Extending the TCP Steady-State Throughput Equation for Parallel TCP Flows

Dragana Damjanovic *, Michael Welzl *, Miklos Telek † and Werner Heiss *
*Institute of Computer Science, University of Innsbruck, Austria
dragana.damjanovic|michael.welzl|werner.heiss@uibk.ac.at
† Budapest University of Technology and Economics, Hungary
telek@hit.bme.hu

***Abstract*—We present a simple extension of the well-known TCP steady-state throughput equation by Padhye et al. which can be used to calculate the throughput of several flows that share an end-to-end path at the same time. The value of this extension, which we show to work well with validations using ns-2 simulations as well as real-life measurements, is its applicability in practice and its ease of use.**

I. Introduction

Almost a decade after its publication in [1], the steady-state throughput equation of TCP by Padhye et al. remains the most widely used method for calculating the throughput that a TCP sender obtains under certain environment conditions. While there is now a wealth of other models available (e.g. [2], [3], [4], [5]), many of which are better in some aspect, none of them seems to strike the same balance between precision and ease of use that makes the equation from [1]the useful tool that it is.

Recently, some effort was made to model a number of TCP flows sharing a bottleneck instead of focusing on just a single one. Examples of such models are [6], [7], [8], [9], [10], [11], [12] and [13]; their practical use is however limited because they assume that the flows under consideration are alone in the network. This is mainly due to loss being expressed as a function of the previous rate of these flows. To the best of our knowledge, [14] is the only model which precisely captures the behavior of a number of TCP flows in the presence of (loss from) other traffic. Being a dynamic model based on differential equations, it is hard to use in practice.

In an effort to enable practical calculation of the throughput of several TCP flows across a real end-to-end Internet path, we present an extension of the equation from [1] to multiple flows. We do this by following the basic approach in [1], but considering a number of senders using an identical path at the same time instead of a single one. It should be noted that end-to-end flows between the same two hosts can sometimes traverse different paths, e.g. when traffic engineering with Equal-Cost Multi-Path (ECMP) is applied; such situations are not captured by our model.

To further strengthen the need for a new model we give two ideas of possible applications. Firstly, the equation from [1] can be used to predict TCP throughput [15]; on the other hand, to predict the throughput of GridFTP [16], which uses several TCPs at the same time to better saturate the network, a new equation which considers parallel flows is needed. Such predictions are especially useful in Grids, where they can be used as a basis for making good scheduling decisions.

Secondly, the equation from [1] is used to calculate the maximum sending rate in the TCP-friendly Rate Control protocol (TFRC) [17]. This leads us to an idea of using our own equation to build a protocol that is as aggressive as n TCP flows, similar to the mulTCP protocol [18] (mulTCP emulates the behavior of n TCP senders). mulTCP is used to differentiate the rates that individual end users in a network should obtain. With a modified TFRC, the same could possibly be achieved with a smooth rate for multimedia streams.

Like the equation from [1], our equation needs the RTT, the number of packets acknowledged per ACK and a loss probability as input. We can distinguish two ways of measuring loss of n parallel flows: 1) a loss rate measured on a per-flow basis (a loss event in any flow is counted), and 2) a loss rate of the cumulative flow (a loss event of the cumulative flow is counted, making no distinction between flows). For example, if two parallel flows experience a loss event during an RTT, two loss events would be counted in the first case whereas only one loss event would be counted in the second case.

We present a first version of our equation which uses per-flow loss measurements as an input in Sections II and II-A. Validation results from simulations are shown in Section III. Measuring loss on a per-flow basis is quite demanding, as flows need to be observed separately. This is not even possible in the TFRC example, where there is only one cumulative flow. This motivated us to develop a refinement of our model which uses a per-aggregate instead of a per-flow loss rate. This equation is presented in Sections IV and IV-A, and corresponding validation results from simulations and real-life measurements are shown in Sections V and VI. Section VIII concludes the paper after a recapulatory comparison of our equations with the Padyhe model in Section VII.

II. The model using a per-flow loss measure

In order to derive an equation for the throughput of several parallel TCP flows we extend the model presented in [1]. We assume that the reader is familiar with this paper and therefore will only repeat preliminary assumptions where needed and shortly repeat necessary definitions.

Consider n parallel TCP flows $f_1, ..., f_n$ starting at time $t = 0$. As in [1] we model TCP's congestion avoidance phase in terms of "rounds", assuming furthermore that the flows are synchronized in terms of rounds (i.e. in a round all flows send their current window size W_f before the next round starts for all flows, see also Figure 1).

For any given time $t \geq 0$ we define N_t as the number of packets transmitted by all flows in the interval $[0, t]$. Let $B_t := N_t/t$ be the cumulative throughput of all flows in that interval. Then we can define the long term steady-state throughput

$$B := \lim_{t\to\infty} B_t = \lim_{t\to\infty} \frac{N_t}{t}.$$

Let W be the cumulative window size of all flows and b be the number of packets acknowledged by a received ACK. TD denotes a "triple duplicate" acknowledgment, i.e. the receipt of four ACKs with the same sequence number. In this section we only consider TD events as loss indication, meaning that the flows stay in the congestion avoidance phase. In the next section we will extend the equation with time-outs.

We assume that all flows share the same path and have the same average RTT. They are mixed (i.e. in a round packet 1 belongs to f_1, packet 2 to f_2, and so on). This is shown in Figure 1. Whenever a flow f experiences a TD loss indication, it halves its window size W_f. For n parallel flows we define a TD-period (TDP) as a period between two consecutive TD loss indications in any flow. We assume that in each TDP just one flow experiences loss. p is defined as the probability that a packet is the first packet lost in a loss event of any flow (p is the probability of a loss event of any flow).

Note that we do not take the same assumption as in [1] (if a packet is lost, all consecutive packets belonging to the same window are lost too). We believe that this decision is well justified: from [19], it is known that the number of packets that are usually lost in a row in the Internet is limited ("single packet losses account for the large majority of the bursts, around 83%, and double losses occupy 12% of the bursts"). Other than the authors of [1], we model multiple flows, which are more aggressive than a single flow, giving their sum (the cumulative flow) a better chance to attain a large window size. This would amplify errors introduced by wrongly assuming a large cluster of packets to be lost. Finally, since we assume packets from flows to be mixed in a round-robin fashion, the probability for a cluster of consecutively dropped packets to affect a single flow decreases as the number of flows increases.

For a period TDP_i let Y_i be the number of packets sent in that period, A_i be the duration of that period and X_i be the number of rounds in that period. It can be shown ([1]) that

$$B = \frac{E[Y]}{E[A]} \ . \tag{1}$$

To derive B we need to take a closer look at how the evolution of the windows size of each flow (W_f), the time between two loss events of a flow (A_f) and the duration of a TD-period of each individual flow influence the development of the cumulative window size (W).

At the end of a TD-period of the cumulative flow just one flow experiences loss, so a flow will not experience loss events at the end of each TD-period. If we assume that loss occurs identically distributed over all flows, the probability that a

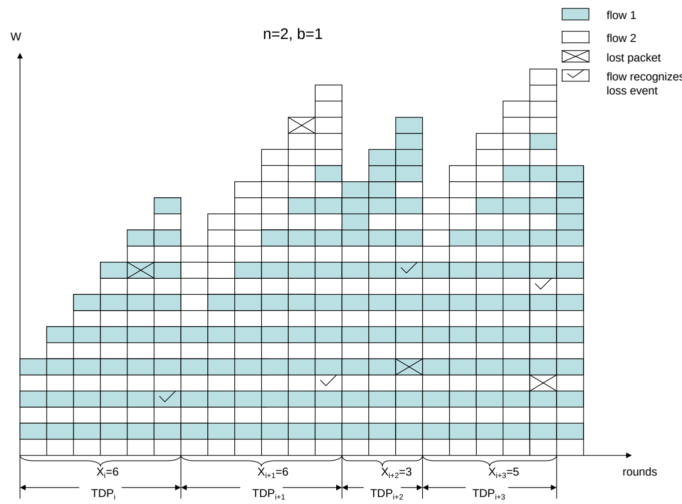


Fig. 1. Triple Duplicate Periods ($TDPs$)

flow experiences loss at the end of a TD-period is $1/n$. The probability that the time between two loss events of a flow (A_f) is k TD-periods ($k = 1, 2, ...$) is equal to the probability that the flow did not lose a packet in $(k-1)$ consecutive TD-periods and in the k-th TD-period it loses a packet:

$$P[loss\ in\ the\ k-th\ TDP] = \frac{1}{n}(1 - \frac{1}{n})^{k-1}. \tag{2}$$

The mean of A_f is:

$$\begin{aligned} E[A_f] &= \sum_{k=1}^{\infty} \left(\frac{1}{n} \left(1 - \frac{1}{n}\right)^{k-1} kE[A] \right) \\ &= \Big(nE[A] \Big) \ . \end{aligned} \tag{3}$$

As in [1], if we consider the duration of each round to be a random variable independent of the window size with the average value RTT and $E[X]$ is the average number of rounds in a TD-period, we have $A_i = \sum_{j=1}^{X_i} d_{ij}$ (d_{ij} is the duration of the j-th round in the i-th TD-period) and:

$$E[A] = E[X]RTT \ . \tag{4}$$

For deriving $E[Y]$ we will examine the evolution of the cumulative window (W) on the sender side, shown in Figure 2. In each round W is incremented by n/b, hence the number of packets sent per round is incremented by n every b rounds. α_i denotes the sequence number of the first packet lost in TDP_i (for simplicity, we assume sequence numbers to begin at 1 for every TD-period). After receiving a triple duplicate acknowledgment a flow recognizes that loss event (receiving the ACK for packet γ_i). We consider that a TD period ends when a flow recognizes a loss event. This can happen in the same round or in the next one; we call this round the "loss round". The total number of packets sent in X_i rounds in TDP_i is $Y_i = \gamma_i$, hence

$$E[Y] = E[\gamma] \ . \tag{5}$$

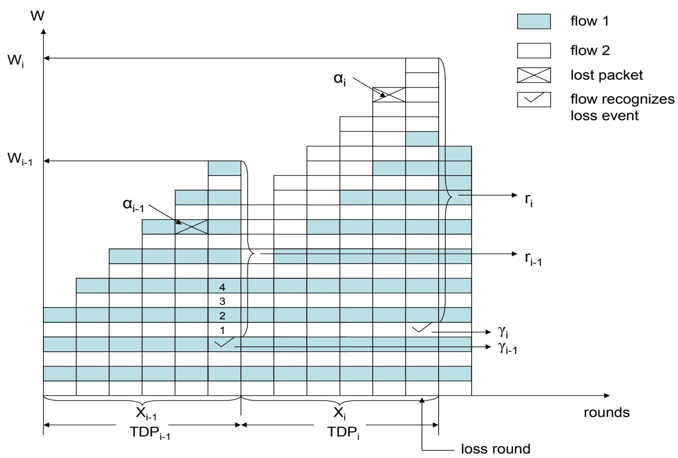


Fig. 2. A Triple Duplicate Period (TDP)

The probability that $\gamma_i = k$ is equal to the probability that $k-1$ packets are not loss indications and the ACK of the k-th packet triggers the fast retransmission:

$$P[\gamma_i = k] = (1-p)^{k-1}p, k = 1, 2, ... \tag{6}$$

And the mean value of γ is:

$$E[\gamma] = \sum_{k=1}^{\infty}(1-p)^{k-1}pk = \frac{1}{p} \tag{7}$$

Looking at the evolution of the window size W we have to assume that in the loss round of each TD-period one flow experiences loss. For the $(i-1)$th TD-period, let this flow be flow m. Flow m does not experience loss in every TD-period, so the TD-periods in which flow m experiences loss are a subset ($\{i_s\}, s = 1, 2...$) of TD-periods of the cumulative flow. For example, in Figure 1 the subset of TD-periods for flow 1 is TDP_i, TDP_{i+2} and for flow 2 it is $\{TDP_{i+1}\}$, $\{TDP_{i+3}\}$.

If $W_{f_{m_{i_s}}}$ is the window size of flow m at the end of the (i_s)th TD-period, and $X_{f_{m_{i_s}}}$ is the number of rounds from the end of $TDP_{i_{s-1}}$ till the end of TDP_{i_s}, during $X_{f_{m_{i_s}}}$ rounds the window size of flow m increases by $\frac{X_{f_{m_{i_s}}}}{b}$ (as in ([1])) and we have:

$$W_{f_{m_{i_s}}} = \frac{W_{f_{m_{i_{s-1}}}}}{2} + \frac{X_{f_{m_{i_s}}}}{b} \ . \tag{8}$$

The number of packets sent in a TD-period is the number of packets sent between two loss events. For the i-th TD-period this includes packets sent in the last round of the $(i-1)$th TD-period, starting from the (γ_{i-1})th packet till the end of the window (r_{i-1} packets) and packets sent in next X_i rounds till the (γ_i)th packet. If flow m is the flow that experiences loss in the $(i-1)$th TD-period and $W_{f_{m_{i-1}}}$ is the window of the flow at the end of the $(i-1)$th TD-period, the window size of the cumulative flow at the beginning of the i-th TD-period is $W_{i-1} - \frac{W_{f_{m_{i-1}}}}{2} + (n-1)$ (flow m reduces its window and the remaining $(n-1)$ flows increase their window size by 1). Every b rounds the window size of the cumulative flow is increased by n and we have:

$$Y_i = r_{i-1} + \sum_{k=0}^{\lfloor X_i/b \rfloor -1}\left(W_{i-1} - \frac{W_{f_{m_{i-1}}}}{2} + (n-1) + nk\right)b - r_i \tag{9}$$

where r_i is the number of packets sent in the loss round after the loss event is recognized.

Assuming that $\{X_{f_{m_{i_j}}}\}$ and $\{W_{f_{m_{i_j}}}\}$ are mutually independent sequences of i.i.d. random variables (as in [1]), from (8) we have:

$$E[W_{f_m}] = \frac{2}{b}E[X_{f_m}]. \tag{10}$$

Taking into account the assumption that a loss occurs identically distributed over all flows we can say that $E[X_{f_m}]$ and $E[W_{f_m}]$ are equal for all flows and from now on we denote them by $E[X_f]$ and $E[W_f]$. From (3) and (4) we have:

$$E[A_f] = nE[X]RTT \tag{11}$$

and the average number of rounds between two loss events of a flow ($E[X_f]$) is

$$E[X_f] = nE[X] \ . \tag{12}$$

Assuming that at the end of each TD-period the window size of a flow experiencing loss is $E[W_f]$ and the window sizes of flows experiencing loss in previous loss events are: $\frac{E[W_f]}{2} + \frac{E[X]}{b}$, $\frac{E[W_f]}{2} + \frac{2E[X]}{b}$ and so on, the mean value of the cumulative window size of all flows is:

$$E[W] = E[W_f] + \sum_{k=1}^{n-1}\left(\frac{E[W_f]}{2} + \frac{kE[X]}{b}\right) \tag{13}$$

Considering (13), (10) and (12) we have:

$$E[W] = \frac{nE[X]}{2b} + \frac{3E[X]n^2}{2b} \tag{14}$$

From (9), assuming that a loss occurs independently distributed over the size of the cumulative window in a loss round, hence $E[r] = \frac{E[W]}{2}$, we have:

$$E[Y] = \left(E[W] - \frac{E[W_f]}{2} + (n-1)\right)E[X] + \frac{nE[X]^2}{2b} - \frac{nE[X]}{2} \tag{15}$$

and considering (5), (7), (10), (12) and (14):

$$\frac{1}{p} = \frac{3n^2E[X]^2}{2b} + \frac{nE[X]}{2} - E[X]. \tag{16}$$

Solving this equation for $E[X]$ we get:

$$E[X] = \frac{2pb - npb + \sqrt{n^2p^2b^2 - 4np^2b^2 + 4p^2b^2 + 24bn^2p}}{6n^2p} \tag{17}$$

and from (14):

$$E[W] = \frac{2pb - npb + \sqrt{n^2p^2b^2 - 4np^2b^2 + 4p^2b^2 + 24bn^2p}}{4bp} + \frac{2pb - npb + \sqrt{n^2p^2b^2 - 4np^2b^2 + 4p^2b^2 + 24bn^2p}}{12bnp} \tag{18}$$

Finally, from (1), (5), (7), (4) and (17) we have:

$$B = 1/RTT \cdot \frac{6n^2}{(2pb - npb + \sqrt{n^2p^2b^2 - 4np^2b^2 + 4p^2b^2 + 24bn^2p})} \tag{19}$$

A. Model with time-outs

We now extend the equation to include time-outs (TO). A loss event experienced by a flow can be a triple-duplicate loss indication or a time-out loss indication, so in a TD-period a flow can be in the slow start phase or in the congestion avoidance phase. We denote with Y_{TDP_i} the number of packets sent by flows that are in the congestion avoidance phase in the i-th TD-period and with R_{TO_i} the number of packets sent by flows that are in the slow start phase in the i-th TD-period. The long term steady-state throughput B_{ext} is:

$$\begin{aligned} B_{ext} &= \frac{E[Y_{TDP}] + E[R_{TO}]}{E[A]} \\ &= E[B_{TDP}] + \frac{E[R_{TO}]}{E[A]} \\ &= E[B_{TDP}] + \frac{E[R_{TO}]}{E[X]RTT} \end{aligned} \tag{20}$$

where $E[B_{TDP}]$ is the throughput of flows that are not in the slow start phase.

We assume that during a TD-period most of the flows are in the congestion avoidance phase and the number of flows in the slow start phase is considerably smaller. We note that this may not be correct when loss is very large, but then, even if most of the flows were in the congestion avoidance phase, they would significantly reduce their window size, and the small total window of the cumulative flow would render the error introduced from this wrong assumption negligible. Taking this assumption we have:

$$E[B_{TDP}] = B\frac{n - E[nTO]}{n} \tag{21}$$

where B is defined in (19) and $E[nTO]$ is the average number of flows that are in the slow start phase. The second part of (20) ($\frac{E[R_{TO}]}{E[X]RTT}$) is the throughput of flows that are in slow start and it equals: $E[nTO]\frac{E[R]}{E[Z^{TO}]}$, where $E[R]$ and $E[Z^{TO}]$, following the notation from [1], are the average number of packets sent by one flow $E[R]$ during a time-out sequence of average duration $E[Z^{TO}]$. $E[R]$ and $E[Z^{TO}]$ are defined as in [1]:

$$E[R] = \frac{1}{(1-p)}, \tag{22}$$

$$E[Z^{TO}] = T * \frac{f(p)}{1-p} \tag{23}$$

where $f(p) = 1 + p + 2p^2 + 4p^3 + 8p^4 + 16p^5 + 32p^6$ and T denotes the initial period of time (in a TO phase) after which the sender retransmits unacknowledged packets.

The main problem for calculating the throughput of parallel TCP flows including time-outs is finding the expression for $E[nTO]$, the number of flows that are in the slow start phase during one TD-period. $E[nTO]$ includes not just a flow that experiences a time-out in the current loss round, but also flows that experienced time-outs in one of the previous TD-periods and that are still in the slow start phase. Let P_{TO} be the probability that a loss ending a TD-period is a time-out loss indication. Then we have:

$$E[nTO] = P_{TO}\frac{E[Z^{TO}]}{E[X] * RTT}. \tag{24}$$

To find the expression for P_{TO} we will observe packets of a flow that experiences loss. The window size of that flow in the loss round, from (10), (12) and (17), is:

$$E[W_f] = \frac{2pb - npb + \sqrt{n^2p^2b^2 - 4np^2b^2 + 4p^2b^2 + 24bn^2p}}{3npb}. \tag{25}$$

We denote by p_{lr} the probability that one of the packets belonging to the loss round is lost. We know that because at least one of the packets of the flow's window is lost, the probability that a packet of that flow is lost in the loss round is at least $\frac{1}{E[W_f]}$. We take this as p_{lr}.

Further on we make the same DropTail loss assumption as in [1]. This may seem to be at odds with our previous assumption of loss events being independent and randomly distributed; however, in accordance with the general assumptions that are central to the design of TCP itself, we consider loss events that cause timeouts to be rare special cases where packets are consecutively dropped (TCP exhibits a Go-Back-N behavior in Slow Start because of this assumption).

The probability that the flow experiences a time-out in the loss round is equal to the probability that less than three packet are successfully sent and received in the loss round ($\sum_{k=0}^{2} A(k)$), or the probability that in the second last round more than three packets are successfully sent and received, but in the last round less than three are successfully sent and received ($\sum_{k=3}^{w} A(k) \sum_{m=0}^{2} A(m)$). Then, the probability that a loss in a window of size w is a TO is:

$$\hat{Q}_{TO}(w) = \begin{cases} 1 & w \leq 3 \\ \sum_{k=0}^{2} A(k) + \sum_{k=3}^{w} A(k) \sum_{m=0}^{2} A(m) & otherwise \end{cases} \tag{26}$$

where $A(k) = (1 - p_{lr})^k p_{lr}$ and hence:

$$\hat{Q}_{TO}(w) = min(1, (1 - (1 - p_{lr})^3)(1 + (1 - p_{lr})^3(1 - (1 - p_{lr})^{w-3})). \quad (27)$$

Considering that a flow experiencing loss has the window size given in (18) we have:

$$B_{ext} = B\frac{n - \hat{Q}_{TO}(E[W_f])\frac{E[Z^{TO}]}{E[X]*RTT}}{n} + \hat{Q}_{TO}(E[W_f])\frac{E[Z^{TO}]}{E[X]*RTT}\frac{E[R]}{E[Z^{TO}]} \quad (28)$$

where B, $\hat{Q}_{TO}(w)$, $E[Z^{TO}]$, $E[R]$, $E[X]$ are given in (19), (26), (23), (22), (17) respectively.

Finally, using $\hat{Q}_{TO}(w)$ and $E[W_f]$ defined in (27) and (25), we obtain the long term steady-state throughput of n TCP flows as a function of RTT, p and b:

$$B_{ext} = \frac{6n^2(1 + \hat{Q}_{TO}(E[W_f])\frac{p}{1-p})}{(2pb - npb + \sqrt{n^2p^2b^2 - 4np^2b^2 + 4p^2b^2 + 24bn^2p})RTT} - \frac{36n^3p\frac{T(1+p+2p^2+4p^3+8p^4+16p^5+32p^6)}{(1-p)}\hat{Q}_{TO}(E[W_f])}{(2pb - npb + \sqrt{n^2p^2b^2 - 4np^2b^2 + 4p^2b^2 + 24bn^2p})^2 RTT^2} \quad (29)$$

III. Validation

We validated the performance of the TCP model presented in the previous section using the ns-2 simulator. With simulations we showed that the presented equation works well in a broad range of conditions. Real background traffic can produce different distributions of loss events. These events include isolated packet losses, burst losses as well as variations of the specific length of the burst. All these loss distributions influence the throughput of an aggregate of TCP flows in a different way. To validate our equation we chose non-correlated loss (each packet is lost with the same probability) as well as bursty loss.

For our ns-2 simulations we used the "dumbbell" topology which is commonly used to study a set of flows sharing the same link. We varied parameters, like the loss percentage and the number of flows, to validate our model in different network conditions.

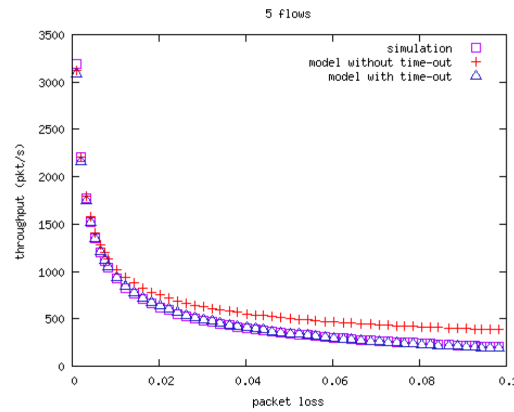
A. Simulation with non-correlated loss

In this scenario, the access links had a bandwidth of 10 Gbit/s and a delay of 1 ms, whereas the bandwidth and delay of the shared link were 1 Gbit/s and 30 ms, respectively. This way, the shared link was always the bottleneck.

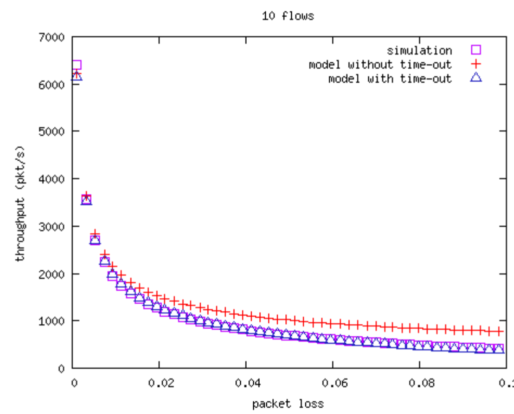
Then, we added uniform random loss to the shared link, where we additionally avoided phase effects by using a RED queue. RED queue was also used on the access links. According to the recommendation in [20], we allowed the algorithm implemented in ns-2 to automatically configure the RED parameters. The amount of loss that was generated by the loss model on the shared link is the parameter that we varied. To validate the equation with different delay, the bottleneck link delay was also varied (30ms and 100ms).

In Figure 3 it can be seen that the equation works well with a broad range of loss event percentages, from a very low loss level to a high one of 10% (even unrealistic in today's Internet). Figure 4 shows that the same good behavior scales with the number of flows. We tested it with up to 100 flows.

In the previous simulations the RTT was around 60ms; Figure 5 shows that the equation also captures the behavior of TCP well when the RTT is longer (around 200ms).



(a)

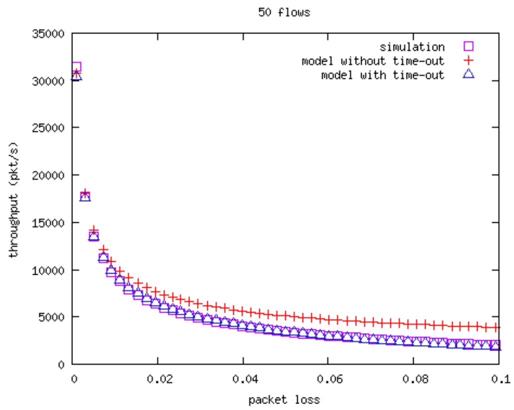


(b)

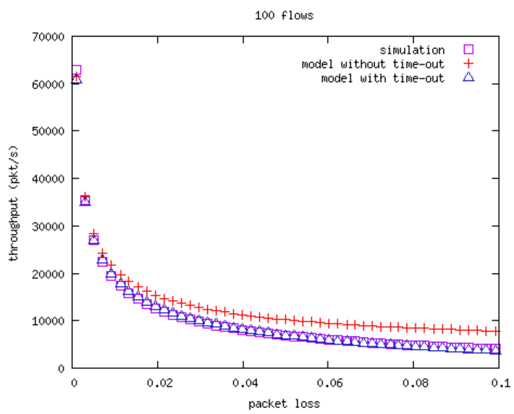
Fig. 3. The model with per-flow loss measure: random loss (each packet is lost with the same probability), 30ms delay on the shared link

B. Simulation with bursty loss

In the second part of our simulations, a DropTail queue was used on the shared link, where the queue length was set to the bandwidth × delay product. Access links had a delay of 1ms and the capacity of 1Gbps. The bottleneck link had a delay of 30ms and the bottleneck capacity was changed to cause a varying amount of loss. It had the values 1Mbps, 2Mbps, 4Mbps, 8Mbps, 16Mbps, 32Mbps, 64Mbps, 128Mbps and 200Mbps. Notably, using different bottleneck capacities influences the RTT (by 26.7% in the 1 Mbit/s case and 0.1% in the 256 Mbit/s case); this makes our result, shown in Figures 6 and 7, somewhat similar to a real-life test where packet loss is a measured parameter.

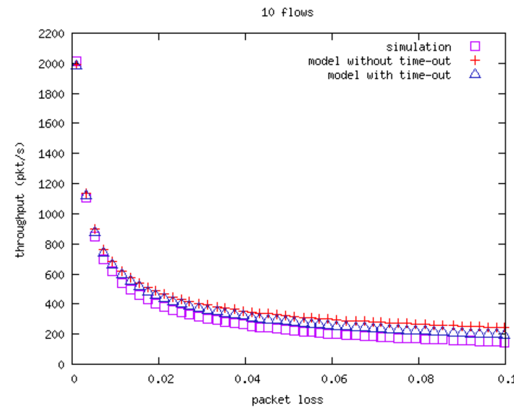


(a)

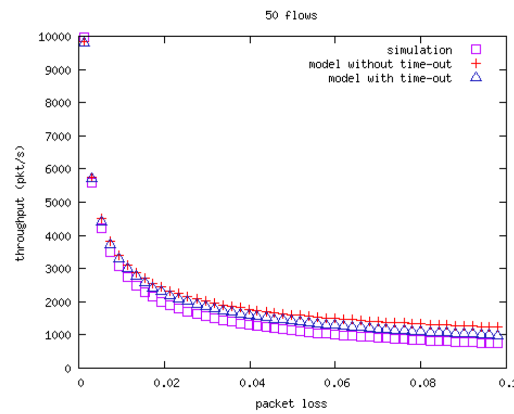


(b)

Fig. 4. The model with per-flow loss measure: random loss (each packet is lost with the same probability), 30ms delay on the shared link



(a)



(b)

Fig. 5. The model with per-flow loss measure: random loss (each packet is lost with the same probability), 100ms delay on the shared link

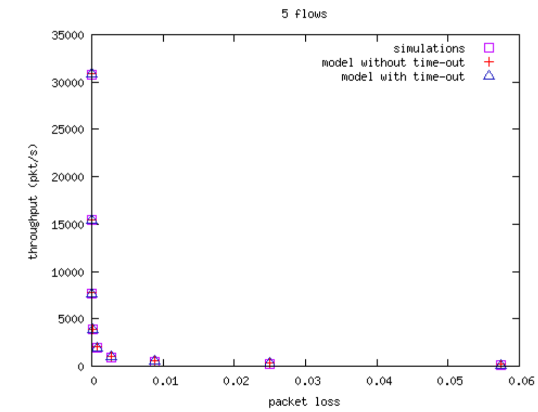
Figures 6 and 7 show that our equation also works well with bursty loss. In Figure 7 we can see that it slightly overestimates the throughput for 100 flows. With bursty loss, the probability that multiple packets are lost in a loss event is higher, and therefore the assumption that we made for calculating the fast-retransmission phase (in a loss event just one packet is lost) is incorrect. This is the effect that we see in this figure. To be able to more precisely capture such a situation, we also need information about real loss. In our second model, besides another refinement, we explore this idea too.

IV. The Model with a Loss Measure of the Cumulative Flow

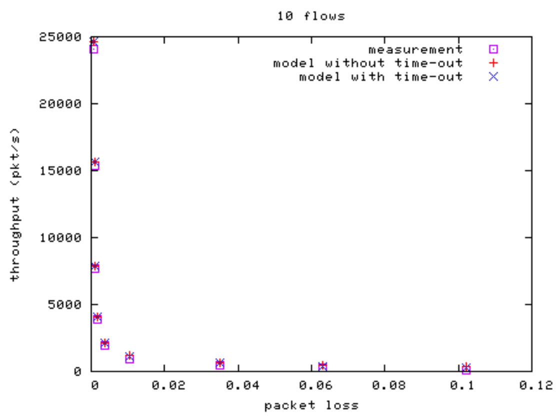
The equation from Section II-A estimates the throughput of n parallel flows sharing the same path, and for this calculation the probability for any flow among these n flows to experience a loss event is needed. To calculate this probability, loss needs to be measured on a per-flow basis. We now explore the possibility of having the same calculation, but using the loss event probability of the cumulative flow. This eliminates the need to make a distinction between flows and can make the measurements significantly easier.

To remind the reader, TCP reduces the congestion window just once in a round-trip time, therefore, just the first lost packet in a round is a loss event indicator. Therefore, for measuring the loss event probability, all consecutive losses after a loss event indicator that are less than round-trip time away from it are neglected. The same principal is used for the loss event measure of the cumulative flow, therefore in a loss event of the cumulative flow more than one real flow can experience a loss event. For a higher loss rate and a larger number of considered flows two loss measuring methods will differ in such a way that a per-flow loss measurement would have a higher rate. Figure 8 shows a comparison of the measurements and the results produced by the equation from Section II-A, using the loss event probability of the cumulative flow instead of the probability for loss to happen in one of the flows, as in Figure 4. As it can be seen, the equation from Section II-A(a) needs to be changed so that it can be used with the loss probability of the cumulative flow.

In this extended equation we will include not just knowledge about the loss event probability but also knowledge about the real loss probability (the probability that a packet is lost). In a loss event of the cumulative flow, one or more flows can experience loss, and here we use information about the real loss probability for estimating how many flows experience loss in one such loss event. As already mentioned, knowledge about how many packets are lost in a loss event would enable a more

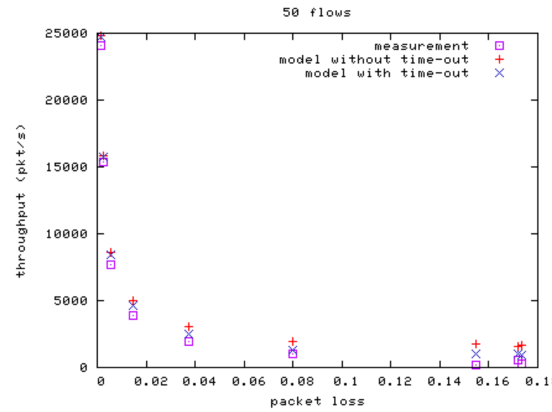


(a)

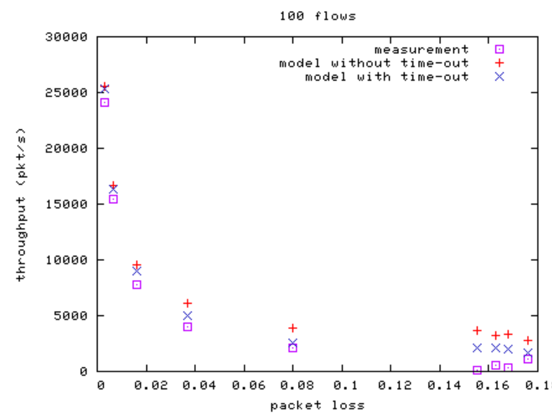


(b)

Fig. 6. The model with per-flow loss measure: DropTail queue, 30ms bottleneck delay



(a)



(b)

Fig. 7. The model with per-flow loss measure: DropTail queue, 30ms bottleneck delay

precise calculation of the timeout probability. Having this extra information leads us to further refine the timeout part of our equation.

For the new equation, p_e is the probability of a loss event of the cumulative flow. It is only counted as one loss event when one or more flows experience loss. With p_r we denote the probability that a packet (belonging to any flow) is lost.

As in Section II, we will observe the development of the window size over a TD-period, where in this case a TD-period is a period between two loss events of the cumulative flow. For the i-th TD-period, Y_i is the number of packets sent in the period, A_i is the duration of the period, and X_i is the number of rounds in the period. As in Section II, we will derive the expression for B starting from equation (1).

The expression for calculating the average duration of a TD-period, expression (4), is not modified. Equation (7), the equation for the number of packets sent in a TD-period, is almost the same. In this case we calculate the number of packets sent between two loss events of the cumulative flow, and therefore we use the loss event probability of the cumulative flow (p_e). We have:

$$E[A] = E[X]RTT \tag{30}$$

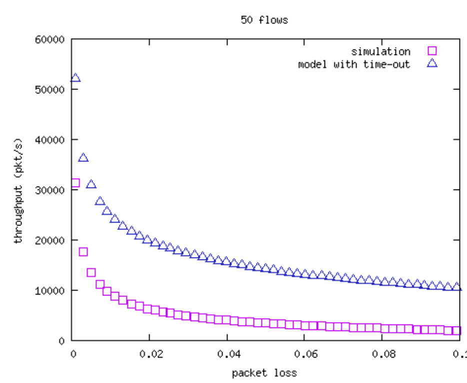


Fig. 8. The equation from Section II-A and the loss event probability of the cumulative flow

and

$$E[Y] = \frac{1}{p_e} \; . \tag{31}$$

Since we now consider loss events of the cumulative flow, in a loss event more than one flow can experience loss. In Section II we assumed that in each loss event just one flow was hit and the probability that a flow was hit in a TD-period was $1/n$. Let j_i be the number of flows, belonging to the cumulative flow, that experience loss at the end of the i-th

TD-period. Assuming that loss is identically distributed over all flows, the probability that a flow is hit in the i-th TD-period is j_i/n. To calculate the probability that there are k TDP between two loss events of an individual flow we take the same assumption as in Section II (equation (2)), but with a different flow hit probability, and we have:

$$P[loss\ in\ the\ k-th\ TDP] = \frac{j_i}{n}\prod_{l=1}^{k-1}(1-\frac{j_{(i-l)}}{n}). \quad (32)$$

If j is the mean number of flows hit in a round, we have:

$$P[A_f = kE[A]] = \frac{j}{n}(1-\frac{j}{n})^{k-1}. \quad (33)$$

The mean value of A_f, the time between two loss events of an individual flow, is:

$$\begin{aligned} E[A_f] &= \sum_{k=1}^{\infty}\left(\frac{j}{n}\left(1-\frac{j}{n}\right)^{k-1}kE[A]\right) \\ &= \left(\frac{nE[A]}{j}\right). \end{aligned} \quad (34)$$

For the i-th TD-period let flows $\{m^e\}, e = 1..j_i$ (subset of n flows) be the j_i flows experiencing loss at the end of the period. As in Section II we will observe the evolution of the window size of an individual flow. Since we do not change any assumption about individual flows, from (8), the evolution of the window size of flow m^e between two loss events of the flow is:

$$W_{f_{(m^e)_{i_s}}} = \frac{W_{f_{(m^e)_{i_{s-1}}}}}{2} + \frac{X_{f_{(m^e)_{i_s}}}}{b}. \quad (35)$$

The window sizes of flows $\{m^e\}, e = 1..j_{i-1}$ are halved at the end of TDP_{i-1}. Accordingly changing equation (9), the number of packets sent by all flows in the i-th TD-period is:

$$Y_i = r_{i-1} + \sum_{k=0}^{\lfloor X_i/b \rfloor -1} b\left(W_{i-1} - \sum_{e=1}^{j_{i-1}}\frac{W_{f_{(m^e)_{i-1}}}}{2} + (n-j_{i-1}) + nk\right) - r_i, \quad (36)$$

where $W_{f_{(m^e)_{i-1}}}, e = 1..j_{i-1}$ are the window sizes of these j_{i-1} flows.

Since assumptions about individual flows are not altered, the steady state equation for an individual flow (equation (10)) is:

$$E[W_f] = \frac{2}{b}E[X_f]. \quad (37)$$

and from (30) and (34) we have:

$$E[X_f] = \frac{nE[X]}{j}. \quad (38)$$

Like in Section II, at the end of a TD-period j flows have the window size $E[W_f]$, another j flows that experience loss in the previous loss event have the window size $\frac{E[W_f]}{2} + \frac{E[X]}{b}$ etc. The mean window size of the cumulative flow is:

$$E[W] = jE[W_f] + \sum_{k=1}^{\frac{n}{j}-1} j\left(\frac{E[W_f]}{2} + \frac{kE[X]}{b}\right) \quad (39)$$

From (37), (38) and (39) we have:

$$E[W] = \frac{nE[X]}{2b} + \frac{3n^2E[X]}{2bj} \quad (40)$$

With the same assumptions as in Section II, from (36) we have:

$$E[Y] = \left(E[W] - j\frac{E[W_f]}{2} + (n-j)\right)E[X] + \frac{nE[X]^2}{2b} - \frac{nE[X]}{2} \quad (41)$$

and including (37), (38), (40) and (31):

$$\frac{1}{p_e} = \frac{3n^2E[X]^2}{2bj} + \frac{nE[X]}{2} - jE[X] \quad (42)$$

Solving this equation for $E[X]$we get:

$$E[X] = \frac{2j^2p_eb - np_ebj + \sqrt{(n^2p_e^2b^2j^2 - 4np_e^2b^2j^3 + 4j^4p_e^2b^2 + 24n^2p_ebj}}{6n^2p_e} \quad (43)$$

and including (40):

$$E[W] = \frac{2j^2p_eb - np_ebj + \sqrt{(n^2p_e^2b^2j^2 - 4np_e^2b^2j^3 + 4j^4p_e^2b^2 + 24n^2p_ebj}}{4bp_ej} \frac{2j^2p_eb - np_ebj + \sqrt{(n^2p_e^2b^2j^2 - 4np_e^2b^2j^3 + 4j^4p_e^2b^2 + 24n^2p_ebj}}{12bnp_e} \quad (44)$$

From (1), (31), (30) and (43) we have:

$$B = \frac{1}{RTT} \frac{6n^2}{2j^2p_eb - np_ebj + \sqrt{(n^2p_e^2b^2j^2 - 4np_e^2b^2j^3 + 4j^4p_e^2b^2 + 24n^2p_ebj}} \quad (45)$$

As already stated, according to [19], loss is usually not a loss burst, so the probability that in a loss event more than one packet of a flow is lost is not high. In the case of a low loss probability we can approximate j with the average number of packets lost in a loss event, and for a higher loss probability the flow rate is mainly governed by time-outs. Therefore we take the approximation: $j = \frac{p_r}{p_e}$. Since j must be less than n, we have $j = min(n, \frac{p_r}{p_e})$.

A. Model with time-outs

To include time-outs in the equation we apply the same method as in Section II-A. Taking expressions (20) and (21), we get the steady state throughput equation with time-outs:

$$B_{ext} = B\ \frac{n - E[nTO]}{n} + E[nTO]\ \frac{E[R]}{E[Z^{TO}]} \quad (46)$$

where $E[R]$ and $E[Z^{TO}]$ are defined in (22) and (23) (just instead of p we have p_e), and $E[X]$ and B are defined in (43) and (45). Since now we also include information about real loss, the expression that yields the number of flows in the slow start phase ($E[nTO]$) differs. We have $E[nTO] =$

$E[nTO'] \frac{E[Z^{TO}]}{E[X]\ RTT}$, where $E[nTO']$ is the number of flows experiencing timeouts in a loss event.

To derive $E[nTO']$, let $pLost_i$ be the number of packets lost in TDP_i and let l_i flows (flows $\{m^e\}, e = 1..l_i$) experience timeouts at the end of the TD-period. $W_{f_{(m^e)_i}}, e = 1..l_i$ are congestion window sizes of these flows at the end of TDP_i. We have $pLost_i \geq \sum_{e=1}^{l_i} W_{f_{(m^e)_i}}$. Assuming that $\{pLost_i\}$ and $\{W_{f_{(m^e)_i}}\}$ are sequences of mutually independent random variables, and taking the average window size of a flow to be $E[W]/n$, and l as the average number of flows experiencing timeouts in a loss event, we have: $pLost \geq l\ \frac{E[W]}{n}$. Since flows experiencing a triple-duplicate loss indication lose much less packets than flows experiencing a timeout, we can approximate the number of flows experiencing a timeout in a loss event as

$$nTO^1 = min(\frac{n\ pLost}{E[W]}, n)\ .$$

In a loss event the average number of packets lost is $pLost = p_r/p_e$. The congestion window size of the cumulative flow ($E[W]$) is given in (44), and the number of flows in the slow start phase in a TD-period is:

$$E[nTO] \quad = \quad min\left(\frac{n\ \frac{p_e}{p_r}}{E[W]}, n\right)\ \frac{E[Z^{TO}]}{E[X]RTT}$$

Using expressions for $E[W]$ from (40) and for $E[Z^{TO}]$ from (23), we have:

$$E[nTO] \quad = \quad min\left(\frac{n\ \frac{p_e}{p_r}}{\frac{nE[X]}{2b} + \frac{3n^2E[X]}{2bj}}, n\right)\ \frac{f(p)T}{(1-p)E[X]RTT} \tag{47}$$

where $f(p) = 1 + p + 2p^2 + 4p^3 + 8p^4 + 16p^5 + 32p^6$ and we approximate $j = min(n, \frac{p_r}{p_e})$.

Finally, the steady state throughput of n parallel flows can be calculated as:

$$B = \frac{n - nTO}{n\ p\ E[X]\ RTT} + \frac{nTO}{f(p)T} \tag{48}$$

where nTO is given in (47) and $E[X]$ is from (43).

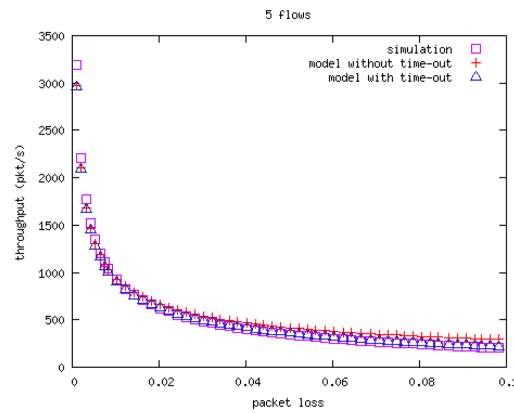
V. Validation of the extended model

For validating the extended equation we use the same simulations as before. Simulation parameters are not modified. For this version of the equation we need the probability of a loss event of a cumulative flow. We measure the loss event probability by counting just one loss event per RTT. If one flow experiences a loss event at time t_{le_1}, loss events of other flows at time $t_{le'} \leq t_{le_1} + RTT$ are ignored. The same principle for calculating the number of loss events is used in [21].

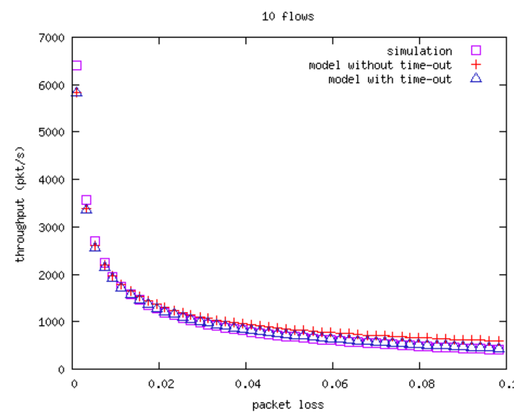
As our validations show the equation performs well with a broad range of loss probabilities. The equation slightly underestimates the throughput for a very small loss probability. The reason for this is the approximation for calculating j (the average number of flows experiencing loss in a loss event of the cumulative flow) as $j = \frac{p_r}{p_e}$, where we assume that each lost packet in a loss event belongs to a different flow. In reality more than one packet can belong to the same flow.

A. Simulation with non-correlated loss

In this set of simulations we use the same setup as in Section III-A. Figures 9 and 10 show validations with random loss and a delay of 30 ms on the bottleneck link. Figure 11 shows that the equation performs well even with a longer RTT (the bottleneck delay was 100 ms).



(a)



(b)

Fig. 9. The model with cumulative flow loss measure: random loss (each packet is lost with the same probability), 30ms delay on the shared link

B. Simulation with bursty loss

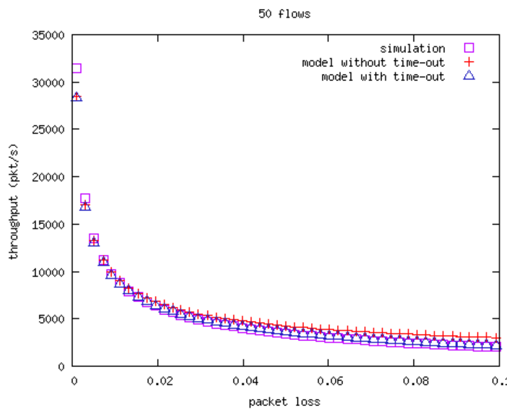
The simulation setup used in this part of the validation is the same as in Section III-B; the equation slightly underestimates the throughput in the case of a low loss probability (Figures 12 and 13) for the same reason as in Section III-B.

VI. Real-life measurements

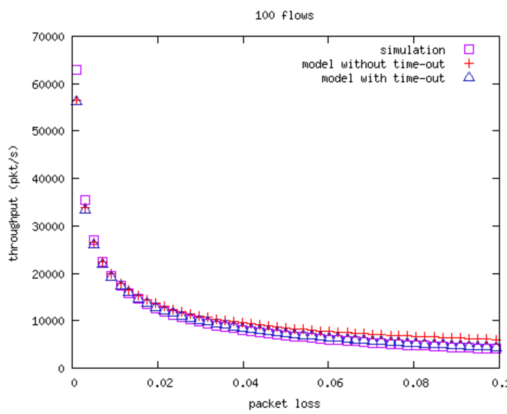
We also validated our extended equation with real-life measurements. We measured the throughput of $n = 1..10$ connections opened between two hosts. All connections started at the same time. We sent data from our site to Texas and Ireland and used web100[1], installed on our site, for monitoring.

On our site the host (Athlon64 3200+, 512 MB RAM, 100 mbit network card) had a Linux kernel version 2.6.17.1 and web100 version 2.5.11. Because of almost no loss in the

[1]http://www.web100.org/

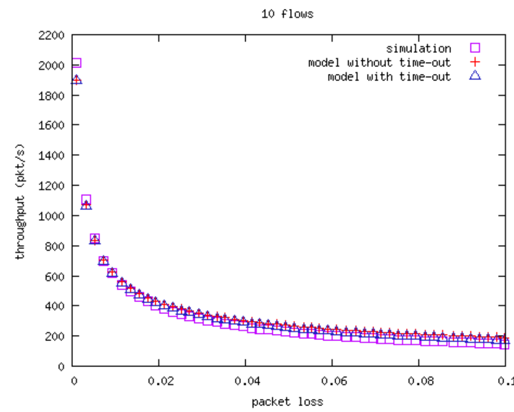


(a)

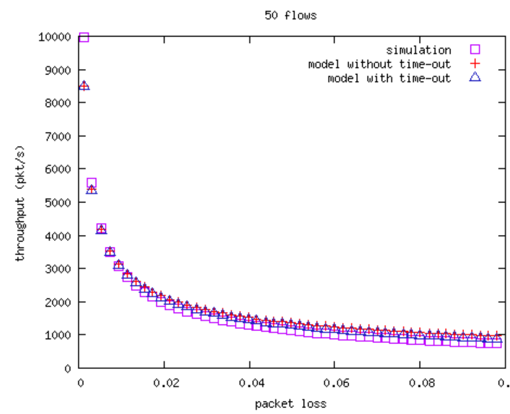


(b)

Fig. 10. The model with cumulative flow loss measure: random loss (each packet is lost with the same probability), 30ms delay on the shared link



(a)



(b)

Fig. 11. The model with cumulative flow loss measure: random loss (each packet is lost with the same probability), 30ms delay on the shared link

network we would have needed to transfer files of 1-2 GB to get sustained steady state TCP behavior. Therefore we set the network card to work with only 10 mbit/s.

The host in Ireland (149.157.192.252) ran Linux and the following TCP parameters were set:

- window scaling was turned on, and the advertised window scaling value was 12
- SACK was enabled

The measurements took from the 30th of May 2008 (16:17) till the 1st of June 2008 (17:28). We transmitted files of 70 MB using HTTP, opening n (1..10) uploads at the same time. For each number n, the measurement was run multiple times. Every transfer lasted at least 700 s (up to 1300 s). A traceroute to this host is listed below:

```
1  192.168.64.1 (192.168.64.1)  0.312 ms  0.395 ms  0.233 ms
2  sr01a.uibk.ac.at (138.232.65.126)  0.463 ms  0.472 ms  0.449 ms
3  rborder.uibk.ac.at (138.232.15.222)  3.639 ms  2.297 ms  0.983 ms
4  Ibk.ACO.net (193.171.19.1)  1.070 ms  0.740 ms  0.562 ms
5  Wien2.ACO.net (193.171.12.209)  9.859 ms  10.099 ms  10.081 ms
6  Wien21.ACO.net (193.171.23.22)  10.332 ms  10.347 ms  10.328 ms
7  aconet.rt1.vie.at.geant2.net (62.40.124.1)  10.340 ms  10.458 ms 10.211 ms
8  so-7-0-0.rt1.pra.cz.geant2.net (62.40.112.6)  16.823 ms  17.091ms  16.699 ms
9  so-6-3-0.rt1.fra.de.geant2.net (62.40.112.38)  24.955 ms  24.838ms  24.688 ms
10 so-5-0-0.rt1.ams.nl.geant2.net (62.40.112.58) 32.074 ms  32.332ms  32.194 ms
11 so-4-0-0.rt1.lon.uk.geant2.net (62.40.112.138) 40.448 ms  40.203ms  40.310 ms
12 62.40.125.126 (62.40.125.126) 51.443 ms 51.084 ms  50.874 ms
13 gige0-1-937-nuim1.cwt.client.hea.net (193.1.236.66)  51.563 ms 51.535 ms  51.675 ms
14 149.157.1.201 (149.157.1.201) 52.067 ms 51.545 ms  51.309 ms
15 149.157.1.11 (149.157.1.11) 52.188 ms 52.191 ms  52.060 ms
```

The host in Texas (129.110.241.44) ran Linux and used the following TCP parameters:

- window scaling was turned on, and the advertised window scaling value was 6
- SACK was enabled

We measured starting from the 9th of May 2008 (15:33) till the 13th of May 2008 (15:35). We performed the same set of HTTP file transfers as to Ireland. Each measurement took at least 800s (up to 2100s). A traceroute to this host is:

```
1  192.168.64.1 (192.168.64.1)  0.283 ms  0.288 ms  0.748 ms
2  sr01a.uibk.ac.at (138.232.65.126)  0.515 ms  0.908 ms  0.621 ms
3  rborder.uibk.ac.at (138.232.15.222)  2.367 ms  0.831 ms  2.147 ms
4  Ibk.ACO.net (193.171.19.1)  1.511 ms  0.971 ms  2.701 ms
5  Wien2.ACO.net (193.171.12.209)  10.053 ms  9.970 ms  10.207 ms
6  Wien1.ACO.net (193.171.23.33)  10.218 ms  10.220 ms  10.332 ms
7  vix2.core01.vie01.atlas.cogentco.com (193.203.0.113)  10.847 ms  10.465 ms
   10.460 ms
8  te1-3.ccr01.muc01.atlas.cogentco.com (130.117.3.21)  17.203 ms  17.073 ms
   16.950 ms
9  te8-3.mpd02.fra03.atlas.cogentco.com (130.117.0.165)  29.953 ms  47.325 ms
   te1-1.ccr01.str01.atlas.cogentco.com (130.117.3.77)  19.212 ms
10 te2-1.ccr01.ams03.atlas.cogentco.com (130.117.2.141)  37.323 ms
   te7-1.mpd02.fra03.atlas.cogentco.com (130.117.3.81)  29.844 ms
   te2-1.ccr01.ams03.atlas.cogentco.com (130.117.2.141)  36.958 ms
11 te3-1.ccr01.ams03.atlas.cogentco.com (130.117.2.202)  37.080 ms
   te7-1.mpd02.lon01.atlas.cogentco.com (130.117.1.118)  45.207 ms
   te1-1.ccr01.ams03.atlas.cogentco.com (130.117.1.162)  37.447 ms
12 te2-1.ccr01.lon01.atlas.cogentco.com (130.117.1.170)  45.074 ms
   te4-1.ccr04.jfk02.atlas.cogentco.com (66.28.4.253)  114.048 ms
   te1-1.ccr01.lon01.atlas.cogentco.com (130.117.1.110)  45.541 ms
13 te2-2.mpd01.ord01.atlas.cogentco.com (154.54.6.18)  140.631 ms
   te3-2.mpd01.bos01.atlas.cogentco.com (130.117.0.185)  119.275 ms
   te9-4.mpd01.ord01.atlas.cogentco.com (154.54.7.81)  141.027 ms
14 te4-1.ccr04.jfk02.atlas.cogentco.com (66.28.4.253)  114.352 ms
   te9-4.ccr02.ord01.atlas.cogentco.com (154.54.7.169)  141.781 ms
   142.155 ms
15 te2-2.ccr02.bos01.atlas.cogentco.com (154.54.5.242)  118.511 ms
   te9-4.ccr02.mci01.atlas.cogentco.com (154.54.6.214)  153.746 ms
   te9-4.mpd01.mci01.atlas.cogentco.com (154.54.7.138)  152.756 ms
16 te3-4.ccr02.dfw06.atlas.cogentco.com (154.54.0.126)  161.841 ms
   te8-4.mpd01.dfw01.atlas.cogentco.com (154.54.5.125)  162.138 ms
   te7-4.ccr02.dfw01.atlas.cogentco.com (154.54.2.113)  248.601 ms
17 te3-4.ccr02.dfw06.atlas.cogentco.com (154.54.0.126)  162.229 ms
```

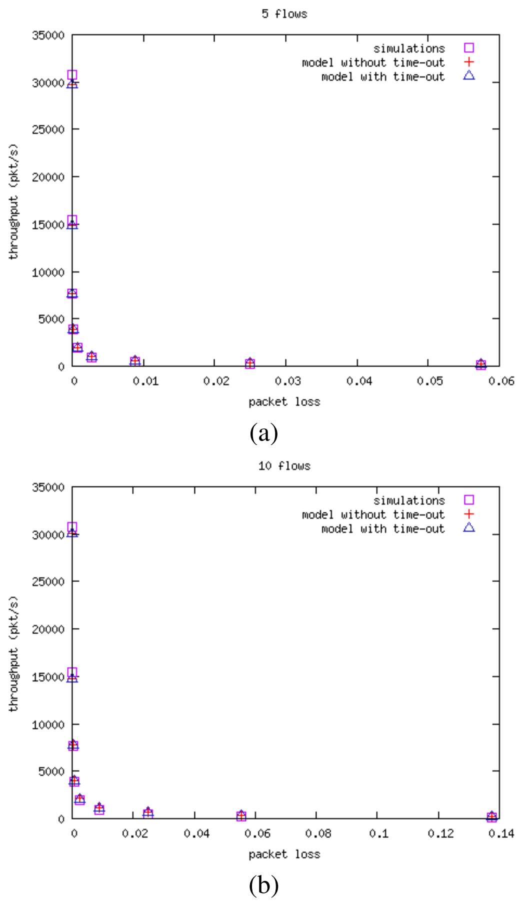


Fig. 12. The model with cumulative flow loss measure: DropTail queue, 30ms bottleneck delay

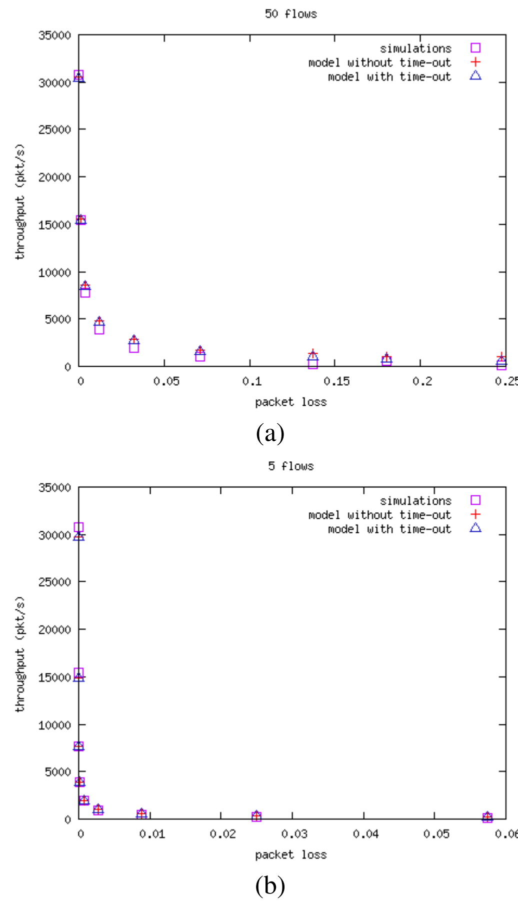


Fig. 13. The model with cumulative flow loss measure: DropTail queue, 30ms bottleneck delay

```
    te2-2.ccr02.mci01.atlas.cogentco.com (154.54.25.77)  151.646 ms
    te2-4.ccr02.dfw06.atlas.cogentco.com (154.54.0.138)  163.848 ms
18  utd.demarc.cogentco.com (38.104.34.26)  162.747 ms
    vl-431-utd-ntg-gwl.northtexasgigapop.org (208.76.224.74)  158.259 ms
    utd.demarc.cogentco.com (38.104.34.26)  162.463 ms
19  utdgw2-vl5-ge-2-2.utdallas.edu (129.110.5.71)  157.882 ms
    vl-431-utd-ntg-gwl.northtexasgigapop.org (208.76.224.74)  157.973 ms
    159.261 ms
```

The loss event probability was measured in the same way as in the case of simulations (not more than one loss event per RTT). Both figures (14 and 15) show that our equation yields a reasonably good estimate of the throughput even with real-life measurements.

VII. Comparison with the equation from [1]

Due to the simplicity of our model, it is tempting to believe that a similar result could also be obtained by simply multiplying the original equation from [1] with the number of flows n. We compared both versions of the equation. On the one hand, we have the equation from Section II-A that uses the loss event probability on a per-flow basis as the only information. On the other hand, the equation from Section IV-A use two pieces of information: real loss and the loss event probability of the cumulative flow. We made a comparison of $n\times$(the equation from [1]) with both equations, applying a corresponding loss event probability measure. We compared $n\times$(the equation from [1]) with the equation from Section II-A, feeding in the loss event probability measured on a per-flow basis, and we compared $n\times$(the equation from [1]) with the equation from Section IV-A, feeding in the loss event probability of the cumulative flow. Having these two completely different ways of measuring loss, we expect that $n\times$(the equation from [1]) calculated with these two loss measures will yield two completely different results.

For both comparisons we used the same simulation as in Section III-A (30 ms bottleneck delay and RED queue). As expected, $n\times$(the equation from [1]) works reasonably well with the measure of the loss event probability of any flow. We observed that this calculation has an error that grows with increasing loss; this is shown in Figure 16. Our model gives a good estimate of the throughput even when the loss probability is as high as 10%.

The equation from Section IV-A uses the loss event probability of the cumulative flow. As we assumed, our equation gives significantly better results than the equation from [1] multiplied by the number of flows in this case (Figure 17).

One special case worth considering is $n = 1$: here, our equation is used for only one flow just like the original equation from [1]. With the first (simpler) model, the two equations only differ in that the model from [1] assumes all packets from a window to be lost when any packet is lost, whereas we do not make this assumption for the congestion avoidance case

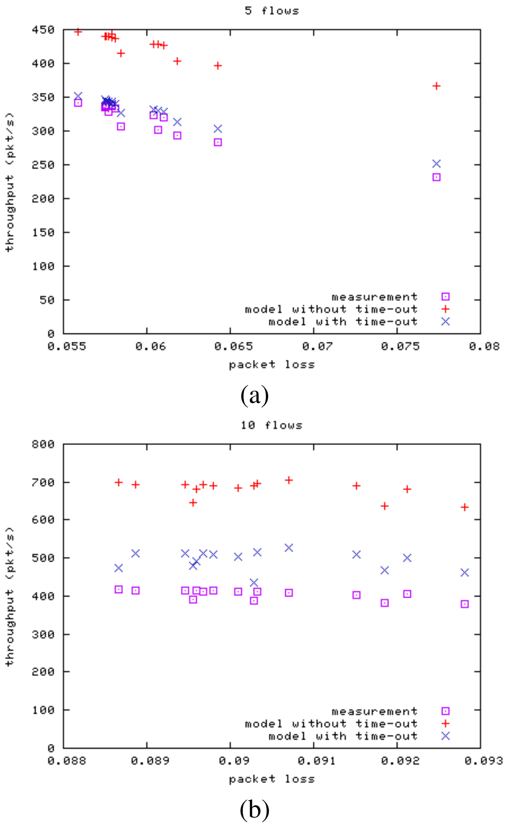


Fig. 14. Measurements Innsbruck - Ireland, using the equation with the loss probability of the cumulative flow

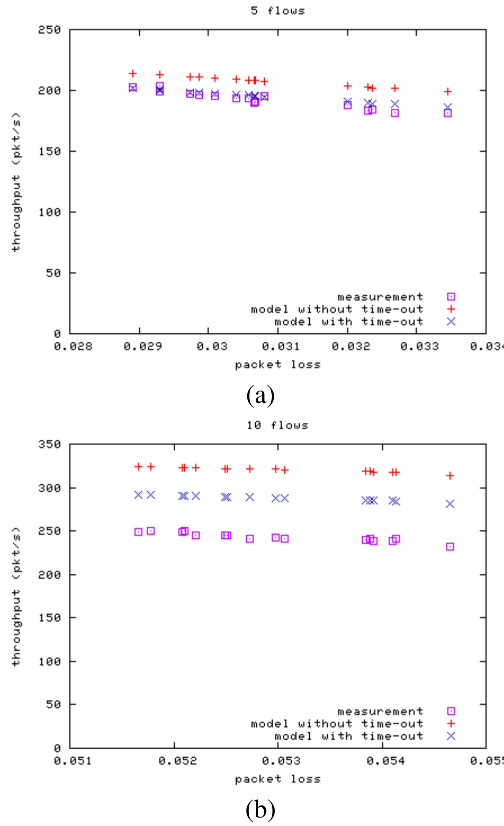


Fig. 15. Measurements Innsbruck - Texas, using the equation with the loss probability of the cumulative flow

and assume significantly clustered losses (causing timeouts) to be a rare special case. As we have argued at the outset of this paper, our assumption seems to be more realistic in today's Internet, and we found that, indeed, our model more precisely matches our real-life measurements even for the $n = 1$ case.

VIII. Conclusion

As our validations show, our equations perform well in a wide range of realistic conditions. Due to their conceptual simplicity, the possibilities for applying our models are manifold.

Arguably, equations II-A and IV-A do not look as easy to use as the model in [1] at first sight. This is however mainly due to repetitive use of the same variables, and things therefore look much easier in the form of an algorithm. The algorithms for calculating our two models are given in the appendix.

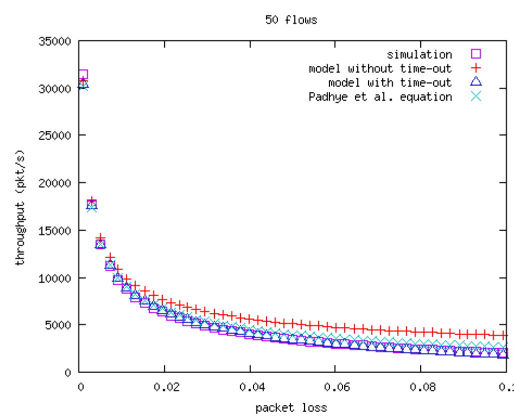


Fig. 16. Comparison: the equation from Section II-A with n×(the equation from [1])

Acknowledgments

We would like to thank Doug Leith and Andrea Fumagalli for giving us access to hosts for our measurements.

Appendix

- n - the number of flows
- p - the loss event probability measured on a per-flow basis
- b - the number of packets acknowledged by a received ACK
- RTT - round-trip time
- T - the initial period of time (in a TO phase) after which the sender retransmits unacknowledged packets
- p_e - the loss event probability of the cumulative flow
- p_r - the probability that a packet is lost

References

[1] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: a simple model and its empirical validation," in *SIGCOMM '98: Proceedings of the ACM SIGCOMM '98 conference on Applications, technologies, architectures, and protocols for computer communication*. ACM Press, 1998, pp. 303–314.

[2] E. Altman, K. Avrachenkov, and C. Barakat, "A stochastic model of TCP/IP with stationary random losses," *IEEE/ACM Trans. Netw.*, vol. 13, no. 2, pp. 356–369, 2005.

[3] V. Misra, W. Gong, and D. F. Towsley, "Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to red." in *SIGCOMM*, 2000, pp. 151–160.

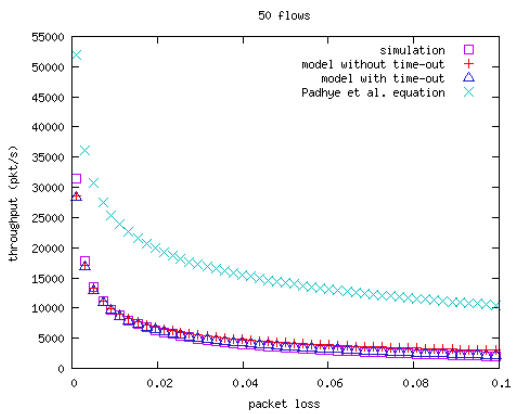


Fig. 17. Comparison: the equation from Section IV-A with n×(the equation from [1])

Algorithm 1 The model using a per-flow loss measure

Require: n, p, b, RTT, T.
Ensure: The throughput B.
$a = sqrt(p * b * (24 * n * n + p * b(n * n - 4 * n + 4)))$
$x = (p * b * (2 - n) + a)/(6 * n * n * p)$
$w = 2 * n * x/b$
$z = T * (1 + 32 * p * p)/(1 - p)$
$q1 = (1-(1-pow(1-1/w, 3)))*(1+pow(1-1/w, 3)*(1-pow(1-1/w, w-3)))$
if $q1 > 1$ **then**
  $q1 = 1$
end if
if $q1 * z/(x * RTT) \geq n$ **then**
  $q = n$
else
  $q = q1 * z/(x * RTT)$
end if
return $(1 - q/n)/(p * x * RTT) + q/(z * (1 - p))$

Algorithm 2 The model using a cumulative loss measure

Require: n, p_e, p_r, b, RTT, T.
Ensure: The throughput B.
$j = p_r/p_e$
$j1 = j$
if $j1 > n$ **then**
  $j1 = n$
end if
$a = sqrt(p_e*b*j1*(24*n*n+p_e*b*j1*(n*n-4*n*j1+4*j1*j1))$
$x = (j1 * p_e * b * (2 * j1 - n) + a)/(6 * n * n * p_e)$
$w = n * x/(2 * b)(1 + 3 * n/j1)$
$z = T * (1 + 32 * p_e * p_e)/(1 - p_e)$
$q1 = j * n/w$
if $q1 > 1$ **then**
  $q1 = 1$
end if
if $q1 * z/(x * RTT) \geq n$ **then**
  $q = n$
else
  $q = q1 * z/(x * RTT)$
end if
return $(1 - q/n)/(p_e * x * RTT) + q/(z * (1 - p_e))$

[4] F. Baccelli and D. Hong, "Interaction of TCP flows as billiards," *IEEE/ACM Trans. Netw.*, vol. 13, no. 4, pp. 841–853, 2005.
[5] Y. Liu, F. L. Presti, V. Misra, D. F. Towsley, and Y. Gu, "Fluid models and solutions for large-scale IP networks," in *SIGMETRICS*. ACM, 2003, pp. 91–101.
[6] E. Altman, D. Barman, B. Tuffin, and M. Vojnovic, "Parallel TCP sockets: Simple model, throughput and validation," in *IEEE INFOCOM*, April 2006.
[7] F. Baccelli, D. R. McDonald, and J. Reynier, "A mean-field model for multiple TCP connections through a buffer implementing RED," *Perform. Eval.*, vol. 49, no. 1-4, pp. 77–97, 2002.
[8] C. Casetti and M. Meo, "A new approach to model the stationary behavior of TCP connections," in *INFOCOM*, 2000, pp. 367–375.
[9] L. Muscariello, M. Mellia, M. Meo, M. A. Marsan, and R. L. Cigno, "Markov models of Internet traffic and a new hierarchical MMPP model," *Computer Communications*, vol. 28, no. 16, 2005.
[10] M. Garetto, R. Lo Cigno, M. Meo, and A. M. Marsan, "Modeling short-lived TCP connections with open multiclass queuing networks," *Comput. Networks*, vol. 44, no. 2, pp. 153–176, February 2004.
[11] Y. Liu, F. Lo Presti, V. Misra, D. Towsley, and Y. Gu, "Fluid models and solutions for large-scale IP networks," in *SIGMETRICS '03: Proceedings of the 2003 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, vol. 31, no. 1. New York, NY, USA: ACM Press, June 2003, pp. 91–101.
[12] M. A. Marsan, M. Garetto, P. Giaccone, E. Leonardi, E. Schiattarella, and A. Tarello, "Using partial differential equations to model TCP mice and elephants in large IP networks," *Networking, IEEE/ACM Transactions on*, vol. 13, no. 6, pp. 1289–1301, 2005.
[13] M. Garetto, R. L. Cigno, M. Meo, and M. A. Marsan, "A detailed and accurate closed queueing network model of many interacting TCP flows," in *INFOCOM*, 2001, pp. 1706–1715.
[14] M. A. Marsan, M. Garetto, P. Giaccone, E. Leonardi, E. Schiattarella, and A. Tarello, "Using partial differential equations to model TCP mice and elephants in large IP networks." in *INFOCOM*, 2004.
[15] Q. He, C. Dovrolis, and M. Ammar, "On the predictability of large transfer TCP throughput," in *SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*. New York, NY, USA: ACM Press, 2005, pp. 145–156.
[16] W. Allcock, J. Bester, J. Bresnahan, A. Chervenak, L. Liming, S. Meder, and S. Tuecke, "GridFTP protocol specification," September 2002.
[17] S. Floyd, M. Handley, J. Padhye, and J. Widmer, "Equation-based congestion control for unicast applications," in *ACM SIGCOMM*. ACM Press, 2000, pp. 43–56.
[18] J. Crowcroft and P. Oechslin, "Differentiated end-to-end internet services using a weighted proportional fair sharing TCP," *SIGCOMM Comput. Commun. Rev.*, vol. 28, no. 3, pp. 53–69, 1998.
[19] E. Brosh, G. Lubetzky-Sharon, and Y. Shavitt, "Spatial-temporal analysis of passive TCP measurements." in *INFOCOM*. IEEE, 2005.
[20] "RED parameters. http://icir.org/floyd/red.html#parameters."
[21] M. Handley, S. Floyd, J. Padhye, and J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol Specification," RFC 3448, Jan. 2003.