Queue Dynamics With Window Flow Control

Ao Tang, *Member, IEEE*, Lachlan L. H. Andrew, *Senior Member, IEEE*, Krister Jacobsson, Karl H. Johansson, Håkan Hjalmarsson, and Steven H. Low, *Fellow, IEEE*

***Abstract*—This paper develops a new model that describes the queueing process of a communication network when data sources use window flow control. The model takes into account the burstiness in sub-round-trip time (RTT) timescales and the instantaneous rate differences of a flow at different links. It is generic and independent of actual source flow control algorithms. Basic properties of the model and its relation to existing work are discussed. In particular, for a general network with multiple links, it is demonstrated that spatial interaction of oscillations allows queue instability to occur even when all flows have the same RTTs and maintain constant windows. The model is used to study the dynamics of delay-based congestion control algorithms. It is found that the ratios of RTTs are critical to the stability of such systems, and previously unknown modes of instability are identified. Packet-level simulations and testbed measurements are provided to verify the model and its predictions.**

***Index Terms*—Congestion control, stability.**

I. Introduction

NETWORK flow control has been studied since at least the 1970s; see [8] and references therein. During the explosive growth of the Internet in the last two decades, congestion control[1] algorithms have attracted much attention, in particular that of Transmission Control Protocol (TCP), which controls the majority of Internet traffic today. During the years, the research focus has changed. Starting with [11] in 1988, the focus was on packet-level "microscopic" properties for simple network topologies, and researchers usually relied on qualitative reasoning, simulations, and analytic models of individual flows [3], [15], [37]. Around 1998, following the seminal paper [18], the focus shifted to fluid models with tools from optimization and control theories, seeking quantitative "macroscopic" understanding for networks with potentially arbitrary topology [21], [25], [31], [35], [36].

Manuscript received December 16, 2008; accepted January 29, 2010; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor V. Misra. Date of publication April 26, 2010; date of current version October 15, 2010. This work was supported by NSF grants EIA-0303620, CNS-0911041, and CCF-0835706, DURIP Grant 53773-MA-RIP, ARO MURI Grant W911NF-08-1-0233, Swedish Research Council, and Australian Research Council grants DP0985322 and FT0991594. Partial and preliminary results have appeared in [13], [33], the Proceedings of IEEE Infocom, Phoenix, AZ, April 2008.

A. Tang is with Cornell University, Ithaca, NY 14853 USA.

L. L. H. Andrew is with the CAIA, Swinburne University of Technology, Hawthorn, Vic. 3122, Australia (e-mail: l.andrew@ieee.org).

K. Jacobsson and S. H. Low are with the California Institute of Technology, Pasadena, CA 91125 USA.

K. H. Johansson and H. Hjalmarsson are with the ACCESS Linnaeus Centre, Electrical Engineering, KTH, Stockholm 100 44, Sweden.

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.2010.2047951

[1]We use the terms flow control and congestion control interchangeably.

However, there is a significant gap between these two types of studies. In general, fluid models are more tractable, but ignore many (sometimes important) details of real protocols, such as the traffic burstiness within a round-trip time (RTT). Current TCPs use window flow control, and rapid changes in the window induce burstiness, like slow-start causing bursts of almost back to back packets. There are also other sources of burstiness such as cross traffic [42]. Most importantly, once such burstiness is generated, it is self-perpetuating due to *ACK-clocking*. Because a new packet is transmitted only when an acknowledgment is received, the bursty pattern of acknowledgments due to bursty transmission will cause the transmissions in the next RTT also to be bursty. This has been verified with simulations and real Internet tests [16].

Another issue is how to model the interactions of queues in a network with multiple links. Existing fluid models assume that each flow has an equal rate at all links it traverses [18]. In reality, this is not the case because, for example, the input rate of a downstream link can never exceed the sum of the capacities of its upstream links. Finally, a packet experiences queueing delays at different links at different times.[2] None of these effects is captured by existing fluid models.

These factors turn out to be important and can sometimes considerably affect system properties [33]. This paper aims at bridging the above-mentioned gap by developing a new model that captures these missing factors. More precisely, it focuses on describing queue trajectories under window flow control, including burstiness at timescales faster than one RTT, and the fact that the rate of a flow at upstream and downstream links can differ.

One of the main applications of this new model is the study of the dynamics of congestion control systems, in particular stability properties. Stability is crucial to ensure that fluctuations due to stochastically varying traffic are damped, and the network operates in a desirable region of the state space. Unstable protocols can amplify small fluctuations in cross traffic into large fluctuations in queue lengths. These can reduce throughput and cause jitter that can be detrimental to interactive services such as voice-over-IP. The dynamics of congestion control algorithms have been studied extensively—in particular, local stability [5], [9], [17], [19], [20], [24], [27], [28], [32], [38] and global stability [1], [6], [10], [41], [45]. However, most existing results are based on models with qualitatively different dynamical properties than the model proposed here. This paper will revisit the stability of the FAST TCP congestion control algorithm [43]. Empirical results [43] suggest that FAST TCP is stable even in the presence of long feedback delays, where previous models pre-

[2]This phenomenon is modeled in network calculus [2], [22]. Our work differs from most network calculus studies in two respects: It explicitly allows feedback control at the sources rather than restricting them to exogenous processes, and it provides an analysis of dynamics rather than focusing on bounds on equilibrium quantities.

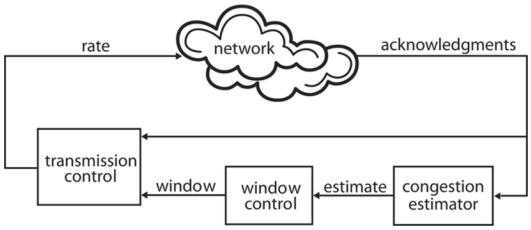


Fig. 1. System view of window-based congestion control.

dict it to be unstable (see [40] and [34, Appendix A]). Application of our new model that captures both sub-RTT burstiness and also the link interactions will resolve the existing discrepancy between experiments and theory and lead to new discoveries.

The intuition behind the model is introduced in Section II-A by considering a single link, including a discussion of the model's well-posedness and how several prior models can be recovered as approximations to it. We then present the general multilink version in Section II-B. In Section III, both simulations and testbed experiments are reported to validate the model, which show excellent accuracy even in transient periods. To illustrate the application of the model, we then specialize to linear stability analysis with feedback delay. Section V shows that in a single-bottleneck network with constant window sizes, the queue size will always stabilize, but that this need not be true in multilink networks. For a ring network in which all flows have the same RTTs, the model predicts a surprising mode of queue oscillation even with constant windows, which is verified by packet-level simulations. Finally, the interaction between the model and congestion control is studied for the single-bottleneck link case in Section VI. Two previously unknown instability modes of FAST TCP (RTT heterogeneity and RTT synchronization) are theoretically predicted, intuitively explained, and again experimentally verified. Sufficient conditions for stability are also provided for cases when many flows share the bottleneck link (Section VI-C).

II. Model

This section presents the new model. To illustrate the intuition, Section II-A first develops the model for the case of a single-bottleneck link. The general case is then presented in Section II-B.

The control structure for window-based transmission control is illustrated in Fig. 1. The dynamics of the endpoint protocol are represented by three blocks: transmission control, window control, and congestion estimator. The system consists of an inner loop and an outer loop. In the outer loop, the window control adjusts the transmission window size (the maximum number of packets that are allowed to be sent out without receiving further acknowledgments) based on the estimated congestion level of the network. This congestion level is estimated based on the ACKs, which carry implicit information in the form of duplicate, missing, and delayed ACKs. The dynamics of the inner loop are given by so-called ACK-clocking, where a new packet is sent when each packet is acknowledged, thereby maintaining the number of outstanding packets equal to the window.

A model for a window-based congestion control system must specify two things: a) the TCP window control algorithm that determines how the congestion signal affects the window; and b) how the congestion signal, based on the queue length, evolves in response to the window sizes. The new model specifies part b), and we will investigate its interaction with existing models for part a) in Section VI.

A network is modeled as L links, indexed by l and with capacities c_l, which carry N TCP flows, indexed by i. Nonbottleneck links (whose queue is always zero) are modeled simply as part of the propagation delay of the upstream link. The following variables are functions of time t; written without explicit time dependence, they denote equilibrium values:

p_l: queueing delay at link l;
w_i: window size[3] of flow i;
x_i: arrival rate of flow i's data at the queue;
d_i: round-trip propagation delay for flow i;
τ_i: RTT for flow i; for flows traversing only link l, $\tau_i = d_i + p_l$.

A. Single-Bottleneck Link Case

Consider a single-bottleneck link shared by multiple TCP flows. As there is a single link, the subscript l will be omitted.

It is well accepted that the length of a FIFO queue integrates the difference between incoming traffic and the link capacity

$$\dot{p}(t) = \frac{1}{c}\left(\sum_{i=1}^{N} x_i(t) - c\right)^{+}_{p(t)} \tag{1}$$

where $(a)^{+}_{b} = \max(a, 0)$ if $b = 0$, and a otherwise.

It remains to find an equation to relate the window (which sources control) with the rate (which affects the queue). This is traditionally done by approximating the rate by the ratio between the corresponding window and RTT, i.e.,

$$x_i(t) = \frac{w_i(t)}{\tau_i(t)}. \tag{2}$$

Although (2) applies to the average over a RTT and is good for equilibrium analysis, it has several limitations for the analysis of dynamics and burstiness. First, there is ambiguity over whether to use $w_i(t-\tau_i)$ and/or $\tau_i(t-\tau_i)$ for $w_i(t)$ and $\tau_i(t)$. Second, it completely ignores ACK-clocking by letting rate directly follow the window's change, while, as noted in Section I, rates are affected by the acknowledgment stream. Third, it implicitly assumes the rate is uniform within one RTT while, in reality, persistent sub-RTT burstiness is prevalent.

The solution is to capture ACK-clocking. The following equation, mentioned in passing in [30], expresses window flow control's goal of equating the packets in flight with the window. More precisely, the window w_i at time $t + d_i + p(t)$ is the integral of the rate x_i from t to $t + d_i + p(t)$. This is because the acknowledgment for the data that is sent at time t arrives at the source at time $t + d_i + p(t)$, after experiencing a queueing delay $p(t)$.[4] Formally

$$\int_{t}^{t+d_i+p(t)} x_i(T)dT = w_i(t + d_i + p(t)). \tag{3}$$

Note that (3) holds for *any* t, and hence it defines a rate function $x_i(t)$ based on the window function. Fig. 2 shows how (3) can be interpreted as a sliding window of such averages.

[3]We assume this is also the number of packets in the network, although a sudden reduction in $w_i(t)$ will not instantly withdraw packets.

[4]If there is a delay τ^f from the source to the link, the argument of $p(\cdot)$ will become $t + \tau^f$. This is considered in the full multilink model.

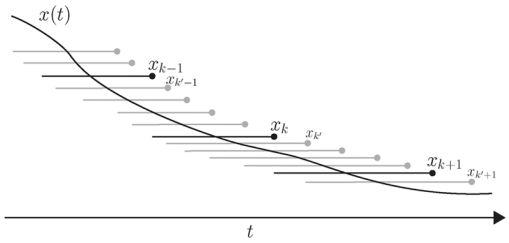


Fig. 2. The input rate of the source into the queue. The sequences $x_k, x_{k'}, \ldots$ that represent averages over an interval $(t_k, t_k + d + p(t_k)], (t_{k'}, t_{k'} + d + p(t_{k'})], \ldots$ are known; we seek the function $x(t)$.

Our proposed model is the mapping from $w(t)$ to $p(t)$ and $x(t)$ implied by (1) and (3). It may be combined with a window control model (such as (25) for FAST TCP) to form a complete system model. Some discussions about its basic properties and relations to existing models are now in order.

1) Uniqueness of Solution: It can be shown that the model supports a unique *equilibrium* value of p and x given w (see footnote to Section IV). However, under certain conditions, there are nonunique periodic orbits for x with a constant w.

Consider a network in which two flows with equal RTTs τ share a bottleneck link of capacity c, and each maintains a constant window of size $c\tau/2$. If the flows alternate between sending at rate c for time $\tau/2$ and sending at rate 0 for $\tau/2$, and if the "ON" periods of flow 1 coincide exactly with the "OFF" periods of flow 2, then the total rate flowing into the bottleneck link is constant c for all time, and (1), (3) is satisfied. It is, however, also satisfied if both sources send constantly at rate $c/2$.

This corresponds to sub-RTT burstiness. For a single link, this burstiness only continues indefinitely if one flow has a RTT that is a rational multiple of another, as discussed in the Appendix. We conjecture that nonequilibrium periodic orbits for p are impossible. The above example relies on the fact that τ (and hence p) is constant, and the Appendix shows that the aggregate rate in the linearization of (1), (3) is asymptotically stable, which implies all periodic orbits of p are equilibria in that model.

2) Relation to Existing Models: We now demonstrate that by making various simplifying assumptions, our new model can be reduced to existing models. Let $H_t(z) = \int_t^z x(T)\,dT - w(z)$. Then, (3) can be written $H_t(t + \tau(t)) = 0$. Different approximations to $H_t(z)$ yield different known models.

a) Ratio Models: Most fluid models take $x_i(t) \approx w_i(t - \Delta_a)/\tau_i(t - \Delta_b)$, for some choice of Δ_a and Δ_b. Applying the right-side rectangle rule to $H_t(t + \tau(t))$ gives $x(t + \tau(t)) = w(t + \tau(t))/\tau(t) + \mathcal{O}(\tau)$ whence

$$x_i(t) \approx w_i(t)/\tau_i(t - \tau_i(\tilde{t})) \tag{4}$$

where $\tilde{t}$ satisfies $\tilde{t} + \tau_i(\tilde{t}) = t$. This is similar to a widely used "ratio link model" (see, e.g., [9] and [26]) which was shown in [40] to be overly pessimistic for large RTTs. More accurate numerical quadrature rules can also be applied. However, even the simplest rules yield approximate models that are of higher complexity than the original system and that, furthermore, tend to be unstable [14]. The use of such model approximations thus seems limited.

By further assuming in (4) that the deviation from the equilibrium rates are negligible, $x_i(t) = x_i + \delta x_i(t) \approx x_i$, we get a static update of the queue in terms of window updates as suggested in [40].

b) "Joint" Models: Taylor expansion of H_t around t yields

$$\begin{aligned} 0 &= H_t(t + \tau(t)) = H_t(t) + H_t'(t)\tau(t) + \mathcal{O}(\tau^2) \\ &= -w(t) + (x(t) - \dot{w}(t))\tau(t) + \mathcal{O}(\tau^2). \end{aligned} \tag{5}$$

Solving for $x(t)$ in this expression gives the rate used by the "joint link model" of [12] as an $\mathcal{O}(\tau_i)$ approximation

$$x_i(t) \approx w_i(t)/\tau_i(t) + \dot{w}_i(t). \tag{6}$$

Ignoring $\dot{w}_i(t)$ in (5) gives $x_i(t) \approx w_i(t)/\tau_i(t)$. If $\dot{w}_i(t) = \mathcal{O}(1/\tau_i(t))$, this is again an $\mathcal{O}(\tau_i)$ approximation, albeit less accurate than (6); otherwise it is $\mathcal{O}(1)$.

c) Models by Padé Approximations: An alternative is to study the linearized model in the Laplace domain, using (14) of Section V. The delay $e^{-s\tau_i}$ in (14) can then be replaced by, for example, different orders of Padé approximation. In this context, a (0,0) Padé approximation (namely $e^{-s\tau_i} \approx 1$) gives the "static link model" introduced in [40], while the "joint link model" [13] corresponds to a (0,1) approximation. A (1,0) approximation yields a time-scaled ratio model. A suitable order of approximation can be chosen. This approach is used with good accuracy in the linear validation example in Section V-A. A nonlinear ODE may finally be "reverse engineered" to approximate the model (1), (3).

All of the above models are based on small τ approximations. However, $\tau(t)$ may not be small; in particular, $\tau(t)$ does not approach zero in the fluid limit of many packets. Thus, (3) can be much better than these existing models. Simulations and experiments in Section III will directly verify this.

B. General Network Case

One important merit of this model is its direct extension to general networks with arbitrary routing and number of links. Therefore, it provides an excellent starting point to investigate possible complex interactions among traffic flows in a general network.

Let $R = (r_{l,i})$ be the $L \times N$ routing matrix, with $r_{l,i} = 1$ if link l is used by flow i, and 0 otherwise. Bidirectional links are modeled as distinct unidirectional links.

At each link l, (1) becomes

$$\dot{p}_l(t) = \frac{1}{c_l}\left(\sum_{i=1}^{N} r_{l,i}x_{l,i}(t) - c_l\right)^+_{p_l(t)} \tag{7a}$$

where $x_{l,i}(t)$ is the arrival rate of flow i to link l at time t. It remains to determine $x_{l,i}(t)$ analogously to (3). This case is significantly more complex since packets experience delays at different instants of time at each link.

Consider a packet $\mathcal{P}$ that is sent at time t by source i. Let $\tau_{l,i}(t)$ be the RTT, i.e., the time between the arrival of $\mathcal{P}$ at link l and the arrival at link l of the packet that is sent as a result of the acknowledgment of $\mathcal{P}$. Similarly, let $\tau^f_{l,i}(t)$ be the delay from t to when $\mathcal{P}$ reaches link l. We can then model the effect of forward transport delay from the location of the source to link l by defining

$$w_{l,i}\left(t + \tau^f_{l,i}(t)\right) = w_i(t) \tag{7b}$$

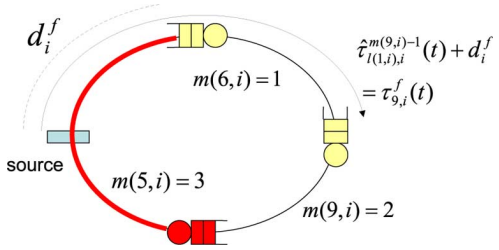


Fig. 3. Notation for multilink delays. Flow i uses three links, which happen to be called 6, 9, and 5. Each is modeled by a "link" consisting of a queue followed by a propagation delay. The delay from the source to its first link is part of the delay associated with the link upstream of the source. The arrow shows the forward propagation delay to the second link, link 9, is $\hat{\tau}^1_{6,i} + d_i^f$.

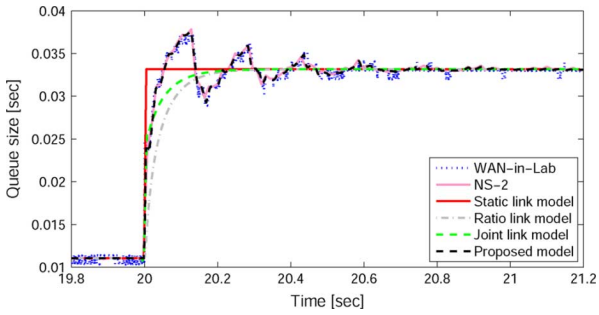


Fig. 4. Single link. Two flows with RTT 14.4 and 128 ms ($d_1 = 3.2$ ms, $d_2 = 117$ ms). Window of short RTT flow increases.

to be the number of packets from source i that arrives at link l during the interval $(t, t + \tau_{l,i}(t)]$. Thus, the instantaneous rate $x_{l,i}(t)$ satisfies

$$\int_t^{t+\tau_{l,i}(t)} x_{l,i}(T)\, dT = w_{l,i}(t + \tau_{l,i}(t)). \tag{7c}$$

It remains to calculate $\tau_{l,i}(t)$ and $\tau^f_{l,i}(t)$. To do this, it is necessary to keep track of the order of the links along each source's path. Let L_i be the number of links in the round-trip path of flow i, $l(m,i)$ be the mth link on that path, and $m(l,i) \in \{1,\dots,L_i\}$ be such that l is the $m(l,i)$th link on path i. Let $p_{l,i}(t) \in \mathbb{R}^{L_i}$ be a cyclic permutation of the queueing delays on the path of flow i, whose nth element is $p^n_{l,i}(t) = p_{l(k,i)}(t)$, where $k \equiv m(l,i) + n - 1 (\mathrm{mod} L_i)$. Thus, the first element is the queueing delay of link l, the second is the queueing delay of the link *downstream* of link l in the ith source's path, and so on, and finally the last element $p^{L_i}_{l,i}(t)$ is the queueing delay of the link directly *upstream* of link l.

The ordered propagation delay $d_{l,i}$ can be defined similarly; $d^n_{l,i}$ represents the propagation delay of the nth link on path i, starting counting from $n = 1$ at l. Thus, $\sum_{k=1}^{L_i} d^k_{l,i} = d_i$. Let $\hat{\tau}^n_{l,i}(t)$, $n = 0,\dots,L_i$ be the delay such that a packet $\mathcal{P}$ that arrives at link l at time t arrives at the link n hops after l on path i at time $t + \hat{\tau}^n_{l,i}(t)$. (Strictly, the packet that arrives may be an ACK or the packet sent in response to the ACK of $\mathcal{P}$.) This is illustrated in Fig. 3.

The total delay, including the queueing at each link, is then

$$\hat{\tau}^n_{l,i}(t) = \sum_{k=1}^{n} \left[p^k_{l,i}(t + \hat{\tau}^{k-1}_{l,i}(t)) + d^k_{l,i} \right]. \tag{7d}$$

The interval of integration in (7c) is then simply

$$\tau_{l,i}(t) = d_i + \sum_{k=1}^{L_i} p^k_{l,i}(t + \hat{\tau}^{k-1}_{l,i}(t)) = \hat{\tau}^{L_i}_{l,i}(t). \tag{7e}$$

Similarly, the forward delay linking $w_{l,i}(t)$ with $w_i(t)$ is

$$\tau^f_{l,i}(t) = \hat{\tau}^{m(l,i)-1}_{l(1,i),i}(t) + d^f_i \tag{7f}$$

where d^f_i is the propagation delay from sender i to $l(1,i)$, the first (explicitly modeled) link on its path.

The complete model of ACK-clocking dynamics for a system of N window-based sources using a network of L links is given by (7). Equation (7a) describes the dynamics of the queue associated with link l. Equation (7c) defines effective rates for flows by the invariant that each flow keeps the number of its outstanding packets equal to its window size. The other equations then define delays carefully, as packets experience different delays at different links in their paths, depending also on the times they are sent out.

III. Model Validation

The model (7) will now be validated by comparing its predictions with packet-level data from NS-2 simulations. To show that the good agreement with NS-2 is not merely because the model captures artifacts introduced by NS-2, the results are also compared to those from WAN-in-Lab testbed [23].

These tests used three of WAN-in-Lab's Cisco 7609 routers connected by 2.5-Gb/s OC48 links, with spools of fiber used to implement delays. Traffic shaping was applied to the OC48 interfaces to obtain the desired link capacities.

Three end-systems were attached to each router by short gigabit Ethernet links. These systems ran Linux 2.6.23, patched to improve the speed of SACK processing. The constant window sizes were obtained using a custom kernel module that disables congestion control and instead sets TCP's window to a fixed value, controllable in real time using a `sysctl`. Both TCP and UDP traffic were generated using iperf.

A. Single-Link Network

For the single-link case, two window-based flows send over a bottleneck link with capacity 100 Mb/s and packets of 1590 bytes, including all link-layer and physical-layer headers. The round-trip delays excluding the queueing delay are $d_1 = 3.2$ ms and $d_2 = 117$ ms. The window sizes are initially $w_1 = 50$ and $w_2 = 550$ packets respectively. After 20 s, w_1 is increased step-wise from 50 to 150 packets.

The dotted line in Fig. 4 corresponds to variation in the one-way RTT when the above scenario is set up and executed in the testbed. The level of the curve is shifted so that it initially matches the equilibrium queue. The light solid line is the queue size in the NS-2 simulation. The dashed black line is the queue size when the model (1), (3) is evaluated with the same step input. The gray dashed-dotted line and the light dashed line correspond to a similar evaluation of the ratio link model (4) and the joint link model (6), respectively. The darker solid line shows the static queue model proposed in [40]. There is a good match between the real testbed data, the NS-2 data, and the model (1), (3) proposed here, while previous models fail to capture the significant oscillations.

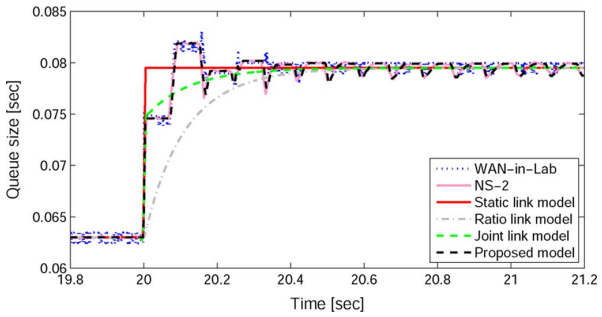


Fig. 5. Single link. Two flows with RTT 73 and 153 ms ($d_1 = 10$ ms, $d_2 = 90$ ms). Window of short RTT flow increases.

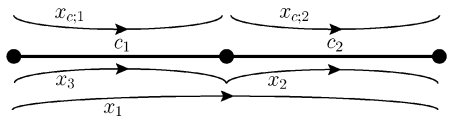


Fig. 6. Network configuration of validation example.

The outcome of a similar experiment is shown in Fig. 5. The setting is identical except that RTTs are $d_1 = 10$ ms, $d_2 = 90$ ms, initial windows are $w_1 = 210, w_2 = 750$, and w_1 is increased by 90 packets at 20 s. As previously, the proposed model (1), (3) shows very good agreement with testbed as well as NS-2 data.

B. Multiple-Link Networks

The general model (7) will now be validated using NS-2 and WAN-in-Lab for the scenario shown in Fig. 6, with three flows sending over a network of two bottleneck links. Flow 1 uses both links, with link 1 upstream of link 2. Flow 2 sends over link 2 only. Note that neither flow 1 nor flow 2 terminates at the end of link 1. To measure the delay on this link, the third artificial single-link flow with window $w_3 = 5$ packets was added to link 1. In some cases, there is also non-flow-controlled cross traffic sent over the individual links.

The physical-layer link capacities in the testbed were set to 80 and 200 Mb/s, giving capacities $c_1 = 72$ Mb/s, $c_2 = 180$ Mb/s. The link round-trip delays excluding queueing are 40 ms for link 1 and 80 ms for link 2. Thus, $d_1 = 120$ ms, $d_2 = 80$ ms, and $d_3 = 40$ ms. Packets had a payload of $\rho = 1448$ bytes. The forward delay from each source to the first bottleneck link it encountered is approximately zero.

At $t = 0$ s, the window of either source 1 or source 2 is increased by 200 packets. The queue sizes of the model are compared to NS-2 data and WAN-in-Lab measurements. To avoid the need for an absolute time reference, time series for each queue were manually aligned to start at $t = 0$ s. Similarly, the level of each curve is shifted so that it initially matches the equilibrium queue. Tests were undertaken for cross traffic on all combinations of links 1 and 2. The results below are the cases whose dynamics are most informative.

1) Case 1: No Cross Traffic: In the first scenario, there are no UDP sources. Initially, the flow-controlled sources have $w_1^0 = 1600$ and $w_2^0 = 1200$ packets. Fig. 7 displays the queueing delays when the window of the first source is increased. There is an immediate response in the first queue, but no transient at all in the second queue. This is because link 1 can still only send data at rate c_1 to link 2, despite its increased arrival rate.

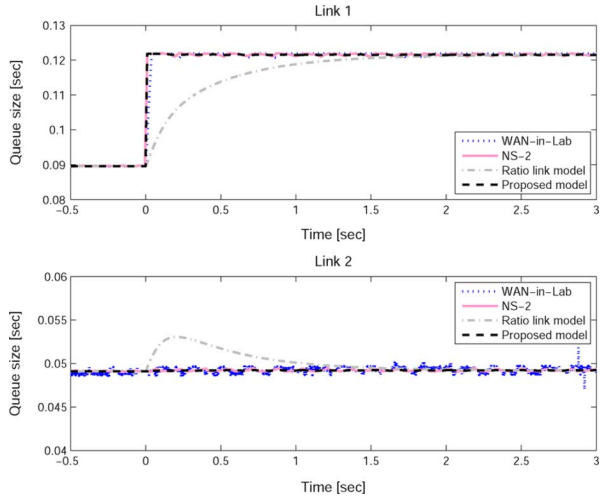


Fig. 7. No cross traffic, step change in window 1.

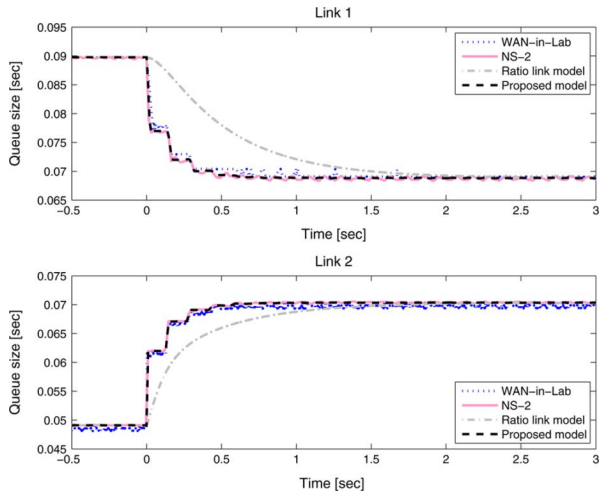


Fig. 8. No cross traffic, step change in window 2.

Our proposed model predicts this effect accurately, while the prediction of the ratio model is qualitatively different.

Fig. 8 shows the response when a step is applied to the second source's window. This time, both queues are affected. When the second queue is increased, the RTT of source 1 is increased, which decreases its sending rate. This decreases the delay at the first queue even though it is "upstream" of the initial perturbation. As before, our proposed model is considerably more accurate than the ratio model.

2) Case 2: Cross Traffic on Link 1: In the second scenario, UDP cross traffic is sent over link 1 using half the capacity, i.e., $x_{c;1}(t) = c_1/2$. Initially $w_1^0 = 1500$ and $w_2^0 = 300$ packets. Fig. 9 shows the queue sizes when the window of source 1 is increased. Unlike in Fig. 7, there is a transient in the second queue because link 1 can temporarily increase the rate at which it forwards source 1 data to link 2, at the expense of the cross traffic. The proposed model captures the difference between these two cases, while the standard ratio model predicts a qualitatively similar transient each time.

Fig. 10 shows the corresponding results when the second source's window is perturbed. The observed behavior and the performance of the models are analogous to the similar scenario

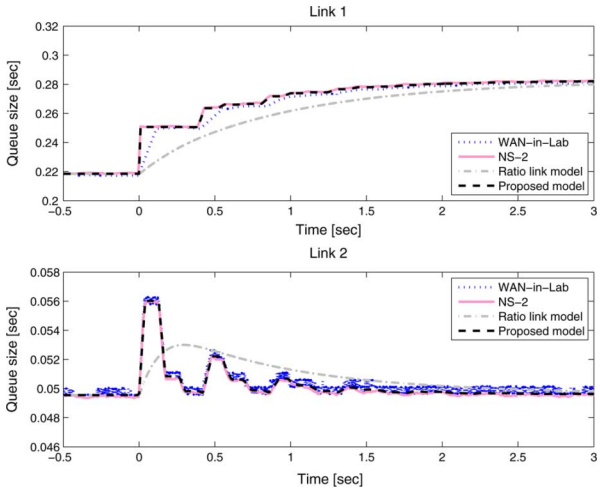


Fig. 9. Cross traffic on link 1, step change in window 1.

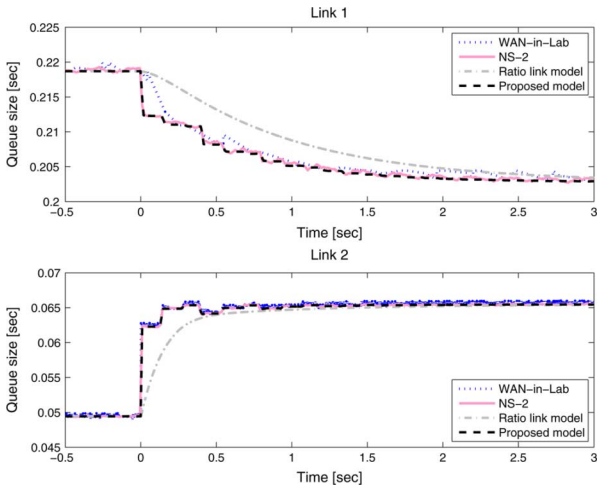


Fig. 10. Cross traffic on link 1, step change in window 2.

in Case 1. Note that the mismatch in queue 1 between the testbed and NS-2 is mainly due to the slow sampling rate (small window, large RTT) of the testbed queue; at sampling instants, the agreement is good.

C. Model Limitations and Strengths

The model accurately captures the effect of small perturbations in the window. It does not model large perturbations that cause the buffer to empty or fill completely nor cause the window to reduce so fast that it becomes less than the number of packets in flight. Since these effects affect the performance of loss-based congestion control more than dynamics on the RTT timescale, it is likely that models that ignore sub-RTT effects may be sufficient to study purely loss-based algorithms such as TCP Reno.

However, for small perturbations, the proposed model shows excellent agreement with packet-level data from both simulations and measurements. It captures both long-term behavior and sub-RTT burstiness effects and different instantaneous rates of a flow at different links along its path.

IV. Model Linearization

In preparation for the sections that follow, this section presents a linearization of the general model (7) around the equilibrium[5] and expresses it in the frequency domain. The corresponding transfer function will later be used for analyzing the stability of the ACK-clocking mechanism both in isolation and with congestion control protocols. The primary approximation in the linearization is that delays in arguments are taken to be their equilibrium values. Henceforth, time-dependent quantities such as $x_{l,i}(t)$ will refer to (small) deviations from the equilibrium.

At links with nonzero equilibrium queues, (7a) becomes

$$\dot{p}_l(t) = \frac{1}{c_l} \sum_{i=1}^{N} r_{l,i} x_{l,i}(t). \tag{8a}$$

Linearizing the derivative of (7c) consists of neglecting time variation of delayed arguments and neglecting the term $x_{l,i}(t)\dot{\tau}_{l,i}(t)$. This gives

$$x_{l,i}(t+\tau_{l,i}) - x_{l,i}(t) + x_i \dot{\tau}_{l,i}(t) = \dot{w}_i \left(t + \tau_{l,i} - \tau^f_{l,i}\right).$$

Since the steady-state RTT of a flow is the same at all links, $\tau_{l,i} = \tau_i$, this becomes

$$x_{l,i}(t+\tau_i) - x_{l,i}(t) + x_i \dot{\tau}_{l,i}(t) = \dot{w}_i \left(t + \tau_i - \tau^f_{l,i}\right). \tag{8b}$$

The total steady-state delay, including the steady-state queueing at each link, is the time average of (7d), given by

$$\hat{\tau}^n_{l,i} = \sum_{k=1}^{n} \left(p^k_{l,i} + d^k_{l,i}\right). \tag{8c}$$

The linearized delay from (7e) then becomes

$$\tau_{l,i}(t) = \sum_{k=1}^{L_i} p^k_{l,i} \left(t + \hat{\tau}^{k-1}_{l,i}\right). \tag{8d}$$

Finally the steady-state forward delay used in (8b) is

$$\tau^f_{l,i} = \hat{\tau}^{m(l,i)-1}_{l(1,i),i} + d^f_i. \tag{8e}$$

In the frequency domain, substituting (8d) into (8b) gives

$$x_{l,i}(s) = x_{l,i}(s)e^{-s\tau_i} - \left(x_i e^{-s\tau_i} \sum_{k=1}^{L_i} p^k_{l,i}(s) e^{s\hat{\tau}^{k-1}_{l,i}} - w_i(s) e^{-s\tau^f_{l,i}}\right). \tag{9}$$

To express (8) in matrix form, let $\mathbf{p} = (p_l)$ be the vector of delays at the links, $\mathbf{w} = (w_i)$ be the vector of windows, and $C = \mathrm{diag}(c_l)$ be the $L \times L$ matrix of link capacities. Furthermore, let P_i be an $L_i \times L$ matrix, whose k,lth entry is 1 if link l is the kth link in the path of flow i, and zero otherwise. This is a permutation matrix with some rows removed. Finally, let

[5]The equilibrium of (7) can be characterized using the utility maximization framework; see [18] and [25]. By associating the equilibrium point of the system with a solution to a convex program, it can be shown to be unique if the routing matrix R is full-rank [29], [43].

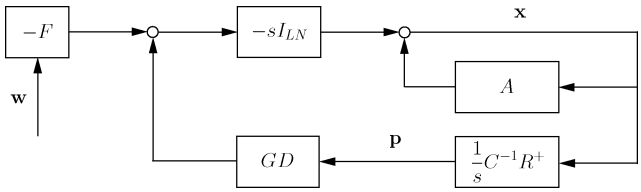


Fig. 11. Block diagram of (10).

$\mathbf{x} \in \mathbb{R}^{LN}$ be a concatenation of $x_{l,i}$, increasing l first; that is $\mathbf{x} = (x_{11}, x_{21}, \ldots, x_{L-1,N}, x_{LN})^T$. Then

$$\mathbf{p}(s) = \frac{1}{s} C^{-1} R^+ \mathbf{x}(s) \tag{10a}$$

$$\mathbf{x}(s) = A\mathbf{x}(s) - s(GD\mathbf{p}(s) - F\mathbf{w}(s)) \tag{10b}$$

where

$$R^+ = [\operatorname{diag}(r_{\cdot,1}) \quad \operatorname{diag}(r_{\cdot,2}) \quad \cdots \quad \operatorname{diag}(r_{\cdot,N})]$$

is an $L \times LN$ expanded routing matrix ($r_{\cdot,i} = P_i^T J_{L_i \times 1}$ is the ith column of the standard delay-free routing matrix, R, whose lth element is 1 if flow i uses link l, and zero otherwise; $J_{a\times b}$ is the $a \times b$ all-ones matrix)

$$A = \operatorname{diag}(e^{-s\tau_i}) \otimes I_L$$
$$G = \operatorname{diag}(x_i e^{-s\tau_i}) \otimes I_L \tag{11}$$

are $LN \times LN$ diagonal matrices, and

$$F = \operatorname{diag}_i \left(\begin{bmatrix} e^{-s\tau^f_{1,i}} & e^{-s\tau^f_{2,i}} & \cdots & e^{-s\tau^f_{L,i}} \end{bmatrix}^T \right)$$

is an $LN \times N$ block-diagonal matrix of the forward delays from each source to each link, with zero entries for links that are not used by flow i.

Finally, the delay matrix $D \in \mathbb{R}^{LN \times L}$ is

$$D = \begin{bmatrix} P_1^* D_1^{-1} K_1 D_1 P_1 \\ P_2^* D_2^{-1} K_2 D_2 P_2 \\ \vdots \\ P_N^* D_N^{-1} K_N D_N P_N \end{bmatrix} \tag{12}$$

where D_i is an $L_i \times L_i$ diagonal matrix, whose kth diagonal element is $\exp(s\tau^f_{l(k,i)})$, where $l(k,i)$ is the kth link on path i, and

$$K_i = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ e^{s\tau_i} & 1 & \ldots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ e^{s\tau_i} & e^{s\tau_i} & \ldots & 1 \end{pmatrix} \tag{13}$$

is the all-ones matrix $J_{L_i \times L_i}$ plus a lower triangular matrix whose lower triangular entries are all $e^{s\tau_i} - 1$.

This corresponds to the block diagram of Fig. 11.

V. Stability With Constant Windows

The dynamics of flow control algorithms are of fundamental interest. In particular, stability of window control despite feedback delay is required in order to ensure that the system operating point is indeed the intended equilibrium (with the desired equilibrium properties such as efficiency and fairness). The stability of the underlying ACK-clocking flow control is an important component of that. This section will demonstrate that the linearization (10) accurately reflects the true dynamics and use it to show that ACK-clocking is always stable in networks with a single bottleneck. However, we also show that queue oscillations can occur in multibottleneck networks, which certainly highlights the importance of our model.

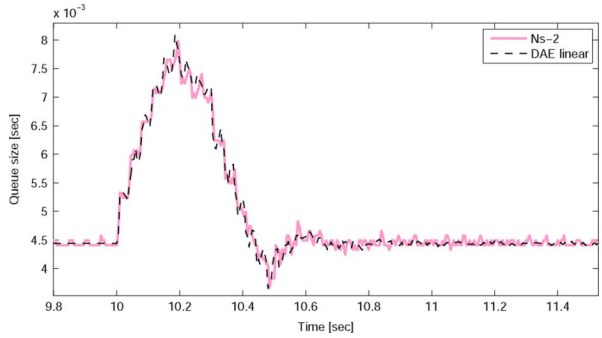


Fig. 12. Single-link, linearized model. Two flows with RTTs 10 and 200 ms. Short RTT flow has an increased window from 10 to 10.3 s. The model was evaluated using a (17,17) Padé approximation to $e^{-s\tau}$.

A. Single Link Is Stable

When there is $L = 1$ link in the model, (10) simplifies considerably since $D^T = R^+ = (1, \ldots, 1) \in \mathbb{R}^N$. Assuming without loss of generality that the forward delay is zero, eliminating $\mathbf{x}$ from (10) gives

$$p(s) = G_{pw}(s)\mathbf{w}(s) = \sum_{i=1}^{N} G_{pw_i}(s) w_i(s) \tag{14}$$

where the ith transfer function element is given by

$$G_{pw_i}(s) = \frac{1}{(1 - e^{-s\tau_i})\left(c + \sum_{n=1}^{N} x_n \frac{\exp(-s\tau_n)}{1-\exp(-s\tau_n)}\right)}. \tag{15}$$

To see the linear model's accuracy, consider two window-based flows that are sending over a bottleneck link with capacity $c = 100$ Mb/s. Initially, $w_1 = 60$ packets and $w_2 = 2000$ packets, with packet size $\rho = 1040$ bytes. Furthermore, $d_1 = 10$ ms and $d_2 = 200$ ms, with no forward delay. The system is started in equilibrium, and w_1 is increased by 10 at $t = 10$ s, and 300 ms later it is decreased back to 60. Fig. 12 shows that the linear model fits the simulations very well.

To be able later to analyze the stability of congestion control algorithms using the Nyquist criterion, it is important to establish that the system (14) is stable. The following theorem shows that this is the case.

Let $\mathbb{C}^+$ be the open right half-plane $\{z : \operatorname{Re}(z) > 0\}$, and $\bar{\mathbb{C}}^+$ be its closure $\{z : \operatorname{Re}(z) \geq 0\}$.

Theorem 1: For all $\tau_i > 0$ and all $0 < x_i \leq c, i = 1, \ldots, N$, such that $\sum_i x_i \leq c$, the function $G_{pw} : \bar{\mathbb{C}}^+ \to \mathbb{C}^{1\times N}$ whose ith element is given by (15) is stable.

Proof: It is sufficient to confirm that [7]:

a) $G_{pw}(s)$ is analytic in $\mathbb{C}^+$;

b) for almost every real number ω

$$\lim_{\sigma\to 0^+} G_{pw}(\sigma+j\omega)=G_{pw}(j\omega);$$

c) $\sup_{s\in\bar{\mathbb{C}}^+}\bar{\sigma}(G_{pw}(s))<\infty$

where $\bar{\sigma}$ denotes the largest singular value.

Conditions a) and b) are satisfied if they hold element-wise. Furthermore

$$\sup_{s\in\bar{\mathbb{C}}^+}\bar{\sigma}(G_{pw}(s))\le\sum_{i=1}^{N}\sup_{s\in\bar{\mathbb{C}}^+}|G_{pw_i}(s)|. \tag{16}$$

Thus, condition c) holds if

$$\inf_{s\in\bar{\mathbb{C}}^+}|1/G_{pw_i}(s)|>0. \tag{17}$$

It is therefore sufficient to establish a)–c) for the ith transfer function element $G_{pw_i}(s)$.

Start with the boundedness condition c). It is sufficient to show that there is no sequence $s_k=\sigma_k+j\omega_k\in\bar{\mathbb{C}}^+$ with $\lim_{k\to\infty}|1/G_{pw_i}(s_k)|=0$. This will be established by showing that the limit evaluated on any convergent subsequence is greater than 0. Consider a subsequence with $\sigma_k\to\sigma$, $\omega_k\to\omega$.

Case 1, $\sigma=\infty$: $1/G_{pw_i}(s_k)\to c>0$.

Case 2, $\sigma\in(0,\infty)$: By the triangle inequality

$$|1-e^{-s_k\tau_i}|\ge|1-|e^{-s_k\tau_i}||\to 1-e^{-\sigma\tau_i}>0. \tag{18}$$

Furthermore, $1/(e^{s_k\tau_n}-1)$ lies on the circle with center $1/(A_k^2-1)+j0$ and radius $A_k/(A_k^2-1)$, where $A_k=|e^{s_k\tau_n}|$. Thus, $\lim_{k\to\infty}\mathrm{Re}(1/(e^{s_k\tau_n}-1))\ge -1/(e^{\sigma\tau_n}+1)$, hence

$$\begin{aligned}\lim_{k\to\infty}\mathrm{Re}\left(c+\sum_{n=1}^{N}\frac{x_n}{e^{s_k\tau_n}-1}\right)&\ge c-\sum_{n=1}^{N}\frac{x_n}{e^{\sigma\tau_n}+1}\\ &=c-\sum_{n=1}^{N}x_n+\sum_{n=1}^{N}\frac{x_ne^{\tau_n\sigma}}{e^{\tau_n\sigma}+1}\ge\sum_{n=1}^{N}\frac{x_n}{1+e^{-\tau_n\sigma}}>0.\end{aligned} \tag{19}$$

Multiplying (18) and (19) gives $\lim_{k\to\infty}|1/G_{pw_i}(s_k)|>0$.

Case 3, $\sigma=0$: Note that $\mathrm{Re}(1/(e^{j\omega_k\tau_n}-1))=-1/2$, so

$$\lim_{k\to\infty}\mathrm{Re}\left(c+\sum_{n=1}^{N}\frac{x_n}{e^{(\sigma_k+j\omega_k)\tau_n}-1}\right)=c-\sum_{n=1}^{N}\frac{x_n}{2}>0. \tag{20}$$

Thus, $\lim_{k\to\infty}|1/G_{pw_i}(s_k)|\neq 0$ except possibly when the first factor of (15) $1-e^{-s_k\tau_i}\to 0$, which occurs when $\omega\tau_i=2\pi m, m\in\mathbb{Z}$. Let $\mathbf{I}_n=1$ if $m\tau_n/\tau_i\in\mathbb{Z}$, and 0 otherwise. Now

$$\begin{aligned}&\lim_{s\to j2\pi m/\tau_i}|1/G_{pw_i}(s)|\\ &\quad=\lim_{s\to j2\pi m/\tau_i}\left|c\big(1-e^{-s\tau_i}\big)+x_i+\sum_{\substack{n=1\\n\neq i}}^{N}x_ne^{-s\tau_n}\frac{1-e^{-s\tau_i}}{1-e^{-s\tau_n}}\right|\\ &\quad=x_i+\sum_{\substack{n=1\\n\neq i}}^{N}x_n\frac{\tau_i}{\tau_n}\mathbf{I}_n>0\end{aligned} \tag{21}$$

using L'Hôpital's rule in the second step when $\mathbf{I}_n=1$. Thus, $\lim_{k\to\infty}|1/G_{pw_i}(s_k)|>0$ for all sequences s_k in $\bar{\mathbb{C}}^+$ for which the limit exists, whence (17) holds, and thus c).

Furthermore, since $1/G_{pw_i}(s)\neq 0, G_{pw_i}(s)$ is also nonsingular in $\bar{\mathbb{C}}^+$, and therefore analytic as its components are analytic. This establishes a). Condition b) holds since $G_{pw_i}(s)$ is analytic in $\bar{\mathbb{C}}^+$. ■

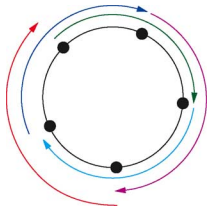


Fig. 13. Two-hop overlapping ring network with five links and five flows. Flows are represented by arrows.

B. Multiple Links May be Unstable

Theorem 1 established that the queue sizes in the single-link model (14) are always asymptotically stable for a fixed window. We now show that this need not be the case for a general network with multiple congested links. This result highlights the possible complication that spatial interaction among flows can bring.

Eliminating $\mathbf{p}$ from (10) gives

$$\mathbf{x}(s)=-(\mathrm{GDC}^{-1}R^+-A)\mathbf{x}(s)+sF\mathbf{w}(s) \tag{22}$$

giving a loop gain of

$$\mathcal{L}_x(s)=\mathrm{GDC}^{-1}R^+-A.$$

The poles of the system (10) are the solutions of the characteristic equation $|I_{LN}+\mathcal{L}_x(s)|=0$, and stability can be studied by the Nyquist criterion.

Note that the approach being taken here is to eliminate $\mathbf{p}$, rather than $\mathbf{x}$ as was done in the single-link case. In the general case, eliminating $\mathbf{x}$ yields a loop gain containing $(I-A)^{-1}$, which contains marginally stable poles. These were canceled by zeros in the single-link case, but need not always be. While $\mathcal{L}_x(s)$ is simpler than $\mathcal{L}_p(s), \mathcal{L}_x(s)$ is of substantially higher dimension: $L\cdot N\times L\cdot N$ instead of $L\times L$.

Now, consider the case of a homogeneous ring network with two-hop flows, as illustrated in Fig. 13. Observe that the number of links and flows are equal, i.e., $N=L$, and index links such that indices i increase clockwise for $i<N$. Furthermore, without loss of generality, let source i be located directly at link i and destination i be located at link $i+2$ (mod N). The assumed homogeneity allows the subscript to be dropped on x_i,τ_i,c_i and $\tau^f_{i+1,i}=\tau^f$ (note that $\tau^f_{i+k,i}=0$ for all $k\neq 1$), and again consider the low-dimensional vector $\mathbf{p}$. Noting that $\mathbf{p}(s)=1/(cs)R^+\mathbf{x}(s)$, multiplying (22) by $1/(cs)R^+$ and substituting (11), we get

$$\begin{aligned}\mathbf{p}(s)&=-e^{-s\tau}\left(\frac{x}{c}R^+D-I_L\right)\mathbf{p}(s)+\frac{1}{c}R^+F\mathbf{w}(s)\\ &=-\mathrm{circ}(l_p(s))\mathbf{p}(s)+\frac{1}{c}R^+F\mathbf{w}(s).\end{aligned}$$

Here $\mathrm{circ}(l_p(s))$ denotes the circulant matrix with first row

$$l_p(s)=\left(0,e^{-s(\tau-\tau^f)}\Big/2,0,\ldots,0,e^{-s\tau^f}\Big/2\right). \tag{23}$$

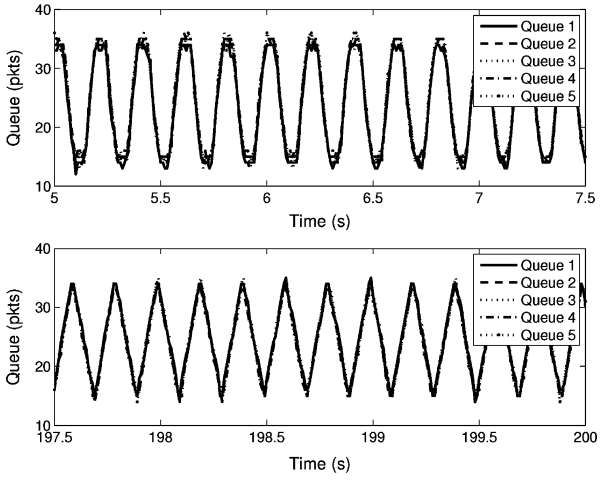


Fig. 14. Oscillating queue size for the network of Fig. 13, in a metastable state ($t \in [5, 7.5]$) and in steady state ($t \in [197.5, 200]$). Parameters: $w = 2401$ packets, $c = 12\,020$ packets/s, $\tau = 399$ ms.

Since the loop gain $\mathrm{circ}(l_p(s))$ is open-loop stable, the Nyquist criterion says that the closed-loop system is strictly unstable if and only if any eigenvalue of $\mathrm{circ}(l_p(j\omega))$ encircles -1 when ω traverses the positive real axis, and marginally stable if no encirclements occur but an eigenlocus passes through -1. The eigenvalues of $\mathrm{circ}(a)$ with $a = (a_0, a_1, \ldots, a_{N-1})$ are simply the coefficients of the discrete Fourier transform (DFT) of the sequence (a_k). Hence, the eigenvalues of the loop gain are

$$\lambda_m(j\omega) = \frac{1}{2}\left(e^{j(-\omega\tau-(-\omega\tau^f+2\pi m/N))} + e^{j(-\omega\tau^f+2\pi m/N)}\right)$$

for $m = 0, 1, \ldots, N-1$. Now, $|\lambda_m(j\omega)| \leq 1$, and thus no strict encirclements can occur. Furthermore, the locus will hit -1 at ω if for feasible parameters[6] there exist odd integers k_1 and k_2 and an integer m and such that both

$$\omega\tau^f - 2\pi m/N = k_1\pi \tag{24a}$$

$$\omega\tau - k_1\pi = k_2\pi. \tag{24b}$$

This suggests that the system may indeed be marginally stable and the queues will oscillate for suitable parameter values, even with constant window sizes. Note that for any ω satisfying (24), $(2n+1)\omega$ also does for all n, indicating that oscillations need not be sinusoidal, even neglecting nonlinearities. Formally establishing marginal stability would require a proof that no zeros cancel the marginally stable poles; instead, we will demonstrate the instability numerically.

Example 1: Instability due to Flow Spatial Interactions: Consider $N = 5$ flows and links and with link delays such that $\tau_f/\tau = 1/4$. For this case, (24) holds for frequency $\omega = 4\pi/\tau$ and $m = 0, k_1 = 1, k_2 = 3$. Fig. 14 shows an NS-2 simulation of this scenario with link capacities $c = 12\,020$ packets/s, link delays set to 100 ms on the forward path and 97.6 ms on the backward path and the common window size set to 2401 packets (giving $\tau = 399.5$ ms in equilibrium). The upper plot shows a snippet of the transient phase where the queues exhibit truncated triangle-wave oscillations that are aligned in phase. The lower plot displays a snippet of the steady state. The queues again exhibit oscillations, of unchanged amplitude, but the trajectories have spread to form untruncated triangle waves. Due to the purely imaginary poles at $\pm j2/\tau$, the model predicts oscillations of frequency $\omega/2\pi = 2/\tau \approx 5$ Hz. This is confirmed by the observations from Fig. 14.

[6]For $N < 3$ links, it is impossible to construct a ring of two-hop flows, and thus the results do not contradict the stability for a single link.

VI. Stability With Adaptive Windows

Motivated by the fact that our proposed model correctly predicts qualitatively different behavior from previous models, we now revisit the stability of the FAST TCP congestion control algorithm [43] in the absence of cross traffic. Empirical results suggest that the algorithm is stable regardless of feedback delay [43]. In particular, it is stable in scenarios that previous models predict to be unstable (see [34]). By using the model derived in this work, we show that for *any* step size γ, there is a (possibly pathological) network in which FAST is unstable, as verified by both analysis and packet level simulations. We finally also provide practical conditions that guarantee FAST to be stable.

FAST TCP is an algorithm that aims to improve TCP Reno's performance especially for networks with large bandwidth delay products [43]. FAST sets the congestion window based on the queueing delay, $p(t-\tau_i)$, seen by the packets. Its continuous time form is

$$\dot{w}_i(t) = -\gamma\frac{p(t-\tau_i)}{(d_i + p(t-\tau_i))^2}w_i(t) + \gamma\frac{\alpha_i}{d_i + p(t-\tau_i)} \tag{25}$$

where $\gamma \in (0, 1]$ is a step size and α_i is a constant measured in packets. Since we are interested in local stability, we linearize and take the Laplace transform of (25). That yields

$$\left(s + \gamma\frac{p}{\tau_i^2}\right)w_i(s) = -\gamma\frac{\alpha_i d_i}{p\tau_i{}^2}p(s)e^{-s\tau_i}. \tag{26}$$

Note that the subscript on the queueing delay has been dropped here since we consider the single-link case where all flows observe the same queueing delay p.

We now have a negative-feedback system, in which w_i is calculated from p by (26), and p from the w_i by (14). The closed-loop stability of the system will be studied via the open-loop transfer function using the Nyquist criterion. Opening the loop at p, the open-loop transfer function becomes

$$L(s) = \sum_{i=1}^{N}\mu_i L_i(s) \tag{27a}$$

where

$$\mu_i = \frac{\alpha_i}{cp} = \frac{\alpha_i}{\sum_{n=1}^{N}\alpha_n} = \frac{x_i}{c} \tag{27b}$$

$$L_i(s) = \frac{d_i\gamma e^{-s\tau_i}}{(\tau_i)^2 s + \gamma p}\frac{T(s\tau_i)}{\sum_{n=1}^{N}\mu_n T(s\tau_n)} \tag{27c}$$

$$T(s\tau) = \frac{1}{1-e^{-s\tau}}. \tag{27d}$$

Note that $\mathrm{Re}(T(j\phi)) = 1/2$ for all ϕ. Define

$$y(\phi) := \mathrm{Im}(T(j\phi)) = -\frac{\sin(\phi)}{2(1-\cos(\phi))}. \tag{28}$$

As the weights satisfy $\sum_i \mu_i = 1$, it follows that $\hat{T} = \sum_{i=1}^N \mu_i T(s\tau_i)$ also lies on the line $\mathrm{Re}(z) = 1/2$. In particular, as ω increases from 0 to $2\pi/\tau_{\max}, \hat{T}$ monotonically traces the line from $1/2 - j\infty$ to $1/2 + j\infty$, where $\tau_{\max} = \max_i \tau_i$.

This model will next be used to predict two previously unknown modes of instability of FAST. To show the existence of an unstable configuration, it is sufficient to consider the case that $p \to 0$. If that case is strictly unstable, then there is also a strictly unstable configuration with $p \neq 0$ since $L(s)$ is continuous with respect to p for $s \neq 0$. When $p \to 0$, the transfer function tends to

$$L(s) = \gamma \sum_{i=1}^N \mu_i \frac{T(s\tau_i)e^{-s\tau_i}/s\tau_i}{\sum_{n=1}^N \mu_n T(s\tau_n)}. \tag{29}$$

A. Instability Due to RTT Heterogeneity

Since each individual flow does not have complete knowledge of the network, we would like to be able to set FAST's parameters, such as γ, so that it will be stable in *all* networks. In the following, we show that is impossible. For any given γ, we construct a network carrying two flows with very different RTTs such that FAST is unstable.

For two flows with equal $\alpha, \mu_1 = \mu_2 = 1/2$ and the loop gain (29) reduces to

$$L(s) = \gamma \frac{\frac{e^{-s\tau_1}}{s\tau_1}(1 - e^{-s\tau_2}) + \frac{e^{-s\tau_2}}{s\tau_2}(1 - e^{-s\tau_1})}{2 - e^{-s\tau_2} - e^{-s\tau_1}}. \tag{30}$$

Instability arises because the loop gain becomes very large near $\omega = 2\pi/(\tau_1 + \tau_2)$ as the heterogeneity increases.

Let $\lambda = \tau_1/(\tau_1 + \tau_2)$ and $\omega(\tau_1, \lambda, \beta) = (2\pi - \beta)\lambda/\tau_1$, so that $\omega = \omega(\tau_1, \lambda, \beta)$ satisfies $\omega\tau_2 = 2\pi - \omega\tau_1 - \beta$. Let

$$L_\lambda(\beta) = L(j\omega(\tau_1, \lambda, \beta)) \tag{31}$$

$$= \gamma \frac{\frac{e^{-j\omega\tau_1}}{j\omega\tau_1}\left(1 - e^{j(\omega\tau_1 + \beta)}\right) + \frac{e^{j\omega\tau_1}e^{j\beta}(1 - e^{-j\omega\tau_1})}{j(2\pi - \omega\tau_1 - \beta)}}{2 - e^{j\omega\tau_1}e^{j\beta} - e^{-j\omega\tau_1}} \tag{32}$$

where each ω in (32) refers to $\omega(\tau_1, \lambda, \beta)$.

The following lemma is proved in [33].

Lemma 2: Let $\omega^* = 2\pi/(\tau_1 + \tau_2)$. Then, (30) satisfies

$$\mathrm{Im}(L(j\omega^*)) > 0. \tag{33}$$

Let $\beta = (2\pi\lambda)^3$. For $0 < \lambda < 1/(2\pi)^2$, (31) satisfies

$$\mathrm{Im}(L_\lambda(\beta)) < 0 \tag{34}$$

and for all $\omega \in [\omega^* - \beta/\tau_1, \omega^*]$, (30) satisfies

$$|L(j\omega)| \geq \frac{\gamma}{411\lambda^2}. \tag{35}$$

For all $\omega > 0$, the general case (14), and hence (30), satisfies

$$\frac{d}{d\omega} \arg(L(j\omega)) \leq 0. \tag{36}$$

The primary result of Lemma 2 is (35), which shows that there is an arbitrarily large positive feedback near ω^*. To see where this arises, consider the limit as $\omega\tau_1 \to 0$. As $\tau_2 \gg \tau_1$, the second term in the numerator of (32) is small, and so to first order

$$L_\lambda(\beta) \approx \gamma \frac{(-1 + \cos(\omega\tau_1))/(j\omega\tau_1) - (\sin(\omega\tau_1))/(\omega\tau_1)}{2 - 2\cos(\omega\tau_1) - j\beta}$$

$$\approx \gamma \frac{1}{\omega\tau_1} \frac{-\omega\tau_1 + j(\omega\tau_1)^2/2}{(\omega\tau_1)^2 - j\beta} \tag{37}$$

using $\sin(a + \beta) - \sin(a) \approx \beta$ for small a, β. The large gain results from the cancellation in the imaginary part of the denominator, leaving only $j\beta$. Physically, this is because there is feedback on the timescale of $\tau_2 \gg \tau_1$; the feedback gain should normally be scaled in inverse proportion to the feedback delay [32], but flow 1 scales its gain in inverse proportion to its own, much smaller RTT.

Theorem 3: For all $\gamma > 0$ and τ_1, there exists a $\tilde{\tau}$ such that for all $\tau_2 > \tilde{\tau}$ the closed-loop with loop gain (30) is unstable.

Proof: First, note that (30) is open loop stable by Theorem 1, except for the pole at the origin introduced by letting $p \to 0$. Thus, by (36) and the Nyquist criterion, a closed-loop system with loop gain (30) is unstable if and only if $L(j\omega) \in (-\infty, -1)$ for some ω (since, when s traverses an infinitesimal semicircle $\epsilon e^{j\phi}$ with ϕ from $-\pi/2$ to $\pi/2$ around the pole $s = 0$, the eigenlocus remains in the right half-plane).

Let $\tilde{\tau} = \max(\sqrt{411/\gamma}, (2\pi)^2)\tau_1$. For any $\tau_2 > \tilde{\tau}$, the right-hand side of (35) exceeds 1, and (33) and (34) of Lemma 2 hold. By (33) and (34), $L(j\omega)$ crosses the real axis for some $\omega \in [\omega^* - \beta/\tau_1, \omega^*]$, and by (36), this crossing must be clockwise and must be a crossing of the negative real axis. By (35) of Lemma 2, this crossing must be to the left of $-1 + j0$, proving the instability. ∎

The following numerical results support Theorem 3. This is the first example to show instability of FAST TCP; previous work failed to show this as it did not explore cases with sufficient heterogeneity in feedback delays [43].

Example 2: Instability Due to Heterogeneous RTTs: Consider two FAST flows with 1040-byte packets sharing a 200-Mb/s bottleneck, with $d_1 = 10$ ms and $d_2 = 303$ ms and FAST parameters $\gamma = 0.5$ and $\alpha = 100$ [43]. The Nyquist plot of (30) in Fig. 15 encircles -1, indicating instability. NS-2 simulations, reported in the three remaining plots, show that there is indeed sustained oscillation at around $1/d_2 \approx 3$ Hz. The variation in *window* size shows that this is not simply packet-level sub-RTT burstiness. The rate of source 2 oscillated between 3380 and 3550 packets/s. For this and Example 3, FAST's multiplicative increase mode was disabled.

Fig. 16 shows results from WAN-in-Lab for two FAST flows with 1500-byte packets sharing a 1-Gb/s bottleneck, with $d_1 = 6$ ms and $d_2 = 130$ ms and FAST parameters $\gamma = 0.5$ and $\alpha = 30$ packets. The sustained oscillation of over 50 packets, at around $1/d_2 \approx 8$ Hz, confirms the instability.

B. Instability Due to RTT Synchronization

The precise timing of the model (29) allows another mode of instability, now to be described. This instability, which is confirmed by NS simulations, is expected not to appear in real networks with jitter as explained in Section VI-C, but affects formal stability proofs.

Consider a network (29) with $N = 2$ flows with equal rate, $\mu_1 = \mu_2 = 1/2$ and RTTs $\tau_1 = 1 + \delta$ and $\tau_2 = k, k = 2, 3, \ldots$.

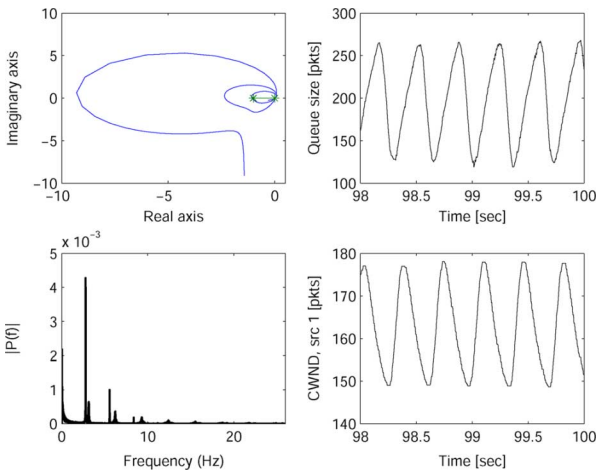


Fig. 15. Instability of FAST due to heterogeneous RTTs: $d_1 = 10$ ms, $d_2 = 303$ ms. Top left: Nyquist plot of loop gain. Top right: Bottleneck queue size. Lower left: Magnitude spectrum (FFT) of queue size, without DC component. Lower right: Window size, source 1.

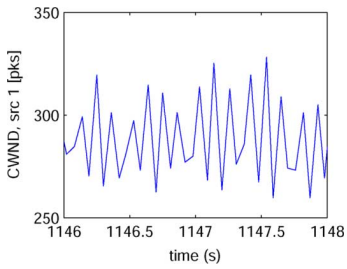


Fig. 16. Window size, source 1: $d_1 = 6$ ms, $d_2 = 130$ ms.

If δ is sufficiently small, this system oscillates with a frequency near $\omega^* \approx 2\pi(1 - \delta/(k+1))$, as will now be shown.

Recall that the individual terms $T(j\omega\tau_i) = 1/2 + jy_i(\omega)$ lie on the line $\{z \in \mathbb{C} \mid \mathrm{Re}(z) = 1/2\}$, as does their convex combination in the denominator of (29). Each of these points moves vertically upwards as ω increases, at a rate depending on τ_i, with all starting from $1/2 - j\infty$ at $\omega = 0$. For $\delta < \pi/k$, $y_1(\omega)$ is negative for $\omega \in (2\pi/(1+\delta), 2\pi)$ and takes all values less than $y_1(2\pi)$. Similarly, $y_2(\omega)$ takes all values greater than $y_2(2\pi/(1+\delta))$ in that interval. Thus, since y_1 and y_2 are continuous on this interval, there exists an ω^* such that $y_1(\omega^*) + y_2(\omega^*) = 0$ making the denominator of $L(s)$ equal 1/2. However, since $\omega^* \approx 2\pi$, for $\delta \ll 1$ the numerator of (29) is approximately

$$\frac{(k+1)/2 + j(k-1)y_1(\omega^*)}{jk\omega^*}.$$

Since $y_1(\omega^*) < y_1(2\pi) \to -\infty$ as $\delta \to 0$, the magnitude of $L(s)$ can be made arbitrarily large by taking δ small. Moreover, this large value occurs when $L(s)$ lies approximately on the negative real axis. This causes the Nyquist curve to encircle $-1+j0$, and the system to be unstable.

In contrast to the previous mode of instability, which was caused by the short-RTT flow overreacting to congestion due to very heterogeneous RTTs, this new mode is caused by precise cancellations when the ratio of the RTTs is approximately rational. In the previous example, the Nyquist curve encircles $-1 + j0$ in its first loop around 0. This time the Nyquist curve encircles $-1 + j0$ in the kth loop around 0, when the cancellation occurs. This corresponds to a frequency the reciprocal of the smaller RTT, rather than the larger. The following NS-2 simulation shows instability due to this mechanism.

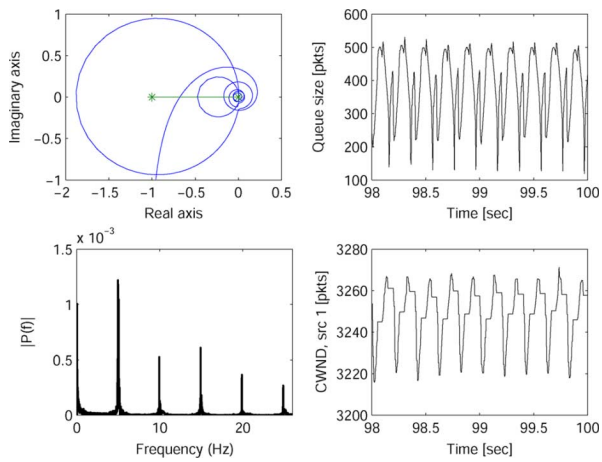


Fig. 17. Instability of FAST due to synchronized RTTs: $\tau_1 = 101$ ms, $\tau_2 = 200$ ms. Upper left: Nyquist plot of loop gain. Upper right: Bottleneck queue size. Lower left: Magnitude spectrum (FFT) of queue size, without DC component. Lower right: Window size, source 1.

Example 3: Instability due to Precisely Matched RTTs: Consider two FAST flows with $\gamma = 1$ and $\alpha = 200$ packets, sharing a bottleneck with $c = 500$ Mb/s, with $d_1 = 94.4$ ms and $d_2 = 193.4$ ms giving $\tau_1 = 101$ ms and $\tau_2 = 200$ ms. The Nyquist plot of Fig. 17 again predicts instability, and NS-2 also exhibits oscillation. As well as the predicted peak at 10 Hz, there is one at 5 Hz because the implementation of FAST freezes the window every alternate RTT, as seen in the bottom right-hand figure, introducing severe nonlinearity.

C. Sufficient Conditions for Stability

With these two mechanisms that can cause instability for arbitrarily small gain γ, one might despair of finding any conditions under which (29) can be shown to be stable. However, it turns out that it is stable when there are many flows, or in the realistic case of networks with slight jitter.

Let $\mu : (0, \tau_{\max}] \to \mathbb{R}^+ \cup \{\infty\}$ be the distribution of RTTs, weighted by the relative rate (x_i/c) of flows with each RTT. For finitely many flows, this is a sum of impulses weighted by the discrete μ_i, but this section will consider general distributions. Sums weighted by μ_i, of the form $\sum_i f_i\mu_i$, are replaced by integrals, denoted

$$\mathcal{M}[f(\tau)] := \int_0^{\tau_{\max}} f(\tau)\mu(\tau)d\tau. \tag{38}$$

With this notation, (29) becomes

$$L(s) = \gamma \frac{\mathcal{M}[T(s\tau)e^{-s\tau}/s\tau]}{\mathcal{M}[T(s\tau)]}. \tag{39}$$

Furthermore, let $\mathrm{sinc}(\theta) = \sin(\theta)/\theta$.

Define $H(\omega)$ as the half-plane under the line through $-1+j0$ with complex argument $\frac{\pi}{2} - \arg(\mathcal{M}[T(j\omega\tau)])$. Formally

$$H(\omega) = \left\{x \middle| \arg(x+1) - \frac{\pi}{2} + \arg\left(\mathcal{M}[T(j\omega\tau)]\right) \in (-\pi, 0)\right\}.$$

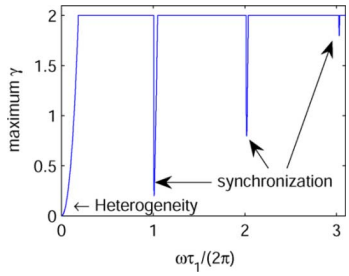


Fig. 18. Maximum γ for which (40) holds for $\tau_1 = 101$ ms, $\tau_2 = 200$ ms, as a function of frequency ω.

Lemma 4: If

$$\gamma \mathcal{M}[\mathrm{sinc}(\omega\tau)] < \mathcal{M}[1 - \cos(\omega\tau)] \tag{40}$$

then for $L(s)$ defined by (39)

$$L(j\omega) \in H(\omega). \tag{41}$$

Proof: By definition, $L(j\omega) \in H(\omega)$ is equivalent to

$$\arg(L(j\omega) + 1) + \arg(\mathcal{M}[T(j\omega\tau)]) \in (-\pi/2, \pi/2) \tag{42}$$

or in other words

$$\begin{aligned} 0 &< \mathrm{Re}\left(\mathcal{M}\left[T(j\omega\tau)\left(1 + \frac{\gamma e^{-j\omega\tau}}{j\omega\tau}\right)\right]\right) \\ &= \mathcal{M}\left[(1 - \cos(\omega\tau))\left(1 - \frac{\gamma \sin(\omega\tau)}{\omega\tau}\right)\right. \\ &\quad \left. - \frac{\gamma \sin(\omega\tau)\cos(\omega\tau)}{\omega\tau}\right]. \end{aligned}$$

Multiplying out the first term and canceling with the final term shows that (42) is satisfied if (40) holds. ■

Remark: The techniques used here, analogous to those discussed in [12], yield significantly tighter bounds than those in existing linear stability analysis of TCP, which is necessary to exploit the increased accuracy of the model.

Fig. 18 shows the maximum γ for which (40) holds (truncated to 2). The frequencies at which instability occurred due to heterogeneity (Example 2) and synchronization (Example 3) are cases where (40) only holds for very small γ. Small perturbations of τ cause the "synchronization" dips to vanish, but for any distribution of τ, (40) is violated at low frequency. The next lemma shows that in a sufficiently low-frequency region, $L(j\omega)$ cannot cross $(-\infty, -1]$.

Lemma 5: For all ω such that $\omega\tau_{\max} \in (0, \pi/3)$, $\mathrm{Im}(L(j\omega)) < 0$.

Proof: For $\omega \in (0, \pi/(3\tau_{\max}))$ and $\tau \le \tau_{\max}$, we have by (28) that $y(\omega\tau) < -\sqrt{3}/2$, whence $\arg(T(j\omega\tau)) \in (-2\pi/3, -\pi/2)$. For those ω and τ, we also have $\arg(\frac{\exp(-j\omega\tau)}{j\omega\tau}) \in (-5\pi/6, -\pi/2)$. Now, the numerator of $L(j\omega)$, $\mathcal{M}[T(j\omega\tau)\frac{\exp(-j\omega\tau)}{j\omega\tau}]$, is a convex combination of points with arguments in $(-3\pi/2, -\pi)$, and so its argument must also be in that interval. Similarly, the argument of the denominator $\mathcal{M}[T(j\omega\tau)]$ is in $(-2\pi/3, -\pi/2)$. Thus

$$\begin{aligned} &\arg(L(j\omega\tau)) \\ &\quad = \arg\left(\mathcal{M}\left[T(j\omega\tau)\frac{\exp(-j\omega\tau)}{j\omega\tau}\right]\right) - \arg(\mathcal{M}[T(j\omega\tau)]) \\ &\quad \in (-\pi, -\pi/6) \end{aligned}$$

whence $\mathrm{Im}(L(j\omega)) < 0$.

We can now state a sufficient condition, in terms of the distribution of RTTs and the value of FAST's step size, for FAST to be stable. It says that if there are many flows, with sufficiently uniformly spread RTTs, then FAST will be stable.

Theorem 6: Let $\tau_0 > 0, 0 < a < \tau_0$ and $0 < h \le 1/(2a)$, and let $\delta < 1$ be such that

$$\delta > \mathrm{sinc}\left(\frac{a\pi}{3\tau_{\max}}\right). \tag{43}$$

Then, for any distribution $\mu(\tau) : (0, \tau_{\max}] \to \mathbb{R}^+ \cup \{\infty\}$ with $\mu(\tau) > h$ for all $\tau \in (\tau_0 - a, \tau_0 + a)$, and for any

$$\gamma < 2ah(1 - \delta) \tag{44}$$

the closed-loop negative feedback system with open-loop transfer function $L(s)$ given by (39) is stable.

Proof: It is sufficient to show that the Nyquist curve does not intersect $(-\infty, -1]$. By Lemma 5, this does not happen for $\omega < \pi/(3\tau_{\max})$.

Since $(-\infty, -1] \cup H(\omega) = \emptyset$, it remains by Lemma 4 to show (40) holds for all $\omega \ge \pi/(3\tau_{\max})$. Since $1 - \cos(\omega\tau) \ge 0$

$$\begin{aligned} \mathcal{M}[1 - \cos(\omega\tau)] &\ge h \int_{\tau_0 - a}^{\tau_0 + a} 1 - \cos(\omega\tau) d\tau \\ &= 2ah(1 - \cos(\omega\tau_0)\mathrm{sinc}(\omega a)). \end{aligned} \tag{45}$$

Similarly, $\mathrm{sinc}(\theta) \le 1$, and hence $\mathcal{M}[\mathrm{sinc}(\omega\tau)] \le 1$.

Now, $0 < a\pi/(3\tau_{\max}) < \pi/2, \mathrm{sinc}(\cdot)$ is decreasing on $[0, \pi/2]$ and $\mathrm{sinc}(\theta) < \mathrm{sinc}(\pi/2)$ for $\theta > \pi/2$. Thus, by (43), $\mathrm{sinc}(\omega a) \le \delta$ for all $\omega \ge \pi/(3\tau_{\max})$. Combining this with (40), (45) and $\cos(\omega\tau_0) \le 1$ shows that (44) implies $L(s)$ is stable. ■

Note that condition (44) is only a sufficient condition and is loose for very peaked distributions μ. Typically, $\mathcal{M}[\mathrm{sinc}(\omega\tau)] \ll 1$ for ω with $\cos(\omega\tau_0) \approx 1$, and vice versa.

VII. Conclusion

This paper has presented a model that describes the queueing process of a communication network with arbitrary topology, in which all data sources use window flow control. It takes into account notable but previously missing factors such as rate burstiness at sub-RTT timescales and different rates of a given flow at different links. The model was built solidly on basic equations that govern the behavior of sources and links. It is a generic model and independent of the source congestion control algorithms. The prediction from the model shows excellent match with experiments even in transient periods and is therefore ideal for accurate studies of the dynamics of such systems. Indeed, focusing on stability of congestion control systems, we are able to clarify existing confusion and provide new findings. All predictions of this model in this paper are verified by packet-level sim-

ulations and experiments. This new model also opens doors to much new work. In particular, it assumes the number of flows is fixed, and it will be important to see the effect of the new model on networks with dynamically arriving and departing flows [4].

APPENDIX
CONDITIONS FOR UNIQUENESS OF RATES

Consider the linearization of (1), (3), following the procedure in Section IV. Note that sustained oscillations in the rate for a constant window correspond to marginally stable (pure imaginary) poles of this linearization. Taking the Laplace transform, eliminating p and solving for x gives the transfer function from windows to rates

$$G_{xw}(s) = \left(\mathrm{diag}\left(\frac{x_i}{c}\right) J + \mathrm{diag}(e^{s\tau_i} - 1)\right)^{-1} \mathrm{diag}(se^{s\tau_i}) \tag{46}$$

where J is an $N \times N$ matrix with entries $J_{i,k} = 1$ for all $i, k = 1, \ldots, N$. Since $\mathrm{diag}(se^{s\tau_i})$ has no poles for finite s, the oscillatory poles of $G_{xw}(s)$ are the nonzero values of s for which the matrix to be inverted is singular. The only imaginary values for which this occurs are when there exist an i and k, both in $\{1, \ldots, N\}$, and integers a and b such that $s\tau_i = j2\pi b$ and $s\tau_k = j2\pi a$. For all other combinations of RTT, the rates are stable.

This linear system admits no nonequilibrium periodic orbits for the aggregate rate $y = \sum_i x_i$. From first principles, the transfer function from w to y, $G_{yw}(s)$, is the solution to

$$G_{yw}(s)\mathrm{diag}\left(\frac{e^{-s\tau_i}}{s}\right)\left(\mathrm{diag}\left(\frac{x_i}{c}\right) J + \mathrm{diag}(e^{s\tau_i} - 1)\right) = (1, \ldots, 1).$$

The poles of G_{xw} correspond to points where the third factor on the left-hand side is singular. At such points, its linearly dependent columns (for which $e^{s\tau_i} = 1$) are all identical. Since the columns of the right-hand side are also identical, the null space of the third factor is contained within that of the right-hand side. Thus, the equation admits a finite solution for $G_{yw}(s)$, and hence $G_{yw}(s)$ has no poles. Thus, sustained oscillations in the individual rates do not cause fluctuations in the total rate into the link or in p.

REFERENCES

[1] T. Alpcan and T. Basar, "Global stability analysis of an end-to-end congestion control scheme for general topology networks with delay," in *Proc. IEEE Conf. Decision Control*, 2003, pp. 1092–1097.

[2] F. Baccelli and D. Hong, "TCP is max-plus linear and what it tells us on its throughput," *Proc. ACM SIGCOMM*, pp. 219–230, 2000.

[3] L. S. Brakmo and L. 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.

[4] T. Bonald and L. Massoulié, "Impact of fairness on Internet performance," in *Proc. ACM SIGMETRICS*, 2001, pp. 82–91.

[5] H. Choe and S. H. Low, "Stabilized Vegas," in *Proc. IEEE INFOCOM*, 2003, pp. 2290–2300.

[6] S. Deb and R. Srikant, "Global stability of congestion controllers for the Internet," *IEEE Trans. Autom. Control*, vol. 48, no. 6, pp. 1055–1060, Jun. 2003.

[7] G. E. Dullerud and F. Paganini, *A Course in Robust Control Theory*. New York: Springer, 2000.

[8] M. Gerla and L. Kleinrock, "Flow control: A comparative survey," *IEEE Trans. Commun.*, vol. COMM-28, no. 4, pp. 553–574, Apr. 1980.

[9] C. Hollot, V. Misra, D. Towsley, and W. Gong, "A control theoretic analysis of RED," *Proc. IEEE INFOCOM*, pp. 1510–1519, 2001.

[10] C. Hollot and Y. Chait, "Nonlinear stability analysis for a class of TCP/AQM schemes," in *Proc. IEEE Conf. Decision Control*, 2001, pp. 2309–2314.

[11] V. Jacobson, "Congestion avoidance and control," in *Proc. ACM SIGCOMM*, 1988, pp. 157–187.

[12] K. Jacobsson, L. L. H. Andrew, A. Tang, S. H. Low, and H. Hjalmarsson, "An improved link model for window flow control and its application to FAST TCP," *IEEE Trans. Autom. Control*, vol. 54, no. 3, pp. 551–564, Mar. 2009.

[13] K. Jacobsson, L. L. H. Andrew, A. Tang, K. H. Johansson, H. Hjalmarsson, and S. H. Low, "ACK-clocking dynamics: Modeling the interaction between windows and the network," in *Proc. IEEE INFOCOM*, 2008, pp. 2146–2152.

[14] K. Jacobsson, "Dynamic modeling of internet congestion control," Ph.D. dissertation, Royal Institute of Technology (KTH), Stockholm, Sweden, 2008.

[15] R. Jain, "Congestion control in computer networks: Issues and trends," *IEEE Network*, vol. 4, no. 3, pp. 24–30, May 1990.

[16] H. Jiang and C. Dovrolis, "Why is the internet traffic bursty in short time scales," in *Proc. ACM SIGMETRICS*, 2005, pp. 241–252.

[17] R. Johari and D. Tan, "End-to-end congestion control for the internet: Delays and stability," *IEEE/ACM Trans. Netw.*, vol. 9, no. 6, pp. 818–832, Dec. 2001.

[18] F. Kelly, A. Maulloo, and D. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability," *J. Opt. Res. Soc.*, vol. 49, no. 3, pp. 237–252, Mar. 1998.

[19] K. Kim, A. Tang, and S. H. Low, "Design of AQM in supporting TCP based on the well-known AIMD model," *Proc. IEEE Globecom*, pp. 3226–3230, 2003.

[20] K. Kim, A. Tang, and S. H. Low, "A stabilizing AQM based on virtual queue dynamics in supporting TCP with arbitrary delays," *Proc. IEEE CDC*, pp. 3665–3670, 2003.

[21] S. Kunniyur and R. Srikant, "End-to-end congestion control: Utility functions, random losses and ECN marks," *IEEE/ACM Trans. Netw.*, vol. 11, no. 5, pp. 689–702, Oct. 2003.

[22] J.-Y. Le Boudec and P. Thiran, *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet LNCS 2050*. Berlin, Germany: Springer, 2001.

[23] G. S. Lee, L. L. H. Andrew, A. Tang, and S. H. Low, "WAN-in-Lab: Motivation, deployment and experiments," in *Proc. PFLDnet*, Marina Del Rey, CA, 2007, pp. 85–90.

[24] S. Liu, T. Basar, and R. Srikant, "Pitfalls in the fluid modeling of RTT variations in window-based congestion control," *Proc. IEEE INFOCOM*, pp. 1002–1012, 2005.

[25] S. H. Low and D. Lapsley, "Optimization flow control, I: Basic algorithm and convergence," *IEEE/ACM Trans. Netw.*, vol. 7, no. 6, pp. 861–874, Dec. 1999.

[26] S. H. Low, F. Paganini, and J. C. Doyle, "Internet congestion control," *IEEE Control Syst. Mag.*, vol. 22, no. 1, pp. 28–43, Feb. 2002.

[27] S. H. Low, F. Paganini, J. Wang, and J. C. Doyle, "Linear stability of TCP/RED and a scalable control," *Comput. Netw. J.*, vol. 43, no. 5, pp. 633–647, 2003.

[28] L. Massoulié, "Stability of distributed congestion control with heterogeneous feedback delays," *IEEE Trans. Autom. Control*, vol. 47, no. 6, pp. 895–902, Jun. 2002.

[29] L. Massoulié and J. Roberts, "Bandwidth sharing: Objectives and algorithms," *IEEE/ACM Trans. Netw.*, vol. 10, no. 3, pp. 320–328, Jun. 2002.

[30] J. Mo, R. La, V. Anantharam, and J. Walrand, "Analysis and comparison of TCP Reno and TCP Vegas," *Proc. IEEE INFOCOM*, pp. 1556–1563, 1999.

[31] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Trans. Netw.*, vol. 8, no. 5, pp. 556–567, Oct. 2000.

[32] F. Paganini, Z. Wang, J. C. Doyle, and S. H. Low, "Congestion control for high performance, stability, and fairness in general networks," *IEEE/ACM Trans. Netw.*, vol. 13, no. 1, pp. 43–56, Feb. 2005.

[33] A. Tang, L. L. H. Andrew, K. Jacobsson, K. H. Johansson, S. H. Low, and H. Hjalmarsson, "Window flow control: Macroscopic properties from microscopic factors," *Proc. IEEE INFOCOM*, pp. 91–95, 2008.

[34] A. Tang, K. Jacobsson, L. L. H. Andrew, and S. H. Low, "An accurate link model and its application to stability analysis of FAST TCP," *Proc. IEEE INFOCOM*, pp. 161–169, 2007.

[35] A. Tang, J. Wang, and S. H. Low, "Counter-intuitive throughput behaviors in networks under end-to-end control," *IEEE/ACM Trans. Netw.*, vol. 14, no. 2, pp. 355–368, Apr. 2006.

[36] A. Tang, J. Wang, S. H. 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.
[37] Z. Wang and J. Crowcroft, "Eliminating periodic packet losses in the 4.3-Tahoe BSD TCP congestion control algorithm," *ACM SIGCOMM Comp. Commun. Rev.*, vol. 22, no. 2, pp. 9–16, Apr. 1992.
[38] G. Vinnicombe, "On the stability of networks operating TCP-like protocols," in *Proc. IFAC*, 2002.
[39] M. Vojnovic and J.-Y. Le Boudec, "On the long-run behavior of equation-based rate control," *IEEE/ACM Trans. Netw.*, vol. 13, no. 3, pp. 568–581, Jun. 2005.
[40] J. Wang, D. X. Wei, and S. H. Low, "Modeling and stability of FAST TCP," in *IMA Volumes in Mathematics and Its Applications*, P. Agrawal, M. Andrews, P. J. Fleming, G. Yin, and L. Zhang, Eds. New York: Springer Science, 2006, vol. 143, Wireless Communications.
[41] Z. Wang and F. Paganini, "Global stability with time-delay in network congestion control," in *Proc. IEEE Conf. Decision Control*, 2002, pp. 3632–3637.
[42] D. Wei, "Microscopic behavior of Internet congestion control," Ph.D. dissertation, California Institute of Technology, Pasadena, CA, 2007.
[43] D. Wei, C. Jin, S. H. Low, and S. Hegde, "FAST TCP: Motivation, architecture, algorithms, performance," *IEEE/ACM Trans. Netw.*, vol. 14, no. 6, pp. 1246–1259, Dec. 2006.
[44] H. Yaiche, R. R. Mazumdar, and C. Rosenberg, "A game theoretic framework for bandwidth allocation and pricing in broadband networks," *IEEE/ACM Trans. Netw.*, vol. 8, no. 5, pp. 667–678, Oct. 2000.
[45] L. Ying, G. Dullerud, and R. Srikant, "Global stability of internet congestion controllers with heterogeneous delays," *IEEE/ACM Trans. Netw.*, vol. 14, no. 3, pp. 579–591, Jun. 2006.

Ao Tang (S'01–M'07) received the B.E. (Hon.) degree in electronics engineering from Tsinghua University, Beijing, China, in 1999, and the Ph.D. degree in electrical engineering with a minor in applied mathematics from the California Institute of Technology (Caltech), Pasadena, in 2006.

He is currently an Assistant Professor with the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY, where his research interests include communication networks, dynamical systems, and network multiresolution analysis.

Dr. Tang was the recipient of the 2006 George B. Dantzig Best Dissertation Award, the 2007 Charles Wilts Best Dissertation Prize, and a 2009 IBM Faculty Award.

Lachlan Andrew (M'97–SM'05) received the B.Sc., B.E., and Ph.D. degrees from the University of Melbourne, Melbourne, Australia, in 1992, 1993, and 1997, respectively.

Since 2008, he has been an Associate Professor with Swinburne University of Technology, Hawthorn, Australia. From 2005 to 2008, he was a Senior Research Engineer with the Department of Computer Science, California Institute of Technology, Pasadena. Prior to that, he was a Senior Research Fellow with the University of Melbourne and a Lecturer at RMIT, Melbourne, Australia. His research interests include performance analysis of congestion control, resource allocation algorithms, and energy-efficient networking.

Dr. Andrew is a Member of the Association for Computing Machinery (ACM). He was co-recipient of the Best Paper Award at the IEEE MASS 2007.

Krister Jacobsson received the M.S. degree in vehicle engineering and the Ph.D. degree in control of communication networks from the Royal Institute of Technology (KTH), Stockholm, Sweden, in 2002 and 2008, respectively.

He is currently a Post-Doctoral Fellow with the California Institute of Technology, Pasadena. His research interests include modeling and control of telecommunication systems.

Karl H. Johansson received the M.Sc. and Ph.D. degrees in electrical engineering from Lund University, Lund, Sweden, in 1992 and 1997, respectively.

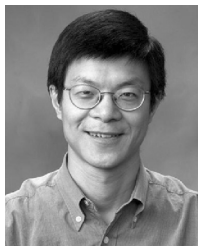
He is Director of the ACCESS Linnaeus Centre and Professor with the School of Electrical Engineering, Royal Institute of Technology, Stockholm, Sweden. He is a Wallenberg Scholar and holds a Senior Researcher Position with the Swedish Research Council. He has held visiting positions at the University of California, Berkeley, from 1998 to 2000 and California Institute of Technology, Pasadena, from 2006 to 2007. His research interests are in networked control systems, hybrid and embedded control, and control applications in automotive, automation, and communication systems.

Dr. Johansson was awarded a six-year Individual Grant for the Advancement of Research Leaders from the Swedish Foundation for Strategic Research in 2005. He received the triennial Young Author Prize from IFAC in 1996 and the Peccei Award from the International Institute of System Analysis, Austria, in 1993. He received Young Researcher awards from Scania in 1996 and from Ericsson in 1998 and 1999. He has been the Chair of the International Federation of Automatic Control (IFAC) Technical Committee on Networked Systems since 2008. He has served on the IEEE Control Systems Society Board of Governors and on the Executive Committees of the European research projects HYCON and RUNES. He is on the editorial boards of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL and the *IET Control Theory & Applications* and previously of the IFAC journal *Automatica*.

Håkan Hjalmarsson received the M.S. degree in electrical engineering and the Licentiate and Ph.D. degrees in automatic control from Linköping University, Linköping, Sweden, in 1988, 1990, and 1993, respectively.

He has held visiting research positions with the California Institute of Technology, Pasadena; Louvain University, Louvain-la-Neuve, Belgium; and the University of Newcastle, Newcastle, Australia. He is a Professor with the School of Electrical Engineering, KTH, Stockholm, Sweden. His research interests include system identification, signal processing, control and estimation in communication networks, and automated tuning of controllers.

Dr. Hjalmarsson received the KTH award for outstanding contribution to undergraduate education in 2001. He has served as an Associate Editor for *Automatica* from 1996 to 2001 and the IEEE TRANSACTIONS ON AUTOMATIC CONTROL from 2005 to 2007. He has been a Guest Editor for the *European Journal of Control* and *Control Engineering Practice*. He is Vice-Chair of the IFAC Technical Committee on Modeling, Identification and Signal Processing.



Steven Low (M'92–SM'99–F'08) received the B.S. degree in electrical engineering from Cornell University, Ithaca, NY, in 1987, and the M.Sc. and Ph.D. degrees in electrical engineering from the University of California, Berkeley, in 1989 and 1992, respectively.

He is a Professor with the Computer Science and Electrical Engineering departments, California Institute of Technology (Caltech), Pasadena, and an Adjunct Professor with Swinburne University of Technology, Hawthorn, Australia.

Dr. Low is a member of the Networking and Information Technology Technical Advisory Group for the U.S. Presidents Council of Advisors on Science and Technology (PCAST). He was a co-recipient of the IEEE William R. Bennett Prize Paper Award in 1997 and the 1996 R&D 100 Award. He was on the Editorial Board of the IEEE/ACM TRANSACTIONS ON NETWORKING from 1997 to 2006 and of *Computer Networks Journal* from 2003 to 2005. He is on the editorial boards of *ACM Computing Surveys* and *NOW Foundations and Trends in Networking*. He is a Senior Editor of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS and a Co-Editor of the Springer book series on *Optimization and Control of Communication Systems: Theory and Applications*.