communication. Both factors may impose problems to higher layer protocols, which are usually designed to operate in wireline networks offering a small round trip time (RTT) and a small delay jitter. These cross-layer interactions between the data link layer of a mobile communication system and higher layers have extensively been studied especially for TCP (e.g. [1], [2], [3], [4]).

The loss probability of a MAC frame in cellular mobile networks may be very high. In order to compensate these losses, Automatic Repeat Request (ARQ) mechanisms are applied on the link layer. In UMTS, a selective repeat ARQ (SR-ARQ) mechanism [5] is applied on the Radio Link Control (RLC) layer. Usually, a cellular mobile network provides sequence integrity and delivers IP packets in-order to higher layers. As a consequence, the loss of a MAC frame will delay the delivery of all subsequent correctly received frames until the lost MAC frame has been retransmitted successfully. This additional delay is usually referred to as *reordering delay* or *resequencing delay*. As the detection of a loss and the retransmission of the corresponding packet take a significant amount of time, the reordering delay accounts for the high delay jitter mentioned above [6].

Resequencing delays have extensively been studied in the past. In [7], Rosberg and Shacham derive the distribution of the resequencing delay and the buffer occupancy in an SR-ARQ system. Their underlying system is a slotted system, where each slot is protected by SR-ARQ and one arriving packet fits into a single slot. A similar assumption was made by Konheim in [8].

In [9], Rossi and Zorzi present an accurate heuristic approach to evaluate the delay of packets in UMTS networks. The authors consider both SR-ARQ and in-order delivery and use their analysis to determine the optimal number of retransmission attempts on the RLC layer. In this paper, we present a similar model for the IP packet delay. We first derive a model of the ARQ mechanism in UMTS and then analytically derive the complementary cumulative distribution function (ccdf) of the IP packet delay. As a major contribution, we use the final result to implement a simple model for the IP packet delay in a network level simulator, and show that it is well suitable to evaluate the performance of transport layer protocols in UMTS networks. This eventually allows the investigation of higher layer protocols and services with respect to their performance in mobile environments without the need to implement a detailed model of the Radio Access Network.

This paper is structured as follows. In section 2, we introduce the considered UMTS system and briefly derive the reordering delay of IP packets in section 3. In section 4, we develop a simple abstract queueing model for the UMTS system. We verify this model in section 5 and show its applicability to transport layer studies. Finally, section 6 concludes the paper.

2 System Model

The system which we will consider in this paper is shown in Fig. 1. It consists of a single cell, where the User Equipment (UE) connects to a Node B of the UMTS Radio Access Network (UTRAN) via a Dedicated Channel (DCH) or a Downlink Shared Channel (DSCH) in the downlink direction and a DCH in the uplink direction. The Node B is connected to the Radio Network Controller (RNC), which itself is connected to the Internet via the 3G-SGSN and the 3G-GGSN of the cellular system's core network. Finally, the UE establishes a data connection with a host in the Internet. The Internet and core network were assumed to introduce a constant delay T_{INet} and not lose any packets.

This reference UTRAN system was modeled in detail with all its relevant protocols. The corresponding model is shown in Fig. 2 together with the UE and fixed network components. The IP packets from an IP source are first delayed by T_{INet} and then buffered

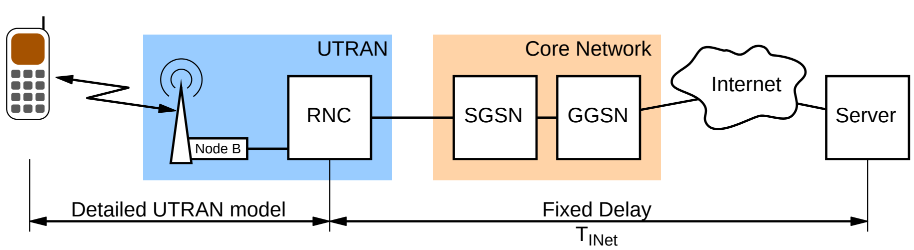

Figure 1. Block diagram of the considered system

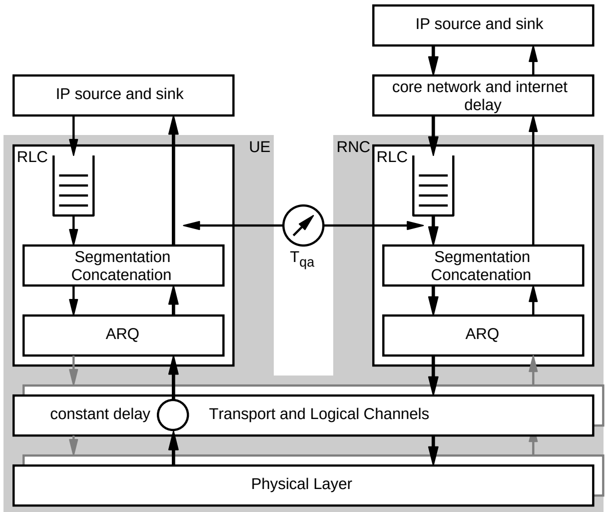

Figure 2. UTRAN reference model including UE and fixed network components

in the bounded FIFO queue of the RNC. Afterwards, they are segmented/concatenated into radio blocks. The following ARQ mechanism protects from frame losses, which may occur on the physical layer. It was configured according to the optimal parameters found in [6], but with a maximum transmission and reception window size of 2048 PDUs. Just before the transmission via a transport channel, a number of radio blocks is grouped together to form a Radio Block Set (RBS), which is then transmitted within one MAC frame. MAC frames were randomly dropped on the physical layer according to a uniform distribution with probabilities $P_{\mathrm{L,DL}}$ and $P_{\mathrm{L,UL}}$ in the downlink and uplink, respectively. The uniform drop distribution is a valid assumption under perfect power control [10] and goes well along with the models of other recent works (e.g. [11], [12], [13]).

3 IP packet reordering delay

In this section, we will examine the IP packet reordering delay as a prerequisite to derive the queueing model of the UMTS radio link in section 4. We will first describe the UMTS data link model in section 3.1 and then derive the delay of an isolated IP packet in section 3.2. Subsequently, we will extend our study to the delay of a sequence of IP packets in section 3.3.

3.1 UMTS data link layer model

Figure 3 shows the segmentation of an IP packet into $b = \lceil \frac{T_{\mathrm{IP}}}{T_{\mathrm{TTI}}} \rceil$ RLC block sets in the UMTS data link. Each RLC block set is transmitted within one Transmission Time Interval (TTI) of length T_{TTI}, corresponding to one MAC frame. Each RLC block set is assumed to be lost with equal and independent probability P_L, where P_L is equal to $P_{\mathrm{L,DL}}$ and $P_{\mathrm{L,UL}}$ depending on the link direction. If a loss is detected, the RLC block set is retransmitted, where the retransmission may be lost again. The probability of n necessary transmission attempts for an RLC block set is denoted with $P_{\mathrm{R},n}$:

$$P_{\mathrm{R},n} = P_L^n(1 - P_L) \quad . \tag{1}$$

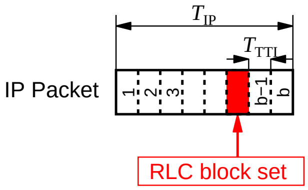

Figure 3. IP Packet segmentation

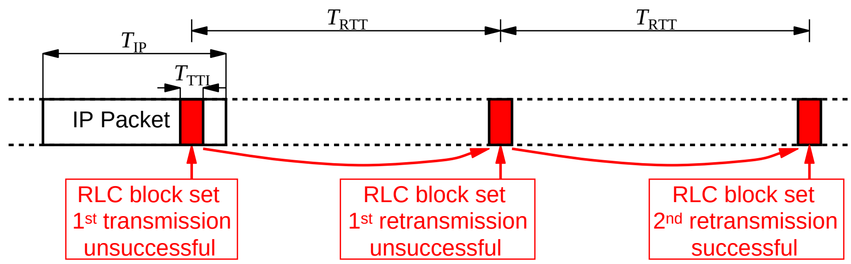

Figure 4. Retransmission of RLC block sets after T_{RTT}

3.2 Delay of an isolated IP packet

The delay of an isolated IP packet is determined by those RLC block sets which need the most retransmission attempts until they are received successfully. If we assume that all retransmissions need the same time, all other RLC block sets which need less retransmissions do not account for the delay of the IP packet. Let n be the number of retransmission attempts of those RLC block sets with the most necessary retransmissions. Let further $P_{\mathrm{IP},n}$ denote the probability that one or more RLC block sets of the IP packet needs n retransmission attempts, and all other RLC block sets need less retransmission attempts. $P_{\mathrm{IP},n}$ can be calculated from $P_{\mathrm{R},i}$ as follows:

$$P_{\mathrm{IP},0} = P_{\mathrm{R},0}^{b} \tag{2}$$

$$P_{\mathrm{IP},1} = P_{\mathrm{R},1}^{b} + \binom{b}{1} P_{\mathrm{R},1}^{b-1} P_{\mathrm{R},0} + \binom{b}{2} P_{\mathrm{R},1}^{b-2} P_{\mathrm{R},0}^{2} + \cdots + \binom{b}{b-1} P_{\mathrm{R},1} P_{\mathrm{R},0}^{b-1} \tag{3}$$

$$\begin{aligned} P_{\mathrm{IP},2} = {} & P_{\mathrm{R},2}^{b} + \binom{b}{1} P_{\mathrm{R},2}^{b-1} (P_{\mathrm{R},0} + P_{\mathrm{R},1}) + \binom{b}{2} P_{\mathrm{R},2}^{b-2} (P_{\mathrm{R},0} + P_{\mathrm{R},1})^{2} \\ & + \cdots + \binom{b}{b-1} P_{\mathrm{R},2} (P_{\mathrm{R},0} + P_{\mathrm{R},1})^{b-1} \ . \end{aligned} \tag{4}$$

In general, we can write:

$$P_{\mathrm{IP},i} = \begin{cases} P_{\mathrm{R},0}^{b} & i = 0 \\ \sum_{k=0}^{b-1} \binom{b}{k} P_{\mathrm{R},i}^{b-k} (1 - P_L^i)^k & \text{otherwise} \end{cases} \tag{5}$$

$$P_{\mathrm{IP},i} = \begin{cases} P_{\mathrm{R},0}^{b} & i = 0 \\ (1 - P_L^{i+1})^b - (1 - P_L^i)^b & \text{otherwise} \end{cases} \tag{6}$$

with $\sum_{l=0}^{i-1} P_{\mathrm{R},l} = 1 - P_L^i$ and $b = \lceil \frac{T_{\mathrm{IP}}}{T_{\mathrm{TTI}}} \rceil$.

Figure 4 illustrates the retransmission of RLC block sets. We assume a lost RLC block set to be retransmitted after one RTT T_{RTT}, which also accounts for the delay it takes the receiver to recognize the loss and notify the sender. Note that T_{RTT} is a model parameter and does not necessarily correspond to the actual RTT of the system. Let X_{IP} be the random variable describing the delay of an isolated IP packet. The pdf f_{IP} of X_{IP} can easily be calculated from $P_{\mathrm{IP},i}$:

$$f_{\mathrm{IP}}(t) = \sum_{i=0}^{\infty} P_{\mathrm{IP},i} \cdot \delta(t - i \cdot T_{\mathrm{RTT}}) \ . \tag{7}$$

Consequently, the cdf $F_{\mathrm{IP}}(t)$ and the ccdf $\bar{F}_{\mathrm{IP}}(t)$ yield to:

$$\bar{F}_{\mathrm{IP}}(t) = 1 - F_{\mathrm{IP}}(t) = 1 - \sum_{i=0}^{\infty} P_{\mathrm{IP},i} \cdot s(t - i \cdot T_{\mathrm{RTT}}) \quad , \tag{8}$$

where $s(t)$ is the step function. Figure 5 shows the ccdf $\bar{F}_{\mathrm{IP}}(t)$. The graph leads to the assumption that $\bar{F}_{\mathrm{IP}}(t)$ has a negative exponential decay. To be more precise, we can assume from the figure that the points $\bar{F}_{\mathrm{IP}}(lT_{\mathrm{RTT}})$ lie on a negative exponential function if l is not too small. To proof this assumtion, we investigate $\bar{F}_{\mathrm{IP}}(t)$ in the logarithmic domain, were the points $\bar{F}_{\mathrm{IP}}(lT_{\mathrm{RTT}})$ supposedly lie on a straight line for large l. The gradient μ of this line yields to:

$$\mu = \lim_{l \to \infty} \frac{\ln(\bar{F}_{\mathrm{IP}}((l+1)T_{\mathrm{RTT}}) - \ln(\bar{F}_{\mathrm{IP}}(lT_{\mathrm{RTT}}))}{T_{\mathrm{RTT}}} \quad . \tag{9}$$

In order to derive the limit, we rewrite eq. (9) as:

$$\begin{aligned}
\mu T_{\mathrm{RTT}} &= \lim_{l \to \infty} \ln \left(\frac{\bar{F}_{\mathrm{IP}}((l+1)T_{\mathrm{RTT}})}{\bar{F}_{\mathrm{IP}}(lT_{\mathrm{RTT}})} \right) = \lim_{l \to \infty} \ln \left(\frac{1 - \sum_{i=0}^{l+1} P_{\mathrm{IP},i}}{1 - \sum_{i=0}^{l} P_{\mathrm{IP},i}} \right) \\
&= \lim_{l \to \infty} \ln \left(\frac{1 - \sum_{i=0}^{l+1} \left((1 - P_L^{i+1})^b - (1 - P_L^{i})^b\right)}{1 - \sum_{i=0}^{l} \left((1 - P_L^{i+1})^b - (1 - P_L^{i})^b\right)} \right) \\
&= \lim_{l \to \infty} \ln \left(\frac{1 - (1 - P_L^{l+2})^b}{1 - (1 - P_L^{l+1})^b} \right) \quad .
\end{aligned} \tag{10}$$

The last expression can be solved by applying the rule of Bernoulli-l'Hospital, and we finally obtain the limit of μ as:

$$\mu = \frac{\ln P_L}{T_{\mathrm{RTT}}} \tag{11}$$

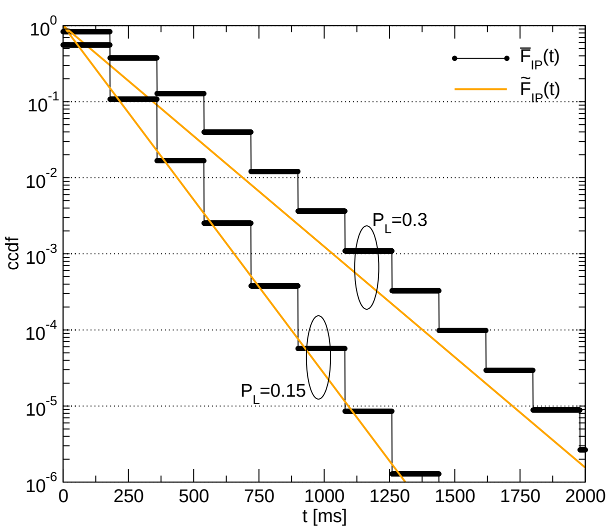

Figure 5. ccdf $\bar{F}_{\mathrm{IP}}(t)$ of the delay of an isolated IP packet for different P_L and its approximation $\tilde{F}_{\mathrm{IP}}(t)$

The above made assumption is thereby proofed and the approximation $\bar{F}_{\mathrm{IP}}(t)$ is now obvious:

$$\bar{F}_{\mathrm{IP}}(t) \approx \tilde{F}_{\mathrm{IP}}(t) = e^{\mu t} = e^{\frac{\ln P_L}{T_{\mathrm{RTT}}} t} \quad . \tag{12}$$

Note that μ is always negative since $P_L < 1$. In [9], Rossi and Zorzi found virtually the same result by heuristic means. Figure 5 compares the graph of the approximations $\tilde{F}_{\mathrm{IP}}(t)$ with $\bar{F}_{\mathrm{IP}}(t)$. The approximation shows a good match to the original ccdf $\bar{F}_{\mathrm{IP}}(t)$, which could even be improved by timeshifting the approximation to the right. In general, we can say that the approximation may even better resemble the behavior of a real UTRAN system, as the hard steps of the original ccdf $\bar{F}_{\mathrm{IP}}(t)$ are avoided.

3.3 Delay of IP packets within sequence

If we assume a sequence of IP packets and not only an isolated IP packet, the reordering delay of packets at the receiver becomes important. Note that if out-of-order delivery were to be assumed, the distribution of the IP packet service time would follow the same distribution as in the isolated packet case.

Let X_i be the random variable describing the delay of the ith IP packet in a sequence if in-order delivery is activated. An example of a sequence is shown in Fig. 6. The sequence may have gaps, and IP packets can arrive at arbitrary time instances. The samples T_i of X_{IP} denote the reordering delay of the ith packet, and t_i denotes the beginning of its service time. Consider the nth packet in Fig. 6. Since in-order delivery is activated, the nth packet may not be delivered to higher layers before any previous packet was correctly received and delivered. That is, the reordering delay of the nth packet corresponds to the maximum of all delays T_i of all previous packets corrected by their arrival times t_i, and the delay T_n of itself. Consequently, X_i follows from the following maximum relation:

$$X_n = \max_{i, i \leq n} (X_{\mathrm{IP}} - (t_n - t_i)) \quad . \tag{13}$$

This equation can be rewritten in a recursive way as follows:

$$X_n = \max(X_{\mathrm{IP}}, X_{n-1} - (t_n - t_{n-1})) \quad . \tag{14}$$

That is, the reordering delay of the nth packet equals the maximum of its own delay due to retransmissions and the delay of the previous packet corrected by the inter-arrival time. Eq. (14) can easily be implemented in a network level simulator, even if the arrival process of IP packets is arbitrary or unknown, by simply keeping track of the service time end of

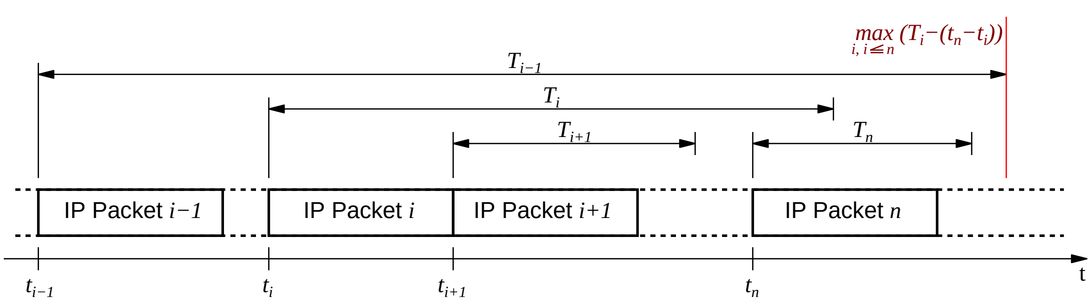

Figure 6. Sequence of IP packets

the last transmitted packet and applying the maximum operation on each new packet.

Note that if an infinitely long sequence of IP packets without gaps were to be considered, $t_i - t_n$ would equal $(i-n)T_{\mathrm{IP}}$. In the latter case, the ccdf $\bar{F}(t)$ of X_i follows as:

$$\begin{aligned}\bar{F}(t) &= 1 - \prod_{n=0}^{\infty}\left(1 - \bar{F}_{\mathrm{IP}}(t + n \cdot T_{\mathrm{IP}})\right) \\ &= 1 - \prod_{n=0}^{\infty}\left(1 - \sum_{i=0}^{\infty} P_{\mathrm{IP},i} \cdot s(t - i \cdot T_{\mathrm{RTT}} + n \cdot T_{\mathrm{IP}})\right) \quad ,\end{aligned} \tag{15}$$

and with eq. (12) the approximation $\tilde{F}(t)$ of $\bar{F}(t)$ yields to:

$$\tilde{F}(t) = 1 - \prod_{n=0}^{\infty}\left(1 - e^{\mu(t+nT_{\mathrm{IP}})}\right) = 1 - \prod_{n=0}^{\infty}\left(1 - e^{\frac{\ln P_L}{T_{\mathrm{RTT}}}(t+nT_{\mathrm{IP}})}\right) \quad . \tag{16}$$

4 UMTS Queueing Model

In this section, we will derive an abstract queueing model for the UMTS data link. The model shall accurately reflect the delay statistics on the IP layer including the correlation properties. In order to derive the model, we consider the arbitrary sequence of IP packets of section 3.3. Each of these IP packets is first stored in the radio network's input buffer, as noted in section 2. The UMTS data link reads the packets from the buffer at the effective line speed. Subsequently, a read packet is delayed by the line speed transmission time and additional processing and propagation delay. Finally, the IP packets experience the reordering delay at the receiver.

This behavior can directly be mapped to the queueing model shown in Fig. 7. A traffic generator generates IP packets, which are first stored in a bounded FIFO queue. Subsequently, a single server delays the IP packets according to the effective line speed R of the radio link. That is, the service time h_{LS} is deterministic and depends on the packet length L_{IP}:

$$h_{\mathrm{LS}} = \frac{L_{\mathrm{IP}}}{R(1-P_L)} \quad . \tag{17}$$

The subsequent infinite server accounts for processing and propagation delays and for the

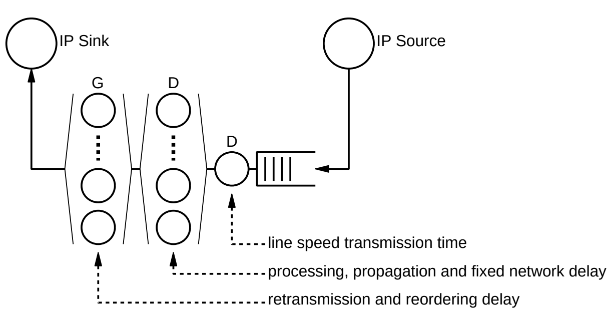

Figure 7. Queueing model of the UMTS radio link

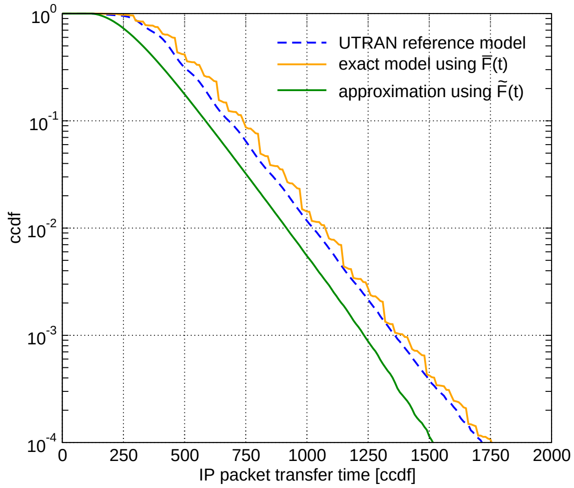

Figure 8. Transfer time of IP packets for different models, $P_{\mathrm{L,DL}} = 0.2$, $P_{\mathrm{L,UL}} = 0.1$

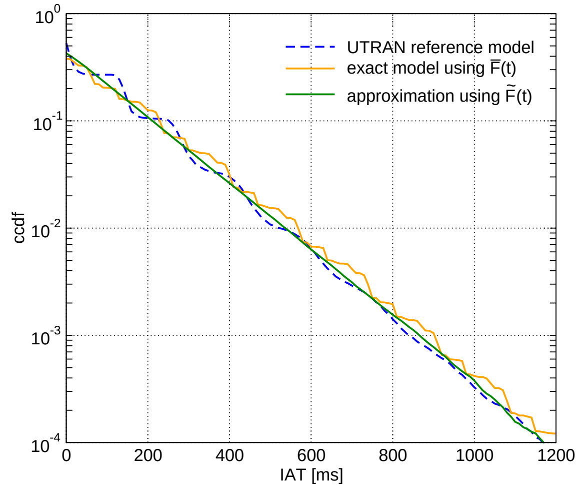

Figure 9. IAT of IP packets at the receiver for different models, $P_{\mathrm{L,DL}} = 0.2$, $P_{\mathrm{L,UL}} = 0.1$

delay introduced by the core network and the Internet. Its service time h_{F} is constant and usually on the order of $50 - 100$ms.

The final infinite server accounts for the reordering delay introduced by the ARQ mechanism. The service time of this infinite server follows a general distribution. In section 3, we discussed its approximation as well as an efficient implementation for simulation environments.

5 Model Verification

5.1 IP packet service time comparison

The model described in section 4 was implemented in a network level simulator both using $\bar{F}(t)$ and $\tilde{F}(t)$ to model the reordering delay. We considered a 256kBit/s DCH in the down- and uplink direction. Bulk data traffic was generated by a greedy TCP NewReno sender with activated window scaling in the downlink direction. All results obtained with the queueing model were compared to results obtained from simulations using the detailed UTRAN model of Fig. 2 and to a link with a constant delay of 55ms and 68ms in the down- and uplink direction, respectively[1].

Figure 8 shows the ccdf of the queue adjusted IP packet delay T_{qa} for all three considered models. T_{qa} is measured as shown in Fig. 2. It does not contain the queueing delay in the RNC, which is mainly determined by the traffic source behavior. Fig. 9 plots the IAT of IP packets delivered to higher layers at the UE. It shows a very good match between the queueing model based on the exact ccdf $\bar{F}(t)$ and the detailed UTRAN model for both the delay T_{qa} and the IAT. The queueing model based on the approximation $\tilde{F}(t)$ shows a good match with the detailed model for the IAT, while it slightly underestimates the delay T_{qa}. This goes back to the underestimation of the delay of an isolated IP packet, which has already been observed in section 3.2. From Fig. 8 it becomes obvious that it could easily be avoided by adding the already mentioned time shift in eq. (12), which would have no impact on the IAT.

[1] This corresponds to the minimum delay of the UMTS radio bearer if no retransmissions occur.

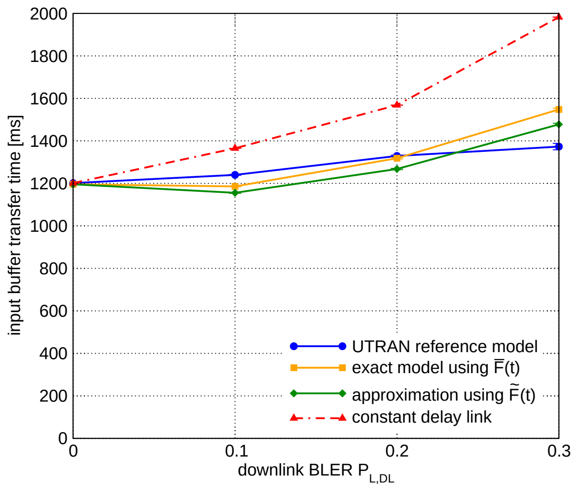

Figure 10. Transfer time of IP packets through input buffer. $P_{\mathrm{L,UL}} = 1/2P_{\mathrm{L,DL}}$

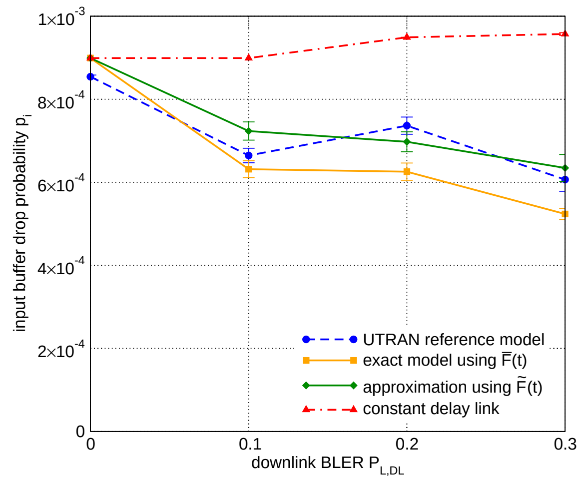

Figure 11. IP packet Loss probability at input buffer. $P_{\mathrm{L,UL}} = 1/2P_{\mathrm{L,DL}}$

5.2 Cross-layer interactions with TCP

Fig. 10 plots the transfer time of IP packets through the downlink input buffer in the RNC in dependence of $P_{\mathrm{L,DL}}$. The size of the input buffer was chosen to be 50kBytes. Plotted is the transfer time for all models considered in the previous section and the constant delay link. The chart reveils a very good match between the proposed models based on $\bar{F}(t)$ and $\tilde{F}(t)$ for reasonable BLERs $P_{\mathrm{L,DL}}$. The overestimation for large $P_{\mathrm{L,DL}}$ results from the fact that at high BLERs the UMTS link layer starts being unreliable. In contrast, the queueing model never looses packets, hence it overestimates the buffer occupancy and the transfer time. Figure 11 shows the loss probability for an IP packet at the same input buffer. Again, we can observe a good match between the detailed model and our proposed queueing model. For both the transfer time and the loss probability, a simplistic link model with a constant delay delivers a strong overestimate.

Finally, we would like to investigate whether our model is suitable for the design of transport layer protocols in mobile environments. We do this by investigating two well known transport protocols, namely TCP Reno [14] and TCP NewReno [15]. One of the key improvements of TCP NewReno over TCP Reno is the better handling of the loss of multiple TCP segments from one transmission window, which expresses itself in a reduced number of TCP timeouts. In the following, we will provoke the loss of multiple TCP segments from one transmission window and verify whether our queueing model accurately reflects the behavior of both TCP variants. We therefore add a second greedy TCP sender, which shares the single end-to-end link and thus the same DCH with the already present TCP sender. As a consequence, packets of both TCP connections flow through the same input buffer in the RNC. This will eventually lead to one TCP connection causing the other TCP connection to lose packets due to a buffer overflow.

Figure 12 shows the number of observed TCP timeouts per transmitted TCP Maximum Segment Sizes (MSS) depending on the size of the RNC input buffer for one of the TCP connections. The figure contains the results obtained with TCP Reno for all four considered network models. For all models, the number of timeouts decreases as the buffer size increases. This is an expected and well known result. It is more interesting

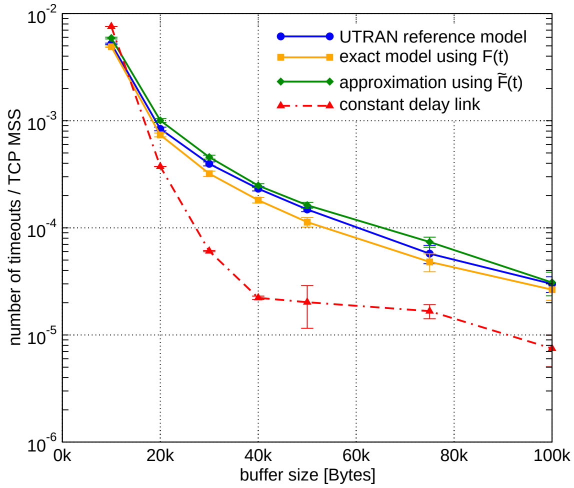

Figure 12. Number of TCP timeouts per MSS for TCP Reno

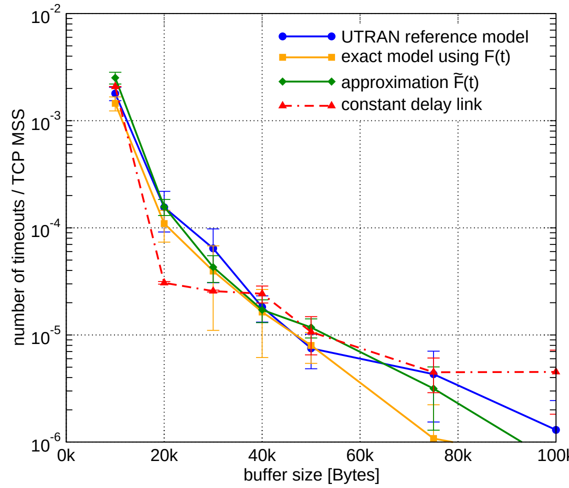

Figure 13. Number of TCP timeouts per MSS for TCP NewReno

to observe that both models according to Fig. 7 deliver almost the same result as the reference UTRAN model according to Fig. 2, while the constant delay link produces an order of magnitude less timeouts.

As discussed before, we expect a decrease in the number of timeouts if we switch from TCP Reno to TCP NewReno. Figure 13 shows the number of TCP timeouts per MSS depending on the buffer size if only NewReno is involved. The graph shows that the proposed queueing model according to Fig. 7 accurately reflects the decrease in the number of timeouts observed in the reference UTRAN model. The decrease can be observed for the model using both $\bar{F}(t)$ and $\tilde{F}(t)$, showing its good suitability to verify or design transport layer protocols in mobile environments.

6 Conclusion

We presented a simple queueing model as a substitute for detailed UTRAN data link layer models. We derived the statistical properties of all servers in the queueing model and implemented the model in an event-driven simulation environment. We showed by simulation, that the model accurately reflects the delay characteristics on the IP layer, including the correlation properties. Consequently, to higher layer protocols, the simple queueing model behaves almost like a real UMTS link. Eventually, our model allows for an easy evaluation of higher layer protocols and services in UMTS environments without the need for detailed UMTS link models.

Acknowledgements

This research was done in cooperation with Alcatel Research and Innovation Department, Stuttgart.The author would like to thank Michael Scharf and Detlef Saß for many fruitful discussions.

References

1. Gurtov, A.: Effect of delays on TCP performance. In: Proc. IFIP Personal Wireless Communication, Lappeenranta, Finland (2001) 87–108
2. Scharf, M., Necker, M.C., Gloss, B.: The sensitivity of TCP to sudden delay variations in mobile networks. In: Proceedings of the 3rd IFIP-TC6 Networking Conference, Lecture Notes in Computer Science (LNCS) 3042, Athens, Greece (2004) 76–87
3. Schüler, J., Gruhl, S., Schwabe, T., Schweigel, M.: Performance improvements for TCP in mobile networks with high packet delay variations. In: Proc. 17th Int. Teletraffic Congress, Salvador, Brazil (2001) 517–528
4. Chan, M.C., Ramjee, R.: TCP/IP performance over 3G wireless links with rate and delay variation. In: Proc. ACM MOBICOM, Atlanta, GA (2002) 71–82
5. Comroe, R.A., Costello, Jr., D.J.: ARQ schemes for data transmission in mobile radio systems. IEEE Journal on Selected Areas in Communications **2** (1984) 472–481
6. Mutter, A., Necker, M.C., Lück, S.: IP-packet service time distributions in UMTS radio access networks. In: Proceedings of the 10th Open European Summer School and IFIP WG 6.3 Workshop (EUNICE 2004), Tampere, Finland (2004) 71–78
7. Rosberg, Z., Shacham: Resequencing delay and buffer occupancy under the selective-repeat ARQ. IEEE Transactions on Information Theory **35** (1989) 166–173
8. Konheim, A.: A queueing analysis of two ARQ protocols. IEEE Transactions on Communications **28** (1980) 1004–1014
9. Rossi, M., Zorzi, M.: An accurate heuristic approach for UMTS RLC delay statistics evaluation. In: Proc. IEEE VTC Spring 2003. Volume 4., Jeju, Korea (2003) 2760–2764
10. Necker, M.C.: Statistical properties of fading processes in WCDMA systems. In: Proc. 2nd International Symposium on Wireless Communication Systems (ISWCS 2005), Siena, Italy (2005)
11. Meyer, M., Sachs, J., Holzke, M.: Performance evaluation of a TCP proxy in WCDMA networks. IEEE Wireless Communications **10** (2003) 70–79
12. Hossain, E., Kim, D.I., Bhargava, V.: Analysis of TCP performance under joint rate and power adaptation in cellular WCDMA networks. IEEE Transactions on Wireless Communications **3** (2004) 865–879
13. Peisa, J., Meyer, M.: Analytical model for TCP file transfers over UMTS. In: Proc. International Conference on Third Generation Wireless and Beyond, San Francisco, CA, USA (2001)
14. Allman, M., Paxson, V., Stevens, W.: TCP congestion control. RFC 2581, IETF (1999)
15. Floyd, S., Henderson, T., Gurtov, A.: The NewReno modification to TCP's fast recovery algorithm. RFC 3782, IETF (2004)