

On the Fair Coexistence of Loss- and Delay-Based TCP

Łukasz Budzisz, *Member, IEEE*, Rade Stanojević, Arie Schlote, Fred Baker, and Robert Shorten

Abstract—This paper presents and develops a novel delay-based additive increase, multiplicative decrease (AIMD) congestion control algorithm. The main features of the proposed solution include: 1) low standing queues and delay in homogeneous environments (with delay-based flows only); 2) fair coexistence of delay- and loss-based flows in heterogeneous environments; 3) delay-based flows behave as loss-based flows when loss-based flows are present in the network; otherwise they revert to delay-based operation. It is also shown that these properties can be achieved without any appreciable increase in network loss rate over that which would be present in a comparable network of standard TCP flows (loss-based AIMD). To demonstrate the potential of the presented algorithm, both analytical and simulation results are provided in a range of different network scenarios. These include stability and convergence results in general multiple-bottleneck networks and a number of simulation scenarios to demonstrate the utility of the proposed scheme. In particular, we show that networks employing our algorithm have the features of networks in which RED AQM's are deployed. Furthermore, in a wide range of situations (including high-speed scenarios), we show that low delay is achieved irrespective of the queueing algorithm employed in the network, with only sender-side modification to the basic AIMD algorithm.

Index Terms—Delay-based congestion control, TCP.

I. INTRODUCTION

CONGESTION control in TCP/IP networks is traditionally handled using packet losses to indicate congestion [1]. An alternative approach to respond to congestion involves the use of network delay. This was first proposed by Jain [2], and since then, there has been much work and debate on this topic [3]–[11]. Delay-based congestion control is conceptually very attractive. Potential benefits include the ability to allocate the network bandwidth between competing sources with: 1) low (zero) packet loss; 2) very low queueing delay; and 3) with full utilization of network links. Networks that exhibit this property are said to operate at the *knee of the throughput-delay curve* [12]. Motivated by these and other potential benefits,

delay-based congestion control remains an active area of research and new algorithms continue to be developed. Recent examples include Fast TCP [3], [13], Microsoft Compound [14] (partially based on delay), more recent delay-based additive increase multiplicative decrease (AIMD) variants [15]–[19], and this present work.

Despite this large body of work, several issues concerning the use of queueing delay remain to be resolved before delay-based congestion control can be deployed. These include: 1) the difficulty in obtaining delay estimates from network measurements [20]; 2) network sampling issues [20]–[23]; 3) the inability of existing delay-based algorithms to maintain a low standing queue [22], [23]; and 4) the inability of delay-based flows to coexist fairly with loss-based flows in mixed environments. These items have been the subject of much discussion by us and other researchers [20]–[22], which we do not repeat here. Rather, we focus the specific issue of coexistence. Note that the issue here is not just simple coexistence; after all, delay-based flows may simply switch to a loss-based mode once a packet loss is detected, thereby solving the fairness problem. The issue that makes coexistence difficult is that delay-based flows must revert back to delay-based operation when loss-based flows are no longer present.

Given these basic observations, our contribution in this paper is to develop and explore a strategy that resolves (at least in part) these issues. In particular, our principal contribution in this paper is to propose a strategy that allows delay-based TCP flows to coexist fairly and without any discernible increase in network loss rate with loss-based flows when they are present, and to revert back to a delay-based behavior whenever loss-based flows are no longer present. While the basic idea underlying this work was presented in [24] and [25], this paper extends and complements this work in a number of ways. These include more comprehensive simulation studies involving realistic network scenarios (multiple bottlenecks, reverse traffic, and the presence of on/off network flows), and a detailed mathematical analysis. Finally, we demonstrate the potential benefits of the proposed strategy when applied to high-speed links.

Our paper is structured as follows. In Section II, we introduce the proposed algorithm, illustrating its efficacy in Section III with a number of simulations in both single and multiple bottlenecks. Then, in Section IV, we provide mathematical analysis that describes the ability of the proposed solution to switch between loss-based and delay-based operation regimes and reflects on stability issues. Next, in Section V, we compare the proposed solution to other existing strategies, namely RED and one of the most recent delay-based congestion control schemes, as well as provide some insights on its application in more versatile scenarios, including high-speed networks. Finally, in Section VI, we discuss constraints limiting the application of the proposed algorithm.

Manuscript received September 21, 2009; revised June 23, 2010; December 20, 2010; and February 07, 2011; accepted April 01, 2011; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor S. Kalyanaraman. Date of publication July 25, 2011; date of current version December 16, 2011.

Ł. Budzisz is with the Telecommunication Networks Group, TU Berlin, 10587 Berlin, Germany (e-mail: Łukasz.Budzisz@tu-berlin.de).

R. Stanojević is with the Institute IMDEA Networks, 28918 Madrid, Spain (e-mail: rade.stanojevic@imdea.org).

A. Schlote and R. Shorten are with the Hamilton Institute, NUI Maynooth, Maynooth, Ireland (e-mail: Arie.Schlote@nuim.ie; Robert.Shorten@nuim.ie).

F. Baker is with Cisco Systems, San Jose, CA 95134 USA (e-mail: fred@cisco.com).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TNET.2011.2159736

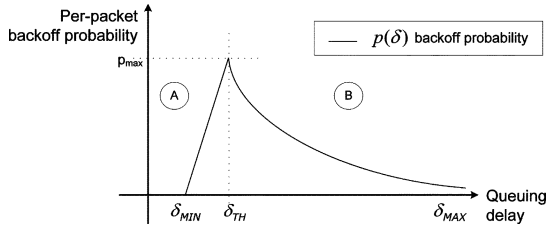


Fig. 1. Per-packet backoff probability as a function of the observed delay.

II. AQM EMULATION TO ENSURE FAIR COEXISTENCE WITH LOSS-BASED FLOWS

Our basic idea is motivated by recent work concerning active queue management (AQM) emulation from end-hosts using delay measurements, called Probabilistic Early Response TCP (PERT) [17]. The basic idea behind PERT is very simple and involves responding to delay in a probabilistic rather than deterministic manner. By judiciously selecting the manner of the probabilistic response, Bhandarkar *et al.* [17] are able to emulate, from end-hosts, the behavior of a range of AQMs. To facilitate such AQM emulation, each PERT flow probes the network for congestion as a normal AIMD flow, but reduces its congestion window in a probabilistic manner that depends on the estimated network delay. We refer to this mechanism as a *backoff policy*. The authors of PERT demonstrate that (in principle) any AQM can be emulated by selecting the backoff policy appropriately. However, it was subsequently shown that their algorithm fails to solve the coexistence problem in a satisfactory manner [25]. In particular, it is shown in this latter paper that PERT can lead to high loss rates when loss-based flows are present, and that the delay-based flows may fail to revert to delay-based operation when the loss-based flows leave the network. Our main contribution here is to present a similar idea that can be used to solve the coexistence problem by carefully choosing the probabilistic backoff function while avoiding many of the side-effects of the PERT strategy. As we shall see, our strategy, referred to as Coexistent TCP (Cx-TCP) further in this paper, avoids problems of adjusting AIMD parameters, keeps the network loss rate low when loss-based flows are present, and ensures that the delay-based flows revert to delay-based operation when loss-based flows are no longer present in the network (termed here as *on/off behavior*), even though these flows do not attempt to sense the presence of such flows directly.

Specifically, our basic idea is to achieve coexistence by carefully selecting the backoff policy to achieve fairness and on/off behavior. In what follows, we assume that queueing delay (δ) and minimum (RTT_{\min}) and maximum round-trip times (RTT_{\max}) can be estimated reliably by all delay-based flows in the network and do not consider the issue of slow start for delay-based flows. These, and other issues that are not within the scope of this paper, are discussed in our previous work and in previous work by other authors [15], [20]–[22].

We select probabilistic backoff strategies of the form depicted in Fig. 1. As can be observed, the per-packet backoff probability function $p(\delta)$ has two parts: a part that increases monotonically with the delay δ (Region A) and a part that decreases monotonically with δ (Region B). This form of AQM emulation has the following properties.

Algorithm 1: Pseudocode for Cx-TCP Algorithm

On receipt of each ACK:
 Estimate the current queueing delay: δ
 Set $p(\delta)$: (only nonzero values are shown)

$$p(\delta) = \begin{cases} p_{\max} \frac{\delta - \delta_{\min}}{\delta_{th} - \delta_{\min}}, & \text{for } \delta_{\min} < \delta \leq \delta_{th} \\ p_{\max} \left(\frac{\delta_{\max} - \delta}{\delta_{\max} - \delta_{th}} \right)^4, & \text{for } \delta_{th} < \delta < \delta_{\max}. \end{cases}$$

Pick a random number rand, uniformly from 0 to 1

if rand < p **then**

reduce cwnd by $0.5 \cdot \text{cwnd}$

else

increment cwnd by $1/\text{cwnd}$

end if

- (i) Assuming that the maximum equilibrium loss rate (p_{\max}) is large enough, the network stabilizes in Region A when only delay-based flows are present.
- (ii) When loss-based flows are present, the network is driven to Region B, and delay-based flows behave as loss-based flows due to the low per-packet backoff probability.
- (iii) When loss-based flows switch off, provided the network is appropriately designed, the network cannot stabilize in this region due to a backward pressure exerted by the probability function. Namely, as the flows experience backoffs, the queueing delay reduces, thereby increasing the per-packet backoff probability, thus making further backoffs more likely. This process continues until the network stabilizes in Region A.

As can be seen, this type of strategy should achieve coexistence of loss- and delay-based AIMD flows, without a discernible increase in network loss rate. Furthermore, the back-pressure described in (iii) should ensure on/off behavior. Related work to that presented here is given in [26] and [27], where the coexistence of different types of congestion control algorithms is studied. The authors analyze there the qualitative properties (uniqueness, existence, optimality, and stability) of equilibria that arise in such networks. Our work in this present paper goes beyond this work. In contrast to purely heterogeneous setup analyzed in [26] and [27], our work focuses on practical methods for designing the protocols with equilibria that have desirable quantitative properties. Namely, when Cx-TCP competes with a loss-based-TCP, the design of Cx-TCP ensures that a Cx-TCP flow converts into a loss-based flow (with a minimal number of non-loss-based backoffs), which means that standard homogeneous model closely approximates the equilibria of the mix of Cx-TCP and loss-based TCP flows (see Section III for the empirical evidence of this fact). In this sense, we engineer some of the equilibria in our network to be unstable and use this as a means to ensure switching between loss-based and delay-based operational modes.

The basic Cx-TCP algorithm is summarized in Algorithm 1. δ is estimated as the difference between the weighted average of the RTT and its minimum observed value. To ensure similar number of RTT samples for flows with different RTTs, the RTT weight is set proportionally to the ratio of the average RTT and the cwnd value for a given flow. Again, the reader should be

TABLE I
SUMMARY OF THE SCENARIO PARAMETERS

PARAMETER	LOW DELAYS	REVERSE TRAFFIC	FEATURES IN HOMOGENEOUS ENVIRONMENTS	FAIRNESS	SCALABILITY	LOSS RATE	ON-OFF SWITCHING
Capacity	25 Mbps	25 Mbps	50 Mbps	25 Mbps	500 Mbps	10-100 Mbps	25 Mbps
RTT	all flows: 100 ms	all flows: 100 ms	30-130 ms, uniformly	30-130 ms, uniformly	30-130 ms, uniformly	30-130 ms, uniformly	30-130 ms, uniformly
Number of flows	1-50(55) flows	forward path: 1-50 reverse path: 20	(1-2): 30 flows (3): 51 flows (20 in at 200 s, 30 out at 400 s)	30 flows in 2 mixes: (20,10) and (10,20)	600 flows: (400,200) mix	30 flows: (20,10) mix	30 flows: 20 delay-based and 10 intermittent [‡] (100-300 s) loss based flows
Web traffic	no	no	no	yes: 10 flows	yes: 100 flows	yes: 10 flows	no
Common parameters	For all tests: buffer size on each link: 1BDP; packet size: 1000 Byte; simulation time: 500 s; All loss based flows (web traffic sessions as well) use Reno TCP.						

[‡] 10 intermittent loss based flows enter the scenario in 0.5 s intervals (100-105 s) and leave in 5 s intervals (250-300 s).

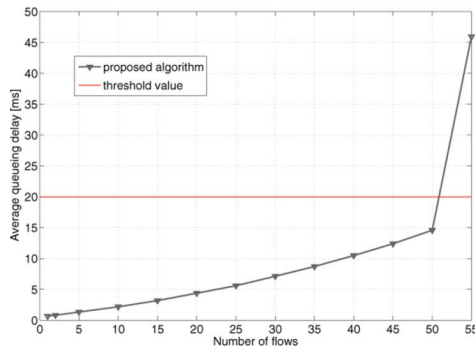


Fig. 2. Proposed algorithm in a single-bottleneck scenario: average queueing delay as a function of the number of flows for a bottleneck capacity of 25 Mb/s.

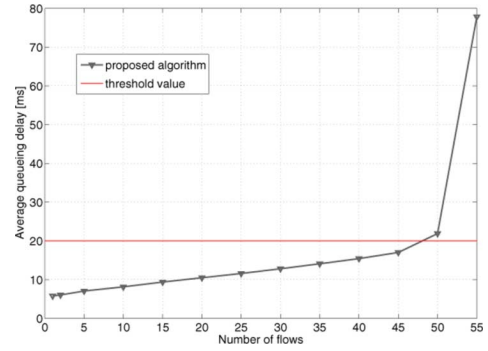


Fig. 3. Performance with 20 flows on the reverse path: average queueing delay as the number of flows on the forward path.

reminded that this paper does not discuss the accuracy of such an estimate.

Comment: Finally, we emphasize that our objective here is to present a simple idea that may be very useful in solving the coexistence problem. In its current embodiment, the idea works best in heavily multiplexed environments with reasonably sized queues, and this is reflected in the experimental results given in the paper. Extensions to light levels of multiplexing have been studied in [28]. Clearly, the queue provides the delay signal, and small buffer sizes will be problematic for the algorithm. However, since our algorithm keeps the delay in the network low, the effects of large buffer sizes are mitigated by the performance of our algorithm.

III. BASIC OPERATION OF ALGORITHM

To illustrate the basic operation of Cx-TCP described in Section II, we now present a number of simple experiments. We begin with a single-bottleneck topology and then progress to a more realistic multiple-bottleneck scenario. Our objectives are: 1) to demonstrate that low-delay networks can be realized in homogeneous environments; 2) to illustrate that almost perfect coexistence can be achieved; 3) to examine the fairness properties of the algorithm in mixed environments; 4) to show that the loss rate is low (not worse than that of just loss-based flows) when loss- and delay-based flows coexist together; and 5) to demonstrate that on/off switching can be achieved. All experiments are constructed with these objectives in mind.

A. Single Bottleneck

We begin with the evaluation of Cx-TCP in a single-bottleneck scenario in a classic *dumbbell* topology. In the following tests, we use ns-2 [29] simulations with the most important settings specified in Table I. As for the Cx-TCP parameters, the algorithm proposed here emulates RED AQM in the region A (Fig. 1). Consequently, one method to select the parameter values for δ_{\min} (5 ms), δ_{th} (20 ms), and p_{\max} (5%) is to use the rules for RED parameter settings [17]. Note that parameter settings provided here are suitable for a range of link capacities and bandwidth. See basic feature and scalability tests for more comments on that matter.

1) *Basic Feature—Low Delays for Homogeneous Networks:* In Fig. 2, we first demonstrate that our algorithm does indeed yield low-delay networks in case all flows are delay-based. As already mentioned in Section II, the maximum equilibrium loss rate is given by p_{\max} (see Fig. 1). This means that the network will revert to a loss-based network if there is a very large number of network flows (related to the available network capacity; here, it is between 50 and 55 flows for a 25-Mb/s bottleneck)—namely, if the required equilibrium loss rate is greater than p_{\max} . This property is desirable as it is well known that estimation of queueing delay is difficult in networks with very large multiplexing of flows [22].

2) *Reverse-Path Traffic:* It is well known that reverse-path traffic can increase the ACK losses for the forward-path flows. To check the influence of this issue on the performance of our algorithm, we introduce 20 long-lived flows using also Cx-TCP on the reverse path. Fig. 3 illustrates the results for the average δ

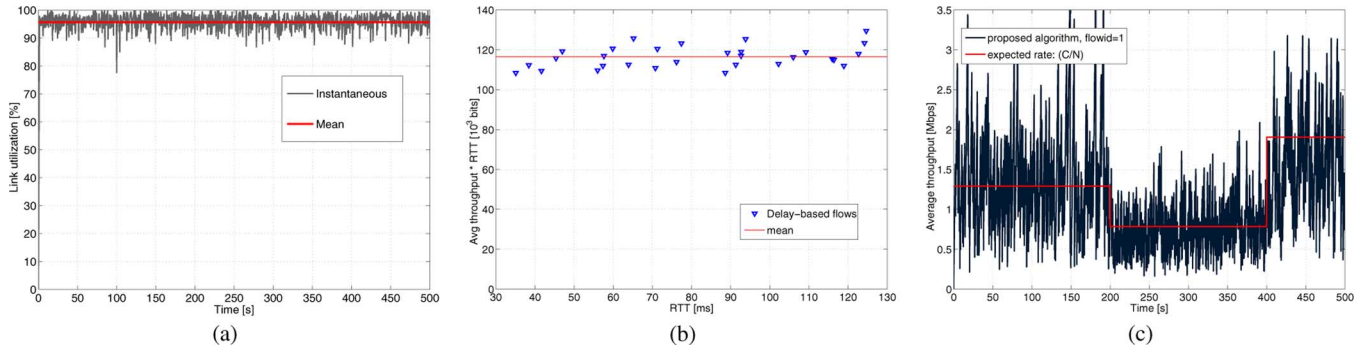


Fig. 4. Illustration of the properties of Cx-TCP in homogeneous scenarios: (a) link utilization, (b) window fairness, and (c) convergence properties.

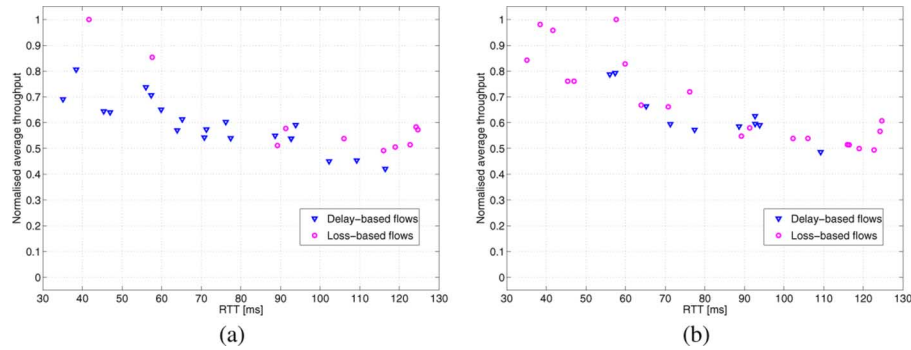


Fig. 5. Coexistence of the delay-based flows and standard TCP flows in terms of the normalized average throughput for the following mixes of flows: (a) the (20, 10) mix and (b) the (10, 20) mix.

as a function of the number of delay-based forward-path flows sharing the same 25-Mb/s bottleneck with 20 delay-based flows on the reverse path. It can be clearly seen that the limit of the operation of the presented algorithm has been decreased to accommodate the reverse-path traffic.

3) *Further Features in Homogeneous Scenarios:* Apart from maintaining low queueing delay when operating in the delay-based mode, Cx-TCP has several interesting features that are illustrated in Fig. 4.

- 1) The proposed algorithm depends on a nonzero average queueing delay. Thus, by the very nature of this hypothesis, we are assuming that the link is close to being fully utilized at all times. This is indeed the case, and for all reported experiments, the average link utilization has been kept at very high level of 95%–97%, e.g., see Fig. 4(a) depicting instantaneous and average link utilization for 30 flows competing on a 50-Mb/s bottleneck.
- 2) The proposed dropping policy (shown in Fig. 1) emulates RED. As such, it strives to achieve *window fairness*, i.e., in the equilibrium state all flows have, on average, the same cwnd. This is illustrated experimentally in Fig. 4(b). Here, 30 flows with RTTs uniformly distributed between 30 and 130 ms all achieve similar average window size.
- 3) Finally, we examine the convergence properties of Cx-TCP in a setting where N varies instantaneously for a given bottleneck capacity (50 Mb/s). This is depicted for a selected flow in Fig. 4(c). Initially, 31 delay-based flows run the proposed algorithm, and then flows either enter (additional 20 flows at time 200 s) or leave (30 flows at time 400 s) the discussed scenario. As

can be seen for the presented flow, the allocation of bandwidth switches instantaneously to the correct equilibria.¹

4) *Fairness:* Next, we examine the ability of delay-based flows to coexist (fairly) with standard loss-based flows. Fig. 5 depicts the normalized average throughput (the fraction of the highest average throughput among all flows over a 500-s-long experiment) for a network with two different mixes of standard TCP and Cx-TCP flows (a total of 30 flows). Note that while there is a bias in favor of the loss-based flows (due to the fact that the delay-based flows experience a small number of non-loss-induced backoffs in the high-queue regime), there is a reasonably fair coexistence of the loss-based and delay-based AIMD flows. Furthermore, the aforementioned bias in favor of loss-based flows can be controlled by carefully selecting the backoff policy. Notwithstanding this latter observation, the experiments nevertheless demonstrate very good coexistence of the delay-based and loss-based flows as measured by the average throughput.

5) *Scalability:* As already mentioned, the parameter settings of the proposed algorithm influence the range of bandwidth in which Cx-TCP can operate. To check that, we run a test in a scenario with a 500-Mb/s bottleneck. To meet the design assumptions, we keep the ratio of C/N from the fairness experiments unchanged. This yields a scenario with a mix of (400, 200) delay- and loss-based flows. As can be seen in Fig. 6, Cx-TCP is able to keep the fairness property constant, despite a slight bias in favor of the loss-based flows (see fairness comments).

¹Due to the limited editorial space, not all result for this particular experiment can be included here. For more results, please refer online to <http://www.hamilton.ie/lukaszc-tcp.html>

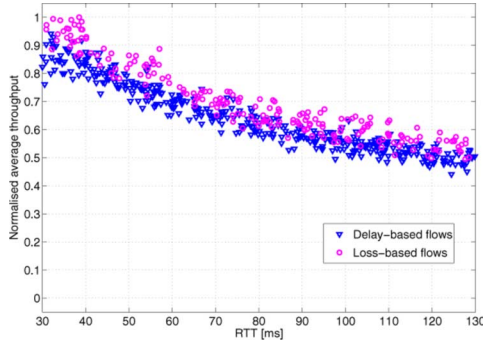


Fig. 6. Scalability of the proposed solution (preserving the C/N ratio): coexistence of the (400, 200) mix of flows in a scenario with a 500-Mb/s bottleneck bandwidth.

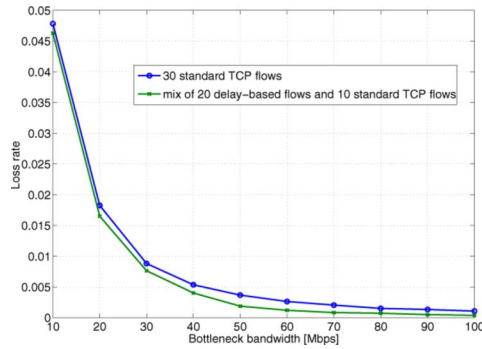


Fig. 7. Comparison in terms of loss rate: 20 delay-based flows coexisting with 10 standard TCP flows and 30 standard TCP flows.

6) *Loss Rate*: Now, we examine the effect of our algorithm on the network loss rate (in the loss-based operation mode, the backoff probability is low enough so that the congestion is controlled by packet losses). To do this, we compare the behavior of 20 delay-based Cx-TCP flows coexisting with 10 standard TCP flows with a scenario in which all 30 flows are standard loss-based TCP. The results are depicted in Fig. 7. Observe that the proposed algorithm does not significantly increase the network loss rate in the presence of loss-based flows. Fair coexistence is achieved without any unnecessary tradeoffs.

7) *On/Off Switching*: Our primary objective in this work was to develop a delay-based algorithm that behaves as a loss-based TCP when competing with loss-based TCP flows, but otherwise reverts to delay-based operation. This behavior is captured in Fig. 8. Here, 30 flows (20 delay-based and 10 intermittent loss-based flows) compete for the available bandwidth. Between 100 and 300 s, when the loss-based flows appear in the network, the delay-based flows behave as standard TCP flows and compete fairly for bandwidth. Otherwise, they strive in a co-operative manner to keep the queueing delay below a certain threshold (δ_{th}). Note also that the mode switching occurs automatically (and swiftly) without any complicated sensing or signal processing to determine whether or not the loss-based flows have left the network. Note also that for this algorithm, the operation mode is decided upon by the average value of δ and not the instantaneous one. Thus, δ_{th} may be exceeded locally, but it is the average value of δ that decides which operation mode to choose. We believe that this mechanism is novel in delay-based congestion control context and has uses beyond the present context.

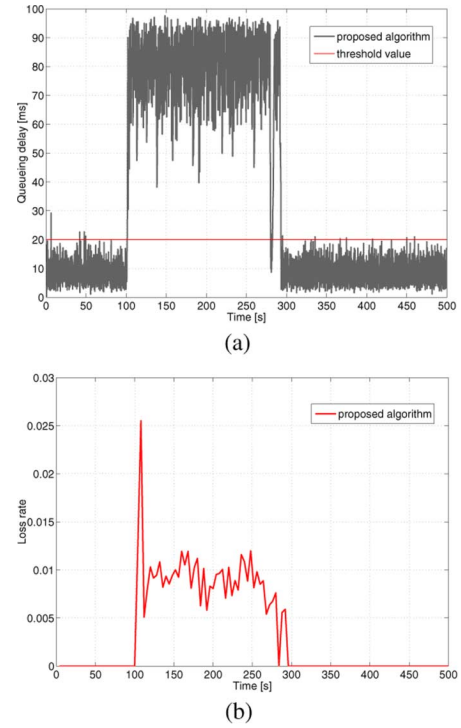


Fig. 8. Coexistence of 20 delay-based flows with 10 loss-based flows switching on and off: (a) queueing delay at the bottleneck; (b) loss rate.

B. Multiple-Bottleneck Scenarios

Next, we examine Cx-TCP in a multiple-bottleneck situation. Objectives stay the same, as stated in the introduction to this section. The scenario under test is the simplest *parking-lot topology* and comprises a cascade of two bottlenecks, with half of all flows traversing both the *first* and *second* bottleneck (and the remaining half living in the second bottleneck only). The first bottleneck has a capacity of 12.5 Mb/s and a 155-packet queue, whereas the second bottleneck has 25-Mb/s capacity and a 310-packet queue, respectively. Each of the bottlenecks introduces 5 ms propagation delay and serves a drop-tail queue. Apart from the flows under evaluation, in the coexistence experiment, 10 mice flows are added on each bottleneck to make the scenario more realistic. All the remaining parameters are the same as for the corresponding tests in Section III-A.

1) *Coexistence With Loss-Based Flows*: First, we compare 30 delay-based flows using Cx-TCP to a mix of (20,10) delay- and loss-based flows split evenly between two bottlenecks. Fig. 9 shows the results for each set in separate rows (the scenario with all flows being delay-based is shown in the upper row, whereas the mix of delay- and loss-based flows is shown in the lower row). In case there are only delay-based flows, the algorithm is able to keep queueing delay below the δ_{th} threshold at each of the bottlenecks (again, readers should be reminded that it is the average value of δ that decides the operating area). We shall see that such a homogeneous network has similar properties to a network in which a RED AQM is deployed. Note also that in presence of the standard TCP flows, the algorithm switches to a loss-based operation, filling the queues at each bottleneck, but preserving the overall fairness with standard TCP counterparts [Fig. 9(f)].

2) *On/Off Switching*: Next, we check the dynamic behavior of Cx-TCP in the multiple-bottleneck scenario with 30 flows (20

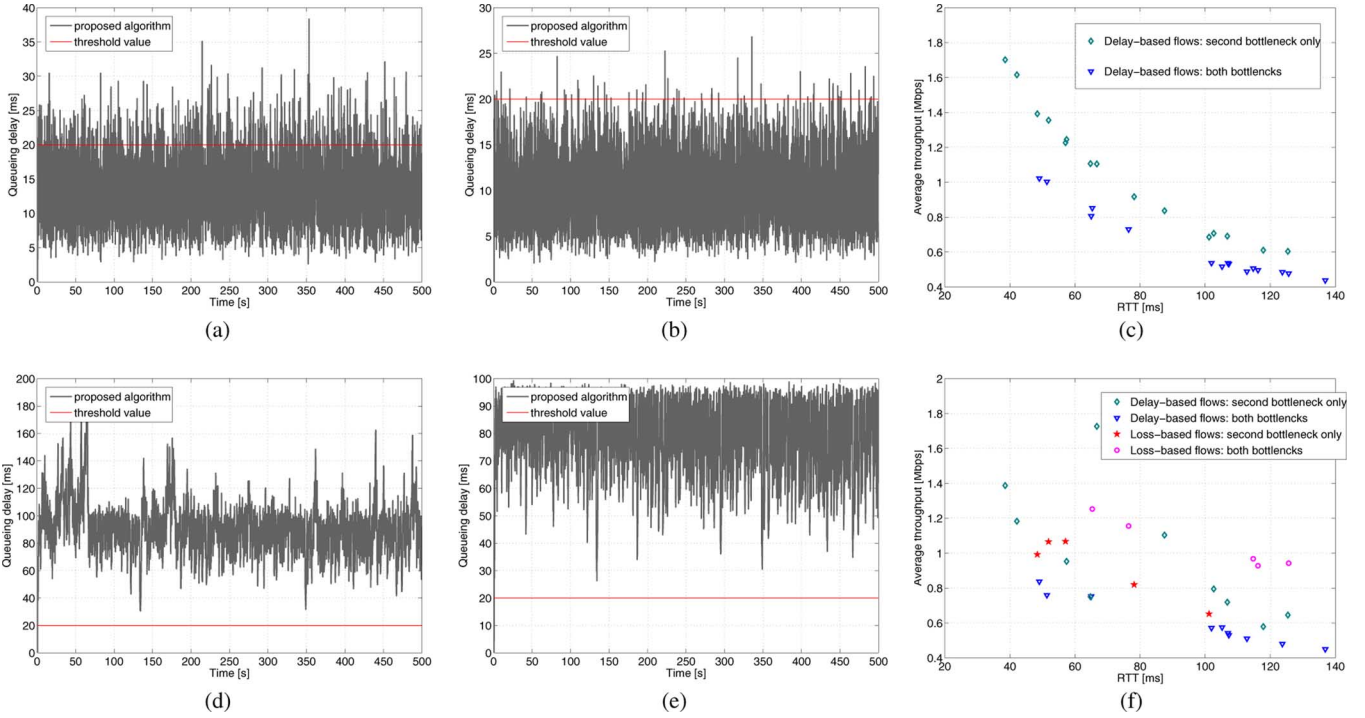


Fig. 9. Coexistence of 20 delay-based flows and 10 standard TCP flows (lower row) in comparison to 30 delay-based flows (upper row) in a multiple-bottleneck scenario: (a), (d) queuing delay at the first bottleneck; (b), (e) queuing delay at the second bottleneck; (c), (f) average throughput.

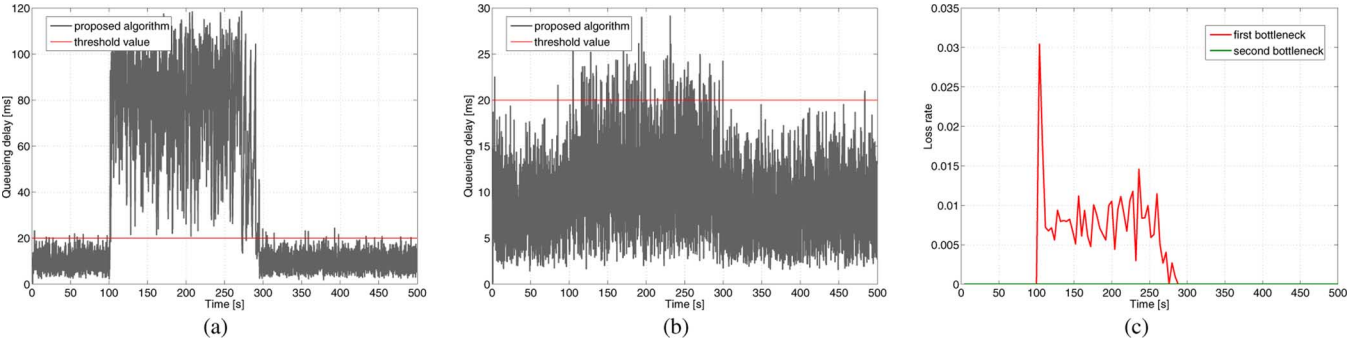


Fig. 10. Coexistence of 20 delay-based flows with 10 loss-based flows switching on and off: queuing delay at (a) the first bottleneck (where the loss-based interference appears), and (b) the second bottleneck; and (c) corresponding loss rates.

delay-based and 10 intermittent loss-based flows) competing for the available bandwidth (Fig. 10). When 10 loss-based flows appear at the first bottleneck (100–300 s), the five delay-based flows living there switch to the long-queueing-delay mode and compete fairly for bandwidth. Meanwhile, the 15 delay-based flows at the second bottleneck stay in the low-queueing-delay regime. Note that the high queueing delay (and nonzero loss rate) operation affects only the bottleneck where the loss-based flow appeared. Otherwise, when there are only delay-based flows (before 100 s and after 300 s), the low-queueing-delay regime is maintained at both bottlenecks.

IV. MATHEMATICAL ANALYSIS

Having outlined the basic properties of our algorithm, we now characterize some of these properties in a mathematical framework. Our starting point is the single-bottleneck scenario. We show, using a fluid (Kelly) like argument, that the on/off behavior is a manifestation of the interaction of stable and un-

stable equilibria in the network. We then extend this result to multiple-bottleneck networks.

Basic Notation: By $\delta(t)$ and $W_i(t)$, where $i = 1, \dots, N$, we denote the queueing delay and congestion window size (cwnd) of flow i at time t .

Basic Model (Single Bottleneck): We essentially use a mean field fluid model to demonstrate the plausibility of our approach. Clearly, an accurate modeling of TCP dynamics merits a much more complicated discrete event model. Nevertheless, our approach follows accepted practice in the community, and these models do provide a reasonable approximation in highly multiplexed situations. Our analysis is based on the following assumption. The evolution of the average window size over one RTT is determined by: 1) the probability of a backoff, and 2) the rate at which the window variable grows. When analyzed over one small time interval Δt , the equation describing this evo-

lution is the same for all flows and is given by $\Delta W_i(t) = \frac{\Delta t}{\text{RTT}_i + \delta(t)} - q_0^i \cdot \frac{W_i(t)}{2}$, where q_0^i is the probability that during the time interval $(t, t + \Delta t)$ a backoff occurred for flow i . Dividing by Δt yields

$$\frac{\Delta W_i(t)}{\Delta t} = \frac{1}{\text{RTT}_i + \delta(t)} - \frac{q_0^i}{\Delta t} \cdot \frac{W_i(t)}{2}. \quad (1)$$

We denote by M_0^i the number of packets the source of flow i with congestion window size of $W_i(t)$ sends in the interval $(t, t + \Delta t)$. Then, $M_0^i = \frac{\Delta t \cdot W_i(t)}{\text{RTT}_i + \delta(t)}$, and we can approximate q_0^i as $q_0^i = 1 - (1 - p)^{M_0^i} \approx p M_0^i = \frac{p \Delta t \cdot W_i(t)}{\text{RTT}_i + \delta(t)}$. In the limit $\Delta t \rightarrow 0$, this yields

$$\dot{W}_i(t) = \frac{1}{\text{RTT}_i + \delta(t)} \left(1 - \frac{p}{2} W_i^2(t) \right). \quad (2)$$

We are going to use this very general equation (2) to model our system also in the multiple-bottleneck case.

We start our analysis by regarding a scenario with N flows competing at a single bottleneck. The dynamic system we are going to analyze is given by (2) for all $i = 1, \dots, N$, and we model queue dynamics as

$$\dot{q} = \begin{cases} \sum_{i=1}^N \frac{W_i}{\tau_i + \delta} - C, & \text{for } q > 0 \\ \left\{ \sum_{i=1}^N \frac{W_i}{\tau_i + \delta} - C \right\}^+, & \text{for } q = 0 \end{cases} \quad (3)$$

where q is the queue length. For any function f that maps into the real numbers, we denote by $\{f\}^+$ the nonnegative part of f , that is $\{f\}^+ = \max\{f, 0\}$. q and δ are coupled as $\delta = \frac{q}{C}$. If we set $\tau_i = \text{RTT}_i + \delta_{\min}$ and abusing notation $\delta = \delta - \delta_{\min}$ and accordingly $q = q - C\delta_{\min}$, then the backoff probability function p is of the form

$$p(\delta) = \begin{cases} K\delta, & \text{for } 0 \leq \delta \leq \delta_{\text{th}} \\ p_0 - \hat{K}\delta, & \text{for } \delta_{\text{th}} \leq \delta \leq \delta_{\max} \end{cases} \quad (4)$$

for positive constants K, p_0, \hat{K} .

Comment: For mathematical convenience, we have assumed a piecewise linear function $p(\delta)$. The following analysis can be carried out for any $p(\delta)$ with the same qualitative properties; namely, it strictly increases in $(0, \delta_{\text{th}})$ and strictly decreases in $(\delta_{\text{th}}, \delta_{\max})$. Some of the details change, but the approach extends in complete generality (a completely general discussion is omitted here due to space limitations).

Comment: It is important to note that the following analysis extends the short mathematical description given in [24] and [25] by including a detailed queue model. In the cited short papers, a brief analysis based on simplified queueing behavior is presented.

Lemma 4.1: The system (2)–(4) can be designed such that it has two equilibria in the positive orthant: one for $\delta \in [0, \delta_{\text{th}}]$ and one for $\delta \in [\delta_{\text{th}}, \delta_{\max}]$.

Proof: Setting (2) to 0 yields $W_i^2 = \frac{2}{p(\delta)}$, and thus $W_i = \sqrt{\frac{2}{p(\delta)}}$, as we are only interested in physically feasible solutions. That implies that, in equilibrium, all window sizes are the same:

$W_1^* = \dots = W_N^* =: W^*$. Now, setting (3) to 0 yields $C = \sqrt{\frac{2}{p(\delta)}} \sum_{i=1}^N \frac{1}{\tau_i + \delta}$. For $\delta \in [0, \delta_{\text{th}}]$, we have $p(\delta) = K\delta$, and thus

$$C = \sqrt{\frac{2}{K\delta}} \sum_{i=1}^N \frac{1}{\tau_i + \delta}. \quad (5)$$

It can be seen that the right-hand side of this equation is strictly decreasing in δ , hence there is at most one δ satisfying (5). By choosing K appropriately, we can assure its existence in the aspired position.

For $\delta \in [\delta_{\text{th}}, \delta_{\max}]$, we have $p(\delta) = p_0 - \hat{K}\delta$, and thus

$$C \sqrt{\frac{p_0 - \hat{K}\delta}{2}} = \sum_{i=1}^N \frac{1}{\tau_i + \delta}. \quad (6)$$

It is easy to see that for positive δ , the left-hand side of this equation is strictly concave, while the right-hand side is strictly convex. Accordingly, there are no more than two possible values for δ to solve (6). Bearing in mind that we need $p_0 - \hat{K}\delta_{\text{th}} = K\delta_{\text{th}}$ for $p(\delta)$ to be continuous, the parameters \hat{K}, p_0 can now be chosen such that the largest value of δ is in $\delta \in [\delta_{\text{th}}, \delta_{\max}]$ while the other is not and will be cut off as a result. ■

Comment: For illustration, it is easily verified using (5) and (6), and the equation for $p(\delta)$, that the following homogeneous network has the properties claimed in Lemma 4.3: $C = 100$ Mbs; $\text{RTT} = 200$ [ms] (for all flows); $N = 400$. $\delta_{\min} = 5$ [ms]; $\delta_{\text{th}} = 20$ [ms]; $\delta_{\max} = 100$ [ms]; $p_{\max} = 0.05$.

Lemma 4.2: The equilibrium with $\delta \in [0, \delta_{\text{th}}]$ is locally asymptotically stable.

Proof: We use the following coordinate transformation: $\tilde{W}_i = W_i - W^*$ for all $i = 1, \dots, N$ and $\tilde{q} = q - q^*$. This shifts the equilibrium the point with all coordinates equal to 0 and enables us to apply a standard Lyapunov argument. Accordingly, we define $\tilde{\delta} = \delta - \delta^*$.

We will further write (3) in the form

$$\begin{aligned} \dot{\tilde{q}} &= \dot{q} + \dot{q}^* \\ &= \dot{q} \\ &= \sum_{i=1}^N \frac{W_i}{\tau_i + \delta} - C \\ &= \sum_{i=1}^N \frac{\tilde{W}_i + W_i^*}{\tau_i + \delta} - C \\ &= \sum_{i=1}^N \frac{\tilde{W}_i}{\tau_i + \delta} + \sum_{i=1}^N \frac{W_i^*}{\tau_i + \delta} - C \\ &= \sum_{i=1}^N \frac{\tilde{W}_i}{\tau_i + \delta} - \sum_{i=1}^N \frac{W_i^* \tilde{\delta}}{(\tau_i + \delta^* + \tilde{\delta})(\tau_i + \delta^*)} + \sum_{i=1}^N \frac{W_i^*}{\tau_i + \delta^*} - C \\ &= \sum_{i=1}^N \frac{\tilde{W}_i}{\tau_i + \delta} - \sum_{i=1}^N \frac{W_i^* \tilde{\delta}}{(\tau_i + \delta^* + \tilde{\delta})(\tau_i + \delta^*)} \end{aligned} \quad (7)$$

where we used the identity $\frac{1}{\tau_i + \delta} = \frac{1}{\tau_i + \delta^* + \tilde{\delta}} = \frac{-\tilde{\delta}}{(\tau_i + \delta^* + \tilde{\delta})(\tau_i + \delta^*)} + \frac{1}{\tau_i + \delta^*}$, and in the last step we used

that at the equilibrium $\sum_{i=1}^N \frac{W_i^*}{\tau_i + \delta^*} = C$. We write (2) in the form

$$\begin{aligned} \dot{W}_i &= \dot{W}_i + \dot{W}_i^* = \dot{W}_i = \frac{1}{\tau_i + \delta} \left(1 - \frac{p(\delta)}{2} W_i^2 \right) \\ &= \frac{1}{\tau_i + \delta} \left(1 - \frac{Kq}{2} W_i^2 \right) \\ &= \frac{1}{\tau_i + \delta} \left(1 - \frac{K\tilde{q} + q^*}{2} (\tilde{W}_i + W_i^*)^2 \right) \\ &= \frac{1}{\tau_i + \delta} \left(1 - \frac{Kq^*}{2} (W_i^*)^2 \right) \\ &\quad - \frac{1}{2} \frac{K}{\tau_i + \delta} \frac{q}{C} (\tilde{W}_i^2 + 2\tilde{W}_i W_i^*) - \frac{1}{2} \frac{K}{\tau_i + \delta} \frac{\tilde{q}}{C} (W_i^*)^2 \\ &= -\frac{1}{2} \frac{K}{\tau_i + \delta} \frac{q}{C} (\tilde{W}_i^2 + 2\tilde{W}_i W_i^*) - \frac{1}{2} \frac{K}{\tau_i + \delta} \frac{\tilde{q}}{C} (W_i^*)^2. \end{aligned} \quad (8)$$

Thus, for all $i = 1, \dots, N$, the systems dynamics can be described by

$$\begin{aligned} \dot{W}_i &= -\frac{1}{2} \frac{K}{\tau_i + \delta} \frac{q}{C} (\tilde{W}_i^2 + 2\tilde{W}_i W_i^*) \\ &\quad - \frac{1}{2} \frac{K}{\tau_i + \delta} \frac{\tilde{q}}{C} (W_i^*)^2 \end{aligned} \quad (9)$$

$$\dot{\tilde{q}} = \sum_{i=1}^N \frac{\dot{W}_i}{\tau_i + \delta} - \sum_{i=1}^N \frac{\delta W_i^*}{(\tau_i + \delta)(\tau_i + \delta^*)}. \quad (10)$$

Now, we show that the function $V : \mathbb{R}^N \times \mathbb{R} \rightarrow \mathbb{R}_+$ given by $V(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}) = \sum_{i=1}^N \frac{\tilde{W}_i^2}{(W_i^*)^2} + \frac{K}{2C} \tilde{q}^2$ is a Lyapunov function with which we can prove asymptotic stability of the equilibrium. To this end, we look at its time derivative

$$\begin{aligned} \dot{V}(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}) &= \sum_{i=1}^N \frac{\tilde{W}_i}{(W_i^*)^2} \dot{W}_i + \frac{K}{C} \tilde{q} \dot{\tilde{q}} \\ &= -\sum_{i=1}^N \frac{\tilde{W}_i}{(W_i^*)^2} \frac{1}{2} \frac{K}{\tau_i + \delta} \frac{q}{C} (\tilde{W}_i^2 + 2\tilde{W}_i W_i^*) \\ &\quad - \sum_{i=1}^N \frac{\tilde{W}_i}{(W_i^*)^2} \frac{1}{2} \frac{K}{\tau_i + \delta} \frac{\tilde{q}}{C} (W_i^*)^2 + \frac{K}{2C} \tilde{q} \sum_{i=1}^N \frac{\tilde{W}_i}{\tau_i + \delta} \\ &\quad - \frac{K}{2C} \tilde{q} \sum_{i=1}^N \frac{\delta W_i^*}{(\tau_i + \delta)(\tau_i + \delta^*)} \\ &= -\sum_{i=1}^N \frac{\tilde{W}_i^2}{(W_i^*)^2} \frac{1}{2} \frac{K}{\tau_i + \delta} \frac{q}{C} (\tilde{W}_i + 2W_i^*) \\ &\quad - \frac{K}{2C^2} \tilde{q}^2 \sum_{i=1}^N \frac{W_i^*}{(\tau_i + \delta)(\tau_i + \delta^*)} \leq 0. \end{aligned} \quad (11)$$

Also, $\dot{V}(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}) = 0$ only in the equilibrium. Thus, by [30, Theorem 3.2.25], the equilibrium is locally asymptotically stable. ■

Comment: Note that the equilibrium (6) can be used to determine the maximum number of flows that can be supported in a stable fashion in the network. To do this, one establishes the maximum N for which a stable equilibrium state exists in the

network—namely, the maximum N such that (6) has a solution (8) to the left of the apex of the probability curve (for a given set of network parameters). See *model validation* for more discussion on this matter. Once such an equilibrium exists, one can also solve the equation for δ^* to find the equilibrium delay.

Lemma 4.3: The equilibrium with $\delta \in [\delta_{th}, \delta_{max}]$ is unstable.

Proof: Here, $p(\delta) = p_0 - \hat{K}\delta$, and the dynamic is described by

$$\dot{W}_i = \frac{1}{\tau_i + \delta} \left(1 - \frac{p_0 - \hat{K}\delta}{2} W_i^2 \right) \quad (12)$$

for all $i = 1, \dots, N$, and

$$\dot{\tilde{q}} = \sum_{i=1}^N \frac{W_i}{\tau_i + \delta} - C. \quad (13)$$

We will now argue that the subset \mathcal{S} of the state space where $\dot{\tilde{q}} > 0$ and $\dot{W}_i > 0$ for all $i = 1, \dots, N$ is a positive invariant set, which means that any solution starting in \mathcal{S} can never leave \mathcal{S} , and that in every neighborhood of the equilibrium there are points in \mathcal{S} that are strictly larger than the equilibrium. This then shows that the equilibrium is not attractive.

We assume that solutions of (12) and (13) are continuous, thus solutions starting in \mathcal{S} cannot leave \mathcal{S} without passing through its boundary. Regard any boundary point (W_1, \dots, W_N, q) of \mathcal{S} . Either (W_1, \dots, W_N, q) is the equilibrium or one of the $N+1$ (12) and (13) is positive. Let $\dot{\tilde{q}} = 0$, then we know that $\dot{W}_i > 0$ for one i , and as W_i increases, $\dot{\tilde{q}}$ is pushed into the positive numbers. Let now $\dot{\tilde{q}} > 0$ and consider any $i = 1, \dots, N$ for which $\dot{W}_i = 0$. As q increases, $p_0 - \hat{K}\frac{q}{C}$ decreases, which increases \dot{W}_i . We conclude that $\mathcal{S} \cup \{(W_1^*, \dots, W_N^*, q^*)\}$ is invariant. We now show that $\mathcal{S} \neq \emptyset$. To this end, regard $q = \lambda q^*$ and $W_i = \sqrt{\frac{2}{p_0 - \hat{K}q}}$ for all $i = 1, \dots, N$. This yields $\dot{W}_i = 0$ for all $i = 1, \dots, N$ and

$$\dot{\tilde{q}} = \sqrt{\frac{2}{p_0 - \hat{K}\lambda q^*}} \sum_{i=1}^N \frac{1}{\tau_i + \frac{\lambda q^*}{\hat{K}}} - C. \quad (14)$$

This converges to $+\infty$ for $\lambda \rightarrow \frac{p_0}{\hat{K}q^*}$, $\lambda < \frac{p_0}{\hat{K}q^*}$. Hence, $\dot{\tilde{q}} > 0$ for all $1 < \lambda < \frac{p_0}{\hat{K}q^*}$, otherwise there would have to be another equilibrium in $[\delta_{th}, \delta_{max}]$, which we excluded.

In order to show that solutions starting in \mathcal{S} are unbounded, we are going to assume the opposite. To this end, let $x_0 \in \mathcal{S}$ and assume that starting in x_0 the system is bounded. Then, the solution has two accumulation points because it needs to have one, and if there was only one, then that would be an equilibrium point. One of these accumulation points needs to be strictly larger than the other in at least one component. Using that all components of the solution are monotonically increasing, it is easy to see that this scenario is impossible. ■

Model Validation: In the previous analysis, we used a fluid-like argument in the dynamic system representation of the model (2) and (3). In such a model, one can easily compute (numerically) the equilibrium states in case it exists.

Comment: This is just a coarse approximation to what is actually happening. The network is certainly not transporting fluid,

but rather discrete packets. It is also discrete event in nature and not continuous. Furthermore, each flow estimates δ and cannot see an exact value of this quantity for a variety of reasons—filtered measurements of δ (difference between weighted average of the RTT and its minimum observed value), different RTTs. Finally, our networks do not support an infinite number of flows, but rather a small number of flows. Nevertheless, the fluid model yields some insights into the qualitative behavior of our network and does approximate in some sense the network dynamics.

Extended Model (Multiple Bottlenecks): We now want to extend our model to handle networks of a more complicated topology. We emphasize that extending our results to this case is nontrivial; see [31] for examples of networks that are single-bottleneck-stable, but become unstable in the multiple-bottleneck case due to queue interactions. In what follows, we reuse the approach given in [32]. Although Wang *et al.* use a different model in [32], we can follow their approach to obtain local stability results if our algorithm uses the following piecewise linear probability function:

$$p(\delta) = K \cdot \delta \quad (15)$$

which is an approximation of the original probability function, shown in Fig. 1. We consider the network consisting of a set of resources $\mathcal{J} = \{1, \dots, J\}$ (links) with capacities $\{C_1, \dots, C_J\}$. We identify each route $\{1, \dots, N\}$ with a subset $i \subset \mathcal{J}$ and assign it the window size $W_i(t)$, which is to be understood as the cumulative sum of the window sizes of the flows along route i and is to take values in the positive real numbers for all $t \in \mathbb{R}_+$. Each link is equipped with a queue. The queue sizes $q_1(t), \dots, q_J(t)$ are also to take values in the positive real numbers for all $t \in \mathbb{R}_+$.

We can model the W_i in the following way

$$\dot{W}_i(t) = \frac{1}{\text{RTT}_i + \delta_i(t)} \left(1 - \frac{p_i}{2} W_i^2(t)\right). \quad (16)$$

This is the same equation as in the single-bottleneck case (2), where we use $\delta_i(t) = \sum_{j \in i} \frac{q_j(t)}{C_j}$ and

$$\dot{q}_j(t) = \begin{cases} \sum_{i: j \in i} \frac{W_i(t)}{\text{RTT}_i + \delta_i(t)} - C_j, & \text{for } q_j > 0 \\ \left\{ \sum_{i: j \in i} \frac{W_i(t)}{\text{RTT}_i + \delta_i(t)} - C_j \right\}^+, & \text{for } q_j = 0 \end{cases} \quad (17)$$

for $i = 1, \dots, N$, $j = 1, \dots, J$. The first equation is familiar from the single-bottleneck case, and the second equation just states that queueing dynamics simply depend on the difference between total arrival rate and capacity at each link, with the constraint that the queue never takes negative values, but rather stays at zero until the dynamic is positive again. Setting (16) and (17) to zero, and assuming that at equilibria we have $\frac{1}{\text{RTT}_i + \delta_i(t)} = \frac{1}{\text{RTT}_i + \delta_i}$ for all $i = 1, \dots, N$, then

$$W_i^* = \sqrt{\frac{2}{p_i^*}} \quad (18)$$

$$\sum_{i: j \in i} \frac{W_i^*}{\text{RTT}_i + \delta_i} = C_j \quad (19)$$

$i = 1, \dots, N$, $j = 1, \dots, J$.

To ease exposition, we give our mathematical proofs for piecewise linear probability functions $p(\delta)$. All results

generalize to the nonlinear case by replacing the piecewise linear function $p(\delta)$ with the appropriate linearizations (Jacobians). This extension is given formally (without proof) as Corollary 4.5. Full details and proofs can be found in [33].

To prove the desired results, we partition our system by choosing two special probability functions, showing certain properties and fitting the two systems together again. At first, we choose

$$p_i(\delta_i) = K\delta_i \quad (20)$$

where δ_i is the sum over all queueing delays experienced by flow i and $K \in \mathbb{R}_+$ for all $i = 1, \dots, N$. Then, we look at

$$p_i(\delta_i) = p_0 - \hat{K}\delta_i \quad (21)$$

where again δ_i is the sum over all queueing delays experienced by flow i and $p_0 > 0$, $\hat{K} \in \mathbb{R}_+$ for all $i = 1, \dots, N$, where we make the additional assumption that $\delta_i \leq \frac{p_0}{\hat{K}_i}$, $i = 1, \dots, N$ such that $p_i(\delta_i)$ is always positive.

In the final step, we allow each flow to switch between the probability functions (20) and (21) in a deterministic manner so that we get an overall probability function

$$p_i(\delta_i) = \begin{cases} K\delta_i, & \text{for } 0 < \delta_i \leq \delta_{\text{th}} \\ p_0 - \hat{K}\delta_i, & \text{for } \delta_{\text{th}} < \delta_i \leq \delta_{\text{max}}. \end{cases} \quad (22)$$

Multiple-Bottleneck Result: In this section, we make a number of simplifying assumptions to keep the analysis short. Most notably, we assume that, close to equilibria, $\text{RTT}_i + \delta(t)$ is approximately constant. This assumption is consistent with the methodology in [32] and [34] and is valid for some network types. Again, a more detailed analysis along the lines of that presented for the single-bottleneck case can be presented, but due to space limitations, this is beyond the scope of the present paper. The analysis itself follows closely [32].

We can now state the main result of this section. This is a direct extension of the result for the single-bottleneck case, albeit with an extremely involved proof.

We assume that the equilibria of our system lie in the appropriate area. Especially, we assume for the stable equilibrium, which is the topic of the following theorem, that $\delta_i^* < \delta_{\text{th}}$ for all $i = 1, \dots, N$. This is a design task as their location depends on the calibration of the algorithm's parameters.

Theorem 4.4: The system described by (16), (17), and (22) has only one asymptotically stable fixed point. Each other fixed point is unstable.

Proof: The proof consists of three parts. First, we will consider a system, where every flow uses (20) as its probability function. We will be able to show that there is a unique globally asymptotically stable fixed point to this system.

To ease the exposition of the final step, in the second step we will consider a system, where all flows use (21) as their probability function. We will be able to show that all equilibria of this system are unstable. In the final step, we consider (22), which will again prove unstable as long as some flows are in the high-delay area.

i) Let every flow use (20) as its probability function. We calculate the equilibrium $(W_1^*, \dots, W_N^*, q_1^*, \dots, q_J^*)$ using

(18) and (19). Next, we consider the following transformation of coordinates $\tilde{W}_i(t) = W_i(t) - W_i^*$ and $\tilde{q}_j(t) = q_j(t) - q_j^*$. We get the following system:

$$\dot{\tilde{W}}_i(t) = -\frac{1}{2} \frac{K}{\text{RTT}_i + \delta_i} \left(\sum_{j \in i} \frac{\tilde{q}_j(t)}{C_j} (\tilde{W}_i(t) + W_i^*)^2 + \sum_{j \in i} \frac{q_j^*}{C_j} (\tilde{W}_i^2(t) + 2W_i^* \tilde{W}_i(t)) \right) \quad (23)$$

$$\dot{\tilde{q}}_j(t) = \sum_{i: j \in i} \frac{\tilde{W}_i(t)}{\text{RTT}_i + \delta_i} \quad (24)$$

as can be seen from

$$\begin{aligned} \dot{\tilde{W}}_i(t) &= \dot{W}_i(t) - \dot{W}_i^* \\ &= \frac{1}{\text{RTT}_i + \delta_i} \left(1 - \frac{p_i}{2} W_i^2(t) \right) \\ &= \frac{1}{\text{RTT}_i + \delta_i} \left(1 - \frac{1}{2} \sum_{j \in i} K \frac{q_j(t)}{C_j} W_i^2(t) \right) \\ &= \frac{1}{\text{RTT}_i + \delta_i} \\ &\quad \times \left(1 - \frac{1}{2} \sum_{j \in i} K \left(\frac{\tilde{q}_j(t)}{C_j} + \frac{q_j^*}{C_j} \right) (\tilde{W}_i(t) + W_i^*)^2 \right) \\ &= \frac{1}{\text{RTT}_i + \delta_i} \left(1 - \frac{1}{2} (W_i^*)^2 \sum_{j \in i} K \frac{q_j^*}{C_j} \right. \\ &\quad \left. - \frac{1}{2} \sum_{j \in i} K \frac{\tilde{q}_j(t)}{C_j} (\tilde{W}_i(t) + W_i^*)^2 \right. \\ &\quad \left. - \frac{1}{2} \sum_{j \in i} K \frac{q_j^*}{C_j} (\tilde{W}_i^2(t) + 2W_i^* \tilde{W}_i(t)) \right) \\ &= -\frac{1}{2} \frac{K}{\text{RTT}_i + \delta_i} \left(\sum_{j \in i} \frac{\tilde{q}_j(t)}{C_j} (\tilde{W}_i(t) + W_i^*)^2 \right. \\ &\quad \left. + \sum_{j \in i} \frac{q_j^*}{C_j} (\tilde{W}_i^2(t) + 2W_i^* \tilde{W}_i(t)) \right) \quad (25) \end{aligned}$$

and

$$\begin{aligned} \dot{\tilde{q}}_j(t) &= \dot{q}_j(t) - \dot{q}_j^* \\ &= \sum_{i: j \in i} \frac{W_i(t)}{\text{RTT}_i + \delta_i} - C_j \\ &= \sum_{i: j \in i} \frac{\tilde{W}_i(t) + W_i^*}{\text{RTT}_i + \delta_i} - C_j \\ &= \sum_{i: j \in i} \frac{\tilde{W}_i(t)}{\text{RTT}_i + \delta_i} + \sum_{i: j \in i} \frac{W_i^*}{\text{RTT}_i + \delta_i} - C_j \\ &= \sum_{i: j \in i} \frac{\tilde{W}_i(t)}{\text{RTT}_i + \delta_i}. \quad (26) \end{aligned}$$

A Lyapunov function for our system is given by

$$V(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}_1, \dots, \tilde{q}_J) = \sum_{i=1}^N \frac{1}{(W_i^*)^2} \tilde{W}_i^2 + \frac{1}{2} \sum_{j=1}^J \frac{K}{C_j} \tilde{q}_j^2. \quad (27)$$

It is clearly positive definite, and furthermore

$$\begin{aligned} \dot{V}(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}_1, \dots, \tilde{q}_J) &= 2 \sum_{i=1}^N \frac{1}{(W_i^*)^2} \tilde{W}_i \dot{\tilde{W}}_i + \sum_{j=1}^J \frac{K}{C_j} \tilde{q}_j \dot{\tilde{q}}_j \\ &= -\sum_{i=1}^N \frac{1}{(W_i^*)^2} \frac{K}{\text{RTT}_i + \delta_i} \tilde{W}_i^2 (\tilde{W}_i + 2W_i^*) \left(\sum_{j \in i} \frac{\tilde{q}_j + q_j^*}{C_j} \right) \\ &\leq 0. \quad (28) \end{aligned}$$

demonstrates the stability of the equilibrium. Using LaSalle's invariance principle, we can show that it is in fact globally asymptotically stable. To this end, we look at the set $\dot{V}^{-1}(0)$. It is

$$\begin{aligned} \dot{V}^{-1}(0) &= \{(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}_1, \dots, \tilde{q}_J) | \dot{V}(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}_1, \dots, \tilde{q}_J) = 0\} \\ &= \{(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}_1, \dots, \tilde{q}_J) | (\tilde{W}_1 = \dots = \tilde{W}_N = 0) \\ &\quad \text{or } (\tilde{q}_1 = -q_1^*, \dots, \tilde{q}_J = -q_J^*)\}. \quad (29) \end{aligned}$$

A simple deliberation shows that the only invariant subset contained in $\dot{V}^{-1}(0)$ is the origin. If we consider a vector from $\dot{V}^{-1}(0)$, which is not the origin, it can be easily seen from (23) and (24) that we get pushed out of $\dot{V}^{-1}(0)$. Thus, the equilibrium is globally asymptotically stable. This concludes this part of the proof.

ii) In the second step, we choose the probability function for every flow to be (21).

We calculate again an equilibrium $(W_1^*, \dots, W_N^*, q_1^*, \dots, q_M^*)$ using (18) and (19). Next, we consider the following transformation of coordinates $\tilde{W}_i(t) = W_i(t) - W_i^*$ and $\tilde{q}_j(t) = q_j(t) - q_j^*$. For $i = 1, \dots, N$ and $j = 1, \dots, J$, we get the following normalized dynamics:

$$\begin{aligned} \dot{\tilde{W}}_i(t) &= -\frac{1}{2} \frac{1}{\text{RTT}_i + \delta_i} \\ &\quad \times \left(-\left(\sum_{j \in i} \tilde{K} \frac{\tilde{q}_j(t)}{C_j} \right) (\tilde{W}_i(t) + W_i^*)^2 \right. \\ &\quad \left. + \left(p_0 - \sum_{j \in i} \tilde{K} \frac{q_j^*}{C_j} \right) (\tilde{W}_i^2(t) + 2W_i^* \tilde{W}_i(t)) \right) \quad (30) \end{aligned}$$

$$\dot{\tilde{q}}_j(t) = \sum_{i: j \in i} \frac{\tilde{W}_i(t)}{\text{RTT}_i + \delta_i} \quad (31)$$

where we used the same trick as for (23) and (24). A Lyapunov function that will prove instability is given by

$$V(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}_1, \dots, \tilde{q}_J) = \sum_{i=1}^N \frac{1}{(\tilde{W}_i^*)^2} \tilde{W}_i^2 - \frac{1}{2} \sum_{j=1}^J \frac{\hat{K}}{C_j} \tilde{q}_j^2. \quad (32)$$

In every neighborhood of the equilibrium, we can find $(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}_1, \dots, \tilde{q}_J) \in R_+^{N+J}$ such that $V(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}_1, \dots, \tilde{q}_J)$ is negative. Furthermore

$$\begin{aligned} \dot{V}(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}_1, \dots, \tilde{q}_J) &= 2 \sum_{i=1}^N \frac{1}{(\tilde{W}_i^*)^2} \tilde{W}_i \dot{\tilde{W}}_i - \sum_{j=1}^J \frac{\hat{K}}{C_j} \tilde{q}_j \dot{\tilde{q}}_j \\ &= - \sum_{i=1}^N \frac{1}{(\tilde{W}_i^*)^2} \tilde{W}_i^2 \frac{1}{\text{RTT}_i + \delta_i} \\ &\quad \times \left(p_0 - \sum_{j \in i} \hat{k} \frac{q_j}{C_j} \right) (\tilde{W}_i + 2\tilde{W}_i^*) \leq 0. \end{aligned} \quad (33)$$

shows the instability of the considered system by the Lyapunov Instability Theorem (see, e.g., [30, Theorem 3.2.37]), which is applicable because there are no invariant sets in $\dot{V}^{-1}(0)$. The argument here is the same as in 1).

- iii) Now, let each flow use (22) as its probability function. Thus, each flow in our system uses one of the probability functions (20) and (21), with the above defined switching rule. Let I^s denote the set of flows using initially (20) and by I^u the set of flows using initially (21). For any partition I^s, I^u of $\{1, \dots, N\}$, (18) and (19) give us an equilibrium of our system. We will show that this equilibrium can only be stable if I^u is the empty set. If I^u is the empty set, then the equilibrium is asymptotically stable. Let us assume $I^u \neq \emptyset$. The dynamic becomes

$$\begin{aligned} \dot{\tilde{W}}_i(t) &= - \frac{1}{2} \frac{1}{\text{RTT}_i + \delta_i} \\ &\quad \times \left(K \sum_{j \in i, j \in I^s} \frac{\tilde{q}_j(t)}{C_j} (\tilde{W}_i(t) + \tilde{W}_i^*)^2 \right. \\ &\quad \left. + K \sum_{j \in i, j \in I^u} \frac{q_j^*}{C_j} (\tilde{W}_i^2(t) + 2\tilde{W}_i^* \tilde{W}_i(t)) \right. \\ &\quad \left. - \left(\sum_{j \in i, j \in I^u} \hat{K} \frac{\tilde{q}_j(t)}{C_j} \right) (\tilde{W}_i(t) + \tilde{W}_i^*)^2 \right. \\ &\quad \left. + \left(p_0 - \sum_{j \in i, j \in I^u} \hat{K} \frac{q_j^*}{C_j} \right) (\tilde{W}_i^2(t) + 2\tilde{W}_i^* \tilde{W}_i(t)) \right) \end{aligned} \quad (34)$$

$$\dot{\tilde{q}}_j(t) = \sum_{i: j \in i} \frac{\tilde{W}_i(t)}{\text{RTT}_i + \delta_i}. \quad (35)$$

The function

$$V(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}_1, \dots, \tilde{q}_J) = \sum_{i=1}^N \frac{1}{(\tilde{W}_i^*)^2} \tilde{W}_i^2 + \frac{1}{2} \sum_{j \in I^s} \frac{K}{C_j} \tilde{q}_j^2 - \frac{1}{2} \sum_{j \in I^u} \frac{\hat{K}}{C_j} \tilde{q}_j^2 \quad (36)$$

is a Lyapunov function that will prove instability. In every neighborhood of the equilibrium, we can find $(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}_1, \dots, \tilde{q}_J) \in R_+^{N+J}$ such that $V(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}_1, \dots, \tilde{q}_J)$ is negative. Furthermore, (37) proves instability of the system by the Lyapunov Instability Theorem [30, Theorem 3.2.37], which is again applicable because there are no invariant sets in $\dot{V}^{-1}(0)$. The argument here is the same as in 1).

Notice how we used findings from (28) and (33) to develop the following:

$$\begin{aligned} \dot{V}(\tilde{W}_1, \dots, \tilde{W}_N, \tilde{q}_1, \dots, \tilde{q}_J) &= 2 \sum_{i=1}^N \frac{1}{(\tilde{W}_i^*)^2} \tilde{W}_i \dot{\tilde{W}}_i + \sum_{j \in I^s} \frac{K}{C_j} \tilde{q}_j^2 - \sum_{j \in I^u} \frac{\hat{K}}{C_j} \tilde{q}_j \dot{\tilde{q}}_j \\ &= - \sum_{i=1}^N \frac{1}{(\tilde{W}_i^*)^2} \frac{1}{\text{RTT}_i + \delta_i} \tilde{W}_i^2 (\tilde{W}_i + 2\tilde{W}_i^*) \\ &\quad \times \left(\left(\sum_{j \in i, j \in I^s} K \frac{q_j}{C_j} \right) + \left(p_0 - \sum_{j \in i, j \in I^u} \hat{K} \frac{q_j}{C_j} \right) \right) \\ &\leq 0 \end{aligned} \quad (37)$$

The stability results for nonlinear probability functions follow from [33, Theorem 6.8]. We state the relevant special case in the following corollary.

Corollary 4.5: Let x^* be a hyperbolic equilibrium of the system described by (16), (17), and arbitrary smooth probability functions $p_i(\delta_i)$, $i = 1, \dots, N$. Then, we have the following.

- 1) x^* is locally asymptotically stable for the system if for all $i = 1, \dots, N$ the functions $p_i(\delta_i)$ are strictly increasing in a neighborhood of x^* .
- 2) x^* is unstable for the system if for one $i = 1, \dots, N$ the function $p_i(\delta_i)$ is strictly decreasing in a neighborhood of x^* .

The assumption of hyperbolicity² cannot be omitted. However, we suppose that it is a generic property of our system. From Corollary 4.5, it is now clear that any equilibrium of our system must be unstable if even one flow is in the high-delay region, while the unique equilibrium in the low-delay region is asymptotically stable.

Comment: The practical interpretation of this result is the following. If we consider Fig. 1, all solutions converge to the unique attractive fixed point left of δ_{th} as long as we stay left of any equilibrium on the right side of δ_{th} . Once we get on the right side of such an equilibrium, as may happen if loss-based flows are present, we might get driven further into the high-

²An equilibrium of a nonlinear system is called hyperbolic if its Jacobian does not have eigenvalues on the imaginary axis. By the continuous dependence of eigenvalues on matrix entries, almost all matrices have this property.

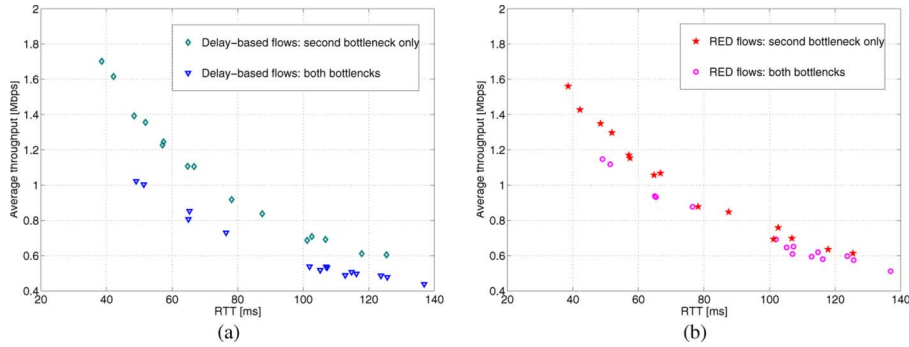


Fig. 11. Comparison of 30 flows in a multiple-bottleneck scenario in terms of the average throughput: (a) all flows delay-based and (b) all flows RED.

delay region. However, due to the fluctuations in the queue sizes driven by the multiplicative backoff behavior of the flows, there is always a positive possibility that we reach a position left of the equilibrium. If this happens, the dynamic drives the solutions to the low-delay equilibrium.

V. CASE STUDIES

We now present a number of case studies to illustrate some important features of our algorithm.

The first two are *comparative studies*. We first show that our algorithm effectively emulates a network of buffers in which RED AQM's are deployed, then briefly compare Cx-TCP to a recently proposed delay-based algorithm that purports to solve the coexistence problem—namely, the PERT algorithm [17].

Next, we present an *application study* in which we show, again briefly, that Cx-TCP may be of use in certain situations where high delays are inevitable (but nevertheless undesirable). Specifically, we examine the High-Speed TCP variant.

A. Comparative Study: Emulation of RED AQM

We first show that networks in which our algorithm is deployed (and in which no loss-based flows are present) have similar characteristics to networks where RED AQM schemes are deployed. These results are consistent with the simulation studies presented by Bhandarkar *et al.* in [17].

We reuse the multiple-bottleneck scenario, discussed previously in Section III-B. The parameters for the RED model are adjusted to be the same as the corresponding parameters of Cx-TCP, namely δ_{\min} , δ_{th} , and p_{\max} . Fig. 11 illustrates the results for the average throughput for 30 flows, half of which traverse both bottlenecks. As can be observed, the performance of the proposed algorithm in terms of average throughput is similar to RED AQM. Consequently, if all flows use Cx-TCP, the queueing delay can be kept at low levels, and the network has similar fairness properties to that of a RED network.

B. Application Study: High-Speed TCP

We now show that our basic idea can also be applied to enhance the performance of high-speed networks. We begin by modifying Sally Floyd's High-Speed TCP (HS-TCP; its details can be found in [35]) to incorporate our backoff strategy. In this situation, our backoff strategy offers a number of benefits. These include not only a low queueing delay, but also enhancing the network fairness properties by reducing the likelihood of flow synchronization (by means of the probabilistic backoff). A series of tests in a single-bottleneck scenario (low delay, fairness,

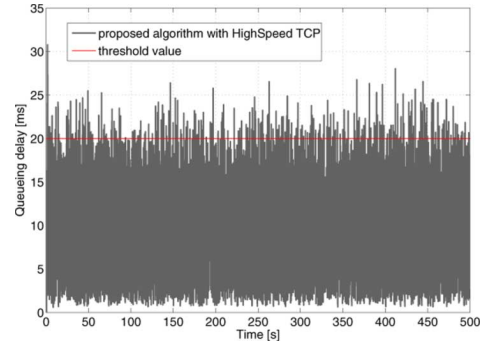


Fig. 12. Delay-based HS-TCP in a single-bottleneck scenario: queueing delay for 30 flows.

loss rate, and on/off switching) with the same parameters as specified in Section III-A has been carried out. Fig. 12 illustrates the basic feature of the presented proposal, the ability to keep the queueing delay low in a homogeneous scenario consisting of flows that use HS-TCP extended with our algorithm.

We examine the ability of delay-based flows to coexist fairly with standard HS-TCP flows. Fig. 13 depicts the normalized average throughput for a network with three different mixes of standard and extended HS-TCP flows (a total of 30 flows). As in the case of standard TCP, we can observe that the proposed algorithm guarantees fair coexistence in terms of average throughput rates, between loss and delay-based flows. Next, we check the network loss rate. We observe in Fig. 14 that HS-TCP flows using the proposed algorithm experience similar loss rates as standard HS-TCP flows. In particular, a comparison of 20 delay-based HS-TCP flows (using the proposed algorithm) coexisting with 10 standard HS-TCP flows with a scenario in which all 30 flows are standard (loss-based) HS-TCP witnesses no significant difference in network loss rate.

Finally, we test dynamic properties of the presented algorithm in a high-speed network scenario consisting of 20 delay-based HS-TCP flows and 10 intermittent standard HS-TCP flows (we repeat analogous on-off switching test from Section III-A). Fig. 15 illustrates the results of such an experiment both in terms of queueing delay and packet loss ratio. It can be clearly seen that the algorithm is able to detect the presence of loss-based flows and then, once these flow are off, is able to revert back to the low-queueing delay operation.

VI. CONCLUSION

In this paper, we have presented a method that can be used to ensure that delay-based AIMD flows operate as loss-based flows

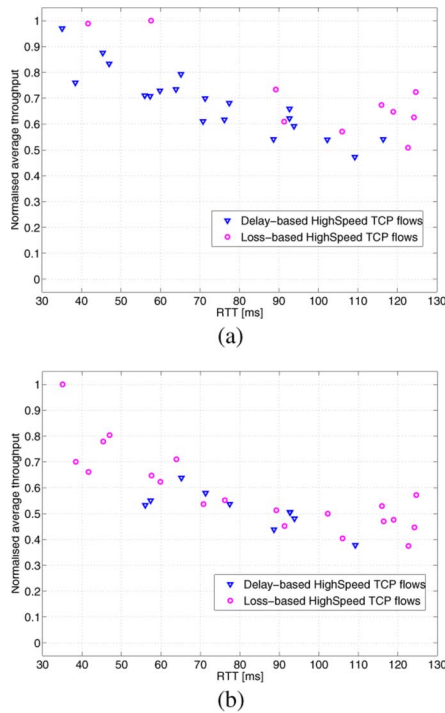


Fig. 13. Coexistence of delay-based HS-TCP flows and standard HS-TCP flows in terms of the normalized average throughput for (a) the (20, 10) and (b) the (10, 20) mixes of flows.

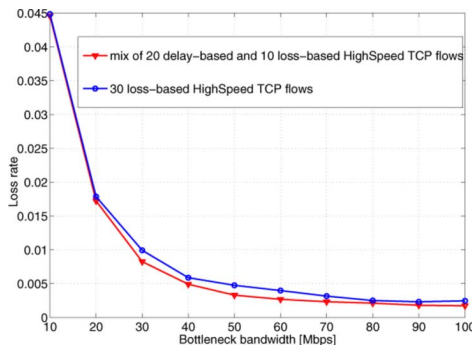


Fig. 14. Comparison in terms of loss rate: 20 delay-based HS-TCP flows coexisting with 10 standard HS-TCP flows and 30 standard HSTCP flows.

when loss-based flows are present in the network, allowing fair coexistence with their loss-based counterparts, and otherwise revert to delay-based operation mode. We demonstrated both in experiments and analytically that the proposed solution guarantees the aforementioned features, and that appropriate adjusting of the maximum equilibrium loss rate permits wide application of Cx-TCP, covering a range of different scenarios, e.g., fast networks. Finally, to conclude the description of the presented algorithm, we denote a number of potential limitations of Cx-TCP. First, our algorithm works best in multiplexed environments with standing queues. In situations where this assumption is not valid, some unfairness may be introduced when loss-based flows are present. A crucial part of the algorithm is the assumption that all flows use the same per-packet backoff probability function and sense the same queueing delay. If this assumption is not valid, unfairness can be introduced. Also, the mathematical analysis presented in Section IV is only valid in situations when the fluid model of TCP provides a description of the queueing dynamics. However, the qualitative description

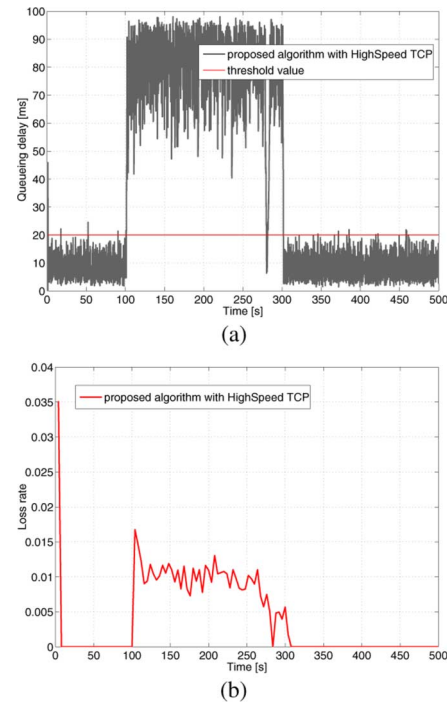


Fig. 15. Coexistence of 20 delay-based HS-TCP flows with 10 loss-based HS-TCP flows (standard HS-TCP) switching on and off: (a) queueing delay at the bottleneck; (b) loss rate.

of the mechanism provided in Section II is valid and is independent of the fluid model.

ACKNOWLEDGMENT

The authors would like to thank F. Wirth with the Institut für Mathematik, Universität Würzburg, Germany, for help with the stability proof for the multiple-bottleneck scenarios.

REFERENCES

- [1] V. Jacobson, "Congestion avoidance and control," in *Proc. ACM SIGCOMM*, 1988, pp. 314–329.
- [2] R. Jain, "A delay-based approach for congestion avoidance in interconnected heterogeneous computer networks," *Comput. Commun. Rev.*, vol. 19, no. 5, pp. 56–71, 1989.
- [3] A. Tang, J. Wang, S. Low, and M. Chiang, "Equilibrium of heterogeneous congestion control protocols," in *Proc. 24th IEEE INFOCOM*, Mar. 2005, vol. 2, pp. 1338–1349.
- [4] L. Brakmo, S. O'Malley, and L. Peterson, "TCP Vegas: New techniques for congestion detection and avoidance," in *Proc. ACM SIGCOMM*, 1994, pp. 24–35.
- [5] L. Brakmo and L. Peterson, "TCP Vegas: End to end congestion avoidance on a global internet," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 8, pp. 1465–1480, Oct. 1995.
- [6] Z. Wang and J. Crowcroft, "Eliminating periodic packet losses in the 4.3-Tahoe BSD TCP congestion control algorithm," *Comput. Commun. Rev.*, vol. 22, no. 2, pp. 9–16, 1992.
- [7] W. Feng and S. Vanichpun, "Enabling compatibility between TCP Reno and TCP Vegas," in *Proc. Symp. Appl. Internet*, 2003, pp. 301–308.
- [8] K. Sriji, L. Jacob, and A. Ananda, "TCP Vegas-A: Solving the fairness and rerouting issues of TCP Vegas," in *Proc. 22nd IEEE IPCCC*, Apr. 2003, pp. 309–316.
- [9] U. Hengartner, J. Bolliger, and T. Gross, "TCP Vegas revisited," in *Proc. 19th IEEE INFOCOM*, 2000, pp. 1546–1555.
- [10] H. Choe and S. Low, "Stabilized Vegas," in *Proc. 22nd IEEE INFOCOM*, 2003, vol. 3, pp. 2290–2300.
- [11] S. Low, L. Peterson, and L. Wang, "Understanding Vegas: A duality model," *J. ACM*, vol. 49, no. 2, pp. 207–235, Mar. 2002.

- [12] R. Jain and K. Ramakrishnan, "Congestion avoidance in computer networks with a connectionless network layer: Concepts, goals and methodology," in *Proc. Comput. Netw. Symp.*, 1988, pp. 134–143.
- [13] J. Wang, D. X. Wei, and S. Low, "Modelling and stability of FAST TCP," in *Proc. 24th IEEE INFOCOM*, 2005, vol. 2, pp. 938–948.
- [14] K. Tan, J. Song, Q. Zhang, and M. Sridharan, "A compound TCP approach for high-speed and long distance networks," Microsoft Research, Redmond, WA, Tech. Rep. MSR-TR-2005-86, 2005.
- [15] D. Leith, J. Heffner, R. Shorten, and G. McCullagh, "Delay-based AIMD congestion control," in *Proc. 5th PFLDnet*, 2007, pp. 1–6.
- [16] D. Leith, R. Shorten, G. McCullagh, L. Dunne, and F. Baker, "Making available base RTT for use in congestion control applications," *IEEE Commun. Lett.*, vol. 12, no. 6, pp. 429–431, Jun. 2008.
- [17] S. Bhandarkar, A. Reddy, Y. Zhang, and D. Loguinov, "Emulating AQM from end hosts," *Comput. Commun. Rev.*, vol. 37, no. 4, pp. 349–360, Oct. 2007.
- [18] K. Kotla and A. Reddy, "Making a delay-based protocol adaptive to heterogeneous environments," in *Proc. 16th IWQoS*, Jun. 2008, pp. 100–109.
- [19] M. Podlesny and C. Williamson, "Providing fairness between TCP NewReno and TCP Vegas with RD network services," in *Proc. 18th IWQoS*, 2010, pp. 1–9.
- [20] J. Martin, A. Nilsson, and I. Rhee, "Delay-based congestion avoidance for TCP," *IEEE/ACM Trans. Netw.*, vol. 11, no. 3, pp. 356–369, Jun. 2003.
- [21] R. Prasad, M. Jain, and C. Dovrolis, "On the effectiveness of delay-based congestion avoidance," in *Proc. 2nd PFLDnet*, 2004, pp. 1–2.
- [22] G. McCullagh, "Exploring delay based TCP congestion control," Master's thesis, Hamilton Institute, NUI Maynooth, Maynooth, Ireland, 2008.
- [23] C. Kellett, R. Shorten, and D. Leith, "A review of delay-based congestion control," Cisco, San Jose, CA, 2006.
- [24] Ł. Budzisz, R. Stanojević, R. Shorten, and F. Baker, "A strategy for fair coexistence of loss and delay-based congestion control algorithms," *IEEE Commun. Lett.*, vol. 13, no. 7, pp. 555–557, Jul. 2009.
- [25] Ł. Budzisz, R. Stanojević, A. Schlote, R. Shorten, and F. Baker, "On the fair coexistence of loss- and delay-based TCP," in *Proc. 17th IWQoS*, 2009, pp. 1–9.
- [26] A. Tang, J. Wang, S. Low, and M. Chiang, "Equilibrium of heterogeneous congestion control: Existence and uniqueness," *IEEE/ACM Trans. Netw.*, vol. 15, no. 4, pp. 824–837, Aug. 2007.
- [27] A. Tang, X. Wei, S. Low, and M. Chiang, "Equilibrium of heterogeneous congestion control: Optimality and stability," *IEEE/ACM Trans. Netw.*, vol. 18, no. 3, pp. 844–857, Jun. 2010.
- [28] D. A. Hayes and G. Armitage, "Improved coexistence and loss tolerance for delay based TCP congestion control," in *Proc. 35th Annu. IEEE LCN*, Oct. 2010, pp. 24–31.
- [29] "The Network Simulator ns-2," [Online]. Available: <http://www.isi.edu/nsnam/ns/>
- [30] D. Hinrichsen and A. Pritchard, *Mathematical Systems Theory I. Modelling, State Space Analysis, Stability and Robustness*. Berlin, Germany: Springer-Verlag, 2004, vol. 48.
- [31] D. Bauso, L. Giarre, and G. Neglia, "AQM stability in multiple bottleneck networks," in *Proc. IEEE ICC*, Jun. 2004, vol. 4, pp. 2267–2271.
- [32] L. Wang, L. Cai, X. Liu, X. S. Shen, and J. Zhang, "Stability analysis of multiple-bottleneck networks," *Comput. Netw.*, vol. 53, no. 3, pp. 338–352, 2009.
- [33] A. Schlote, "Stability of TCP models," Diplomarbeit, Julius-Maximilians, Universität Würzburg, Würzburg, Germany, 2009.
- [34] S. H. Low, F. Paganini, J. Wang, and J. C. Doyle, "Linear stability of TCP/RED and a scalable control," *Comput. Netw.*, vol. 43, pp. 633–647, Dec. 2003.
- [35] S. Floyd, "Highspeed TCP for large congestion windows," RFC 3649, 2003 [Online]. Available: <http://www.ietf.org/rfc/rfc3649.txt>



Łukasz Budzisz (S'05–M'09) received the M.Eng.Sc. degree in electronics and telecommunication from the Technical University of Łódź, Łódź, Poland, in 2003, and the Ph.D. degree in signal theory and communications from the Technical University of Catalunya, Barcelona, Spain, in 2009.

He has worked as a Research Fellow with the Hamilton Institute, Maynooth, Ireland. Currently, he is a Postdoctoral Researcher with the Technical University of Berlin, Berlin, Germany. His research interest include network congestion control, mobility, and wireless networks.



Rade Stanojević received the B.Sc. degree in mathematics from University of Nis, Nis, Serbia, in 2003, and the Ph.D. degree from the Hamilton Institute, NUI Maynooth, Ireland, in 2007.

Since 2010, he has been a Staff Researcher with the IMDEA Networks Institute, Madrid, Spain. His current research interests span performance evaluation, network economics, and energy-aware computing.

Dr. Stanojević was awarded the ACM SIGMETRICS 2008 Kenneth C. Sevcik Outstanding Student Paper Award and the IEEE IWQoS 2009 Best Paper

Award for his work on decentralized cloud control.



Arie Schlote received the Diploma in mathematics from the University of Würzburg, Würzburg, Germany, in 2010, and is currently pursuing the Ph.D. degree at the Hamilton Institute, Maynooth, Ireland, where he works on Markov chains and their application to road traffic modeling.

His research interests include stability theory of dynamical systems and matrix theory.



Fred Baker a Cisco Fellow, has been active in technology development and Internet Standardization since the 1980s. He participated in early development of IEEE 802.1d switching and IP routing. In the IETF, he has written or edited RFCs on a variety of topics and chaired both working groups and the IETF itself. At Cisco, his group supports research at universities. He is looked to for research advice and mentorship both within and outside the company. At this time, he is the IETF's Voting Member on the U.S. NIST Smart Grid Interoperability Panel,

a member of the SGIP's Architecture Committee, and co-chair of the IPv6 Operations Working Group.



Robert Shorten received the M.Eng.Sc. and Ph.D. degrees from University College Dublin, Dublin, Ireland, in 1993 and 1996, respectively.

From 1993 to 1996, he held a Marie Curie Fellowship to conduct research at the Daimler-Benz Research Institute for Information Technology, Berlin, Germany. In 1996, he was invited as a Visiting Fellow to the Center for Systems Science, Yale University, New Haven, CT, to work with Prof. K. S. Narendra. Since returning to Ireland in 1997 as the recipient of a European Presidential Fellowship, he has built up a

research group in the area of switching systems. He is an Editor of the *IEEE Proceedings on Control* and a co-founder of the Hamilton Institute, NUI Maynooth, Ireland.