Universität Stuttgart

INSTITUT FÜR KOMMUNIKATIONSNETZE UND RECHNERSYSTEME
Prof. Dr.-Ing. Dr. h. c. mult. P. J. Kühn

Copyright Notice

Institute of Communication Networks and Computer Engineering
University of Stuttgart
Pfaffenwaldring 47, D-70569 Stuttgart, Germany
Phone: ++49-711-685-8026, Fax: ++49-711-685-7983
email: mail@ikr.uni-stuttgart.de, http://www.ikr.uni-stuttgart.de

Parameter Selection for HSDPA Iub Flow Control

Marc C. Necker
Institute of Communication Networks and Computer Engineering,
University of Stuttgart, Pfaffenwaldring 47, D-70569 Stuttgart, Germany
Email: necker@ikr.uni-stuttgart.de

Andreas Weber
Alcatel SEL AG, Research and Innovation
Lorenzstr. 10, D-70435 Stuttgart
Email: Andreas.Weber@alcatel.de

***Abstract*— The recently emerging High Speed Downlink Packet Access (HSDPA) enhances conventional WCDMA systems according to the UMTS standard with data rates of up to 14MBit/s in the downlink direction. This is achieved by using adaptive modulation and coding as well as a fast Hybrid Automatic Repeat Request (HARQ) mechanism. This functionality is implemented close to the air interface in the Node B. In addition to the data buffer in the RNC, this requires a second data buffer in the Node B. Consequently, a flow control mechanism is needed which controls the amount of data to be transmitted from the RNC's buffer to the Node B's buffer. The spatial separation of RNC and Node B imposes significant signaling constraints and control dead time limitations to the flow control mechanism. Additionally, due to the time-varying nature of the radio channel, the data rate towards a particular user may be highly variable. In this paper, we study the impact of the flow control on system performance. In particular, we consider the parameter choice for a previously presented algorithm and highlight some inherent tradeoffs.**

I. Introduction

In the past years, WCDMA networks based on the UMTS standard have widely been deployed. In these systems, several Node B base stations are connected to a Radio Network Controller (RNC) via the Iub interface. The RNC implements all relevant radio protocols, such as the Radio Link Control (RLC) and the MAC-d, while the Node B is a mere slave device, responsible for the actual physical transmission on the air interface. As the RNC and the attached Node Bs are usually distributed over several sites, the data link between a Node B and the RNC introduces a significant additional delay.

HSDPA introduces an additional functional layer in the protocol stack, namely the MAC-hs layer. The MAC-hs functionality is implemented in the Node B, which allows a much faster reaction on errors and variations of the channel quality, compared to protocols implemented in the RNC. This allows for fast adaptations of the modulation and coding scheme, fast scheduling and for a powerful HARQ mechanism [1].

As the HSDPA functionality is distributed, two separate data buffers are required in the RNC and Node B, respectively. Consequently, a data flow control is needed which controls the amount of data to be transmitted from the RNC's buffer to the Node B's buffer. This flow control is typically located in the Node B and signals to the RNC the amount of data to be transmitted. Its goal is to keep the buffer level in the Node B at an adequate level. If the Node B's buffer is too full, the Round Trip Time (RTT) on the RLC layer is unnecessarily increased, causing problems to RLC protocols and other higher layer protocols. On the other hand, a minimum buffer level should always be maintained to prevent the buffer from running empty and thus wasting air interface resources.

Scheduling in mobile communication systems has widely been addressed. A general introduction and in-depth study of scheduling and QoS in HSDPA is provided by Gutiérrez in [2]. In [3], Kolding investigates the performance of Proportional Fair (PF) scheduling in HSDPA systems under non-ideal channel condition reporting. In [4], Aniba and Aissa enhance the PF approach to provide fairness, when channel conditions towards different users are heterogeneous.

The issue of flow control in HSDPA systems has rarely been addressed so far. In [5], Legg presents an optimized Iub flow control algorithm. In [6], we presented a generic Iub flow control algorithm and studied the impact of signaling constraints on the delay performance of IP traffic. In particular, we investigated the impact of the dead time of the flow control loop and the signaling load. We showed by means of simulation that the signaling constraints may dominate the delay performance over other system parameters and algorithms, such as the maximum number of HARQ retransmissions or the scheduling algorithm. Finally, we explored possibilities to improve the performance of the flow control under the given constraints.

In this paper, we present a detailed parameter study of the flow control algorithm in [6]. We investigate the impact of key parameters of the flow control on the delay and throughput performance for different traffic characteristics and different variations of the algorithm. Eventually, we highlight some necessary trade-offs when choosing the parameter set.

Our paper is structured as follows. Section II introduces the investigated system and the corresponding model. In section III, the flow control algorithm from [6] is briefly reviewed. Section IV presents the simulation scenario and studies the influence of the various flow control parameters on the system performance. Finally, section V concludes the paper.

II. System Model

A. System Overview

The basic scenario is shown in Fig. 1. We consider a single-cell environment, where several User Equipments (UEs) connect to the Node B via a High Speed Downlink Shared Channel (HS-DSCH) in the downlink and a dedicated channel (DCH) in the uplink direction. The Node B is

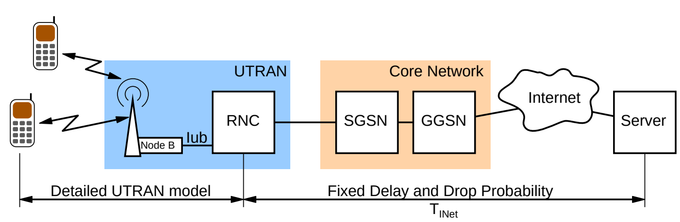

Fig. 1: Architecture of the considered 3G network

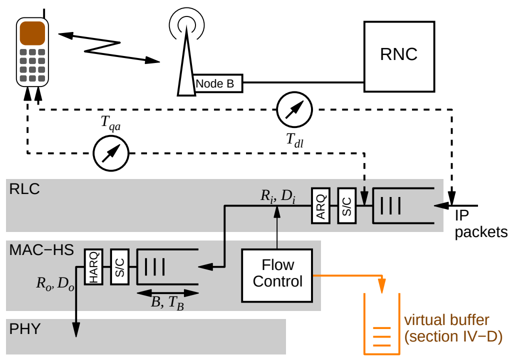

Fig. 2: Flow control overview

connected to the RNC, which itself is connected to the Internet via the 3G-SGSN and 3G-GGSN of the cellular system's core network. The UEs establish a data connection with a host in the Internet. The Internet and core network were assumed to introduce a constant delay $T_{\text{INet}} = 20$ms in each direction and not lose any IP packets.

B. Simulation Model

The HSDPA network was modeled with all its relevant RLC, MAC-d and MAC-hs protocols using an event-driven simulation tool based on the IKR SimLib [7]. The physical layer was based on BLER-curves obtained from link level simulations including HARQ. Transport formats (TF) on the MAC-hs layer were selected based on the channel quality such that the BLER is 10%. We assumed ideal conditions for the reporting of Channel Quality Indicators (CQI) from the mobile terminals, i.e. zero delay. The maximum number of MAC-hs retransmissions and RLC retransmissions was limited to $R_{\text{max,hs}} = 4$ and $R_{\text{max,rlc}} = 10$, respectively. The maximum RLC window size was assumed to be unlimited in order to avoid side effects in the results. We neglect the convergence layer, as it only introduces a very small overhead in a single-cell environment.

III. Flow Control and Scheduling

A. RNC / Node B flow control

The general concept of the flow control is shown in Fig. 2 for one data connection: IP packets arriving at the RNC are first stored in RNC input buffers with one buffer per data connection. The RNC segments and concatenates, respectively, incoming data packets into RLC blocks (S/C). These RLC blocks are protected by the RLC layer's ARQ mechanism and transmitted to the Node B, where they are stored in individual Node B buffers, also known as *HS priority queues*. For our simulations we have chosen the memory large enough to avoid side-effects caused by limited buffer sizes. Furthermore, we assumed that all data flows have the same priority.

In [6], we presented a generic flow control algorithm including some performance enhancements. Its goal is to tune the buffer level so that a predefined queuing time T_w in the Node B's buffer is obtained. That is, the flow control tries to keep the buffer level B_w of every data flow at a value of:

$$B_w = R_o \cdot T_w \quad , \tag{1}$$

where R_o is the bit rate of RLC frames transmitted for the first time over the radio channel. Hence, R_o corresponds to the data connection's effective channel bit rate. This value has to be measured and, in order to be accurate enough, this measurement value has to be averaged over a certain period of time T_m. However, the longer this measurement period, the more obsolete is this value. If no measurement values are available, R_o is set to a predefined value R_{def}.

The original algorithm in [6] calculated the data rate R_i at which data blocks are transmitted from the RNC to the Node B according to the following formula:

$$R_i = \max(0, \underbrace{R_o}_{\text{proportional term}} + \underbrace{\alpha\frac{B_w - B}{T_u}}_{\text{differential term}}) \ . \tag{2}$$

This equation contains a term proportional to the measured output rate R_o and a term depending on the difference between the desired and the actual buffer level. The enhancements to this algorithm in [6] introduced a virtual buffer keeping track of the granted but not yet received resources. Additionally, the proportional term in eq. (2) was limited to an exponential average $R_{\text{o,max}_{\text{new}}}$. In total, the equations of the improved flow control are:

$$R_{\text{o,max}_{\text{new}}} = (1-\beta)R_{\text{o,max}_{\text{old}}} + \beta R_o \tag{3}$$

$$\tilde{R}_o = \max(R_o, R_{\text{o,max}_{\text{new}}}) \tag{4}$$

$$R_i = \max\left(0, \tilde{R}_o + \alpha\frac{B_w - B}{T_u}\right) \ . \tag{5}$$

For $\alpha = 0$, the differential term vanishes. Alike, for $\beta = 0$, the proportional term vanishes if we initialize $R_{\text{o,max}}$ to 0.

The Node B signals T_u/TTI_{RLC} resource grants to the RNC at the end of each update interval. The update interval has to be small enough to allow the flow control to accurately follow the channel dynamics. On the other hand, it has to be large enough to keep the signaling load between RNC and Node B at a reasonable level. Due to protocol delays, the resource grants are not used instantaneously but with a certain delay T_p, also known as *dead time*.

IV. Performance Evaluation

A. Simulation Scenario

We consider five independent terminals which perform bulk data transfer in the downlink. TCP NewReno with window scaling was used as transport protocol. Additionally, we consider three terminals, performing ping operations every ten seconds with a different amount of data. Ping user 1 transmits one packet of size 80 Bytes in the downlink direction, Ping user 2 one packet of size 1500 Bytes and Ping user 3 two packets each having a size of 1500 Bytes. The ping operations were interleaved by $3\ 1/3$ seconds.

Terminal mobility was modeled taking into account both slow and fast fading. All mobiles move at a speed of $v =$ 30km/h, corresponding to the well-known 3GPP-Scenario Vehicular 30. The mobiles periodically experience the same slow fading profile, where each mobile starts at a different position of the profile in order to obtain independent channel conditions. In the Node B, we assumed a Round Robin (RR) scheduler, which equally serves all users in a cyclic manner.

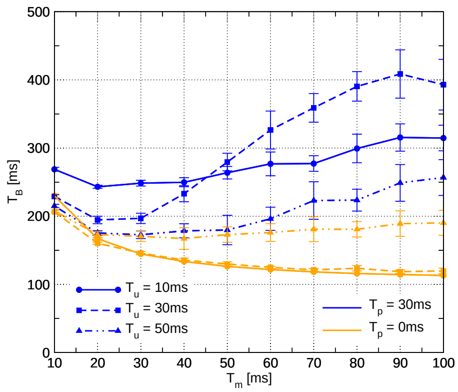

Fig. 3: Mean Node B queuing delay T_B for different dead times T_p and different update periods T_u, $T_w = 100$ms, TCP user

Throughout this paper, we chose the control loop's dead time T_p and update period T_u to be rather realistic than optimal. In particular, we chose $T_p = 30$ms and $T_u = 50$ms.

B. *Influence of measurement period*

Figure 3 plots the mean waiting time in the Node B buffer queue T_B for different dead times and different update periods over the measurement period. If the flow control operated perfectly, T_B should be equal to the target delay T_w. While the choice of the measurement period is less critical for a small dead time, it is a crucial parameter as we go to a realistic dead time of $T_p = 30$ms. In general, we can say that T_m must not be chosen too large, but should rather be chosen on the order of the update period.

C. *Influence of flow control dynamics*

Figure 4–6 plot the delay T_{dl} of an IP packet in the downlink direction for the three considered Ping users and the MobileRR Vehicular 30 scenario. T_{dl} includes all queuing delays in the UTRAN, as shown in Fig. 2. The default data rate R_{def} was set to 100kbps, which corresponds to the transfer of 125 Bytes from the RNC to the Node B within one 10ms flow control period. Since a ping transfer is initiated only every 10 seconds, the RNC–Node B data transfer always starts with a data rate of R_{def}. Consequently, the delay of an 80 Byte long packet of Ping user 1 does not depend on the remaining flow control parameters, as seen in Fig. 4. If the amount of data transmitted with each ping becomes larger, the delay increases, since more than one flow control period is necessary to transfer the data from the RNC buffer to the Node B buffer. This becomes obvious in Fig. 5 and Fig. 6 for Ping user 2 and 3, respectively.

In general, the ping delay decreases as the target delay T_w and α increase. This can directly be explained from eqs. (1) and (2), where R_i is defined to be proportional to $\alpha(R_o \cdot T_w - B)$. Consequently, a larger α and a larger T_w both lead to a quicker increase of the transfer rate R_i from its initial value R_{def} and a faster transmission of the ping data.

In contrast, the behavior with a TCP user is essentially different, since it exhibits a greedy traffic characteristic. Hence, the RNC queue usually does not run empty, and the delay T_{dl} is not an adequate metric for evaluating the performance of the flow control anymore. We will therefore look at the queue adjusted IP packet delay T_{qa}, which is measured as shown in Fig. 2. It does not take into account the RNC queuing delay, mainly determined by the traffic source behavior. Note that for an interactive Ping user the total end-to-end delay is most important, while for a bulk data transfer a short RTT between the RNC and the UE is advantageous.

Figure 7 plots T_{qa} over T_w and α. We can distinguish several regions in the graph. We will first consider relatively large values of α, i.e. $\alpha \geq 0.5$. For $T_w \leq 50$ms, we observe a very large delay, caused by the α-weighted differential term in eq. (2). This is related to the update period T_u of 50ms and the dead time T_p of 30ms. If the target waiting time T_w is smaller $T_u + T_p$, the Node B buffer may not hold enough data in order to be able to quickly react to data rate variations on the air interface. This eventually causes the Node B buffer to drain and the differential term in eq. (2) to transfer too much data from the RNC to the Node B. This becomes less of an issue as α decreases, leading to a decrease of T_{qa}.

For $T_w \geq 100$ms, the Node B buffer can hold enough data for a continuous data flow. For any particular α, we can observe a linear increase of T_{qa} proportional to T_w. Note that T_{qa} is substantially larger than T_w due to additional delay in-between the RNC and the Node B and possible retransmissions. We can also observe an increase of T_{qa} for small values of α, since the flow control is more inert. Finally, for values of T_w between 50ms and 100ms, we can observe the transition between the regions described above.

D. *Virtual Buffer Extension*

In this section, we will study the performance of the enhanced flow control algorithm according to eqs. (3)–(5). We will first study the influence of the measurement period T_m and the weighting factor β if we set α to 1, which was found to be a good value in the previous section. Longer measurement periods lead to more inaccurate measurements. On the other hand, the same applies if T_m is too short. In general, we expect a measurement period on the order of the update period T_u to be a good choice, since this would measure the data transmitted since the last resource grant was issued.

Figure 8 plots T_B (cmp. Fig. 2) over T_m and β. Ideally, this value would equal the desired target T_w of 100ms. For small β, the differential term in the flow control equations becomes dominant, and the influence of T_m is reduced. As β increases, the proportional term gains influence. We can observe an increase in the waiting time as the measurement period increases, since the flow control cannot react to data rate variations as quickly as necessary.

Additionally, Fig. 9 plots T_{dl} for Ping user 3. As only a small amount of data has to be transmitted within each ping, β only has a minor impact on the delay. On the other hand, the impact of the measurement period is much bigger. If T_m is chosen much larger than the mean transmission time of the ping data, the delay increases since the measurement period also includes those time periods, where no data was transmitted, thus reducing the measured output rate R_o.

So far, we used the previously optimized value of the differential term's weighting factor $\alpha = 1$. With β, we introduced

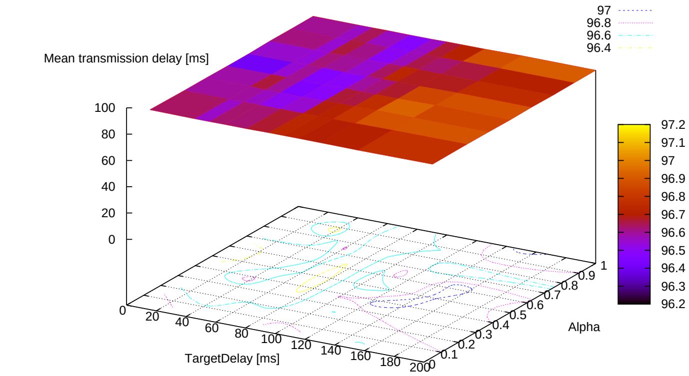

Fig. 4: Downlink delay T_{dl} for Ping User 1 in dependence of α and T_w

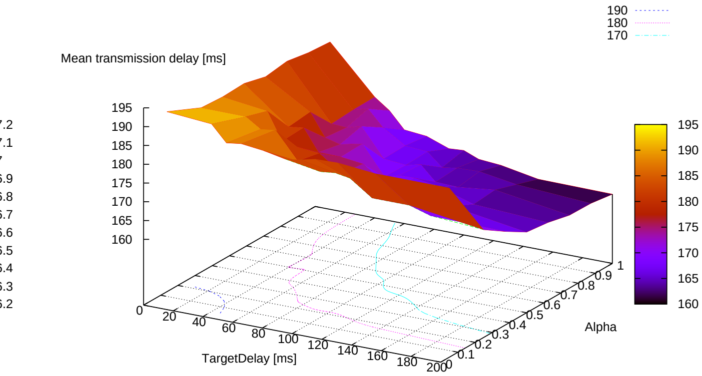

Fig. 5: Downlink delay T_{dl} for Ping User 2 in dependence of α and T_w

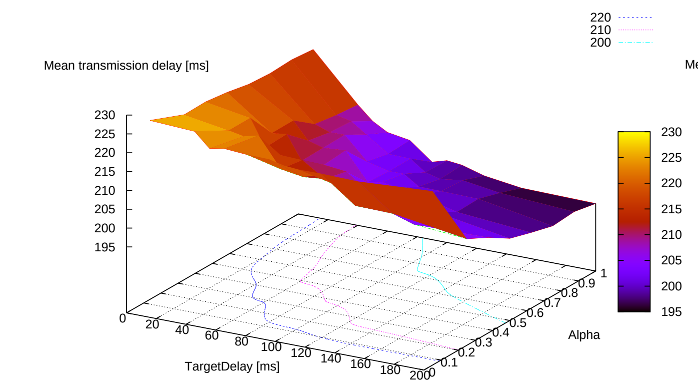

Fig. 6: Downlink delay T_{dl} for Ping User 3 in dependence of α and T_w

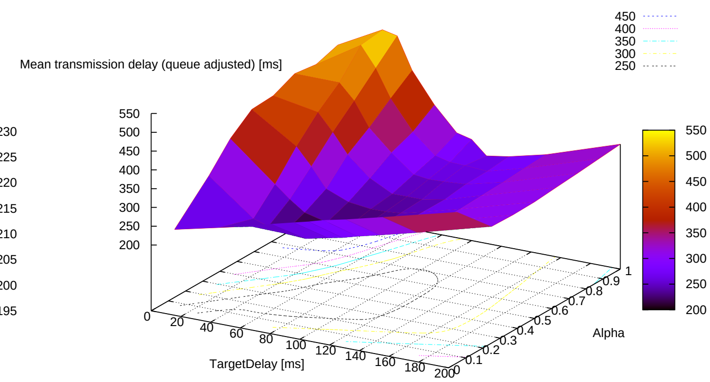

Fig. 7: Downlink queue adjusted delay T_{qa} for a TCP User in dependence of α and T_w

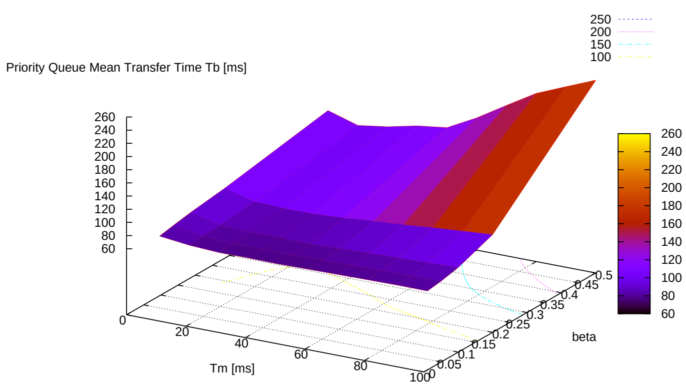

Fig. 8: Mean Node B queuing delay T_B for a TCP user, $\alpha = 1.0$, $T_w = 100$ms

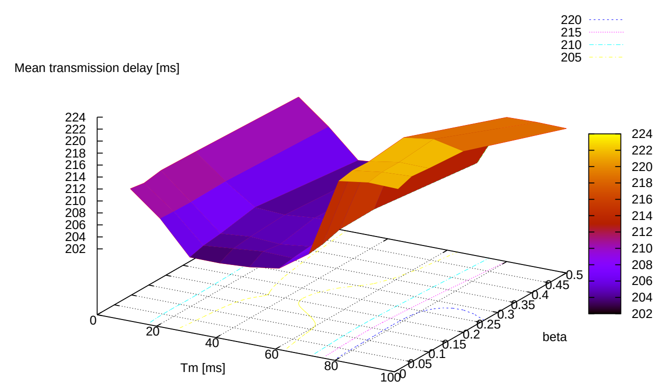

Fig. 9: Downlink delay T_{dl} for Ping User 3, $\alpha = 1.0$, $T_w = 100$ms

an additional weighting factor for the dynamics of the proportional term. It is therefore necessary to jointly consider both weighting factors. Figure 10 plots T_B in dependence of α and β. Similar to the case of the plain flow control algorithm, small values α deliver a bad performance. As we increase α beyond 0.2, we can observe a better delay performance. Also, we can see an interconnection of α and β. Note that the 95%-quantile of the queueing delay plotted in Fig. 11 goes well along with the mean delay with respect to the parameters α and β.

In order to bring light into the choice of α and β, we study the fraction of data blocks which have to wait longer than 20ms in the Node B priority queue. For the case of a greedy traffic source, and assuming a desired target delay of $T_w = 100$ms, all data blocks should be waiting longer than 20ms. Due to the dead time of $T_p = 30$ms, any data block waiting shorter than 30ms is an indication of a priority queue which is about to run empty. The flow control should avoid this if there is enough data available in the RNC, since an empty Node B queue eventually wastes air interface resources.

Figure 12 plots the fraction of RLC data blocks, which have

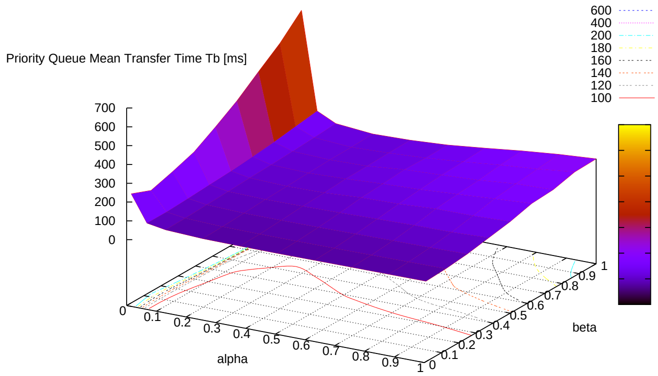

Fig. 10: Mean Node B queuing delay T_B for a TCP user, $T_m = 50$ms, $T_w = 100$ms

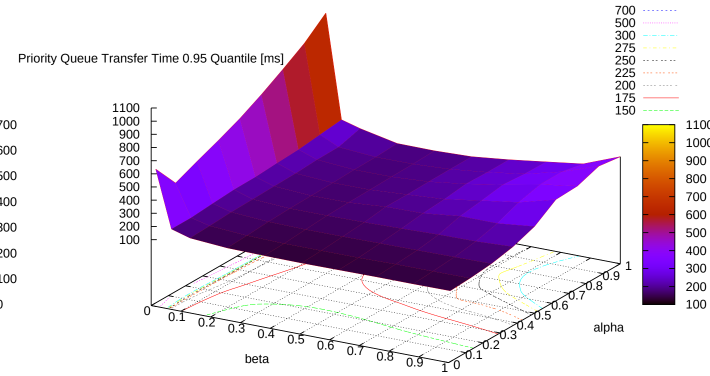

Fig. 11: 95% quantile of the Node B queueing delay T_B for a TCP user, $T_m = 50$ms, $T_w = 100$ms

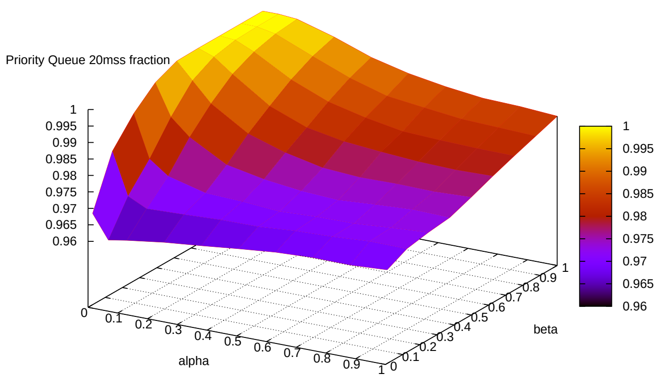

Fig. 12: Fraction of packets waiting longer than 20ms in Node B for a TCP user, $T_m = 50$ms, $T_w = 100$ms

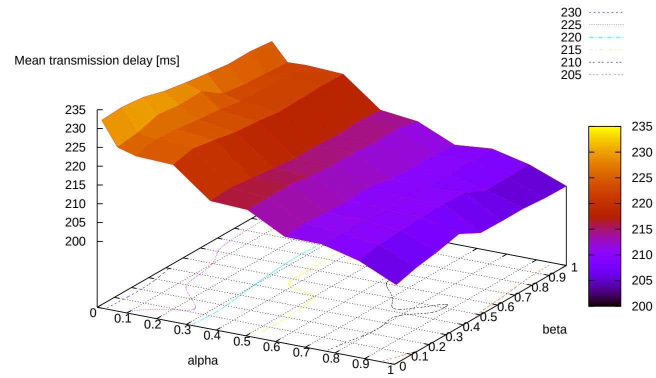

Fig. 13: Downlink delay T_{dl} for Ping user 3, $T_m = 50$ms, $T_w = 100$ms

to wait longer than 20ms in the Node B priority queue over α and β. The graph indicates that it is desirable to chose β as large as possible, while smaller values of α seem to be more interesting. On the other hand, the downlink delay of Ping user 3 in Fig. 13 again shows better performance for larger α, as we already observed in Fig. 6.

To summarize our results, there are many tradeoffs in the parameter selection. On the one hand we saw that it is desirable to emphasize the dynamic term in the flow control equations and neglect the proportional term when a good tracking of the desired target waiting time T_w is important. This gives the flow control a better chance to react to fluctuations of the data rate towards a particular user. On the other hand, emphasizing the proportional term avoids unnecessary buffer under-runs, which may eventually lead to a performance degradation.

V. Conclusion

In this paper, we studied the parameter selection of a generic flow control algorithm and its enhancements from [6]. We focused on a realistic set of parameters for the update period and the control loop dead time. In general, the parameter selection is subject to many tradeoffs, which makes it difficult to give a general recommendation on the parameter selection. In our underlying scenario, it is advantageous to put emphasis on the differential regulation relative to the Node B buffer level rather than a proportional regulation proportional to the measured output rate. For different traffic characteristics, such as for example streaming traffic, other parameter choices might be more advantageous. Further studies need to focus on cross-layer interactions with higher layer protocols and services. Eventually, the impact on the user-perceived performance is important, such as the page loading times with web services.

References

[1] R. A. Comroe and D. J. Costello, Jr., "ARQ schemes for data transmission in mobile radio systems," *IEEE Journal on Selected Areas in Communications*, vol. 2, no. 4, pp. 472–481, July 1984.

[2] P. J. A. Gutiérrez, "Packet scheduling and quality of service in HSDPA," Ph.D. dissertation, Aalborg University, Department of Communication Technology Institute of Electronic Systems, Aalborg University Niels Jernes Vej 12, DK-9220 Aalborg ?st, Denmark, October 2003.

[3] T. Kolding, "Link and system performance aspects of proportional fair scheduling in WCDMA/HSDPA," in *Proc. Vehicular Technology Conference 2003 (VTC 2003-Fall)*, vol. 3, October 2003, pp. 1717–1722.

[4] G. Aniba and S. Aissa, "Adaptive proportional fairness for packet scheduling in HSDPA," in *Global Telecommunications Conference (GLOBECOM 2004)*, vol. 6, December 2004, pp. 4033–4037.

[5] P. J. Legg, "Optimised Iub flow control for UMTS HSDPA," in *Proc. IEEE Vehicular Technology Conference (VTC 2005-Spring)*, Stockholm, Sweden, June 2005.

[6] M. C. Necker and A. Weber, "Impact of Iub flow control on HSDPA system performance," in *Proc. Personal Indoor and Mobile Radio Communications (PIMRC 2005)*, Berlin, Germany, September 2005.

[7] *IKR Simulation Library*. [Online]. Available: http://www.ikr.uni-stuttgart.de/Content/IKRSimLib/