Technical University Berlin

Telecommunication Networks Group

Fuzzy Explicit Window Adaptation: a Method to Further Enhance TCP Performance

Michael Savorić

savoric@ee.tu-berlin.de

Berlin, May 2003

TKN Technical Report TKN-03-010

TKN Technical Reports Series

Editor: Prof. Dr.-Ing. Adam Wolisz

Abstract

In the current Internet, standard TCP congestion control is based on segment loss events possibly assisted by simple explicit router feedback mechanisms like Explicit Congestion Notification (ECN). The performance of TCP congestion control can be improved if a more sophisticated Router Congestion Feedback (RCF) mechanism than ECN is used. One of such approaches is Explicit Window Adaptation (EWA). EWA uses TCP's built-in flow control mechanism to inform TCP senders about the current load in a network path. To reach this, an EWA-capable router is able to decrease the advertised receiver window in TCP acknowledgments if necessary in order to avoid congestion and packet losses. Since the EWA algorithm performed in a router has some shortcomings under certain conditions, a new EWA-related approach is developed. This approach, called fuzzy explicit window adaptation (FEWA), replaces parts of the EWA algorithm in a router with a more accurate fuzzy-based calculation.

In this technical report, FEWA is described in detail. Using two simulation scenarios with different load in the network, the performance of standard TCP is compared with the performances of standard TCP assisted by EWA and standard TCP assisted by FEWA. The simulation results show that standard TCP assisted by FEWA outperform both other approaches in throughput over loaded routers and considerably decreases the number of packet losses in congested routers.

Keywords: TCP, congestion control, flow control, distributed network information sharing, fuzzy control

Contents

Chapter 1

Introduction

In today's Internet, standard TCP connections perform congestion control triggered by segment loss events, e.g., a retransmission timer timeout or the reception of duplicated acknowledgments (dupacks). In addition, explicit feedback mechanisms from routers, e.g., Explicit Congestion Notification (ECN) [10] as a very simple one-bit-information (no congestion/congestion) mechanism, can be used to assist TCP's congestion control by informing TCP senders early enough about (impending) congestion in a router. In this context, it is interesting to investigate other explicit feedback mechanisms from routers that are able to provide TCP senders with more information about the current congestion situation in the network. One of such approaches is the Explicit Window Adaptation (EWA) [5] which is based on TCP's built-in flow control. EWA uses the advertised receiver window in TCP acknowledgments to provide TCP senders with the information which sending window can be currently supported by the network in order to avoid congestion in the network. Although EWA is able to improve the performance of TCP connections [5] it has some shortcomings in cases where a router is lowly loaded for a longer period of time. Therefore, I have adapted the basic EWA algorithm by replacing parts of it with a more appropriate algorithm based on the fuzzy control theory. This new EWA-related algorithm is called Fuzzy Explicit Window Adaptation (FEWA).

The remainder of this technical report is organized as follows. Chapter 2 briefly describes the EWA mechanism and pictures some shortcomings of parts of the EWA algorithm in a router. An improvement of the EWA algorithm, called Fuzzy Explicit Window Adaptation (FEWA), based on a fuzzy congestion controller is explained in Chapter 3. The simulation model used for the performance evaluation of standard TCP, standard TCP assisted by EWA, and standard TCP assisted by FEWA is described in Chapter 4. Chapter 5 contains the results of this performance evaluation. Finally, Chapter 6 concludes this technical report and gives and outlook to my future activities in this research area. Appendix A contains the statistical evaluation of the simulation results. In Appendix B, the linguistic rules and the parameter values of the FEWA fuzzy controller are depicted.

Chapter 2

Explicit Window Adaptation (EWA)

The Explicit Window Adaptation (EWA) approach [5] has been developed to explicitly inform the senders of TCP connections about the currently available bandwidth over a bottleneck link in a transparent way; the source codes of a TCP sender and a TCP receiver are kept unchanged. In this chapter, I describe the basic algorithm of EWA, explain some shortcomings of EWA, and depict some possible improvements of EWA.

2.1 EWA Algorithm

A bottleneck router with EWA-capabilities measures its current queue length Q_i and computes the current mean queue length $\overline{Q}_i$ for every measurement interval i. Q_i, $\overline{Q}_i$, and the last but one computed mean queue length $\overline{Q}_{i-1}$ are then used to calculate a new sending window for each TCP connection traversing this router:

$$\text{sending window} = \max\{\text{MSS}, \alpha \cdot \log_2(B - Q_i)\} \tag{2.1}$$

where B is the maximum queue length in the bottleneck router and MSS is the maximum segment size. B and Q_i are expressed in number of packets, MSS is expressed in number of bytes. The logarithmic expression in Equation (2.1) is introduced to reflect that TCP connections which are in slow start are able to send twice as much segments in their next round trip time (RTT) interval than now, i.e., the queue length can exponentially grow in the near future. In addition, Equation (2.1) ensures that every TCP connection is always allowed to send at least one TCP segment with a MSS.

The alterable factor α in Equation (2.1) is introduced to better utilize the bottleneck link if only a few TCP connections are transferring segments over the bottleneck router. α is updated every 10 milliseconds as follows:

$$\alpha = f(\alpha, \overline{Q}_i) = \begin{cases} \alpha + w_{\text{up}} & \text{if } \overline{Q}_i < \text{threshold}_{\text{low}} \\ \alpha \cdot w_{\text{down}} & \text{if } \overline{Q}_i > \text{threshold}_{\text{high}} \end{cases} \tag{2.2}$$

with

$$\overline{Q}_i = \frac{127}{128} \cdot \overline{Q}_{i-1} + \frac{1}{128} \cdot Q_i \tag{2.3}$$

The initial value for the utilization factor α is set to 1, the parameters w_{up} to additively increase and w_{down} to multiplicatively decrease α are set to $1/8$ and $31/32$, and the low and high thresholds for the mean queue length are set to 20 % and 60 % of the maximum queue length B.

The calculated sending window is transferred to each TCP sender by using the advertised receiver window in the TCP acknowledgments. In a TCP acknowledgment, the advertised receiver window is only replaced by the calculated sending window of the router if the sending window is smaller than the advertised receiver window:

$$\text{advertised receiver window} = \min\{\text{sending window}, \text{advertised receiver window}\} \tag{2.4}$$

With this explicit congestion feedback information, the TCP senders are able to react more adequately to the current load in the bottleneck router than it is possible with other mechanisms, e.g., Explicit Congestion Notification (ECN) or Random Early Detection (RED).

The main idea of EWA or related approaches [3], using existing flow control functionalities of TCP to increase the overall performance of TCP's end-to-end congestion control, is very interesting and promising. EWA and related approaches require—at least—that the segments and acknowledgments of the considered TCP connections pass through the same bottleneck router. If this necessary condition of symmetrical routing can be guaranteed for all routers in the path from a TCP sender to its TCP receiver, such an approach can be extended to an improved end-to-end congestion control assisted by a virtual flow control between the routers and the TCP senders.

2.2 Shortcomings of EWA

The basic EWA algorithm described in [5] does not consider the case that the queue of a router with EWA-capabilities is below the lower threshold for a longer period of time. In this case, the utilization factor α perpetually increases without an upper bound. If the formerly lowly loaded router is highly loaded sometimes, it takes dozens or hundreds of measurement intervals to decrease the utilization factor α to an appropriate value. Thus, the history of the load of the router is too heavily weighted in the calculation of the utilization factor α of the EWA algorithm. This drawback of the EWA algorithm can be softened but not completely prevented if an upper bound for the utilization factor α is introduced. It can be only prevented if an adapted calculation of the utilization factor α is used. One possible solution for this adapted calculation of the utilization factor α is a fuzzy-based calculation which considers only the queue statistics of the last two measurement intervals. This new calculation is described in detail in Chapter 3.

2.3 Improvements of EWA

The feedback function (2.1) of EWA is one example function to calculate the sending window in a bottleneck router. Other feedback functions are conceivable. These new feedback function should adopt the well-chosen logarithmic expression from the EWA feedback function, but can adapt or replace the calculation of the utilization factor α. The FEWA algorithm described in Chapter 3 is an example approach for the latter case.

EWA and related approaches can be easily adapted so that symmetrical routing is not longer a necessary condition for its usage. This can be done by introducing a new TCP header option that carries the calculated sending window of the routers in the path from the TCP sender to the TCP receiver. The TCP receiver puts the minimum of this sending window and its currently supported receiver window as the new advertised receiver window in a normal TCP acknowledgment and sends it to the TCP sender. For this variant of EWA the TCP protocol in the end systems have to be slightly changed.

Chapter 3

Fuzzy Explicit Window Adaptation (FEWA)

In this chapter, I describe a new EWA-related approach, called FEWA, whose queue-operating point is changed and whose calculation of the utilization factor α is based on a fuzzy controller.

3.1 FEWA Algorithm

The FEWA algorithm is similar to the EWA algorithm with one main exception: the utilization factor α (cf. Equation (2.2)) is calculated by a fuzzy controller depending on the current and the last but one measured queue length in a router:

$$\alpha = f(Q_i, Q_{i-1}) \tag{3.1}$$

This fuzzy controller is related to the fuzzy controllers used in the Fuzzy Explicit Rate Marking Adaptation (FERMA) [13] and in the more conservative FERMA-Modification (FERMAM) [12]. These approaches are Available Bit Rate (ABR) congestion controllers located in Asynchronous Transfer Mode (ATM) switches. All linguistic rules and the membership functions of FERMAM are reused for FEWA and only a few parameters of the FERMAM fuzzy controller are adapted.

The most important part of a fuzzy control system is the fuzzy logic controller (FLC) [7, 8, 9]. With the FLC the human train of thoughts can be adapted in a simple way by using linguistic variables with their set of linguistic values and a small number of linguistic rules or relational expressions. One of the linguistic rules, linguistic rule R5, in the nonlinear FEWA fuzzy congestion controller (FCC) is for example (see Appendix B.1):

If the queue (length) is short and
the rate of change is increasing slowly, (R5)
then the utilization factor should be high.

"queue (length)", *"rate of change"*, and *"utilization factor"* are called linguistic variables; *"short"*, *"increasing slowly"*, and *"high"* are examples of their valid linguistic values.

The design of a FLC is split into two parts: First, the linguistic rules are set (surface structure) and second, the membership functions of the linguistic variables are determined (deep structure), quite often by a more intuitive and pragmatic choice.

Depending on the input parameter x the membership function m_V of a linguistic variable V denotes the weights $w_{v_k} \in [0,1]$ of the valid linguistic values v_k, $1 \leq k \leq n_V$:

$$m_V(x) = (w_{v_1}, \ldots, w_{v_{n_V}}) \in [0,1]^{n_V} \tag{3.2}$$

Here, each membership function m_V is represented by a composition of n_V membership functions of the fuzzy sets F_{v_k} of its corresponding linguistic values v_k:

$$\begin{aligned} m_V(x) &= m_{v_1}(x) \otimes \ldots \otimes m_{v_{n_V}}(x) \\ &= (w_{v_1}, \ldots, w_{v_{n_V}}) \end{aligned} \tag{3.3}$$

with the tensor operator $\otimes$. Let $\oplus$ denote the s-norm operator $\max$. For an FLC with a single output variable Y with the membership function $m_Y(y) = m_{y_1}(y) \otimes \ldots \otimes m_{y_{n_Y}}(y)$ its weighted membership function

$$\begin{aligned} m_Y^*(y) &= w_{y_1} \cdot m_{y_1}(y) \oplus \ldots \oplus \\ & \quad w_{y_{n_Y}} \cdot m_{y_{n_Y}}(y) \end{aligned} \tag{3.4}$$

represents the outcome of the L related fuzzy rules $v^{(l,1)} \times \ldots \times v^{(l,n)} \rightarrow y^{(l)}$, $1 \leq l \leq L$, with n linguistic values $v^{(l,1)}, \ldots, v^{(l,n)}$ of n linguistic variables $V^{(1)}, \ldots, V^{(n)}$ and their weights $w^{(l,1)}, \ldots, w^{(l,n)}$, dependent on the input-value vector $(x_1, \ldots, x_n)$ of the FLC, i.e., for $1 \leq l \leq L$, $1 \leq j \leq n$ and exactly one k, $1 \leq k \leq n_{V^{(j)}}$, it holds:

$$\begin{aligned} v^{(l,j)} &= v_k^{(j)} \\ w^{(l,j)} &= w_{v_k^{(j)}} = m_{v_k^{(j)}}(x_j) \end{aligned} \tag{3.5}$$

If none of the linguistic values $v^{(l,j)}$ of a linguistic variable $V^{(j)}$ are used in a linguistic rule l, i.e., the linguistic rule l is independent of $V^{(j)}$, then $w^{(l,j)}$ is set to 1.

Applying the often used norms $\min$ and $\max$ the weights w_{y_k}, $1 \leq k \leq n_Y$, of Y can be expressed by:

$$w_{y_k} = \max_{y^{(l)} = y_k} \{\min\{w^{(l,1)}, \ldots, w^{(l,n)}\}\} \tag{3.6}$$

The weights of all linguistic values of the linguistic variables are used to determine the demanded fuzzy value by a weighted analysis of the linguistic rules with a fuzzy inference engine. Due to computational simplicity the membership function of a linguistic variable is often triangular or trapezoidal shaped (cf. Figures 3.1, 3.2 and Appendix B.2).

Similar to EWA, the time of the control interval of FEWA is set to 10 ms. In every control interval i FEWA measures the current queue length Q_i and its current growth rate $G_i = Q_i - Q_{i-1}$ or its current fractional growth rate $\Delta G_i = G_i/B$, respectively. The membership function of FEWA regarding the queue length is shown in Figure 3.1.

The input range of the membership function of the fractional growth rate ΔG (see Figure 3.2) has been changed to substantial lower values because of the chosen target queue length $QT = \lfloor 0.33 \cdot B \rfloor$ at each router with FEWA-capabilities in the network.

Before every control interval i the weights of all linguistic values are set to 0. Then the weights w_{Q_k}, $1 \leq k \leq 6$, for the linguistic values of the current queue length Q_i and the weights w_{G_k}, $1 \leq k \leq 5$, for the linguistic values of the current fractional queue growth rate ΔG_i are determined. These weights are used to calculate the weights w_{α_k}, $1 \leq k \leq 6$, for the linguistic values "very

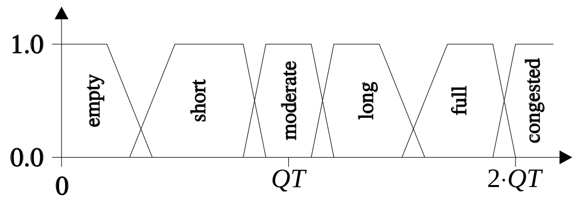

Figure 3.1: Membership function of the queue length Q of FEWA

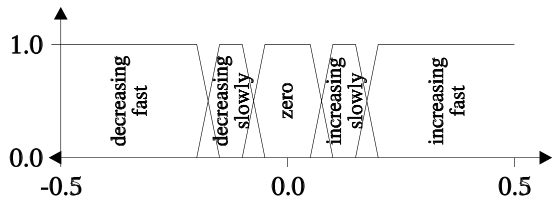

Figure 3.2: Membership function of the fractional queue growth rate ΔG of FEWA

very little", "very little", "little", "medium", "high", and "very high" of the utilization factor α (cf. Appendix B.1). For the linguistic rule (R5), for instance, the last step of this computation can be expressed by (cf. Equation (3.6)):

$$w_{\alpha_5} = \max\{w_{\alpha_5}, \min\{w_{Q_2}, w_{\Delta G_4}\}\} \tag{3.7}$$

There may be other linguistic rules which have an influence on w_{α_5}.

For FERMA, FERMAM, and for the fuzzy controller of FEWA, I choose a different formula to calculate the weights of the linguistic values of α. The linguistic rule (R5), for instance, can then be computationally expressed by:

$$w_{\alpha_5} = w_{\alpha_5} + w_{Q_2} \cdot w_{\Delta G_4} \tag{3.8}$$

There may be other linguistic rules which have an influence on w_{α_5}, too.

By the weights w_{α_k}, $1 \leq k \leq 6$, the membership function m_α of α is converted to m^*_α (cf. Equation (3.4)). The weighted linguistic values of α are combined by a special fuzzy calculation, called center of gravity (COG), to the new utilization factor α (cf. [9]):

$$\alpha = \frac{\int y \cdot m^*_\alpha(y)\, dy}{\int m^*_\alpha(y)\, dy} \tag{3.9}$$

Analog to FERMA and FERMAM, the utilization factor α is determined by a simplified calculation where only the weights of the linguistic values of α are considered. Then, Equation (3.9) can be transformed into

$$\alpha = \frac{\sum_{k=1}^{6} w_{\alpha_k} \cdot \alpha_k}{\sum_{k=1}^{6} w_{\alpha_k}} \tag{3.10}$$

for a simplified computation with $\alpha_1 = 2.00$, $\alpha_2 = 5.00$, $\alpha_3 = 8.00$, $\alpha_4 = 10.00$, $\alpha_5 = 15.00$, and $\alpha_6 = 20.00$ in the current version of the FEWA fuzzy controller. The values for the α_k, $1 \leq k \leq 6$, are chosen to set the utilization factor α to a low value of 2 ($= \alpha_1$) if the queue is congested. If the queue is empty, the utilization factor is set to a high value of 20 ($= \alpha_6$). If the queue is neither congested nor empty, the utilization factor α is set to a value between 2 and 20 according to the linguistic rules of the FEWA FLC (cf. Appendix B.1).

Compared to the FLC of FERMAM, only these a_i's are adapted for the FLC of FEWA. Similar to FERMAM, these values are selected to achieve a moderate queue length Q in the neighborhood of the target queue length QT. But due to the more complex congestion control in IP-based networks compared to ATM networks and the less available information in IP routers compared to ATM switches, e.g., the number of TCP flows and their current congestion control mode (slow start, congestion avoidance) are unknown, the target queue length in IP routers cannot be met as well as in ATM switches (cf. [13, 12]).

Figure 3.3 shows the control surface of the current FEWA fuzzy controller for the utilization factor α and the control surface of the current FEWA algorithm for the sending window with $B = 99$ packets and $QT = 32$ packets, respectively.

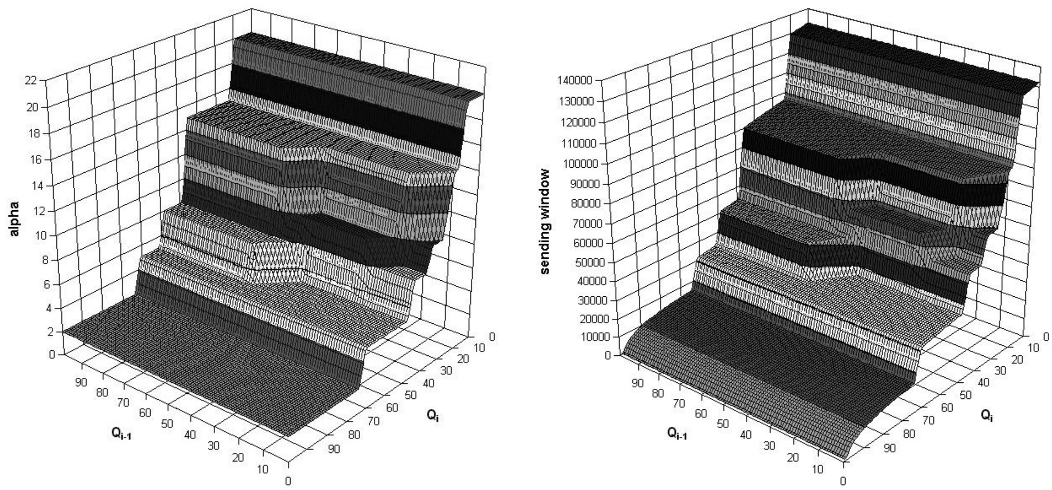

Figure 3.3: Control surfaces of the FEWA fuzzy controller for the utilization factor α and of the FEWA algorithm for the sending window

The fuzzy controller of FEWA is developed in such a manner that the current queue length Q_i has the main influence on the calculation of the utilization factor α or the sending window, respectively; the current (fractional) growth rate $(\Delta)G_i$ is used to refine this calculation. For the FEWA algorithm the described membership functions of the linguistic variables (see Appendix B.2) and the values of the utilization factor are the result of a more intuitive and pragmatic choice and not of an analytic approach. Nevertheless, this selection provides promising results.

It should be noted that due to the differences in the congestion feedback functions of FERMAM and FEWA, in the FEWA fuzzy controller the values for the parameters α_i can not be independently chosen from the maximum queue length B in a router. Therefore, these fuzzy controller parameters have to be carefully determined for each maximum queue length B in FEWA-capable routers in a network.

3.2 Practicability of FEWA

In a real IP router there is no need for a complex and computational intensive fuzzy inference engine in the FLC. After the linguistic rules are found and the linguistic values are tuned by a simulator the control surface is known and can be stored as a lookup table for selected sampling points requiring only a few kilobytes of read-only memory (ROM). In combination with a simple interpolation algorithm FEWA can be implemented in such a way with a very fast response time.

Chapter 4

Simulation Model

In this chapter, the basic simulation model including the considered network topology with the location of EWA/FEWA-capable routers and the used traffic model are described.

4.1 Network Topology

The network topology of the prototype simulation model is shown in Figure 4.1. This hierarchical structured network topology consists of a public Internet part, one core network (CN), two radio access networks (RANs), and six radio cells (RCs) with a (W)LAN access technology.

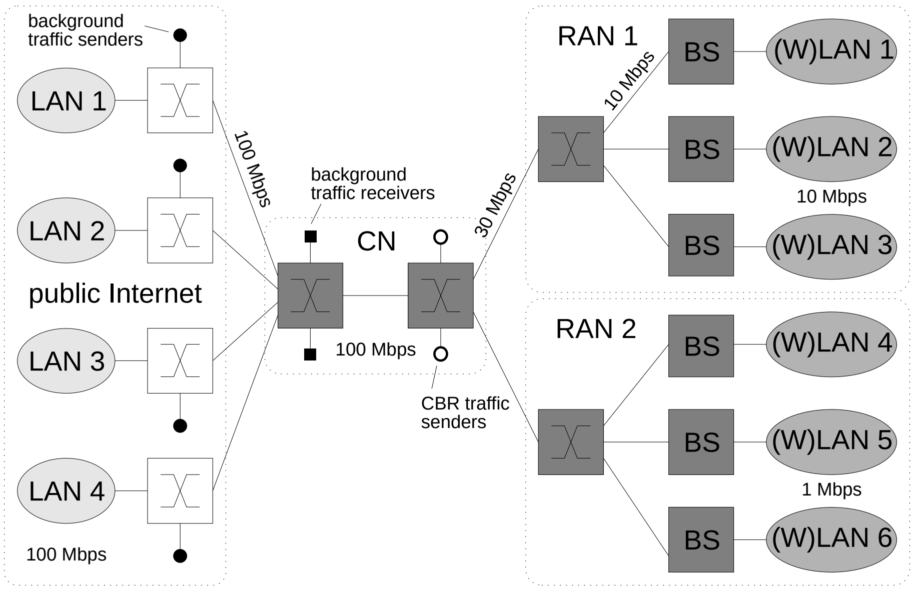

Figure 4.1: Simulated network topology

The public Internet part of the simulation model consists of four LANs and four routers. Each LAN has a bandwidth of 100 Mbps and a delay of $5 \cdot 10^{-7}$ seconds. All links in the public Internet have a bandwidth of 100 Mbps. The delay between the routers in the public Internet and the ingress router of the core network is set to 50 ms. The link between the ingress router and the egress router of the core network has a bandwidth of 100 Mbps and a delay of 5 ms. The ingress routers of the RANs are connected to the egress router of the core network with links with a bandwidth of 30 Mbps and a delay of 5 ms. The base stations ((W)LAN routers) are connected to the ingress RAN router by links with a bandwidth of 10 Mbps and a delay of 5 ms. Each (W)LAN in RAN 1 has a bandwidth of 10 Mbps while each (W)LAN in RAN 2 has a bandwidth of 1 Mbps. The delays in the (W)LANs are $5 \cdot 10^{-7}$ seconds. The packet error rate p in the (W)LANs is adjustable to investigate standard TCP, EWA, FEWA, and other approaches, e.g., Freeze-TCP [2], in different network scenarios. In addition, also the influence of moving TCP receivers with adjustable handover-rates and handover-latencies on the performance of these approaches can be investigated. But in this technical report, the performance of standard TCP and standard TCP assisted by EWA or assisted by FEWA is investigated in a network scenario with reliable links in the (W)LANs, immobile TCP receivers, and only the mean network load is varied.

All router queues in the simulation scenario are droptail queues. These router queues have a maximum queue length limit of 100, i.e., every router queue can store at most $B = 99$ IP packets. The target queue length of FEWA is set to $QT = 32$.

In each of the LANs in the public Internet, n TCP senders are located. The dedicated TCP receivers are distributed over the (W)LANs in the two radio access networks (RANs). 16 background TCP senders are connected to the routers in the public Internet, 4 to each of them. The dedicated background TCP receivers are connected to the ingress router of the core network. 12 Background UDP senders are connected to the egress router of the core network, while the dedicated UDP receivers are uniformly distributed over the (W)LANs of the RANs.

4.2 Location of EWA/FEWA-Capable Routers

Both routers in the core network and the routers in the radio access networks are equipped with the EWA/FEWA approach. These additional router functionality can be optionally switched on and off to simulate end-to-end Internet congestion control based on standard TCP or Internet congestion control based on standard TCP and assisted by EWA or assisted by FEWA.

4.3 Traffic Model

Properly characterizing traffic loads for interactive Internet users is a difficult undertaking. In order to represent some mixture of different applications, I decided to use one TCP-based application class to generate WWW traffic and one UDP-based application class to generate voice traffic.

The first application class includes TCP connections whose number of segments to send is determined by a WWW traffic model [11]. This traffic model is derived from real HTTP traces in corporate and educational environments and uses three abstraction levels: The session level, the page level, and the packet level. Here, a simplified version of this WWW traffic model is used which consists only of the first two levels. In every WWW session a log-normally distributed number of WWW pages with Pareto-distributed page sizes are sent. In addition, the page sizes are limited by an upper bound of 1 MB. The time between the pages, i.e., the inter-connection or reading time, is gamma distributed. The load in the network can be easily adjusted by using the exponentially distributed session inter-arrival time with a different parameter. The distributions and parameters chosen for the stochastic variables of the simplified WWW traffic model are shown in Table 4.1.

Table 4.1: Distributions and parameters for the stochastic variables of the simplified WWW model

Stochastic variable	Distribution	Distribution parameter(s)
Inter-session time	Exponential	$\mu = 5.0$ s
Pages per session (pps)	Lognormal	$\mu = 25.807$ pps $\sigma = 78.752$ pps
Inter-page time (reading time)	Gamma	$\mu = 35.286$ s $\sigma = 147.390$ s
Page size	Pareto (limited)	$\alpha = 1.7584$ $\beta = 30458$ Bytes

This traffic model is used for all TCP connections whose receivers are located in the (W)LANs of the RANs. The background traffic TCP connections in the public Internet part of the simulation model use the same distribution for their pages per session and their page size as shown in Table 4.1. But the inter-session time and the inter-page time are deterministically set to zero.

The second application class includes UDP data streams which send every 20 ms an UDP packet with a payload length of 36 bytes. This class is used to model traffic with a constant bit rate (CBR) like voice-over-IP.

The whole simulation model is implemented in ns-2 (version 2.1b9) [1]. It is part of a larger framework in which different congestion feedback mechanisms can be investigated in different network scenarios.

Chapter 5

Performance Evaluation

In this chapter, first the metrics used for the performance evaluation of standard TCP, standard TCP assisted by EWA, and standard TCP assisted by FEWA, are defined and explained. After that, the simulation results of this performance evaluation are shown and summarized.

5.1 Performance Evaluation Metric

To compare the simulation results of the pure end-to-end congestion control approach with the simulation results of the two explicit router congestion feedback (RCF) approaches, two different overall mean throughput computations for TCP connections are used. If n TCP connections are considered and each of these TCP connections has sent s_i segments in duration d_i, $1 \leq i \leq n$, then the two overall mean throughput calculations work as follows:

- Computation of the overall mean throughput ($\overline{T}_1$): The sum of sent segments of all n TCP connections is divided by the overall duration of these TCP connections:

$$\overline{T}_1 = \frac{\sum_{i=1}^{n} s_i}{\sum_{i=1}^{n} d_i}$$

- Computation of the connection-oriented mean throughput ($\overline{T}_2$): For each of the n TCP connections a mean throughput t_i, $1 \leq i \leq n$, is calculated. All these mean throughput values are then used to compute the overall mean throughput of the TCP connections by a normal non-weighted arithmetic mean calculation independent of the number of segments sent by each of the TCP connections:

$$\overline{T}_2 = \frac{1}{n} \cdot \sum_{i=1}^{n} \frac{s_i}{d_i} = \frac{1}{n} \cdot \sum_{i=1}^{n} t_i$$

The former throughput calculation is the more important one, since with this throughput metric the overall throughput of the considered TCP connections can be evaluated. The latter calculation gives a connection-oriented mean throughput which can be understood as the mean throughput a single of all considered TCP connections can expect.

I use these two throughput metrics to calculate the overall and connection-oriented mean throughputs separately for each (W)LAN by considering all TCP connections with receivers in each of the (W)LANs. In addition, I count the number of packet losses in the base stations of the (W)LANs as another performance evaluation metric.

5.2 Performance Evaluation Results

In the following, two simulation scenarios are considered which differ in their mean load in the network: the first simulation scenario uses $n = 24$ TCP senders in each LAN of the public Internet to generate WWW traffic in the network, the second one doubles this value. In each scenario the performance metrics of standard TCP is compared with the performance metrics of standard TCP assisted by EWA or standard TCP assisted by FEWA. In addition, for each simulation scenario the queue length process and the queue length histogram of each downstream queue in the base stations in RAN 2 are shown.

Currently, ten simulation runs per scenario case are performed. Every run for each simulation scenario with either a lower or a higher mean network load considers a simulated time of 18000 seconds.

The last column Δ in the following Tables 5.1 and 5.2 denotes whether the simulation results are statistically significantly different or not. A '+' or '-' denotes that standard TCP assisted by EWA/FEWA or standard TCP is significantly better. A '=' means that with the simulation results no significant difference between standard TCP assisted by EWA/FEWA and standard TCP can be concluded. The details of the statistical evaluation of the simulation results can be found in Appendix A.

5.2.1 Scenario 1: TCP Newreno, Lower Mean Network Load

In this scenario with a lower mean network load the base stations in RAN 1 are not congested at all, i.e., no packets are lost, and the base stations in RAN 2 are rarely congested. The throughputs of TCP connections traversing the base stations in RAN 1 are comparable for standard TCP and standard TCP assisted by EWA, standard TCP assisted by FEWA is slightly worse. For the throughputs of TCP connections traversing the base stations in RAN 2 standard TCP assisted by FEWA outperform standard TCP as well as standard TCP assisted by EWA in most of the throughput metrics. For these TCP connections the throughput gain of standard TCP assisted by FEWA is approximately 4 % compared to both standard TCP and standard TCP assisted by EWA.

Compared to standard TCP, the EWA algorithm considerably decreases the number of packet losses in congested routers by 47 %. This amount of packet losses can be further decreased by 94 % if the EWA algorithm is replaced by the FEWA algorithm in these routers.

5.2.2 Scenario 2: TCP Newreno, Higher Mean Network Load

In this scenario with a higher mean network load the base stations in RAN 1 are not congested at all, i.e., no packets are lost, and the base stations in RAN 2 are more often congested than in the previous scenario. For standard TCP connections traversing the uncongested base stations in RAN 1 the throughput is comparable to the throughputs of TCP connections assisted by FEWA. The throughput of TCP connections assisted by EWA is slightly worse. For TCP connections traversing the base stations in RAN 2 standard TCP assisted by FEWA outperform both other approaches. For

Table 5.1: Simulation results of scenario 1 — $\overline{T}_1$ is the overall mean throughput, $\overline{T}_2$ is the connection-oriented mean throughput, Δ denotes the statistical significance of the simulation results

		standard TCP	EWA	FEWA	Δ
$\overline{T}_1$ of TCP connections [segments/second]	(W)LAN 1	63.48	63.38	63.27	==
	(W)LAN 2	63.48	63.25	62.98	=−
	(W)LAN 3	63.26	63.61	63.51	==
	(W)LAN 4	47.33	47.44	49.18	=+
	(W)LAN 5	47.03	47.60	49.23	++
	(W)LAN 6	47.08	47.33	49.19	++
$\overline{T}_2$ of TCP connections [segments/second]	(W)LAN 1	58.56	58.55	58.50	==
	(W)LAN 2	58.50	58.45	58.31	=−
	(W)LAN 3	58.40	58.58	58.57	+=
	(W)LAN 4	46.02	45.83	46.30	−+
	(W)LAN 5	45.82	45.92	46.23	=+
	(W)LAN 6	45.89	45.78	46.21	=+
mean number of packet losses in base station ((W)LAN router)	(W)LAN 1	0.00	0.00	0.00	==
	(W)LAN 2	0.00	0.00	0.00	==
	(W)LAN 3	0.00	0.00	0.00	==
	(W)LAN 4	5865.50	4161.90	256.10	++
	(W)LAN 5	6160.80	3993.50	242.10	++
	(W)LAN 6	6211.10	4275.30	302.30	++

these TCP connections the throughput gain of standard TCP assisted by FEWA is approximately 9 % compared to standard TCP and approximately 8 % compared to standard TCP assisted by EWA.

Compared to standard TCP, the EWA algorithm considerably decreases the number of packet losses in congested routers by 60 %. This amount of packet losses can be further decreased by more than 86 % if the EWA algorithm is replaced by the FEWA algorithm in these routers.

5.2.3 Queue Lengths in Base Stations of RAN 2

Figure 5.1 shows the queue length process in the downstream queue of base station 4, one of the base stations in RAN 2, for both simulation scenarios and for a measurement interval of 60 s. In the scenario with a lower mean network load and compared to standard TCP, standard TCP assisted by EWA is able to slightly decrease the maximum queue length in these base stations. But in the scenario with a higher mean network load, standard TCP and standard TCP assisted by EWA are both not able to prevent packet losses due to congestion. For both simulation scenarios, standard TCP assisted by FEWA reaches considerably lower maximum queue lengths in this measurement interval.

Figure 5.2 shows the histograms of the queue length in the downstream queues of the base stations

Table 5.2: Simulation results of scenario 2 — $\overline{T}_1$ is the overall mean throughput, $\overline{T}_2$ is the connection-oriented mean throughput, Δ denotes the statistical significance of the simulation results

		standard TCP	EWA	FEWA	Δ
$\overline{T}_1$ of TCP connections [segments/second]	(W)LAN 1	54.93	54.63	54.64	$-=$
	(W)LAN 2	54.82	54.63	54.80	$==$
	(W)LAN 3	54.77	54.77	54.63	$==$
	(W)LAN 4	33.90	34.11	36.75	$=+$
	(W)LAN 5	33.47	34.34	36.71	$++$
	(W)LAN 6	33.43	34.09	36.82	$++$
$\overline{T}_2$ of TCP connections [segments/second]	(W)LAN 1	54.24	54.21	54.19	$==$
	(W)LAN 2	54.29	54.22	54.17	$=-$
	(W)LAN 3	54.25	54.25	54.20	$==$
	(W)LAN 4	37.68	37.40	38.20	$-+$
	(W)LAN 5	37.43	37.50	38.22	$=+$
	(W)LAN 6	37.45	37.43	38.21	$=+$
mean number of packet losses in base station ((W)LAN router)	(W)LAN 1	0.00	0.00	0.00	$==$
	(W)LAN 2	0.00	0.00	0.00	$==$
	(W)LAN 3	0.00	0.00	0.00	$==$
	(W)LAN 4	30681.00	19212.90	2628.60	$++$
	(W)LAN 5	32242.10	18759.30	2695.90	$++$
	(W)LAN 6	32066.50	18907.80	2608.80	$++$

4–6, the base stations in RAN 2, for both simulation scenarios. For standard TCP, these queues have a mean queue length of 8.30 in the scenario with a lower mean network load and 20.23 in the scenario with a higher mean network load. If standard TCP is assisted by EWA, these values can be reduced to 7.80 and 18.62. Standard TCP assisted by FEWA reaches the lowest mean queue lengths of 6.68 and 17.60, respectively.

In the simulated network topology and with the given traffic model the base stations in the RAN 2 are lowly to moderately loaded most of the simulated time. Therefore, the utilization factor α in the EWA algorithm is increased to such high values in periods of time with a lower load in the base stations that it can not be decreased as fast as necessary in periods of time with highly loaded or even congested base stations. Hence, the EWA algorithm is not able to adequately control the sending window of the TCP senders to avoid congestion and packet losses in the base stations. Only the FEWA algorithm is able to early react on impending congestion in the base stations and adequately limit the sending window of TCP senders that only a few packets are lost due do congestion in the base stations.

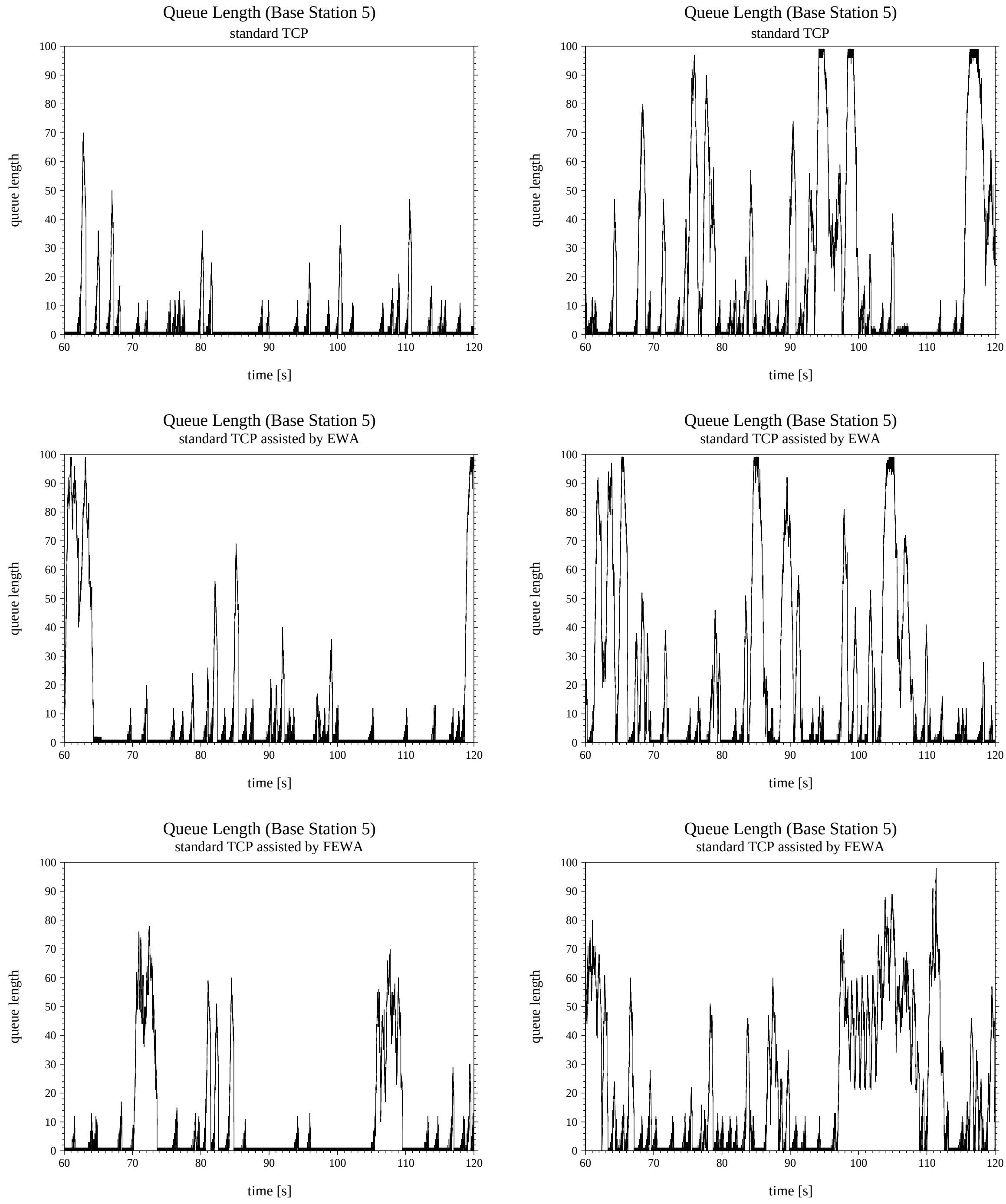

Figure 5.1: Queue length process in the downstream queue of base station 4 in RAN 2 for standard TCP, standard TCP assisted by EWA, and standard TCP assisted by FEWA (lower mean network load on the left side, higher mean network load on the right side)

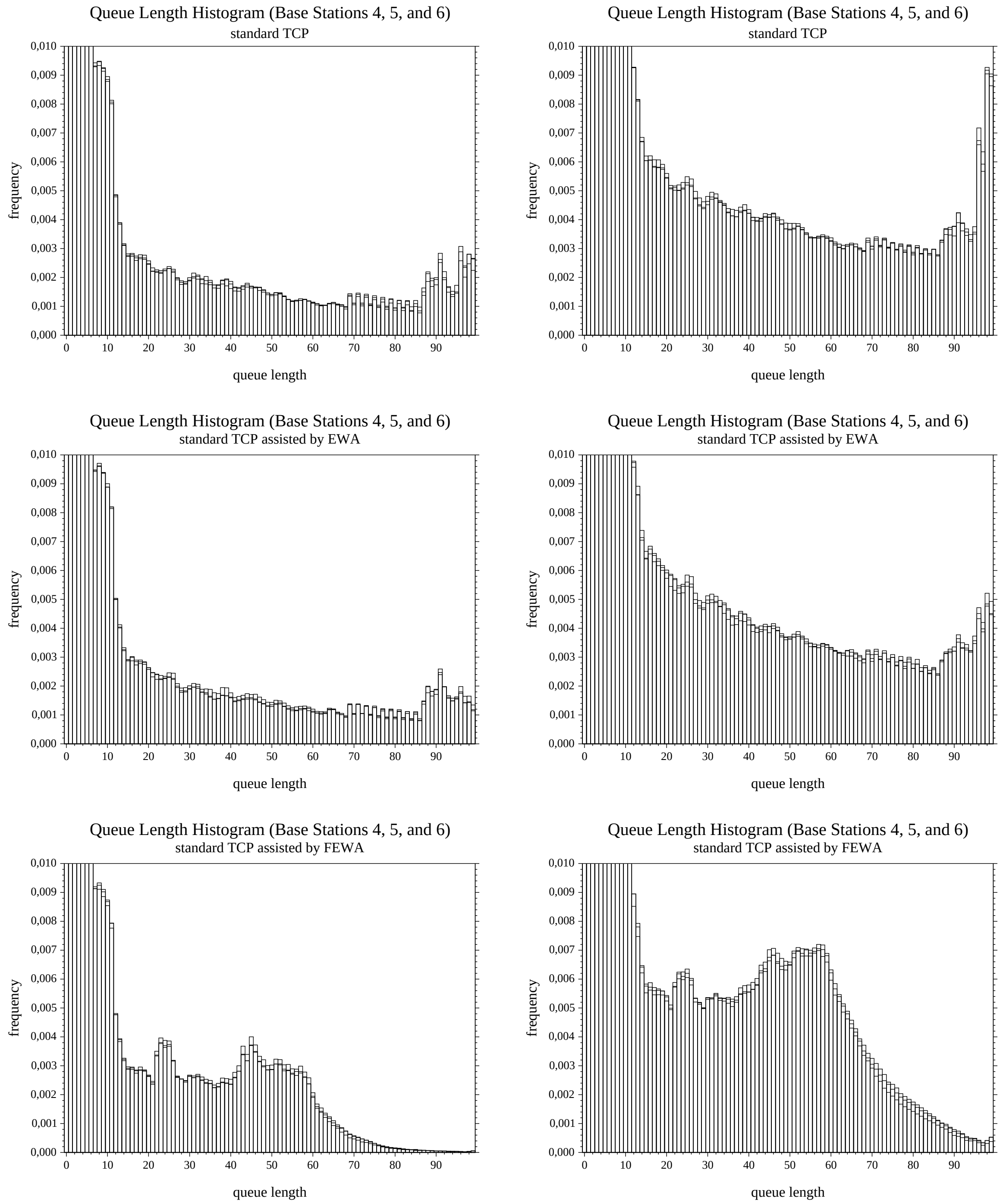

Figure 5.2: Histograms of the queue length in the downstream queues of the base stations in RAN 2 for standard TCP, standard TCP assisted by EWA, and standard TCP assisted by FEWA (lower mean network load on the left side, higher mean network load on the right side)

Chapter 6

Conclusion and Outlook

In this technical report, I have investigated the performance of standard TCP and standard TCP either assisted by EWA or assisted by FEWA. Using two simulation scenarios with different network loads, I achieve the following results: If routers in the network are lowly to moderately loaded and congestion is likely to happen, the throughput of TCP connections traversing these routers is increased if TCP's end-to-end congestion control is assisted by an explicit congestion feedback from routers. In this case, FEWA is able to even outperform EWA with a throughput increase of 4–8 % and a large decrease in the number of packet losses in congested routers by up to 94 %. If routers in the network are not congested at all, the throughput of TCP connections traversing these routers is not decreased or only slightly decreased compared to standard TCP if TCP's end-to-end congestion control is assisted by EWA or assisted by FEWA. Thus, EWA and FEWA only reduce the sending window of TCP senders if it is necessary in order to avoid packet losses due to (impending) congestion in routers.

Considering these results, the implementation of FEWA in routers to assist TCP's end-to-end congestion control is highly recommended. FEWA reaches considerable performance gains with a low additional computational complexity in routers (cf. Section 3.2). Therefore, FEWA is another example that fuzzy-based congestion control algorithms are able to more adequately react on changing load in routers than approaches based on other control-theoretic assumptions, e.g., time series.

Currently, I consider the performance of FEWA for different wireless access technologies, e.g., WLAN and UMTS, and for fixed and mobile TCP receivers. In the case of mobile TCP receivers, also a combined approach of FEWA and Freeze-TCP is investigated. After that, I will consider the following new approach for TCP congestion control: If all routers—or at least the bottleneck routers—in a network are equipped with FEWA, the overall performance of TCP connections can be significantly increased if the congestion control algorithm in a standard TCP sender is slightly changed by adapting the semantic of the advertised receiver window in a TCP acknowledgment: The current advertised receiver window in a TCP acknowledgment is not just an upper bound for the sending window of a TCP sender, it is the new sending window. In this case, an additional pacing mechanism is needed to avoid a bursty sending behavior of this adapted TCP sender. I call this new TCP congestion control Enhanced TCP (ETCP). Since FEWA is developed to work with standard TCP senders, the use of ETCP senders might have an influence on the load of FEWA-capable routers. Thus, it might be possible that the FEWA control algorithm has to be adapted. For example, Equation (2.1) can be replaced by a more accurate formula that reflects that all TCP senders perform the ETCP congestion control; TCP's slow start and congestion avoidance mechanism are not used any longer. Then, also an adaptation of the α-calculation in the FEWA algorithm might be necessary.

In addition, it might be interesting to investigate how the fuzzy controller of the FEWA algorithm can be combined with other existing approaches, e.g., RED or Explicit Control Protocol (XCP) [6]. In combination with RED a modified FEWA fuzzy controller can then be used to determine the packet loss probability. And the Efficiency Controller (EC) of XCP can be replaced by a modified FEWA fuzzy controller to reach a higher performance of aggregated traffic over a router with XCP-capabilities.

Appendix A

Appendix: Statistical Evaluation

A.1 Statistical Evaluation Method

For the simulated scenarios with a lower or a higher mean network load the results of standard TCP are statistically compared with the results of standard TCP assisted by EWA or standard TCP assisted by FEWA.

The statistical evaluation method used for this comparison is called the t-test for unpaired observations of two alternatives and is described in detail in [4]. The main idea of this method is to compute a confidence interval for the difference of the mean values of both alternatives for a given confidence level. Then the decision criterion is:

- If the confidence interval includes zero, then the results of the two alternatives can not be distinguished.
- If the confidence interval is above/below zero, then the first/second alternative reaches significant higher values.

Tests with confidence intervals give not only a yes-no answer like other hypothesis tests, they also give an answer to the question how precise the decision is. A narrow confidence interval indicates that the precision of the decision is high whereas a wide confidence interval indicates that the precision of the decision is rather low.

This t-test for unpaired observations of two alternatives is used for the statistical evaluation of the simulation results for both the overall mean throughput ($\overline{T}_1$) and the connection-oriented mean throughput ($\overline{T}_2$) for TCP connections with receivers in each of the (W)LANs and the mean number of packet losses in the base station of each (W)LAN.

The values for this statistical evaluation are produced by ten independent simulation runs of the considered simulation scenario.

A.2 Statistical Evaluation Results

In the following Tables A.1 and A.2 the statistical evaluation of the simulation results are shown. For each confidence interval also the confidence level (0.90, 0.95 or 0.99) is depicted. If the simulation results for standard TCP, standard TCP assisted by EWA, or standard TCP assisted by FEWA are

Table A.1: Statistical evaluation of the simulation results of scenario 1

		standard TCP ↔ EWA	standard TCP ↔ FEWA
$\overline{T}_1$ of TCP connections [1 − α : conf()]	(W)LAN 1	$0.90 : (-\ 0.20, +\ 0.39)$	$0.90 : (-\ 0.05, +\ 0.45)$
	(W)LAN 2	$0.90 : (-\ 0.29, +\ 0.75)$	$0.90 : (+\ 0.06, +\ 0.93)$
	(W)LAN 3	$0.90 : (-\ 0.75, +\ 0.05)$	$0.90 : (-\ 0.66, +\ 0.15)$
	(W)LAN 4	$0.90 : (-\ 0.33, +\ 0.10)$	$0.99 : (-\ 2.18, -\ 1.53)$
	(W)LAN 5	$0.99 : (-\ 0.86, -\ 0.28)$	$0.99 : (-\ 2.53, -\ 1.87)$
	(W)LAN 6	$0.90 : (-\ 0.47, -\ 0.03)$	$0.99 : (-\ 2.53, -\ 1.71)$
$\overline{T}_2$ of TCP connections [1 − α : conf()]	(W)LAN 1	$0.90 : (-\ 0.14, +\ 0.17)$	$0.90 : (-\ 0.07, +\ 0.19)$
	(W)LAN 2	$0.90 : (-\ 0.18, +\ 0.27)$	$0.90 : (+\ 0.03, +\ 0.36)$
	(W)LAN 3	$0.90 : (-\ 0.33, -\ 0.03)$	$0.90 : (-\ 0.35, +\ 0.01)$
	(W)LAN 4	$0.95 : (+\ 0.07, +\ 0.31)$	$0.99 : (-\ 0.49, -\ 0.08)$
	(W)LAN 5	$0.90 : (-\ 0.20, +\ 0.01)$	$0.99 : (-\ 0.59, -\ 0.23)$
	(W)LAN 6	$0.90 : (-\ 0.03, +\ 0.24)$	$0.99 : (-\ 0.52, -\ 0.12)$
mean number of packet losses in base station [1 − α : conf()]	(W)LAN 1	$0.90 : (-\ 0.00, +\ 0.00)$	$0.90 : (-\ 0.00, +\ 0.00)$
	(W)LAN 2	$0.90 : (-\ 0.00, +\ 0.00)$	$0.90 : (-\ 0.00, +\ 0.00)$
	(W)LAN 3	$0.90 : (-\ 0.00, +\ 0.00)$	$0.90 : (-\ 0.00, +\ 0.00)$
	(W)LAN 4	$0.99 : (+\ 1109.80, +\ 2297.40)$	$0.99 : (+\ 5040.42, +\ 6178.40)$
	(W)LAN 5	$0.99 : (+\ 1872.27, +\ 2462.33)$	$0.99 : (+\ 5630.94, +\ 6206.46)$
	(W)LAN 6	$0.99 : (+\ 1328.33, +\ 2543.27)$	$0.99 : (+\ 5332.74, +\ 6484.87)$

not significantly different even for the confidence level 0.90, then the confidence interval for the confidence level 0.90 is stated.

Table A.2: Statistical evaluation of the simulation results of scenario 2

		standard TCP ↔ EWA	standard TCP ↔ FEWA
$\overline{T}_1$ of TCP connections $[1-\alpha$: conf()]	(W)LAN 1	$0.90 : (+\ 0.02, +\ 0.58)$	$0.90 : (-\ 0.02, +\ 0.59)$
	(W)LAN 2	$0.90 : (-\ 0.02, +\ 0.40)$	$0.90 : (-\ 0.22, +\ 0.26)$
	(W)LAN 3	$0.90 : (-\ 0.22, +\ 0.22)$	$0.90 : (-\ 0.08, +\ 0.36)$
	(W)LAN 4	$0.90 : (-\ 0.50, +\ 0.08)$	$0.99 : (-\ 3.34, -\ 2.36)$
	(W)LAN 5	$0.99 : (-\ 1.36, -\ 0.37)$	$0.99 : (-\ 3.66, -\ 2.81)$
	(W)LAN 6	$0.99 : (-\ 1.10, -\ 0.22)$	$0.99 : (-\ 3.86, -\ 2.92)$
$\overline{T}_2$ of TCP connections $[1-\alpha$: conf()]	(W)LAN 1	$0.90 : (-\ 0.07, +\ 0.14)$	$0.90 : (-\ 0.06, +\ 0.16)$
	(W)LAN 2	$0.90 : (-\ 0.03, +\ 0.16)$	$0.90 : (+\ 0.05, +\ 0.14)$
	(W)LAN 3	$0.90 : (-\ 0.11, +\ 0.09)$	$0.90 : (-\ 0.02, +\ 0.22)$
	(W)LAN 4	$0.95 : (+\ 0.05, +\ 0.51)$	$0.99 : (-\ 0.77, -\ 0.27)$
	(W)LAN 5	$0.90 : (-\ 0.24, +\ 0.09)$	$0.99 : (-\ 1.03, -\ 0.56)$
	(W)LAN 6	$0.90 : (-\ 0.13, +\ 0.18)$	$0.99 : (-\ 1.00, -\ 0.52)$
mean number of packet losses in base station $[1-\alpha$: conf()]	(W)LAN 1	$0.90 : (-\ 0.00, +\ 0.00)$	$0.90 : (-\ 0.00, +\ 0.00)$
	(W)LAN 2	$0.90 : (-\ 0.00, +\ 0.00)$	$0.90 : (-\ 0.00, +\ 0.00)$
	(W)LAN 3	$0.90 : (-\ 0.00, +\ 0.00)$	$0.90 : (-\ 0.00, +\ 0.00)$
	(W)LAN 4	$0.99 : (+10263.57, +12672.63)$	$0.99 : (+26936.75, +29168.05)$
	(W)LAN 5	$0.99 : (+12304.24, +14661.37)$	$0.99 : (+28421.23, +30671.17)$
	(W)LAN 6	$0.99 : (+11431.56, +14885.85)$	$0.99 : (+27769.39, +31146.01)$

Appendix B

FEWA Fuzzy Controller

In this chapter, I present the linguistic rules of the FEWA algorithm. Furthermore, I state the chosen parameters of the membership functions $m_{\Delta Q}$ and $m_{\Delta G}$ of the linguistic variables ΔQ and ΔG.

B.1 Linguistic Rules of FEWA

If the queue (length) is empty,
then the utilization factor should be very high. (R1)

If the queue (length) is short and
the rate of change is decreasing fast, (R2)
then the utilization factor should be high.

If the queue (length) is short and
the rate of change is decreasing slowly, (R3)
then the utilization factor should be high.

If the queue (length) is short and
the rate of change is zero, (R4)
then the utilization factor should be high.

If the queue (length) is short and
the rate of change is increasing slowly, (R5)
then the utilization factor should be high.

If the queue (length) is short and
the rate of change is increasing fast, (R6)
then the utilization factor should be medium.

If the queue (length) is moderate and
the rate of change is decreasing fast, (R7)
then the utilization factor should be high.

If the queue (length) is moderate and
the rate of change is decreasing slowly, (R8)
then the utilization factor should be medium.

If the queue (length) is moderate and
the rate of change is zero, (R9)
then the utilization factor should be medium.

If the queue (length) is moderate and
the rate of change is increasing slowly, (R10)
then the utilization factor should be medium.

If the queue (length) is moderate and
the rate of change is increasing fast, (R11)
then the utilization factor should be little.

If the queue (length) is long and
the rate of change is decreasing fast, (R12)
then the utilization factor should be little.

If the queue (length) is long and
the rate of change is decreasing slowly, (R13)
then the utilization factor should be little.

If the queue (length) is long and
the rate of change is zero, (R14)
then the utilization factor should be very little.

If the queue (length) is long and
the rate of change is increasing slowly, (R15)
then the utilization factor should be very little.

If the queue (length) is long and
the rate of change is increasing fast, (R16)
then the utilization factor should be very little.

If the queue (length) is full and
the rate of change is decreasing fast, (R17)
then the utilization factor should be very little.

If the queue (length) is full and
the rate of change is decreasing slowly, (R18)
then the utilization factor should be very little.

If the queue (length) is full and
the rate of change is zero, (R19)
then the utilization factor should be very little.

If the queue (length) is full and
the rate of change is increasing slowly, (R20)
then the utilization factor should be very little.

If the queue (length) is full and
the rate of change is increasing fast, (R21)
then the utilization factor should be very little.

If the queue (length) is congested,
then the utilization factor should be very very little. (R22)

B.2 Parameters of FEWA

Each row k of the following Tables B.1 and B.2 shows the angle points $(x_{k,1}; y_{k,1}), \ldots, (x_{k,4}; y_{k,4})$ of the membership function m_{v_k} of the linguistic value v_k, $1 \leq k \leq n_v$.

Table B.1: Parameters of the linguistic variable $\Delta Q = Q/QT$ (see Figure 3.1)

k	$m_{\Delta Q_k}$
1	$(\ 0.00; 1), (\ 0.00; 1), (\ 0.20; 1), (\ 0.40; 1)$
2	$(\ 0.30; 0), (\ 0.50; 1), (\ 0.80; 1), (\ 0.90; 0)$
3	$(\ 0.80; 0), (\ 0.90; 1), (\ 1.10; 1), (\ 1.20; 0)$
4	$(\ 1.10; 0), (\ 1.20; 1), (\ 1.40; 1), (\ 1.60; 0)$
5	$(\ 1.50; 0), (\ 1.70; 1), (\ 2.00; 1), (\ 2.00; 1)$
6	$(\ 1.90; 0), (\ 2.00; 1), (\ 2.00; 1), (\ 2.00; 1)$

Table B.2: Parameters of the linguistic variable $\Delta G = G/B$ (see Figure 3.2)

k	$m_{\Delta G_k}$
1	$(-1.00; 1), (-1.00; 1), (-0.20; 1), (-0.15; 0)$
2	$(-0.20; 0), (-0.15; 1), (-0.10; 1), (-0.05; 0)$
3	$(-0.10; 0), (-0.05; 1), (\ 0.05; 1), (\ 0.10; 0)$
4	$(\ 0.05; 0), (\ 0.10; 1), (\ 0.15; 1), (\ 0.20; 0)$
5	$(\ 0.15; 0), (\ 0.20; 1), (\ 1.00; 1), (\ 1.00; 1)$

Notice:

- If $(x_{k,1}; y_{k,1})$ is equal to $(x_{k,2}; y_{k,2})$, then $m_{v_k}(x) = y_{k,1}$ for all $x \leq x_{k,1}$.
- If $(x_{k,3}; y_{k,3})$ is equal to $(x_{k,4}; y_{k,4})$, then $m_{v_k}(x) = y_{k,4}$ for all $x \geq x_{k,4}$.

Acknowledgments

The work presented in this technical report has been supported by a research contract with Ericsson Eurolab, Aachen, Germany.

Many thanks to Cornelius Mahlo (mahlo@cs.tu-berlin.de) who has implemented the traffic models for ns-2 (version ns-2.1b9). In addition, together with me he is the developer of a ns-2 framework in which the performance of different congestion feedback mechanisms in routers, e.g., EWA, FEWA, or XCP, can be investigated and evaluated under various network conditions.

Acronyms

ABR Available Bit Rate

AIMD Additive Increase Multiplicative Decrease

ATM Asynchronous Transfer Mode

CBR Constant Bit rate

CWND Congestion Window Size

EC Efficiency Controller

ECN Explicit Congestion Notification

ETCP Enhanced TCP

EWA Explicit Window Adaptation

FCC Fuzzy Congestion Controller

FERM Fuzzy Explicit Rate Marking

FERMA FERM Adaptation

FERMAM FERMA-Modification

FEWA Fuzzy Explicit Window Adaptation

FLC Fuzzy Logic Controller

IP Internet Protocol

MSS Maximum Segment Size

RCF Router Congestion Feedback

RED Random Early Detection

RTT Round Trip Time

TCP Transmission Control Protocol

UDP User Datagram Protocol

XCP Explicit Control Protocol

Bibliography

[1] K. Fall and K. Varadhan. *The ns Manual*. VINT Project, http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf, 2001.

[2] T. Goff, J. Moronski, D.S. Phatak, and V. Gupta. Freeze-TCP: A true end-to-end TCP enhancement mechanism for mobile environments.

[3] Packeteer Inc. Controlling TCP/IP bandwidth. http://www.packeteer.com/PDF_files/resources/tcp_bw.pdf, November 1998.

[4] R. Jain. *The Art of Computer Systems Performance Analysis*. Wiley & Sons, 1991.

[5] L. Kalampoukas, A. Varma, and K. K. Ramakrishnan. Explicit window adaptation: A method to enhance TCP performance. *IEEE/ACM Transactions on Networking*, 10(3):338–350, June 2002.

[6] D. Katabi, M. Handley, and C. Rohrs. Congestion control for high bandwidth-delay product networks. Proceedings of ACM SIGCOMM'02, August 2002.

[7] C. C. Lee. Fuzzy logic in control systems: Fuzzy logic controller—part I. *IEEE Transactions on Systems, Man and Cybernetics*, 20(2):404–418, March 1990.

[8] C. C. Lee. Fuzzy logic in control systems: Fuzzy logic controller—part II. *IEEE Transactions on Systems, Man and Cybernetics*, 20(2):419–435, March 1990.

[9] A. Pitsillides, Y.A. Şekercioglu, and G. Ramamurthy. Effective control of traffic flow in ATM networks using fuzzy explicit rate marking (FERM). *IEEE Journal on Selected Areas in Communications*, 15(2):209–225, 1997.

[10] K. Ramakrishnan, S. Floyd, and D. Black. The addition of explicit congestion notification (ECN) to IP. RFC 3168, September 2001.

[11] A. Reyes-Lecuona, E. Gonzáles-Parada, E. Casilari, and J. Casasola. A page-oriented WWW traffic model for wireless system simulations. In *Proceedings ITC 16*, pages 1271–1280, 1999.

[12] M. Savorić and U. Krieger. The performance of a fuzzy flow-control scheme for WWW data transfer in a wireless ATM network. In *Proc. of ITC Specialist Seminar on Mobile Systems and Mobility*, pages 145–156, March 2000.

[13] M. Savorić, U. Krieger, and A. Wolisz. The impact of handover protocols on the performance of ABR flow-control algorithms in a wireless ATM network. *European Transactions on Telecommunications (ETT)*, 11:419–430, August 2000. Special Issue on Service Quality Control in Multimedia Wireless Networks.