

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Adaptive Active Queue Management for TCP Friendly Rate Control (TFRC) Traffic in Heterogeneous Networks

Rahim Rahmani^a and Christer Åhlund^b

^aMid Sweden University, ^bLuleå University of Technology
SWEDEN

1. Introduction

Proposals to handle differentiated and guaranteed services in Internet have not provided the benefits that users and operators are expecting. Its complexity, with a large number of interconnected networks, is difficult to handle in an efficient way. This is due to the resource heterogeneity in terms of technologies and the inconsistent implementation of quality of services (QoS) in different networks. Despite several research activities in the area of QoS, Internet is still basically a best-effort network, which it is likely to stay, also in the far future. Streaming-based servers utilizing UDP for the underlying transport need to use some form of congestion control [1] to ensure the stability of Internet as well as the fairness to other flows, like those using TCP. The TCP-Friendly Rate Control (TFRC) is such a congestion control scheme appropriate for UDP. In this paper we target the question of how to optimize network and user-perceived performance in a best-effort network. In particular, we focus on the impact of end-to-end performance of the queue management scheme utilized in a wireless network.

The first of the basic assumptions in this study is that the wireless last hop constitutes the bottleneck of the end-to-end (E2E) path. TFRC is intended for applications such as streaming media, where a relatively smooth sending rate is of importance. TFRC measures loss rate by estimating the loss event ratio [2], and uses this measured rate to determine the sending rate in packets per RTT. When a bottleneck is shared with large-packet TCP flows, this has consequences for the rate achievable by TFRC. In particular a low bandwidth small-packet TFRC flow, sharing a bottleneck with high-bandwidth large-packet TCP flows, may be forced to slow down, even though the nominal rate of the TFRC application in bytes per second is less than the total rate of the TCP flows. This is fair only if the network limitation is defined by the number of packets per second, instead of bytes per second. In the TFRC protocol the small-packets are intended for flows that need to send frequent small quantities of information. It intends to support applications better, which should not have their sending rates in bytes per second decreased because of the use of small packets. This is restricted to applications, which do not send packets more often than every 10 ms. The TFRC Small-Packet (TFRC-SP) variant is motivated partly by the approach in Ref. [3], with

the argument that it is reasonable for voice over IP (VoIP) flows to assume that the network limitation is in bytes rather in packets per second, that the sending rate is the same in bytes per second as for a TCP flow with 1500 byte packets, and that the packet drop rate is the same. An application using TFRC-SP can have a fixed packet size or may vary its packet size in response to congestion.

Wireless channels suffer from bursty error losses reducing TFRC throughput, because TFRC incorrectly interprets packet loss as a sign of congestion. The maximum increase of TFRC rate, given fixed RTT, is estimated to be 0.14 packets per RTT and 0.22 packets per RTT with history discounting [4]. It takes four to eight RTTs for TFRC to halve its sending rate in the presence of persistent congestion.

Active Queue Management (AQM) intends to achieve high link utilization without introducing an excessive delay into the E2E path. For good link utilization it is necessary for queues to adapt to varying traffic load. The AQM has been subject to extensive research in the Internet community lately, and a number of methods to control the queue size have been proposed. An increase in RTT not only degrades the control performance of an AQM algorithm, but also leads to instability in the network.

The work described in this chapter examines throughput, aggressiveness, and smoothness of TFRC with varying bandwidth. A dynamic model of TFRC enables applications to address the basic feedback nature of AQM.

To demonstrate the generality of the proposed method an analytic model is described and verified by extensive simulation of different AQM.

2. Adaptive AQM (AAQM) Algorithm

AAQM is a light weight algorithm aimed at maximizing the flow of packets through the router, by continuously computing the quotient between the number of arriving and departing packets. The algorithm applies probabilistic marking of incoming packets to keep the quotient between arriving and departing packets just below 1 to minimize the queue length and maximize the throughput. Minimizing the queue length means minimizing the delay and packet loss.

2.1 Formal description

This part contains a more formal description regarding how the AAQM algorithm works. The goal of the AAQM algorithm is to keep the packet arrival rate as close to the packet departure rate as possible in order to maximize the flow of packets through the queue. The packet arrival rate is determined by the end systems sending packets and can for this reason be controlled by probabilistic marking of arriving packets. The packet departure rate is determined by the packet size and bandwidth of the outgoing link. The packet arrival rate is denoted by A and the packet departure rate is denoted by B , both of which are measured over T seconds. The utilization U of the flow of packets through the queue is defined by equation (1).

$$U = \frac{A}{B}, \quad B \neq 0 \quad (1)$$

If in the equation above , $U>1$ the queue grows, which can result in a queue overflow and packet loss. Similarly, if, in the equation above , $U<1$ the queue shrinks, which could result in a link underutilization and wasted bandwidth. Therefore to maximize the flow, U should be as close to 1 as possible. The required benefit is that when $U=1$ a packet arrives for each departing packet causing the link to be fully utilized, and if U is just below 1 the queue will simultaneously shrink minimizing both loss and delay. Now, if the total running time of the algorithm is divided into N non-overlapping time slots of size T seconds then the utilization of each time slot n is given by equation (2).

$$U_n = \frac{A_n}{B_n}, \quad B_n \neq 0, n = 1 \dots N$$

(2)

Now the packet marking probability P will be adjusted depending on the value of U during each time slot n . If $U>1$ then P can be increased to reduce the arrival rate and thus decrease U . Likewise, if $U<1$ then P can be decreased to increase the arrival rate and thus increase U . Equation (3) shows how the packet marking probability is adjusted.

$$P_{n+1} = P_n + f(U_n), \quad n = 1 \dots N$$

$$P_0 = 0$$

(3)

The function f is given by equation (4) and, depending on U , increases or decreases the packet marking probability P .

$$f(x) = \begin{cases} -P_1, & x < 1 \\ P_2, & x > 1 \end{cases}$$

(4)

$T, P1$ and $P2$ are constants that need to be determined beforehand.

2.2 Parameter Tuning

Of the variables described above only $P1, P2$ and T are user adjustable. Two sets of recommended values for these parameters are presented in Table 1. The first row of parameters in Table 1 gives the algorithm a little better utilization at the cost of higher loss and delay. In the second row of parameters gives the algorithm a little lower utilization but improved loss and delay characteristics. The recommended parameters have been found by performing several simulations using NS[13] of the network using different parameter settings.

P ₁	P ₂	T
0.700	0.900	0.010
0.700	1.000	0.010

Table 1. Recommended AAQM Parameter.

When tuning the algorithm for use in a network some basic aspects must be kept in mind. A larger T will give a larger time interval over which the quotient U is computed. This will

in turn lead to a better prediction of the flow through the queue but on the downside it will also lead to a slower reaction to changes in the flow. Slow reactions to changes in the flow can lead to high delay and packet loss. When choosing P_1 and P_2 it should be kept in mind that $P_1 < P_2$ leads to low utilization, delay and packet loss. Similarly $P_1 > P_2$ gives high utilization, delay and packet loss.

3. Simulation setup

We use the methods presented in [9] for the study of the average queue time to have a well-tuned AQM

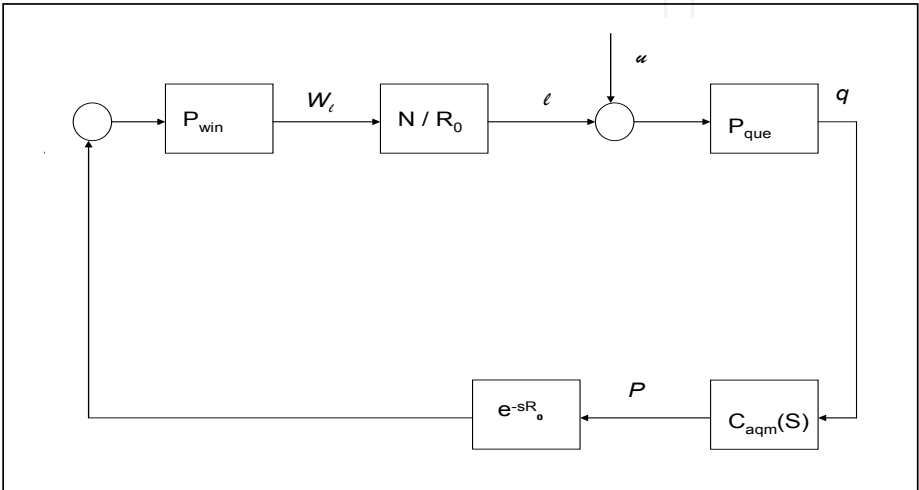


Fig. 1. Linearization of AQM with long-lived TCP traffic l and unresponsive traffic u .

with a queue averaging time less than RTTs. Figure 1 presents a block diagram of the linearized AQM feedback system, where $c_{aqm}(s)$ denotes the transfer function of the AQM controller, w_l the congestion window size in packets, R_0 the round-trip time, N the number of long-lived TCP connections, p_{win} the window size, and p_{que} the queue transfer function. The symbol l is used for long-lived TCP traffic (FTP) and u for unresponsive flows when describing how these flows impact on the queue length q , loss probability p , and arrival rates of both l and u . In this work we focus on randomly created short and long-lived flows as well as their impact on AQM. For applications needing to maintain a slowly changing sending rate, the equation-based congestion control is the most appropriate. This kind of application is the source of short-lived flows. It is assumed that they never escape the slow start phase of TCP, so their window sizes are increasing exponentially rather than linearly as during TCP's congestion-control phase. As a model of a short-lived flow we used the sending rate of TFRC [4] which is a rate based congestion control mechanism ensuring fairness when it co-exists with TCP flows [4]. Its throughput equation is a version of the throughput equation for conformant TCP Reno [4]:

$$X = \frac{B}{RTT \sqrt{\frac{2bp}{3}} + t_{RTO} \left(3 \sqrt{\frac{3bp}{8}} p (1 + 32 p^2) \right)},$$

(5)

where X is the transmission rate in byte/sec, as a function of the packet size B in bytes, RTT is the round-trip time in seconds, p the steady-state loss event rate, t_{RTO} is the TCP retransmission timeout value in seconds, and b the number of packets acknowledged by a single TCP acknowledgement. TFRC uses this equation to adjust the sender rate to achieve the goal of TCP friendliness.

The sender can calculate the retransmit timeout value t_{RTO} using the usual TCP algorithm:

$$t_{RTO} = SRTT + 4 * RTT_{var}, \quad (6)$$

where RTT_{var} is the variance of RTT , and $SRTT$ is the round trip time estimate. Different TCP congestion control mechanisms use different clock granularities to calculate retransmit timeout values, so it is not sure that TFRC accurately can model a typical TCP flow. Unlike TCP, TFRC does not use this value to determine whether to retransmit or not. Hence the consequences of this inaccuracy are less serious. In practice the simple empirical heuristic of $t_{RTO} = 4RTT$ works reasonably well to provide fairness with TCP [4]. For the dynamics of the model for TFRC behavior, we use the simplified and modified model presented in [10]. This model ignores the TCP timeout mechanism and is described by the nonlinear differential equation:

$$\dot{X}(t) = \frac{1}{R(t)} - \frac{X(t)}{2} \cdot \frac{X(t-R(t))}{R(t-R(t))} p(t-R(t)), \quad (7)$$

$$\dot{Q} = \begin{cases} -C + \frac{N(t)}{R(t)} X(t), & q > 0 \\ \max \left\{ 0, -C + \frac{N(t)}{R(t)} X(t) \right\}, & q = 0 \end{cases} \quad (8)$$

where X is the average packet rate (packets/s). The RTT is calculated as $R(t) = q(t)/C + T_p$, where q is the average queue length (packets), C is the link capacity (packets/s), and T_p is the propagation delay. The parameter p is the probability of a packet mark, the $1/R(t)$ term models additive increase of the packet rate, while the $X(t)/2$ term models multiplicative decrease of the packet rates in response to a packet marking p . Eq. (8) models the bottleneck queue length using the accumulated differences between packet arrival rates $N(t)X(t)/R(t)$ and the link capacity C , where $N(t)$ is the load factor (number of traffic sessions).

For linearization of Eqs. (7) and (8) and the estimation of some of the parameters Ref. [10] defines plant dynamics of the AQM feedback control as:

$$P(s) = \frac{C^2/2N}{(s + \frac{2N}{R^2C})(s + \frac{1}{R})}, \quad (9)$$

In Eq. (5) $C^2/2N$ is the high-frequency plant gain, which is an important parameter in the design of AQM control schemes, since it affects the stability, transient response, and steady-state performance. The variation of the queue length depends on the input rate, output rate, and the AQM controller.

The design of adaptive AQM algorithm should be based on the mean RTT [11] since realistic network conditions have heterogeneous flows with different RTTs. The link capacity C is measured by keeping track of departed packets, and an estimate of the TFRC load N/R is inferred from measurements of the dropping probability p . The TFRC throughput $N/RC = \sqrt{p/2}$ provides a mean for estimating N/R .

For the parameterization of AQM dynamics the generalized fluid description of AQM dynamics relating instantaneous queue length q and loss probability p from [12] is used which is described as follow:

$$\begin{aligned}\chi_{am} &= f_{\theta}(\chi_{am}, q) \\ p &= g(\chi_{am}, q)\end{aligned}\quad (10)$$

Where χ_{am} denotes AQM state and f and g describe the AQM dynamic behavior. In RED [5], χ_{am} is the average queue length (RED bases its decision whether to mark a packet or not on the average queue length). In REM [6] χ_{am} is the marking probability and the link price and in BLUE [7] χ_{am} is the packet loss and link utilization history used to manage congestion.

In AAQM [8] χ_{am} is the quote between the numbers of arrived and departed packets within a timer event. This kind of event occurs at fixed, evenly spaced, time intervals, see [8]. The quote is used as the utilization factor of the queue. For tuning of the AAQM the control parameters of the AAQM (P_1 , P_2 , T) are linked to the network parameters N , R and C . In the absence of knowledge of these parameters we can develop an on-line estimation of the link capacity C and the TFRC load N .

The link capacity \hat{C} is estimated using the model in [12] extended with computation of the packet arrival rate. The degree of TFRC traffic utilization of the link changes depending on the competing traffic, and to smooth the TFRC capacity \hat{C} a low-pass filter (LPF) and the fluid mechanism is used:

$$\dot{\theta}_C = -K_C \theta_C + K_C \hat{C}, \quad (11)$$

where θ_C denotes the estimated capacity and K_C the filter time constant. To estimate the TFRC load we use the TFRC fluid model presented by Eqs. (7) and (8) and derive the following steady-state TFRC relationship:

$$0 = \frac{1}{R} - \frac{X_0^2}{2R} P_0, \quad (12)$$

$$0 = \frac{N}{R} X_0 - C, \quad (13)$$

Where X_0 is the equilibrium congestion packet rate and P_0 is the equilibrium drop probability. From formulas (12) and (13) we can obtain $\sqrt{p_0/2} = N/RC$. In the same way as we estimated the link capacity, we can smooth the estimate of the N/RC by using the LPF:

$$\dot{\theta}_{rc}^n = -K_{rc}^n \theta_{rc}^n + K_{rc}^n \sqrt{p/2}, \quad (14)$$

Where θ_{rc}^n is the smoothed estimate of the N/RC and K_{rc}^n the filter time constant.
For self-tuning AQM uses the parameters estimated in Eqs. (11) and (14) described as:

$$\begin{aligned}\chi_{am} &= f_{\theta}(\chi_{am}, q), \\ p &= g_{\theta}(\chi_{am}, q),\end{aligned}$$

(15)

Where f_{θ} and g_{θ} show the explicit dependency of AQM dynamics using the estimate variables $\theta = (\theta_C, \theta_{rc}^n)$.

When the 802.11 MAC layer approaches saturation, contention delays induced by deferred count-down timers, an increased contention window and retransmissions affecting the performance of TFRC and TCP [16]. TFRC is unaware of the MAC layer congestion, rendering in that the TFRC sender may overestimate the maximum sending rate. This congests the MAC layer, and the wireless network consequently reaches a sub-optimal stable state with respect to the throughput and round-trip time [17][18]. Because of the lack of interaction between TFRC and the 802.11 MAC layer, a Rate Estimator (RE) is added to TFRC [17]. The RE approximates the saturation capacity of the MAC layer, and by limiting the sending rate, MAC layer congestion can be avoided.

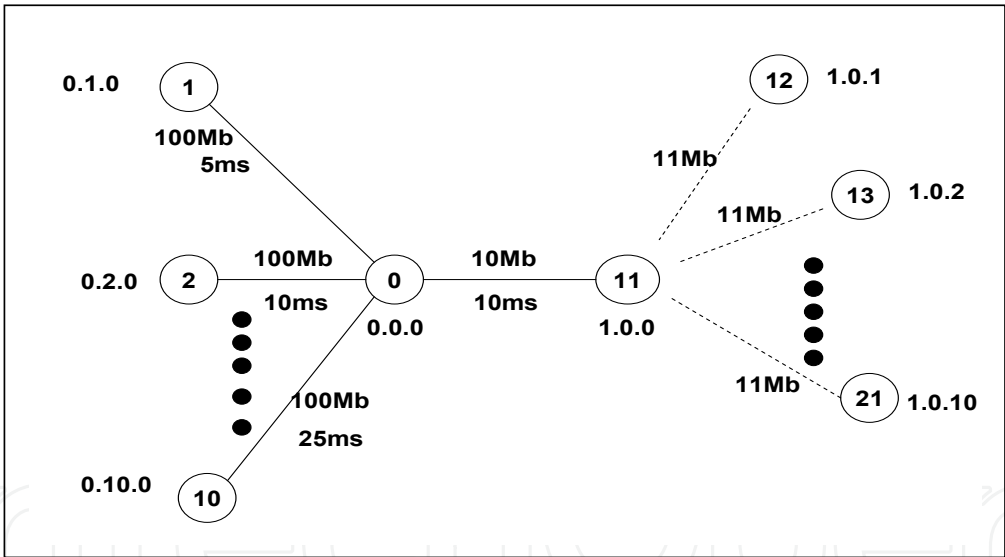


Fig. 2. Configuration of the simulated network

4. Simulations

The simulation results presented obtained using the NS-2 simulation tool [13]. Figure 2 depicts the topology used. The solid lines symbolize wired links and the dashed lines wireless links. Nodes numbered 1-10 are fixed; those numbered 12 - 21 are mobile, and the node numbered 11 is the 802.11b base station (infrastructure mode). The link between the nodes number 0 and 11 represents the virtual bottleneck link running the AQM algorithms. An access point has two interfaces an 802.11 wireless interface to transmit/receive frames on the air and a wired interface. The disparity in channel capacity of these two interfaces makes

the access point a significant potential bottleneck link. The virtual bottleneck represents an access point.

Data are originating at the nodes numbered 1 - 10 and received by the nodes numbered 12 - 21. Each link carries both a TFRC and a TCP flow using a Pareto traffic generator (to generate aggregate traffic that exhibits a long-range dependency). The traffic generators start randomly after one second of the simulation to avoid a deterministic behavior and lasts for 100 seconds. Each simulation was run 30 times with different seeds for the random number generator. TCP-SACK [14] is used with Selective Acknowledgments (SACK), allowing a receiver to acknowledge out-of-order segments selectively. The TFRC flows are modeled as short-lived small packets web flows, and the TCP flows as a mix of short-lived flows and long-lived FTP flows. The sources of the short-lived web flows are modeled according to Ref. [15].

In table 2 the different parameters settings used are listed. The TFRC and TCP timer granularity used, i.e. the tick value, is set to 500 ms, and the TCP minimum retransmission timeout to 1 s. The throughput at the bottleneck link, the queue size (in packets), and the drop probability are used for performance evaluation throughout the simulations.

Queues	Queue sizes (packets)	Numbers of web sessions
BLUE	5	400
Drop-Tail	10	800
RED	50	1600
REM	100	
AAQM		

Table 2. List of parameter settings used in different scenarios and packetsize 14 bytes.

The parameter settings used in the AQM algorithms are shown in Tab. III. For Blue the values were obtained from Ref. [7], for RED from Ref. [5], for REM from Ref. [6], and for AAQM from Ref. [8].

BLUE	freeze_time=10 ms	D1=0.001	d2=0.0002	
RED	Wq=0.002	minth=20% of the queue size	maxth=80% of the queue size	maxp=1
REM	Gamma=1	Phi=1.001	Bo=20	Updtime=0.0
AAQM	P1=0.700	P2=1.000	T=0.010	

Table 3. Parameter setting for the simulated aqm

5. Performance evaluation

To achieve satisfactory control performance, the design goals of AQM algorithms are responsiveness and stability. Bursty sources are used for the performance evaluation with varying queue sizes managed by the AQM algorithms. The bursts generated require an AQM algorithm to efficiently and quickly adapt to the current situation to maintain a high overall throughput and to avoid dropping more packets than necessary. As a result the drop rate and throughput are compared for the different algorithms.

Figures 3 and 4 depict the results for the drop rate and overall throughput respectively. Four queue sizes are used 5, 10, 50 and 100 packets. The curves are plotted using a 95% confidence interval

Figures 3 shows that Drop-Tail (DT) exhibits a high drop rate. This is due to the fact that all packets are dropped when the queue is full so it affects all flows. All sources will then decrease its sending rate at approximately the same time. This also means that all sources will increase the sending rate about the same time rendering in full queues once more and this will create oscillation. and beyond a high drop rate low throughput is also exhibited.

Figures 3(a) and 3(b) show AAQM and RED have similar drop rate but in figures 3(c) and 3(d) RED has higher drop rate than AAQM.

The instantaneous queue length of RED is controlled in range of \min_{th} and \max_{th} . In order to be effective a RED queue must be configured with a sufficient amount of buffer space to accommodate an applied load greater than the link capacity from the instant in time when the applied load decreases at the bottleneck link in response to congestion notification. RED must ensure that congestion notification is given at a rate which sufficiently suppresses the transmitting sources without underutilizing the link. When a large number of TCP flows are active the aggregate traffic generated is extremely bursty. Bursty traffic often defeats the active queue management techniques used by RED since queue length grow and shrink rapidly before RED can react. Exactly that happens in figures 3(c), 3(d), 4(c) and 4(d).

On the other hand the queues controlled by REM and BLUE are often empty. When the link capacity is low AAQM regulates the queue length, where REM and BLUE oscillate between an empty buffer and its limit of queue size. As a result REM and BLUE show poor performance under a wide range of traffic environments.

Drop-Tail and REM must be configured with a sufficient amount of buffer space in order to accommodate an applied load greater than the link capacity from the instant that congestion is detected, using the queue length trigger, to the instant when the applied load decreases in response to congestion notification.

In general, the bias against bursty traffic and the tendency towards global synchronization can be eliminated by maintaining a stable packet loss over time. The steady-state control performance of each AQM algorithm was evaluated and the packet loss rate studied at three different traffic loads.

As shown in figure 3 and figure 4 the flow dynamics are severely oscillating and it takes a long time to stabilize to a steady-state. The AAQM controller is able to compensate the oscillatory of the flow dynamics and given satisfactory control performance such as a fast and stable control dynamics. AAQM show the most robust steady-state control performance, independent of traffic loads, in terms of relatively small mean value of the packet loss rate as well as its variance.

In the experiments we allowed the AQM schemes to converge to steady state when there were 400 web sessions, then we increased the web session number to 800 and 1600 to study

the performance of the AQM schemes when the number of short flows increases. From figures 4(a) and 4(b) it can be seen that AAQM has 1% better throughput than RED when the number of sessions is 1600, and much higher throughput than BLUE, REM and DropTail. In figure 4(c) AAQM has 3% better throughput than RED, and from figure 4(d), it can be seen that AAQM has 36% better throughput than RED.

The results from Figures 3 and 4 show that AAQM has the best responsiveness to congestion as well as the most robust steady-state control.

6. Related work

The work presented in [9] studies the effects of unresponsive flows on AQM. It shows that the queue averaging time is a result of a trade-off between AQM responsiveness and the robustness of the uncontrolled flows. The average queue time results in a smooth or stable congestion feedback, which introduces jitter in the queuing delay due to variation in the unresponsive flows. Three types of flow types were considered: short-lived TCP, Markov on-off UDP, and traffic with long-range dependencies (e.g. ftp). Our work instead focuses on short-lived flows and uses a more realistic model for VoIP traffic by using TFRC-SP. Also [9] does not study the impact of unresponsive flows on the AQM algorithms, while we do. Our focus is on the responsiveness of UDP flows with co-existent TCP flows.

The work presented in [19] surveys two adaptive and proactive AQM algorithms using a classical proportional-integral-derivative feedback control to achieve stability and responsiveness. The TCP flows are modeled as long-lived FTP flows. In our work the flows are modeled as the mix of long-lived flows and short-lived flows to fulfill the design goal of an adaptive AQM to interact with a realistic flow composition.

In reference [22] the authors argue in favor of rate-based AQM for high-speed links. Also in that work a proportional-integral controller for the AQM scheme is used. The design goal was to match the aggregate rate of TCP flows to the available capacity while minimizing the queue size. We study the integration of TFRC-SP and UDP in co-existence with TCP in heterogeneous networks.

The work presented in [23] uses a token-bucket model as a virtual queue (VQ) with a link capacity less than the actual link capacity. If a packet arrives, it is placed in a queue in the VQ if there is space available. Otherwise the packet is dropped. Accordingly the algorithm is able to react at an earlier stage, even before the queue grows, making it very sensitive to the traffic load and round trip time. However, the utility functions are much different from ours due to the AQM control parameters. AAQM uses control law and link utilization in order to manage congestion. The action of the control law in AAQM is to mark incoming packets in order to maintain the quotient between arriving and departing packet as close to one as possible.

The study in reference [24] focused just on the RED and the parameter setting of RED was based on heuristics. It also studied RED against disturbances on the wireless access network. Only one type of flow types was considered: short-lived TCP. We study UDP in co-existence with TCP and their impact on DT, RED, REM, BLUE and AAQM.

The work presented in [25] by using of proxy AQM between access point for WLAN and wired network. The proxy reduces the overhead of the access point by implementing the AQM functionality at the gateway. In the work they extended the RED/ARED scheme to a

proxy mode by calculating the average queue length and updating p_{ma} of ARED. They measured only a number of TCP flows.

In reference [26] a channel-aware AQM scheme is presented. This new approach provides congestion signals for flow control not only based on the queue length but also the channel condition and the transmission bit rate. For the performance evaluation of the new AQM in multi-rate WLAN the bit rate of the wireless node in manuals fixed at different levels (in sequence of 2M, 1M, 11M, and 5.5 Mbps). Two TCP flows were considered. The main idea in the work was to design an AQM for flow control in multi-rate WLAN.

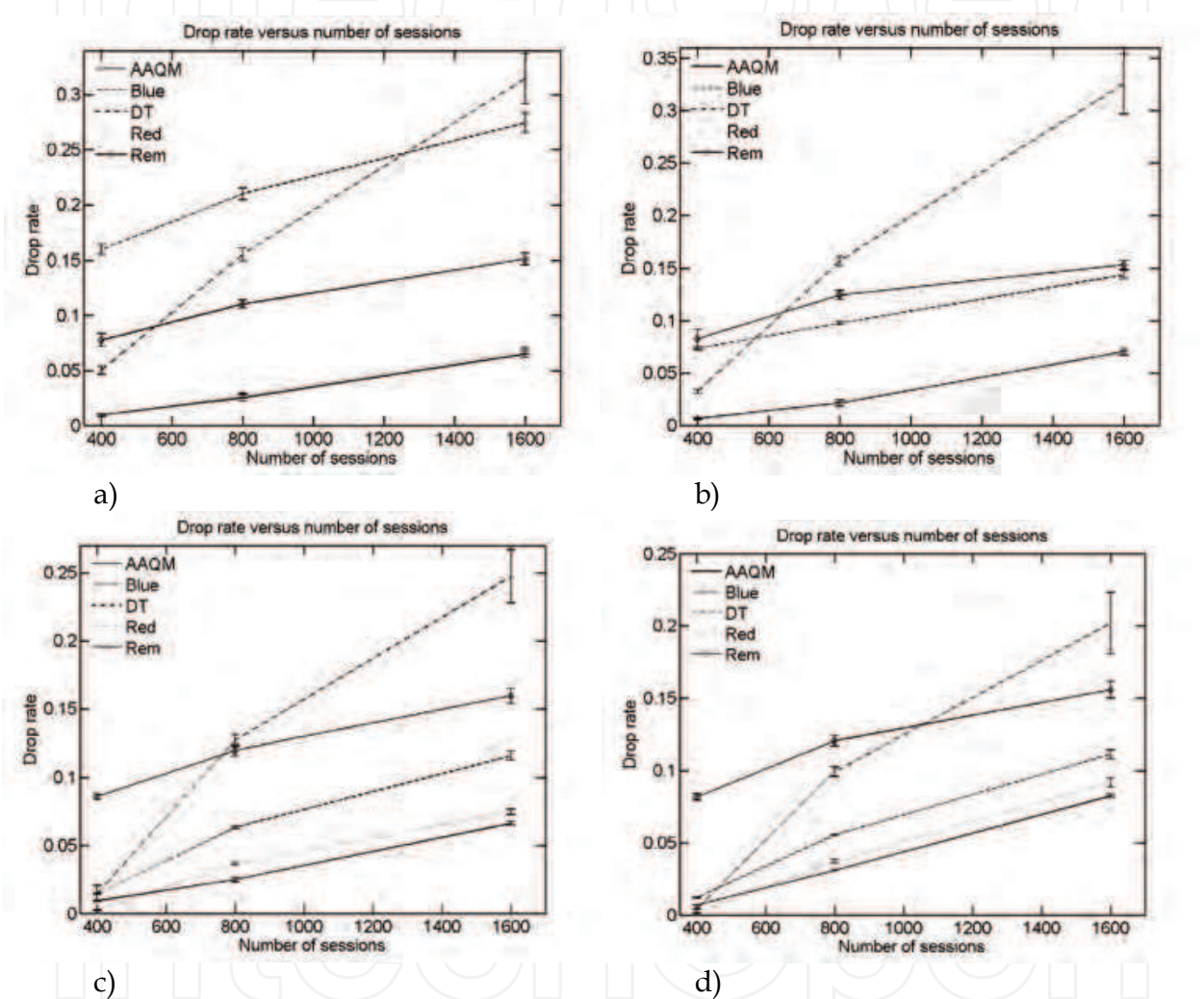


Fig. 3. Packet loss rate for a packet size of 14 bytes; (a) queue size 5 packets, (b) queue size 10 packets, (c) queue size 50 packets, (d) queue size 100 packets.

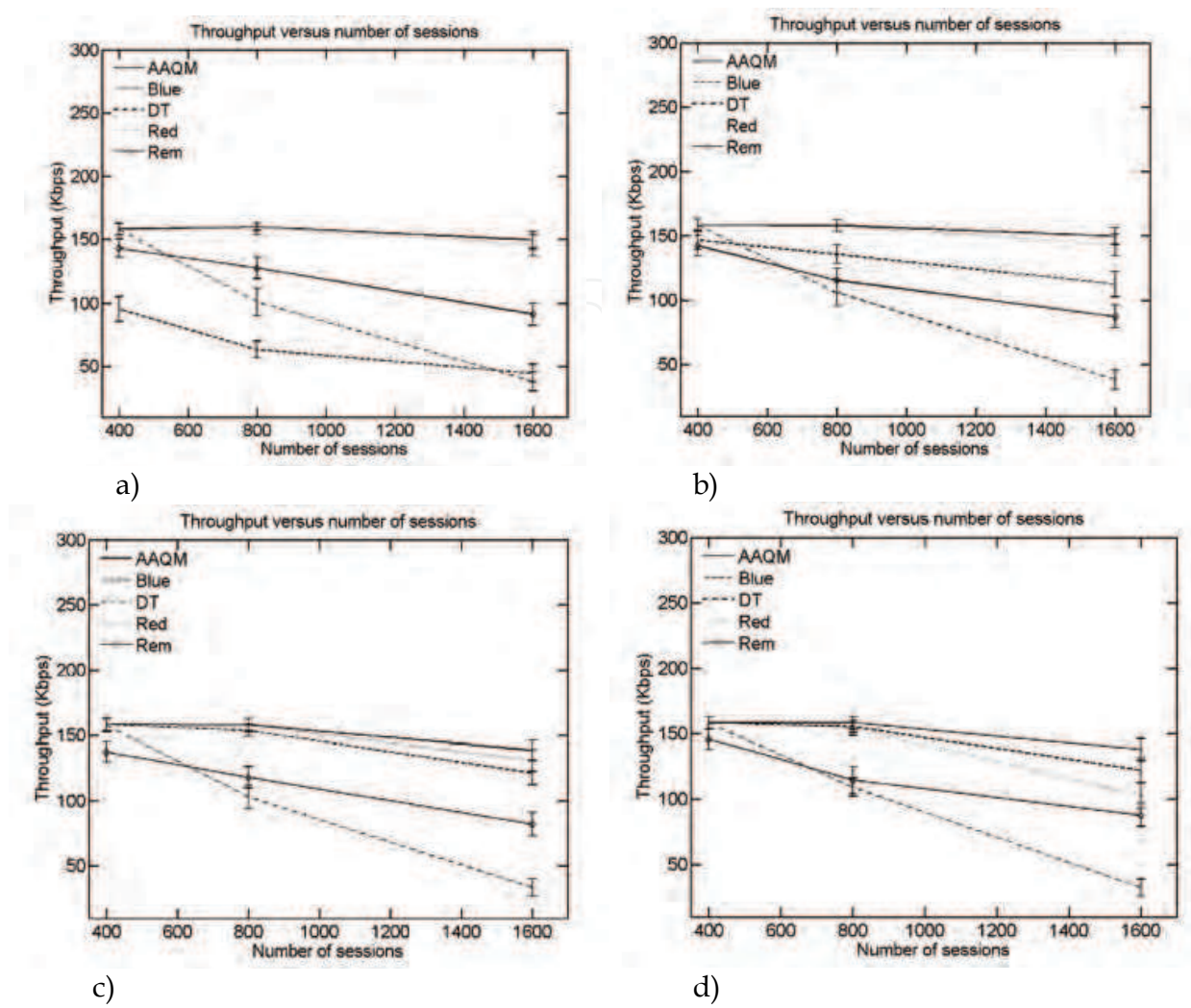


Fig. 4. Throughput of the link for a packet size of 14 bytes; (a) queue size 5 packets, (b) queue size 10 packets, (c) queue size 50 packets, (d) queue size 100 packets.

7. Conclusions and future work

The present work studies AQM algorithms for competing small-packet TFRC flows and TCP flows in heterogeneous networks. We investigate the effects caused by unresponsive flows using TFRC on the AQM performance that is measured using responsiveness and stability. Through simulations it is shown that with suitable design of the AQM scheme the end-to-end performance can be maintained for TFRC flows consisting of small packets. It is shown that the control performance of Drop-tail, RED, and BLUE are very sensitive to the traffic load and round trip time. With REM the links suffer in utilization as the buffer size increases. In particular AAQM shows a stable queue length with low and smooth packet loss rates independent of the traffic load.

Future works will particularly investigate the integration of the Guaranteed TFRC (GTFRC) [20] with the DCCP protocol [21] and co-existence of DCCP with the Stream Control Transmission Protocol (SCTP). The impact of these protocols on the AQM performance and QoS will be studied in heterogeneous networks. An additionally study will be focused on optimization of the buffer space requirements.

8. References

- [1] S. Floyd, "Congestion Control Principles" RFC2914, September 2000.
- [2] M. Handley, J. Padhye, S. Floyd, and J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol Specification". RFC3448, January 2003.
- [3] RFC 3714.
- [4] S. Floyd, M. Handley, J. Padhye, and J. Widmer. "Equation-based congestion control for unicast application". In proc. of ACM SIGCOMM'00, Aug. 2000.
- [5] Sally Floyd and Van Jacobson, "Random Early Detection Gateways for Congestion Avoidance", IEEE/ACM Transactions on Networking, vol. 1, Number 4, August 1993.
- [6] Sanjeeva Athuraliya, "A Note on Parameter Values of REM with Reno-like Algorithms", Networking Laboratory, Caltech, March 2002.
- [7] Wu-chang Feng, Dilip Kandlur, Debanjan Saha and Kang Shin, "BLUE: A New Class of Active Queue Management Algorithms", Department of EECS, University of Michigan, Ann Arbor, MI 48105.
- [8] R. Rahmani, "Adaptive Active Queue Management in Heterogeneous Networks" in 26th ITI 2004, Croatia, June 2004.
- [9] C.V. Hollot, Y. Liu, V. Misra, D. Towsley "Unresponsive flows and AQM Performance", In Proceedings IEEE, 2003.
- [10] V. Misra, W. Gong, and D. Towsley, "Fluid based analysis of a network of AQM routers supporting TCP flows with an application to RED, " in Proceedings of ACM/SIGCOMM, 2000.
- [11] C. Hollot, V. Misra, D. Towsley, and Gong, "Analysis and design of controllers for AQM routers supporting TCP flows," IEEE Transactions on Automatic Control, vol. 47, no. 6, 2002.
- [12] H. Zhang, C. Hollot, D. Towsley, and V. Misra, "A self-tuning structure for adaptation in TCP/AQM networks, " Department of computer sciences, UMass, Amherst, Tech. Rep., July 2002.
- [13] The NS-2 Simulator available at <http://www.isi.edu/nsnam/ns/>.
- [14] RFC 2018.
- [15] Anja Feldmann, Anna C., P. Huang and W. Willinger "Dynamics of IP traffic: A study of the role of variability and the impact of control" sigcomm99, pub acm, pages 301-313, <http://www.acm.org/sigcomm/sigcomm99/papers/session8-3.html>.
- [16] Saikat Ray, Jeffrey Carruthers, and David Starobinski, "RTS/CTS-induced Congestion in Ad-Hoc Wireless LANs" in Proceedings of IEEE WCNC, Mar. 2003, pp. 1516-1521.
- [17] Mingzhe Li, Choong-Soo Lee, Emmanuel Agu, Mark Claypool, and Robert Kinicki, "Performance Enhancement of TFRC in Wireless Ad Hoc Networks, Distributed Multimedia Systems (DMS)", September 2004.
- [18] Henghua Fu, Xiaoqiao Meng, and Songwu Lu, "A Transport Protocol For Supporting Multimedia Streaming in Mobile Ad Hoc Networks", IEEE JSAC, December 2004.
- [19] S. Ryu, C. Rump, C. Qiao "Advances in Active Queue Management (AQM) Based TCP Congestion Control" Telecommunication Systems Journal Kluwer publisher 25:3/4, p 317-351, 2004.

- [20] E. Lochin, L. Dairaine, G. Jourjon "gTFRC: a QoS-aware congestion control algorithm", The 5th International Conference on Networking (ICN'2006) Mauritius - April, 2006.
- [21] S. Floyd, E. Kohler, J. Padhye : " Profile for DCCP Congestion Control ID 3: TFRC Congestion Control ", RFC 4342, IETF (2006).
- [22] J. Aweya, M. Ouellette, D. Y. Montuno, K. Felske, "Rate-based proportional-integral control scheme for active queue management" International Journal of Network Management John Wiley & Sons Publisher 16: p 203-231, 2006.
- [23] S. Kunniyur, R. Srikant, " An Adaptive Queue(AVQ) Algorithm for Active Queue Management" ACM Transactions on Networking Vol.12 No.2 April 2004.
- [24] F. Zheng, J. Nelson, " An H_∞ approach to congestion control design for AQM routers supporting TCP flows in wireless access networks" Computer Networks vol. 51 p 1684-1704, 2007.
- [25] S. Yi, M. Kappes, S. Grag, X. Deng, G. Kesidis, C. R. Das, " Proxy-RED: An AQM Scheme for Wireless Local Area Networks" 0-7803-8784-8/\$20.00 2004, IEEE.
- [26] Y. Xue, H. V. Nguyen, K. Nahrstedt " CA-AQM: ChannelAware Active Queue Management for wireless Networks" ICC 2007, IEEE.

IntechOpen



Trends in Telecommunications Technologies

Edited by Christos J Bouras

ISBN 978-953-307-072-8

Hard cover, 768 pages

Publisher InTech

Published online 01, March, 2010

Published in print edition March, 2010

The main focus of the book is the advances in telecommunications modeling, policy, and technology. In particular, several chapters of the book deal with low-level network layers and present issues in optical communication technology and optical networks, including the deployment of optical hardware devices and the design of optical network architecture. Wireless networking is also covered, with a focus on WiFi and WiMAX technologies. The book also contains chapters that deal with transport issues, and namely protocols and policies for efficient and guaranteed transmission characteristics while transferring demanding data applications such as video. Finally, the book includes chapters that focus on the delivery of applications through common telecommunication channels such as the earth atmosphere. This book is useful for researchers working in the telecommunications field, in order to read a compact gathering of some of the latest efforts in related areas. It is also useful for educators that wish to get an up-to-date glimpse of telecommunications research and present it in an easily understandable and concise way. It is finally suitable for the engineers and other interested people that would benefit from an overview of ideas, experiments, algorithms and techniques that are presented throughout the book.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Rahim Rahmani and Christer Ahlund (2010). Adaptive Active Queue Management for TCP Friendly Rate Control (TFRC) Traffic in Heterogeneous Networks, Trends in Telecommunications Technologies, Christos J Bouras (Ed.), ISBN: 978-953-307-072-8, InTech, Available from: <http://www.intechopen.com/books/trends-in-telecommunications-technologies/adaptive-active-queue-management-for-tcp-friendly-rate-control-tfrc-traffic-in-heterogeneous-network>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen