2nd POLISH-GERMAN TELETRAFFIC SYMPOSIUM PGTS 2002
9th Polish Teletraffic Symposium

COMBINED MODELLING OF TCP AND MULTI-RED IN DIFFSERV NETWORKS

Matthias Baumann, Dmitry Marandin, Samer Sulaiman
TU Dresden, Chair for Telecommunications
{baumann | marandin | sulaiman}@ifn.et.tu-dresden.de

Abstract
This paper presents a new numerical method to analyse the steady-state TCP throughput in DiffServ networks. The method couples models for TCP, marking device, and multi-RED queue, establishing a closed network model. We assess the accuracy and limits of the technique and discuss parameter settings of marking devices.

1 Introduction

Dependable provision of quality-of-service (QoS) in integrated IP networks is still an issue of ongoing research. The framework "Integrated Services Architecture" (IntServ, [2]) relies on a connection-oriented network operation, thus allowing resource reservation in order to guarantee bit rate or delay bounds. The framework "Differentiated Services Architecture" (DiffServ, [1]) tries to overcome scalability problems of IntServ by superimposing single traffic flows to a (small) number of aggregated flows. Packet classification in network nodes then becomes feasible even in core networks, since little or no flow-specific information has to be processed.
Due to the appealing simplicity of the DiffServ approach, much effort has been invested in assessing its performance: is it possible to guarantee user-requested bit rates by co-ordinating i) TCP's congestion control mechanisms, ii) traffic marking and/or shaping schemes at network edges, and iii) active queue management in network nodes?
Predicting steady-state TCP throughput has been subject of a large number of contributions, e.g. the seminal work [12]. Recently, a number of researchers have modelled and evaluated the performance of TCP connections with token bucket or time-sliding window marking in a DiffServ environment [11][13][14]. In these contributions, it has been assumed that packet loss probabilities for in-profile (IN) and out-of-profile (OUT) packets are given as input for the numerical analysis. Analytical techniques based on fluid-flow models [8][10] and iteration-based approaches [5] can be used to analyse combined models of TCP end systems and network nodes with active queue management. To our knowledge, there are no techniques extending these results to the combined analysis of TCP and RED in DiffServ environments (Recently,

work on fluid-flow analysis of DiffServ networks has been published in [4]).
The paper is organized as follows. First, we review approaches to model the TCP throughput in non-DiffServ networks. Secondly, we propose a new method to analyse TCP throughput in DiffServ networks with multi-RED queue management. We compare numerical with simulation results, and discuss parameter settings for marking devices.

2 TCP Models

Methods to analyse TCP throughput can be subdivided into two groups. The first one uses a heuristic approach assuming that packet loss events occur evenly spaced over time. This leads to a periodic evolution of the congestion window size allowing simple calculation of the values of interest. Some contributions of this group consider loss detection by time-out (TO) and by triple duplicates, e.g. [12], whereas others neglect the influence of TOs, e.g. [7]. Results are the mean congestion window size

$$EW = 3/4 W_{max} = \sqrt{3 / (2 p_L)} \text{ (in packets)}, \quad (1)$$

and an estimation for the probability $P_{TO} \approx \min(1, 9/(4EW))$ that a loss only can be detected by a TO (W_{max}: maximum congestion window size, p_L: packet loss probability). The second group assumes loss indications arriving at the sender with negative-exponentially distributed distances [8][10]. The result is a higher expected congestion window size of $EW' = \sqrt{2 / p_L}$. If losses do not occur evenly spaced but say with first a long distance in time and then sometimes with smaller distance in order to reach the same mean distance as in the periodic case, then the gain in amount of transmitted packets during the long period is larger then the loss during the subsequent shorter periods.
From measurement and simulation results in the above contributions as well as from additional simulation studies with the network simulator *ns-2* we conclude:
a) If the frequency of TOs tends towards zero, then the first analysis underestimates mean congestion window size and therefore throughput. TO frequency becomes low, if e.g. end systems with selective acknowledgements are used (SACK, [7]).
b) The frequency of TOs often is severely underestimated by all proposed techniques, if Tahoe or Reno TCP variants are used [11][12][14]. As the initial result (equ. (1)) of congestion window size without TO is too low, while underestimating the throughput reduction by TOs, the final results are often quite accurate when compared to measurements or simulations.
c) In QoS-oriented IP networks, network nodes will apply active queue management like multi-RED which reduces the frequency of burst packet losses. Additionally, TCP implementations will evolve towards variants avoiding TOs under most circumstances. Thus, it seems reasonable to neglect the influence of TOs.
d) All known techniques to model the marking devices [11][13][14] apply the heuristic approach of assuming periodic congestion window evolution. As this reduces

model accuracy, we normally would need more rigorous approaches. We still use the heuristic approach, as it provides useful results (see Section 6). An analysis of marking devices with non-periodic TCP behaviour remains as topic for further research.

3 TCP with Two-Colour TSW Marker

The time-sliding window (TSW) marker has been introduced in [3]. It calculates an exponentially weighted number of packets received during a constant time T, thus measuring the packet rate over a time window of T. Packets exceeding the committed (target) information rate CIR are proportionally marked as OUT. The exact algorithm is described in [3]. For TCP analysis with bucket markers see e.g. [4][13].
Our TCP throughput analysis is based on the ideas presented in [11][13][14] which all rely on assuming a periodic evolution of TCP's congestion window. As proposed in [3], we investigate two parameter settings. For the first one, we request that the marker time constant T is short (T in the order of one round-trip time RTT). This implies that the *current* probability of marking a packet as IN depends on the *current* window size W. In the second case, we assume a substantially larger time constant T which yields a marking probability depending on the *mean* window size EW. For our analysis, we need to determine the *mean* probability p_m of marking packets IN.

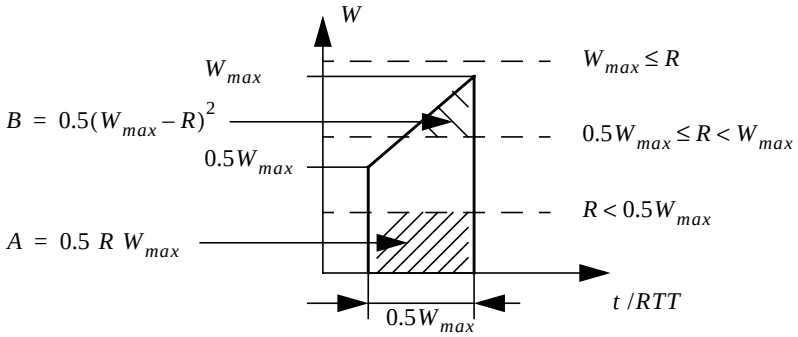

Figure 1: Congestion and reservation window size.

Setting 1 (short time T): Depending on the ratio between the maximum window size W_{max} and the reservation window size $R = CIR\ RTT/k$, (k: packet size in bits), three cases for calculating p_m can be distinguished (see e.g. areas A and B in Fig. 1):

$$p_m = \begin{cases} 1 & \text{if } W_{max} \le R \\ 1 - \dfrac{1/2(W_{max} - R)^2}{3/8 W_{max}^2} & \text{if } \dfrac{1}{2}W_{max} \le R < W_{max} \\ \dfrac{1/2\ R\ W_{max}}{3/8 W_{max}^2} & \text{if } R < \dfrac{1}{2}W_{max} \end{cases} \qquad (2)$$

The relation between average packet loss probability p_L and loss probabilities p_{IN}, p_{OUT} for IN and OUT packets, respectively, is given by

$$p_L = p_m p_{IN} + (1 - p_m) p_{OUT} . \tag{3}$$

Together with equ. (1), we derive ($\delta = p_{OUT} - p_{IN}$, $\gamma = 4p_{OUT} - p_{IN}$)

$$W_{max} = \begin{cases} \sqrt{\frac{8}{3p_{IN}}} & \text{if } W_{max} \le R \\ \frac{4R\delta + 2\sqrt{4R^2\delta^2 + 2\gamma(1 - 0.5R^2\delta)}}{\gamma} & \text{if } \frac{1}{2}W_{max} \le R < W_{max} \\ -\frac{a}{2} + \sqrt{\frac{a^2}{4} + \frac{8}{3p_{OUT}}} & \text{if } R < \frac{1}{2}W_{max}, \text{ with } a = \frac{4R(p_{IN} - p_{OUT})}{3p_{OUT}} \end{cases} \tag{4}$$

For a large time constant T, only two cases have to be considered ($EW = 3/4\ W_{max}$):

$$p_m = \begin{cases} 1 & \text{if } 3/4\ W_{max} \le R \\ 4R\,/(3W_{max}) & \text{otherwise} \end{cases} \tag{5}$$

Together with equs. (3) and (1), this leads to

$$W_{max} = \begin{cases} \sqrt{\frac{8}{3p_{IN}}} & \text{if } 3/4\ W_{max} \le R \\ -\frac{a}{2} + \sqrt{\frac{a^2}{4} + \frac{8}{3p_{OUT}}} & \text{otherwise, with } \quad a = \frac{4R(p_{IN} - p_{OUT})}{3p_{OUT}} \end{cases} \tag{6}$$

Both for large and small values of T, the mean throughput B of a TCP connection finally follows from equs. (4) or (6), and

$$B = k\frac{3W_{max}}{4RTT} . \tag{7}$$

4 Multi-RED Queueing Model

The Random Early Detection (RED) buffer management scheme has been proposed to accompany a transport-layer congestion control protocol such as TCP. If the average queue occupation exceeds a minimum threshold, RED starts dropping packets. The rate of packet drop increases linearly, as the average queue occupation increases until it reaches a maximum threshold. Above the maximum threshold, all packets are dropped.

Multi-RED extends RED to handle two classes of packets. Service discrimination between classes can be achieved in different ways. The first one is to use two thresholds to decide when to begin dropping packets, the threshold for OUT packets being lower than that for IN packets (see Fig. 2). Another way is to set the same thresholds for both classes, but to use drop probabilities that increase at different rates as the mean queue length increases.

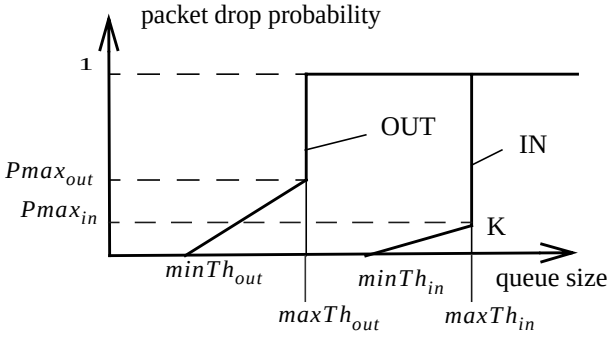

Figure 2: Parameters of a multi-RED queue.

We have used the model for calculating drop probabilities for IN and OUT packets, presented in [9]. Assume that packets arrive in the queue according to a Poisson process with rate λ. Packets are marked IN with probability p_m and OUT with probability $1 - p_m$. Both types of packets require a service exponentially distributed with parameter μ. The total offered load is denoted $\rho = \lambda/\mu$, the buffer size is K. Then according to [9], drop probabilities for IN and OUT packets are given by

$$p_{IN} = 1 - \sum_{n=0}^{K} \alpha^{in}(n)\pi(n)\,;\; p_{OUT} = 1 - \sum_{n=0}^{K} \alpha^{out}(n)\pi(n)\,, \tag{8}$$

where $\pi(n)$ is the probability that n packets will be in the buffer. It can be found based on queueing theory and Markov chain analysis particularly.

$$\pi(n) = \pi(0)\rho^n \prod_{i=0}^{n-1} \alpha(i)\,;\; \pi(0) = \left[\sum_{n=0}^{K} \rho^n \prod_{i=0}^{n-1} \alpha(i) \right]^{-1} \tag{9}$$

The value $\alpha(n) = p_m\alpha^{in}(n) + (1 - p_m)\alpha^{out}(n)$ is the probability that a packet is accepted, $\alpha^{in}(n)$ and $\alpha^{out}(n)$ depicting the probabilities that IN and OUT packets are accepted, when the queue contains n packets.

$$\alpha^{in}(n) = \begin{cases} 1 & \text{if } n < minTh_{in} \\ 1 - Pmax_{in}\dfrac{n - minTh_{in}}{maxTh_{in} - minTh_{in}} & \text{if } minTh_{in} \le n \le maxTh_{in} \\ 0 & \text{if } n > maxTh_{in} \end{cases} \tag{10}$$

$$\alpha^{out}(n) = \begin{cases} 0 & \text{if } n < minTh_{out} \\ 1 - Pmax_{out}\dfrac{n - minTh_{out}}{maxTh_{out} - minTh_{out}} & \text{if } minTh_{out} \le n \le maxTh_{out} \\ 1 & \text{if } n > maxTh_{out} \end{cases} \tag{11}$$

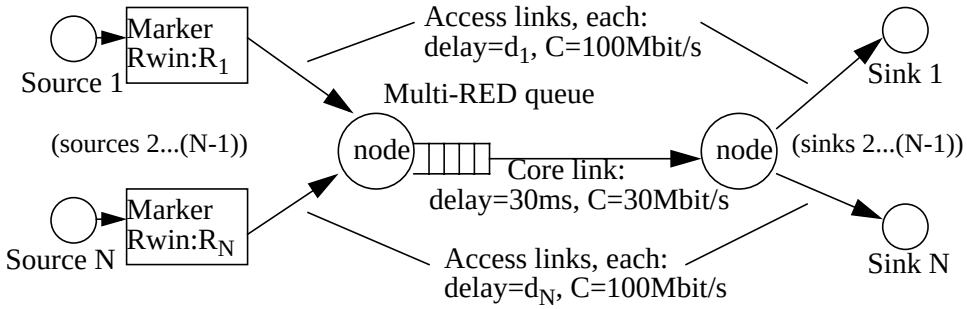

Figure 3: Network model.

5 Network Model

The investigated network structure is depicted in Fig. 3. All sources are "greedy" TCP sources, the round-trip time for connection i is $RTT_i = 2(30\text{ms} + 2d_i)$. TCP connections may have different reservation windows R_i, and different round- trip times RTT_i, therefore creating different marking probabilities $p_{m,i}$, and mean bit rates B_i. Round-trip times are approximated as constant values. Total input packet rate λ of the queueing system and total marking probability p_m are given by

$$\lambda = \frac{1}{k}\sum_i B_i, \text{ and } p_m = \sum_i B_i p_{m,i} \,/\, \sum_i B_i . \tag{12}$$

The queueing model derives dropping probabilities p_{IN}, p_{OUT} from total input rate and marking probability. We assume that all TCP connections experience the same mean packet loss probability as given by equ. (3). This is justified by the operation of RED queues which randomly drop packets from different connections. Only if the current packet rate of a connection constitutes a too large fraction of the total packet rate, the connection in question can modulate the loss process.

It seems to be natural to find the steady-state solution by an iteration, starting with a given pair of values for p_{IN}, p_{OUT} (see [5] for a non-DiffServ network). This poses two problems. Firstly, it has to be ensured that the iteration does find the equilibrium system state (two unknowns). In this paper, the problem is circumvented by restriction to "undersubscribed" networks, where $p_{IN} \to 0$. Nevertheless, a simple iteration – as applicable to find the steady-state state probabilities in a discrete-time Markov chain – often cannot find the steady-state solution. We applied a binary search in order to calculate the equilibrium value of p_{OUT}:

$$p_{OUT} = f_1(f_2(p_{OUT})) \quad \to \quad f_1(f_2(p_{OUT})) - p_{OUT} = 0 . \tag{13}$$

In equ. (13), $f_1(.)$ is the transfer function of the queueing model (equ. (9)), and $f_2(.)$ relates to equ. (12). The binary search is stopped, if none of the throughputs B_i is changing by more than a fraction of ε (examples in Section 6 use $\varepsilon = 0.01$). This approach also might be suitable for a solution with regard to the single unknown p_L (instead of the pair p_{IN}, p_{OUT} of unknowns) in the oversubscribed case.

6 Numerical Results

For the examples, $N = 50$ sources are connected over uniform access delays of $d_i = 30\text{ms}$ each, which yields uniform round-trip times of 180ms. The TSW markers are set to committed information rates of $CIR_i = (i - 0.5)\ 20\ \text{kbit/s}, 1 \le i \le 50$. The total reserved rate thus is 25Mbit/s, creating an undersubscribed core link. All TSW windows are set to $T_i = 1\text{s}$ (default in *ns-2* and used, e.g., in [14]) which leads to a marking behaviour "short TSW window". Packet size is $k = 8000\text{bit}$. Fig. 4 shows a comparison between analysis and simulation. The ladder results have been obtained with different TCP variants Reno, Newreno and Sack [7], confidence level is 95% .

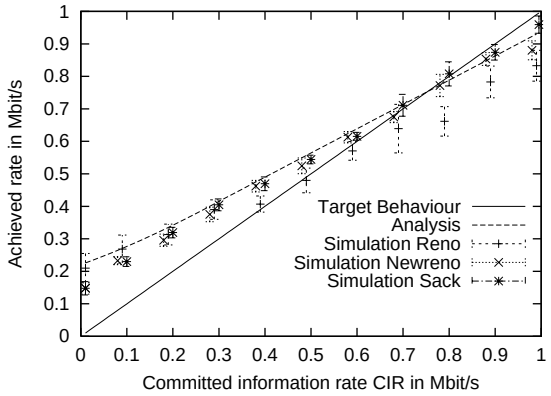

Figure 4: Achieved bit rates with short TSW window ($T_i = 1\text{s}$).

We observe two differences between analysis and simulation. For all connections with low CIR, analysis predicts a higher throughput than simulation. These connections send a high fraction of OUT packets, thus suffering from a higher p_L and therefore higher TO probability (burst losses at high p_L). The analytical model does not include TO behaviour. For connections with high CIR, analysis and simulation for Newreno / Sack match quite well. Closer review shows that TCP Reno also produces frequent TOs. We conclude that the analysis yields good accuracy if TOs are rare.
Fig. 4 also shows that connections with high CIR cannot achieve their target rates. According to equ. (1), high rates require a low packet loss probability. Thus, these connections cannot use OUT packets. There are two possibilities to avoid sending of OUT packets: a) long TSW windows, or b) marking packets as OUT if the peak rate (instead of the mean rate) of TCP's "sawtooth" is exceeded ($R = 4/3\ CIR\ RTT/k$).
Fig. 5 presents results from analysis and simulation of a) and b) with TCP Sack. All connections achieve their target rates, but both solutions introduce drawbacks. Marking with respect to TCP's peak bit rate requires that the additional bit rate is reserved, since users can continuously send IN packets at this rate (worst case). Long TSW windows allow emission of large bursts which requires large node buffers. We assume that a marking scheme like dual token bucket could avoid these problems.

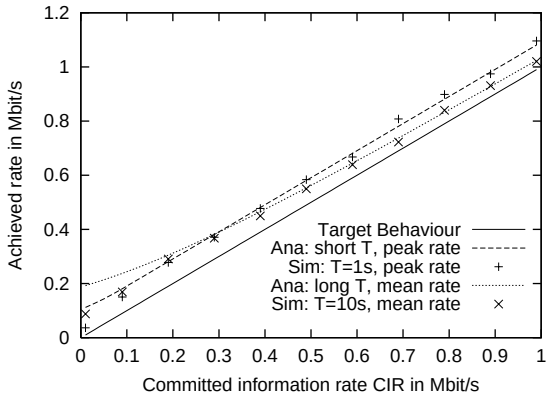

Figure 5: Achieved rates with peak rate marking / long TSW window (TCP Sack).

7 Conclusions

In this paper, we have presented a numerical method to analyse steady-state TCP bit rates in DiffServ networks. The results show reasonable accuracy if TOs are rare. The analysis is fast (e.g. approx. 10s run-time compared to 20min for simulation with small confidence intervals). We show that either large TSW window sizes or measuring against TCP peak rate are necessary in order to guarantee QoS for TCP connections with high CIR. We see the following topics for further work: investigation of networks with inhomogeneous RTT, and modelling of dual token buckets.

References

[1] S. Blake, D. Black, M. Carlson, et al. RFC 2475 – An architecture for differentiated services. IETF, 1998.

[2] R. Braden, D. Clark, and S. Shenker. RFC1633 – Integrated Services Architecture. IETF, June 1994.

[3] D. Clark and W. Fang. Expl. allocation of best-effort pkt delivery. IEEE/ACM ToN, p.362-373, Aug 1998.

[4] Y. Chait, C. V. Hollot, and V. Misra. Providing Throughput Differentiation for TCP Flows Using Adaptive TwoColor Marking and Multi-Level AQM. In Proceedings of IEEE Infocom 2002, New York, June 2002.

[5] Claudio Casetti and Michela Meo. A new approach to model the stationary behavior of TCP connections. In Proceedings of Infocom 2000, pages 367-375, Tel Aviv, March 2000.

[6] K. Fall and S. Floyd. Comparison of Tahoe, Reno and Sack TCP, 1995.

[7] S. Floyd and K. Fall. Promoting the use of end-to-end cong. contr. IEEE/ACM ToN, p.458-472, Aug 1999.

[8] P. Kuusela, P. Lassila, J. Virtamo, and P. Key. Modeling RED with idealized TCP sources. In 9th IFIP Conference on Performance Modelling and Evaluation of ATM & IP Networks 2001, Budapest, 2001.

[9] M. May, J.-C. Bolot, A. Jean-Marie, and C. Diot. Simple performance models of differentiated services schemes for the Internet. In Proceedings of IEEE Infocom 1999, New York, 1999.

[10] V. Misra, W.-B. 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 ’00, Stockholm, 2000.

[11] N. C. Malouch and Z. Liu. On steady state analysis of TCP in networks with differentiated services. In Proceedings of ITC-17 2001, Salvador da Bahia, December 2001.

[12] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose. Modeling TCP throughput: a simple model and its empirical validation. In Proceedings of ACM SIGCOMM 98, 1998.

[13] S. Sahu, P. Nain, D. Towsley, et al. On achievable service differentiation with token bucket marking for TCP. Technical report, UMASS CMPSCI 99-72, 1999.

[14] I. Yeom and A.L. Narasimha Reddy. Modeling TCP behavior in a differentiated services network. IEEE/ACM Networking, 9(1):31-46, February 2001.