This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2007 proceedings.

An Accurate Link Model and Its Application to Stability Analysis of FAST TCP

Ao Tang*, Krister Jacobsson†, Lachlan L. H. Andrew*, Steven H. Low*
* California Institute of Technology, Pasadena, CA 91125, USA
†Royal Institute of Technology (KTH), Stockholm, SE-129 32, Sweden

***Abstract*— This paper presents a link model which captures the queue dynamics when congestion windows of TCP sources change. By considering both the self-clocking and the link integrator effects, the model is a generalization of existing models and is shown to be more accurate by both open loop and closed loop packet level simulations. It reduces to the known static link model when flows' round trip delays are similar, and approximates the standard integrator link model when the heterogeneity of round trip delays is significant. We then apply this model to the stability analysis of FAST TCP. It is shown that FAST TCP flows over a single link are always linearly stable regardless of delay distribution. This result resolves the notable discrepancy between empirical observations and previous theoretical predictions. The analysis highlights the critical role of self-clocking in TCP stability and the scalability of FAST TCP with respect to delay. The proof technique is new and less conservative than the existing ones.**

I. Introduction

In the field of network congestion control [17], [22], one line of work of fundamental interest is the dynamics of congestion control protocols. Stability is crucial to ensure that the system operating point is indeed the intended equilibrium, with the desired efficiency and fairness. Local stability [6], [11], [13], [14], [15], [16], [18], [19], [20], [23] and global stability [1], [9], [10], [26], [28] of congestion control protocols have both attracted much attention.

Until recently, stability analyses have been based predominantly on rate-based (rather than window based) source models, and, almost without exception, on integrator link (queue) models. In these models, sources control their data rates explicitly and the rate of change of the queueing delay is proportional to the difference between the aggregate incoming traffic and link capacity. Typically, these results show that the system is stable when round trip delays do not exceed some upper bound. This is in line with the intuition that increased feedback delay may have a destabilizing effect on a closed loop system.

There are however features of TCP which play important roles on stability but are not captured in the rate-based/ integrator model. Current TCPs are window based algorithms, meaning that each sender controls its window size — an upper bound on the number of packets that have been sent but not acknowledged. The actual rate of transmission is controlled or "clocked" by the stream of received acknowledgments (ACKs): a new packet is transmitted only when an ACK is received, thereby keeping the number of outstanding packets constant and equal to the window size. Therefore, sources control the amount of data they inject into the network rather than the rate of doing so. Intuitively, this "volume control" is safer in terms of stability than "rate control". This *self-clocking* has the consequences that the queue size traces the changes in the window much faster than the integrator queue model predicts, and that the sending rate cannot exceed the capacity over long time frames. Both enhance stability.

For the case where flows' round trip delays coincide and no non-window-based cross traffic is present, the self-clocking effect is dominant: the total ACK rate cannot exceed the link capacity at any time and hence a window change translates into a proportional change in the queue. Indeed, as an example, homogeneous FAST TCP [27] flows are reported empirically to be stable with any delay, in stark contrast to the prediction of the integrator link model which asserts instability when delay is large enough. This observation motivated the proposal of a static link model to capture self-clocking effect under TCP window control [24], [25]. Using the static link model, it has been theoretically shown that FAST TCP flows are always stable for the homogeneous case [8], [24], [25]. As the static model fails to hold in heterogeneous cases, we certainly need a better model to study the general case.

It turns out that both the integrator link model and the static link model are incomplete; the former tends to lag the true dynamics yielding conservative stability results, while the latter leads the true dynamics yielding optimistic predictions (See examples 1, 2, 3 and 5). After introducing basic notations and preliminary knowledge, we will show in Section II that a natural combination of these two models leads to a more accurate one. Open loop experiments are provided to validate this joint model and reveal its underlying intuition. To illustrate its application, we then specialize in Section III to the stability of FAST TCP. Using the joint link model, we prove that FAST running over a single link is stable for any heterogeneous delays, and hence resolve the discrepancy between previous experimental results and existing theoretical predictions. Closed loop experiments are also reported where accurate predictions on stability region is obtained and verified with packet level simulations[1]. We conclude in Section IV and provide some possible directions to extend the current work.

II. Model and Notation

To capture the self-clocking effect in window based congestion control, we avoid working directly with the sources'

[1]To the best of our knowledge, the current status of research on congestion control protocols can provide quantitative results on equilibrium, while for dynamics, most works focus on qualitative study and have not been able to compare predictions with packet level simulations quantitatively.

0743-166X/07/$25.00 ©2007 IEEE

sending rates as states of the protocol. Instead, we use the sources' window sizes and the bottleneck queue size to represent the state of the closed loop system.

A. Preliminaries

We consider N window-based TCP sources sending over a bottleneck link with capacity c. Let $w_n(t)$ denote the congestion window of source n at time t, $n \in \{1, \dots, N\}$. Let a packet that is sent by source n at time t appear at the bottleneck queue at time $t + \tau_n^f$. This *forward delay* τ_n^f models the amount of time it takes to travel *from* source n *to* the link, and it accounts for the constant forward latency but not queuing delays. The *backward delay* $\tau_n^b(t)$ is defined in the same manner: it is the time from when a packet arrives at the link to when the corresponding acknowledgment is received at source n. Note that the backward delay includes the time-dependent queuing delay at the bottleneck queue. The *round trip delay* $\tau_n(t)$ seen by source n is the elapsed time between when a packet is sent and when the corresponding acknowledgment is received; naturally $\tau_n(t) = \tau_n^f + \tau_n^b(t)$. The *latency* of source n is denoted d_n and is defined as the minimum achievable round trip delay, i.e. the round trip delay when the bottleneck queue is empty.

The queueing delay of the bottleneck link is denoted by $p(t)$, and $c > 0$ is the capacity of the link. The queuing delay observed by the nth source at time t is $q_n(t)$; it relates to the queue delay by $q(t) = p(\tilde{t})$, where $\tilde{t}$ solves $t = \tilde{t} + \tau_n^b(\tilde{t})$.

The bottleneck link may also carry non-window-based traffic such as User Datagram Protocol (UDP) traffic. Let $x_c(t) \in [0, c]$ be the rate (averaged over a suitable time interval) at which non-window-based cross-traffic is sent over the link. This implies that the *available bandwidth*, shared between the window based sources sending over the link, is $c - x_c(t)$.

Whenever a time argument of a variable is omitted it represents its equilibrium value, e.g., $p(t)$ corresponds to the queueing delay variable at time t while p is its equilibrium value. When working in discrete time the convention $w(k) := w(t_k)$ will be used.

B. Link model and validation

As described in Section I, previous work differs in how the dynamic map between the window sizes and the buffer size is modeled. Most existing literature on window based congestion control, see [2], [4], [11], [16], [17], [29], assumes that the sending rate is proportional to the window size divided by the round trip delay and may further model the queue as a simple integrator, integrating the excess rate at the link, i.e.,

Integrator link model:

$$\dot{p}(t) = \frac{1}{c}\left(\sum_{n=1}^{N} \frac{w_n(t - \tau_n^f)}{d_n + p(t)} + x_c(t) - c\right). \tag{1}$$

The model (1) does not, however, take into account the "self-clocking" characteristic of window based schemes, where the sending rate is regulated by the rate of the received ACKs. To model this phenomenon, an alternative model is used in [24], [27] and implicitly in [21], based on the empirical observation that, due to "self-clocking", transient effects are negligible. The relation between the window size and the buffer size is then described by the algebraic relation[2]

Static link model:

$$\sum_{n=1}^{N} \frac{w_n(t - \tau_n^f)}{d_n + p(t)} = c - x_c(t). \tag{2}$$

In (2), a change in any of the sources' congestion windows w_n results in a proportional change in the queueing delay p one forward delay τ_n^f later. This is in contrast to the integrator model (1) where a window change gives a smooth queue transition. Obviously the two models are fundamentally different. We will now combine their main features and arrive at a more accurate joint model.

1) Joint link model: The joint model is justified heuristically in this section, with the aim of emphasizing the underlying intuition. See [12] for a rigorous derivation, where the same model results from a thorough and detailed analysis of the system at the packet level.

To understand the difference between (and the accuracy of) the integrator model (1) and the static model (2), the key is to examine cases where flows have heterogeneous round trip delays or where non-window-based cross traffic such as UDP exist. Consider the case of N flows sending over the bottleneck link with constant window sizes, and consider the response to a change in window size by a system initially in equilibrium.

The long term effect of a window change is that the queue integrates the excess rate. This is well known, and captured by the integrator model (1).

The short term effect of a window change is more complex, and often wrongly ignored. Since the link is fully utilized in equilibrium, the queue's immediate response to a window change (that is, transmission control injects an extra packet or discards an ACK) is a proportional change in the queue size; this will occur one forward delay τ_n^f after the window w_n is changed. If there is no cross traffic and the sources' share the same round trip time (RTT), there is no further transient. This is due to the fact that sources' sending rates are auto-regulated by their individual ACK rates which sum up to the capacity; subsequently the queue input rate will equal its output rate (capacity). This is in line with the static queue model (2) which neglects transient behavior. However, in the case when there is stationary UDP cross traffic, *sources can affect their ACK rates over time intervals greater than one RTT* (actually it is a function of the amount of cross traffic, see [12]). This is also true when no cross traffic is present but the heterogeneity among sources' RTT is large enough. As individual flows operate on their individual RTT time scales and it takes one RTT before a queue change affects the queue input rate, from the perspective of flows with small round times, flows with larger RTTs can be considered as non-responsive cross traffic and the system is hence transient in this case as well.

[2]The original model was presented in discrete time for multiple bottlenecks, here we use its continuous time version used in e.g. [8] and we consider a single bottleneck.

Adopting the standard fluid flow approximation, that packets transmitted from a source form a continuous smooth flow, and motivated by the previous discussion we capture both the short and long term behavior in a single model,

Joint link model:

$$\dot{p}(t) = \frac{1}{c}\left(\sum_{n=1}^{N}\left(\frac{w_n(t-\tau_n^f)}{d_n+p(t)} + \dot{w}_n(t-\tau_n^f)\right) + x_c(t) - c\right) \quad (3)$$

which can be seen as a superposition of (1) and (2). The derivative term $\dot{w}(t-\tau_n^f)$ models the immediate proportional change in the queue size due to a window change. Note that it is the window size and its corresponding time derivative *only* that have delayed variable arguments, which furthermore are identical. This is rigorously motivated in the full model derivation in [12]. A similar model was also implicitly used for flow control stability analysis in [3].

Consider the Laplace domain transfer function from the window size of flow n to the queue derived from the linearized version of (3)

$$G_{pw_n}(s) = \frac{\mathcal{L}[p(t)]}{\mathcal{L}[w_n(t)]} = \frac{\frac{1}{c}\left(s+\frac{1}{d_n+p}\right)e^{-s\tau_n^f}}{s+\frac{1}{c}\sum_{i=1}^{N} w_i/(d_i+p)^2}. \quad (4)$$

For the the case with homogeneous delay $d_n = d$ and no cross traffic, applying the identity $\sum_{n=1}^{N} w_n/(d+p) = c$ shows that the transfer function zero and pole will cancel and hence that the map is a pure delay, scaled by $1/c$. This agrees with the description in the previous discussion as well as in the model (2). Moreover, the dynamics will be more distinguishable with increasing heterogeneity among the sources and cross traffic as in (1). Finally, note that (4) is open loop stable, as expected due to self-clocking.

2) Open loop validation experiments: The accuracy of the joint model (3) as well as its similarities and differences with the integrator model (1) and the static model (2) is illustrated in the following open loop examples. A closed loop experiment will be reported in Section III-E.

The models (1), (2) and (3) are validated with packet level data generated by using the NS-2 FAST TCP module [7]. In each experiment we consider 20 window based flows with static windows, sending over a single bottleneck link. Non-bottleneck links provide configurable forward and backward delays. The sources are window based and the window sizes are initially set to the same constant size and are *not* updated dynamically, i.e. there is no dynamic feedback except for the ACK-clocks. The system is started in equilibrium and perturbed at time $t = 10\,\text{s}$ by a 10% step change in the first source's congestion window. In all experiments a packet size of 1040 bytes is used.

Example 1: Homogeneous sources

All 20 window based flows share the same latency $d_n = 200\,\text{ms}$ and the bottleneck link capacity is set to $c = 100\,\text{Mbit/s}$. The window size is $w_n = 125$ packets. Source 1, which is subject to the window change, has a forward delay of $\tau_1^f = 100\,\text{ms}$. The solid gray line in Figure 1 shows the queue size when the system is simulated in NS-2. The black dashed, dotted and solid lines show the integrator model (1), the static model (2) and the joint model (3) respectively. The fit of the

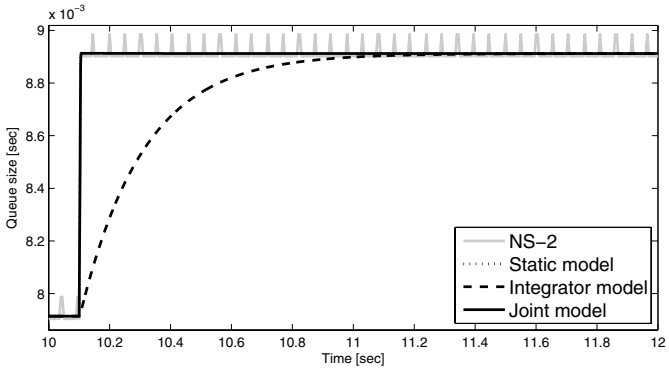


Fig. 1. Homogeneous sources. Both the joint model and the static model agree with the NS-2 simulation, but the integrator model lags significantly.

static and the joint models is excellent (neglecting packet level "noise"); they are identical in this scenario. This suggests that the true dynamics in this case is indeed a pure delay. Also observe that the integrator model lags the NS-2 simulation. Note that it takes 100 ms before the window change affects the queue, as predicted by the models.

Example 2: A cross traffic scenario

This scenario is as in Example 1 but with bottleneck link capacity $c = 500\,\text{Mbit/s}$ which is also shared by 400 Mbit/s of UDP traffic. In this case the dynamics are clearly distin-

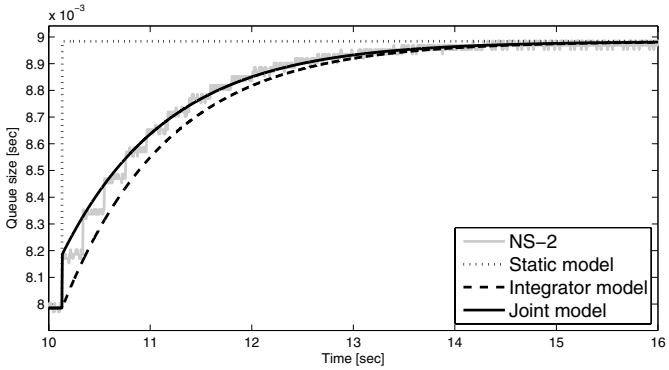


Fig. 2. Cross traffic. Both the joint model and the integrator model agree with the NS-2 simulation, but the static model leads significantly.

guishable; it takes over four seconds before the queue settles again or over twenty round trip times. The static model is too rapid in this case, as expected, while the other two are both good. The joint model captures the rapid initial rise in queue size, and initially tracks the upper envelope of the staircase simulation results, while the integrator model tracks the lower envelope. From 1 s after the transient, the joint model tracks the mean while the integrator lags slightly.

Example 3: Heterogeneous sources

This scenario is as in Example 1 but with $d_1 = 50\,\text{ms}$, distributed such that $\tau_1^f = 25\,\text{ms}$, for the first source, and

$d_n = 250$ ms for the remaining 19 sources, and $w_n = 135$ packets.

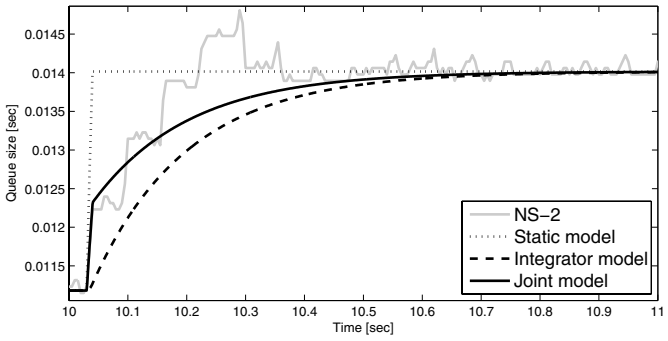


Fig. 3. Heterogeneous sources. The joint model captures both the initial jump and the protracted rise; the integrator model misses the former, while the static model misses the latter.

In this case there is a more pronounced initial increase in the queue followed by a transient phase which dies out after about 250 ms. This corresponds to the time before the ACK-clocks of the high delay sources adjust to the new conditions.

In summary, these three examples, from different aspects, demonstrate that while the integrator link model may lag the true dynamics and the static link model can lead the true dynamics a lot, the joint model (3) succeeds in modeling the two main system characteristic of "self-clocking", i.e., the short term proportional change, and the long term integrating effect that are present in the system. For rigorous derivation of the model and more validation experiments, see [12].

III. Linear Stability of FAST TCP

FAST [27] is a high speed TCP variant that uses delay as its main control signal. So far, all experiments with FAST TCP have operated at a stable equilibrium regardless of what the round trip delays are. This section will use the joint link model (3) to show that FAST is indeed locally stable for a single bottleneck with default step size.

Unlike [24], [25], this analysis is for flows with heterogeneous delays. In this case, there is no longer a natural time step, and it becomes simpler to work in continuous time.

A. Window model of FAST TCP

The sending rate of FAST TCP is implicitly adjusted via the congestion window mechanism. Each sender updates its window size in discrete time according to

$$w_n(k+1) = (1-\gamma_n)w_n(k) + \gamma_n \frac{d_n}{d_n + \hat{q}_n(k)} w_n(k) + \gamma_n \alpha_n \tag{5}$$

where $\alpha_n \in \mathbb{Z}^+$ and γ_n are protocol parameters [27]. This update is performed *once* every RTT. The queuing delay $q_n(k)$ is *estimated* by the source and the kth estimate is denoted by $\hat{q}_n(k)$. The window algorithm operates in time scale of RTTs while the estimator is such that it operates on a time scale of packets; for the case of high bandwidth and latency, the estimator operates at a much faster time scale than the window algorithm. This work will therefore ignore estimator dynamics and use $\hat{q}_n(k) = q_n(k)$ in (5).

To obtain a continuous time approximation of the window control, first rewrite (5) as

$$\begin{aligned} \frac{w_n(k+1) - w_n(k)}{\tau_n(k)} &= \frac{\gamma_n}{\tau_n(k)} \cdot \frac{d_n}{d_n + q_n(k)} w_n(k) \\ &\quad - \frac{\gamma_n}{\tau_n(k)} w_n(k) + \frac{\gamma_n \alpha_n}{\tau_n(k)}. \end{aligned} \tag{6}$$

Using a first order Euler approximation of the derivative and applying the identity $\tau_n(t) = d_n + q_n(t)$ yields the continuous time window update

$$\dot{w}_n(t) = -\gamma_n \frac{q_n(t)}{(d_n + q_n(t))^2} w_n(t) + \gamma_n \frac{\alpha_n}{d_n + q_n(t)}. \tag{7}$$

Linearizing (7) around (w, q), and noting that in equilibrium $\alpha_n/q = w_n/(d_n + q)$ $[= (c - x_c)\alpha_n / \sum_m \alpha_m]$, gives

$$\dot{\tilde{w}}_n(t) = -\gamma_n \frac{q}{{\tau_n}^2} \tilde{w}_n(t) - \gamma_n \frac{\alpha_n d_n}{q {\tau_n}^2} \tilde{q}_n(t) \tag{8}$$

where $\tilde{w}_n$ and $\tilde{q}_n$ represent perturbed variables. Adopting the standard convention that whenever a round trip time, or forward and backward delay, appears in a variable argument, it is replaced by its equilibrium value [18], [25], yields

$$\tilde{q}_n(t) = \tilde{p}(t - \tau_n^b) \tag{9}$$

around equilibrium.

B. Loop Gain

Combining the Laplace transform version of the linear window dynamics (8) and the communicated price (queuing delay) (9) with the frequency domain version of the linear queue dynamics (4), results in a negative feedback system with open loop transfer function

$$L(s) = \sum_{n=1}^{N} \mu_n L_n(s) \tag{10}$$

where

$$\mu_n = \frac{\alpha_n}{cq} = \frac{\alpha_n}{c \sum_m \alpha_m} \tag{11}$$

$$L_n(s) = \frac{s + \frac{1}{\tau_n}}{s + \frac{1}{\hat{\tau}}} \frac{d_n \gamma_n e^{-\tau_n s}}{\tau_n^2 s + \gamma_n q} \tag{12}$$

$$\frac{1}{\hat{\tau}} = \sum_{n=1}^{N} \mu_n \frac{1}{\tau_n}. \tag{13}$$

When $x_c = 0$, we can interpret $\hat{\tau}$ as a weighted harmonic mean value of τ_n. In particular, when all flows have equal α_n, giving $\mu_n = 1/N$, $\hat{\tau}$ is the harmonic mean of τ_n.

C. Stability analysis

For notational simplicity, we will consider the case $\gamma_n = \gamma$ in the rest of this paper. Intuitively, for given parameters, the stability degrades as $q \to 0$, as each $L_n(s)$ will have larger

gain and lose more phase. We now focus on the case of $q \to 0$, i.e., the open loop transfer function tends to

$$L(s) = \sum_n^N \mu_n \frac{s + \frac{1}{\tau_n}}{s + \frac{1}{\hat{\tau}}} \frac{\gamma e^{-s\tau_n}}{s\tau_n}. \quad (14)$$

Define $H(\omega)$ as the half plane under the line that passes $-1 + j0$ with slope $1/(\omega\hat{\tau})$. Formally, we have

$$H(\omega) = \left\{ x \mid \arg(x+1) - \arctan\left(\frac{1}{\omega\hat{\tau}}\right) \in (-\pi, 0) \right\}. \quad (15)$$

Lemma 1: If

$$\sum_{n=1}^N \frac{\mu_n}{\tau_n} \left(\cos(\omega\tau_n) - \frac{\sin(\omega\tau_n)}{\omega\tau_n} \right) + \frac{1}{\hat{\tau}\gamma} > 0 \quad (16)$$

then in the limit $\alpha_n \to 0$, with α_n/α_m fixed for all m, n,

$$L(j\omega) \in H(\omega). \quad (17)$$

Proof: The limit $\alpha_n \to 0$ gives $q \to 0$ since $q = \sum_m \alpha_m$, and by definition $L(j\omega) \in H(\omega)$ becomes equivalent to

$$\arg(L(j\omega) + 1) - \arctan\left(\frac{1}{\omega\hat{\tau}}\right) \in (-\pi, 0). \quad (18)$$

Substituting (14) and noting that

$$\arg\left(j\omega + \frac{1}{\hat{\tau}}\right) + \arctan\left(\frac{1}{\omega\hat{\tau}}\right) = \frac{\pi}{2}, \quad (19)$$

condition (18) can be further rewritten as

$$\arg\left(\sum_{n=1}^N \mu_n \left(j\omega + \frac{1}{\tau_n} \right) \frac{\gamma e^{-j\omega\tau_n}}{j\omega\tau_n} + j\omega + \frac{1}{\hat{\tau}} \right) \in \left(-\frac{\pi}{2}, \frac{\pi}{2} \right) \quad (20)$$

which is equivalent to

$$\begin{aligned} \mathrm{Re}\left(\sum_{n=1}^N \mu_n \left(j\omega + \frac{1}{\tau_n} \right) \frac{\gamma e^{-j\omega\tau_n}}{j\omega\tau_n} + j\omega + \frac{1}{\hat{\tau}} \right) &> 0 \\ \iff \sum_{n=1}^N \frac{\mu_n}{\tau_n} \left(\gamma\cos(\omega\tau_n) - \frac{\gamma\sin(\omega\tau_n)}{\omega\tau_n} \right) + \frac{1}{\hat{\tau}} &> 0. \end{aligned}$$

Dividing by $\gamma > 0$ gives the hypothesis of the lemma. ■

The construction used for Lemma 1 is depicted in Figure 4, for $\tau_1 = 1$, $\tau_2 = 5$, $\mu_1 = \mu_2 = 1/2$, at $\omega\hat{\tau} = 3$.

An analogous lemma for the general case of $\alpha_n \geq 0$ is given in Appendix A.

Remark: The techniques used here are significantly different from ones in the existing literature on linear stability of TCP, in two aspects. First, the usual approach is to find a convex hull that contains all individual $L_n(j\omega)$ curves and then argue any convex combination of them is still contained by the convex hull. See, e.g., [23], [18], [6]. However, the proof of Lemma 1 deals directly with $L(j\omega)$ instead of $L_n(j\omega)$. Second, for each ω, a separate region is found to bound $L(j\omega)$ away from the interval $(-\infty, -1]$. That is, the half plane $H(\omega)$ defined by (15) depends on ω. In existing works, convex regions are typically used to bound the whole curves and hence are independent of ω. One exception is [20], where the frequency range is divided into two parts and different convex regions are used in the two parts. These two features lead to tighter bounds, which is necessary for the analysis of this problem. ■

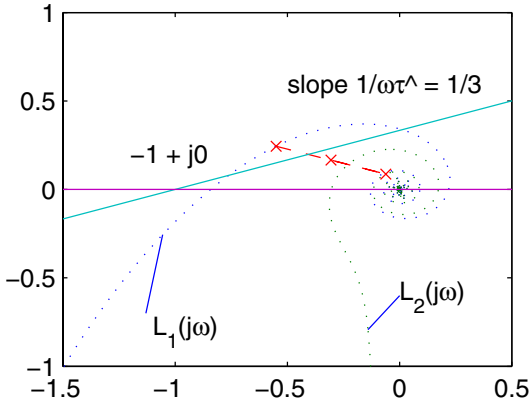


Fig. 4. An example of a line of slope $1/(\omega\hat{\tau})$ which bounds $L(j\omega)$, denoted by the center cross. Note that the individual terms $L_n(j\omega)$, denoted by the individual crosses, are not all below this line.

We are now ready to prove the following main theorem.

Theorem 2: If $\gamma \leq 0.94$, the model (10) for FAST TCP operating over a single link is locally stable.

Proof: Define $F(\theta) = \cos(\theta) - \sin(\theta)/\theta$ and denote its minimal value by $F_{\min}$. Then

$$\sum_{n=1}^N \frac{\mu_n}{\tau_n} \left(\cos(\omega\tau_n) - \frac{\sin(\omega\tau_n)}{\omega\tau_n} \right) \geq F_{\min} \sum_{n=1}^N \frac{\mu_n}{\tau_n} = \frac{F_{\min}}{\hat{\tau}}. \quad (21)$$

By (16), if $\gamma < -1/F_{\min}$, then the condition of Lemma 1 is satisfied and the Nyquist curve cannot encircle -1, since

$$(-\infty, -1] \cap \bigcup_{\omega > 0} H(\omega) = \emptyset.$$

It is straight forward to find $F_{\min}$ is -1.0631 which is attained when $\theta = 2.7437$, the smallest positive solution to $\tan(\theta) = \theta/(1-\theta^2)$. ($F(\theta)$ is plotted in Figure 5). Therefore, the assumption on γ guarantees

$$\gamma < 0.94 < \frac{1}{1.0631} = -\frac{1}{F_{\min}}. \quad (22)$$

■

Remark: Note that an effective value of $\gamma = 0.5$ is used for the FAST implementation [27]. For this case, Theorem 2 immediately establishes FAST TCP's stability for any pattern of round trip delays. This fully explains the fact that FAST TCP has been stable for all experimental cases studied. ■

Theorem 2 is proved by finding a uniform bound for all flows' τ_n. If we have more detailed knowledge about the round trip delay distribution, we may achieve even better bounds. Let us now consider some special cases.

1) $N = 1$: If there is a single flow, $\mu_1 = 1$ in (14) and the joint link model degenerate to the static link model. In this case, FAST is stable for all $\gamma < \pi/2$. In this case, (22) is loose simply because the frequency, $\omega = 2.7437/\tau$, which

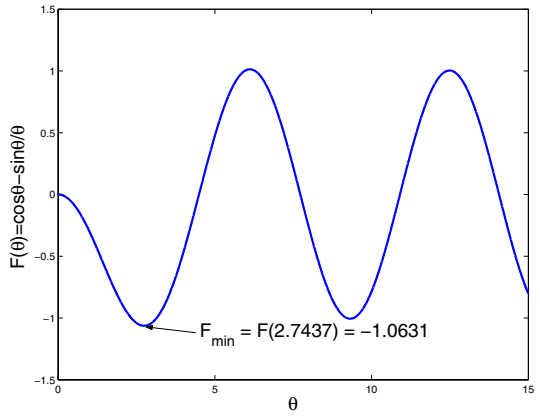


Fig. 5. $F(\theta) = \cos(\theta) - \sin(\theta)/\theta$

k	1	1.5	2	5	10	20
Stability bound	0.940	1.052	1.294	1.164	0.947	0.945

TABLE I
MAXIMUM γ FOR STABILITY FOR CASES WITH TWO FAST FLOWS WITH DIFFERENT RTT; HETEROGENEITY, k; AND $q = 0$

minimizes $F(\omega\tau)$ does not coincide with a frequency at which the Nyquist plot of $L(s)$ crosses the real axis.

2) $N = 2$: Consider two FAST flows with $\mu_1 = \mu_2$ (corresponding to the current practice that all FAST flows share the same α). Write $\tau_1 = k\tau_2$, where k measures the heterogeneity. Define

$$F(\theta, k) = \cos(\theta) - \frac{\sin(\theta)}{\theta} + \frac{1}{k}\left(\cos(k\theta) - \frac{\sin(k\theta)}{k\theta}\right)$$

and let its minimal value over θ be $F_{\min}(k)$. It then follows that a sufficient condition for stability is

$$\gamma < \frac{k+1}{k}\frac{1}{-F_{\min}(k)}. \tag{23}$$

Table I lists the stability bounds for a few values of k. It is straightforward to show that the bound first increases and then decreases in k and asymptotically when $k \to \infty$, the bound is again 0.94, the same as the case of $k = 1$.

3) $N = \infty$: In reality, the link is likely to be shared by many flows. It is then interesting to find the statistical mean value of the stability bound for those scenarios. We will now consider the case of many flows with continuously distributed RTTs, letting $\alpha_n \to 0$ with α_m/α_n fixed.

Let $M(\tau) = \sum_{\tau_n \le \tau} \alpha_n/cq$, and let all τ_n be in the range $\Omega = (\underline{\tau}, \bar{\tau})$, with $\bar{\tau}$ possibly infinite. If there are many flows with RTTs drawn from a continuous distribution, then applying $\mu(\tau) = M'(\tau)$ to (16) gives

$$\int_\Omega \frac{\mu(\tau)}{\tau}\left(\cos(\omega\tau) - \frac{\sin(\omega\tau)}{\omega\tau}\right) d\tau + \frac{1}{\gamma}\int_\Omega \frac{\mu(\tau)}{\tau}\, d\tau > 0. \tag{24}$$

Noting that

$$\frac{d}{d\theta}\frac{\sin(\theta)}{\theta} = \frac{\cos(\theta)}{\theta} - \frac{\sin(\theta)}{\theta^2},$$

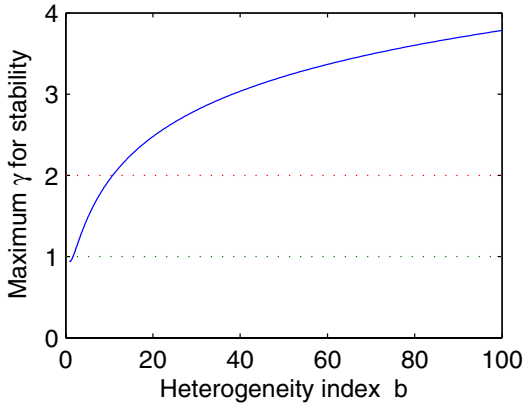


Fig. 6. Maximum value for γ for stability with RTTs in $(1, k)$.

and setting $\theta = \omega\tau$, (24) becomes

$$\frac{1}{\gamma} > -\frac{\displaystyle\int_{\omega\Omega} \mu\left(\frac{\theta}{\omega}\right)\left(\frac{\sin(\theta)}{\theta}\right)' d\theta}{\displaystyle\int_{\omega\Omega} \frac{\mu(\theta/\omega)}{\theta}\, d\theta} \tag{25}$$

$$= -\frac{\left[\mu\left(\frac{\theta}{\omega}\right)\frac{\sin(\theta)}{\theta}\right]_{\omega\underline{\tau}}^{\omega\bar{\tau}} + \displaystyle\int_{\omega\underline{\tau}}^{\omega\bar{\tau}} \frac{\mu'(\theta/\omega)}{\omega}\frac{\sin(\theta)}{\theta}\, d\theta}{\displaystyle\int_{\omega\underline{\tau}}^{\omega\bar{\tau}} \frac{\mu(\theta/\omega)}{\theta}\, d\theta} \tag{26}$$

where $\omega\Omega = (\omega\underline{\tau}, \omega\bar{\tau})$ and $(\cdot)'$ denotes derivative. This must hold for all $\omega > 0$.

As an example, assume RTTs follow a uniform distribution. As units of time are arbitrary, this can be modeled without loss of generality as

$$\mu(\tau) = \begin{cases} 1/(k-1) & \tau \in (1, k) \\ 0 & \text{otherwise,} \end{cases} \tag{27}$$

with $k > 1$. In that case, (26) becomes

$$\frac{1}{\gamma} > \max_{\omega>0}\left(-\left[\frac{\sin(\theta)}{\theta}\right]_{\omega}^{\omega k} \Big/ \int_{\omega}^{\omega k} \frac{d\theta}{\theta}\right) \tag{28}$$

$$= \max_{\omega>0}\left(\frac{\sin(\omega)k - \sin(\omega k)}{\omega k \log(k)}\right). \tag{29}$$

As shown in Appendix B, the right hand side approaches $-F_{\min}$ as $k \to 1$, in accordance with (22), while for $k > 1$ the bound is strictly looser as shown in Figure 6, and asymptotes to $1/\log(k)$ for large k. Already for $k = 1.69$, the upper bound exceeds 1, while for $k = 10.79$ it becomes 2. If $\gamma > 2$ then the discrete rule (5) becomes intrinsically unstable since each update overshoots by more than the current error, and so it is not helpful to increase γ beyond 2 in the continuous time model.

Note that in this case, $\mu(\tau)$ reflects both the distribution of RTTs and also differences in α values of different flows. If all FAST flows use the same α which is true for the current implementation, then $\mu(\tau)$ is the distribution of RTTs.

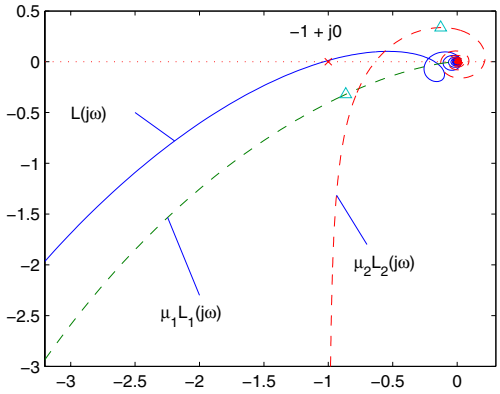


Fig. 7. Nyquist plot of the system of Section III-D. Triangles show the unstable frequency, $\omega = 1$.

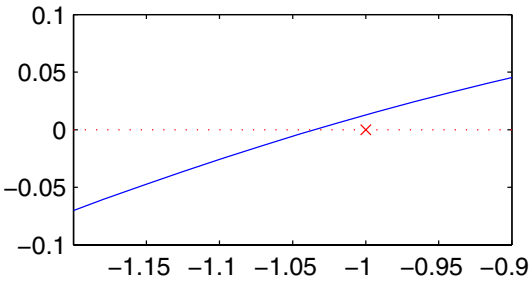


Fig. 8. Close-up of the Nyquist plot of the system of Section III-D.

D. A counter-example

Because the above argument proves stability for γ so close to 1, it is tempting to seek to show stability for all $\gamma \leq 1$. However, the following example breaks that hope. It also illustrates how heterogeneity can potentially hurt stability.

Example 4: A counter example
Consider a network with a single bottleneck link carrying two flows. The flows have RTTs $\tau_1 = 0.1$ ms and $\tau_2 = 2750$ ms, with $\alpha_1 = 1$ and $\alpha_2 = 10^8$ with $c = 10^{18}$ packets per second. This gives $q = 10^{-10}$, $\mu_1 = 10^{-8}$, $\mu_2 = 1 - \mu_1$ and $\hat{\tau} = 2749$ ms.[3]

With $\gamma = 1$, in contrast to the implemented $\gamma = 0.5$, this extreme system is unstable. Instability arises because of the high heterogeneity between the RTTs of the flows, and the greater heterogeneity between the weights given to the flows.

The Nyquist curve for this network with $\gamma = 1$ is shown with the solid line in Figure 7. The dashed lines show the individual curves $\mu_1 L_1(j\omega)$ and $\mu_2 L_2(j\omega)$, and the triangles show these curves for the frequency $\omega = 1$ at which $L(j\omega)$ first crosses the real axis. The magnified view of this curve near the point $-1+j0$ in Figure 8 shows that the Nyquist curve does indeed encircle -1 and the resulting system is unstable.

In this example, most of the weight is given to flow 2, and the instability occurs when $\omega\tau_2 \approx 2.75$, minimizing the curve in Figure 5 with $F(\omega\tau_2) \approx -1.06$. Although τ_1 is very small, μ_1 is even smaller, making the coefficient μ_1/τ_1 in (16) negligible, and allowing (16) to be violated by the τ_2 term. However, the extra factor of $1/\tau_1$ provided by the numerator of the first factor in (12) allows the imaginary part of $\mu_1 L_1(j)$ to balance that of $\mu_2 L_2(j)$ where the curve crosses the axis.

This example shows that two flows are sufficient to cause instability, even though a network with a single flow (or multiple homogeneous flows) is always stable. It is also possible to construct a network of three flows with slightly less extreme parameters ($\mu_1 = 1.2 \times 10^{-5}$, $\mu_2 = 0.982$, $\mu_3 = 0.0179$, $\tau_1 = 300$ ms, $\tau_2 = 150.1$ ms and $\tau_3 = 0.2$ ms). The final Nyquist plot looks very similar to that of Figure 7.

E. (De)stabilized FAST TCP: closed loop experiment

In this subsection, we will use cases with $\gamma > 1$ to compare stability predictions of all three models. The objectives here are twofold. First to investigate the critical step size for FAST to maintain stability. This can potentially suggest a larger step size for quicker response. Second, by comparing three models' predictions, this closed loop experiment will further strengthen the validation results in Section II-B.1 where open loop experiments are reported.

Example 5: Closed loop validation
Two FAST TCP flows share a single link with capacity of 10 000 pkt/s. The propagation delays of the two flows are 400 ms and 700 ms, respectively. Both flows use $\alpha = 50$. The open loop transfer functions for all three models and the critical step size (γ_c) for stability predicted by those models are summarized below. The integrator model predicts a critical step size much smaller than the one from the static model, while the joint model yields a prediction in between as expected.

- Integrator model: $\gamma_c = 1.23$

$$L(s) = \sum_{n=1}^{N} \mu_n \frac{1/\tau_n}{s + 1/\hat{\tau}} \frac{d_n \gamma_n e^{-\tau_n s}}{\tau_n^2 s + \gamma_n q} \tag{30}$$

- Static model: $\gamma_c = 1.80$

$$L(s) = \sum_{n=1}^{N} \mu_n \frac{d_n \gamma_n e^{-\tau_n s}}{\tau_n^2 s + \gamma_n q} \tag{31}$$

- Joint model: $\gamma_c = 1.69$

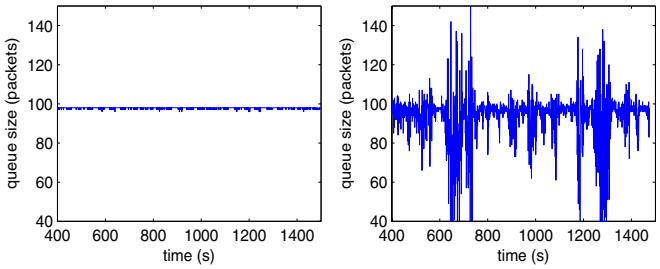
$$L(s) = \sum_{n=1}^{N} \mu_n \frac{s + 1/\tau_n}{s + 1/\hat{\tau}} \frac{d_n \gamma_n e^{-\tau_n s}}{\tau_n^2 s + \gamma_n q} \tag{32}$$

We now report NS-2 packet level simulations [7].[4] Figure 9 shows the queue trajectories with $\gamma = 1.23$ and 1.80, the critical step size predicted by the integrator link model and the static link model. It is clear that the queue is not stable with $\gamma = 1.80$, which means the static model is too optimistic for stability analysis. We further show queue trajectories with

[3]The parameters in this subsection are quite extreme. The example here is primarily of theoretical interest as it gives an upper bound on the γ which can yield guaranteed stability. We report unstable cases with realistic parameters in Section III-E.

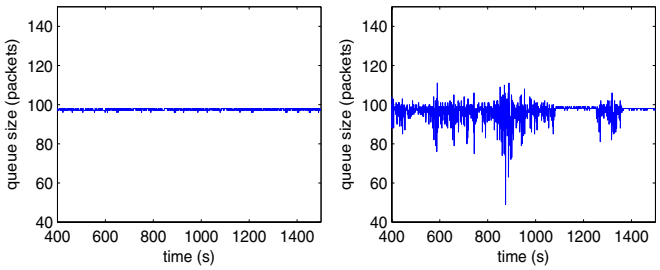
[4]To validate the link model, the code was modified to update the window once per RTT, and for modeling simplicity the RTT estimate was evaluated over 0.1 RTT. All queue trajectories are plotted after initial transients, to emphasize the local stability of the congestion avoidance phase.

$\gamma = 1.65$ and 1.75 in Figure 10. The case with $\gamma = 1.65$ is still stable which suggests that the integrator model is too conservative. Note the queue starts to oscillate with $\gamma = 1.75$, suggesting that the critical step size ($\gamma_c = 1.69$) predicted by the new joint model is surprisingly accurate.



(a) $\gamma = 1.23$ (b) $\gamma = 1.80$

Fig. 9. Queue trajectories with critical step sizes predicted by the integrator link model and the static link model.



(a) $\gamma = 1.65$ (b) $\gamma = 1.75$

Fig. 10. Queue trajectories around critical step sizes.

IV. Conclusion

We have presented a link model which captures the queue dynamics when congestion windows of TCP sources change. The model is shown to be much more accurate than existing ones. It agrees with the known static link model when flows' round trip delays are similar, and approximates the standard integrator link model when the heterogeneity of round trip delays is significant. Using this new model, we have shown that FAST TCP is always stable for networks with a single bottleneck link. This extends the existing stability result on homogeneous FAST flows to cases with heterogeneous delays and resolves the notable discrepancy between empirical observations and existing theoretical predictions. The analysis highlights the critical role of self-clocking in TCP stability and the scalability of FAST TCP with respect to delay. Throughout this paper, various open loop and closed loop simulations are used to validate our predictions. In particular, we are able to predict stability region of the closed loop system accurately compared to packet level simulations.

There are several possible directions in which to extend this work. For example, we have assumed the number of flows is fixed and it is certainly very interesting to see the effect of this more accurate link model on networks with dynamically arriving and departing flows [5]. The model itself is of use for any window based TCP algorithm, and it would be useful to investigate its impact on other protocols. So far, we have only looked at its implications for stability. It will be of great interest to see its prediction of performance, such as queue distribution. Finally, the model and analysis remain to be extended to general networks which can potentially have multiple congested links.

Acknowledgments

The authors thank David Wei and Fernando Paganini for valuable discussions. This is part of the Caltech FAST Project supported by NSF, Caltech Lee Center for Advanced Networking, ARO, AFOSR, and Cisco.

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 *Proceedings of IEEE Conference on Decision and Control*, 2003.

[2] E. Altman, C. Barakat and V. Ramos. Analysis of AIMD protocols over paths with variable delay. In *Proceedings of IEEE Infocom*, 2004.

[3] L. L. H. Andrew, S. V. Hanly and R. G. Mukhtar. CLAMP: A system to enhance the performance of wireless access networks. In *Proceedings of IEEE Globecom*, pp. 4142-4147, 2003.

[4] F. Baccelli and D. Hong. AIMD, fairness and fractal scaling of TCP traffic. In *Proceedings of IEEE Infocom*, 2002.

[5] T. Bonald and L. Massoulie. Impact of fairness on Internet performance. In *Proceedings of ACM Sigmetrics*, June 2001

[6] Hyojeong Choe and S. H. Low. Stabilized Vegas. In *Proceedings of IEEE Infocom*, 2003.

[7] T. Cui and L. Andrew. FAST TCP simulator module for ns-2, version 1.1. Available ⟨http://www.cubinlab.ee.mu.oz.au/ns2fasttcp⟩.

[8] J. Choi, K. Koo, J. Lee and S. H. Low. Global stability of FAST TCP in single-link single-source network. In *Proceedings of IEEE Conference on Decision and Control*, 2005.

[9] S. Deb and R. Srikant. Global stability of congestion controllers for the Internet. *IEEE/ACM Transactions on Automatic Control*, 48(6):1055-1060, June 2003.

[10] C. Hollot and Y. Chait. Nonlinear stability analysis for a class of TCP/AQM schemes. In *Proceedings of IEEE Conference on Decision and Control*, 2001.

[11] C. Hollot, V. Misra, D. Towsley and W. Gong. A control theoretic analysis of RED. In *Proceedings of IEEE Infocom*, 2001.

[12] K. Jacobsson, H. Hjalmarsson, and N. Möller. ACK-clock dynamics in network congestion control – an inner feedback loop with implications on inelastic flow impact. In *Proceedings of the 45th IEEE Conference on Decision and Control*, San Diego, USA, December 2006.

[13] R. Johari and D. Tan. End-to-end congestion control for the Internet: delays and stability. *IEEE/ACM Transactions on Networking*, 9(6):818-832, December 2001.

[14] K. Kim, A. Tang and S. H. Low. Design of AQM in supporting TCP based on the well-known AIMD model. In *Proceedings of IEEE Globecom*, 2003.

[15] K. Kim, A. Tang and S. H. Low. A stabilizing AQM based on virtual queue dynamics in supporting TCP with arbitrary delays. In *Proceedings of IEEE CDC*, 2003.

[16] S. Liu, T. Basar and R. Srikant. Pitfalls in the fluid modeling of RTT variations in window-based congestion control. In *Proceedings of IEEE Infocom*, 2005.

[17] S. H. Low, F. Paganini, and J. C. Doyle. Internet congestion control. *IEEE Control Systems Magazine*, 22(1):28–43, Feb. 2002.

[18] S. H. Low, F. Paganini, J. Wang, and J. C. Doyle. Linear stability of TCP/RED and a scalable control. *Computer Networks Journal*, 43(5):633–647, 2003.

[19] L. Massoulie. Stability of distributed congestion control with heterogeneous feedback delays. *IEEE Transactions on Automatic Control*, 47(6): 895-902, June 2002

[20] F. Paganini, Z. Wang, J. C. Doyle and S. H. Low. Congestion control for high performance, stability and fairness in general networks. *IEEE/ACM Transactions on Networking*, 13(1):43-56, February 2005.

[21] R. Shorten, F. Wirth and D. Leith. Modelling TCP in droptail and other environments. *Automatica*, To appear, 2007.

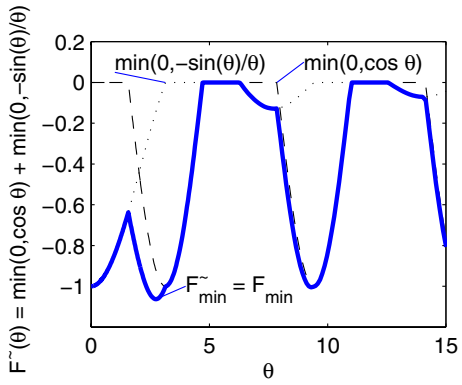


Fig. 11. Plot of $\tilde{F}(\theta)$ along with $\min(0, \cos(\theta))$ and $\min(0, -\sin(\theta)/\theta)$.

[22] R. Srikant. *The Mathematics of Internet Congestion Control*. Birkhauser, 2004.

[23] G. Vinnicombe. On the stability of networks operating TCP-like protocols. In *Proceedings of IFAC*, 2002.

[24] J. Wang, A. Tang, and S. H. Low. Local stability of FAST TCP. In *Proceedings of IEEE Conference on Decision and Control*, Dec. 2004.

[25] J. Wang, D. X. Wei, and S. H. Low. Modeling and stability of FAST TCP. In *IMA Volumes in Mathematics and its Applications*, Volume 143: Wireless Communications. P. Agrawal, M. Andrews, P. J. Fleming, G. Yin, and L. Zhang (eds.), Springer Science, 2006.

[26] Z. Wang and F. Paganini. Global stability with time-delay in network congestion control. In *Proceedings of IEEE Conference on Decision and Control*, December 2002.

[27] D. Wei, C. Jin, S. H. Low, and S. Hegde. FAST TCP: motivation, architecture, algorithms, performance. *IEEE/ACM Transactions on Networking*, To appear in 2007.

[28] L. Ying, G. Dullerud and R. Srikant. Global stability of Internet congestion controllers with heterogeneous delays. In *Proceedings of American Control Conference*, 2004.

[29] Y. Liu, F. L. Presti, V. Misra, D. Towsley, and Y. Gu. Fluid models and solutions for large-scale IP networks. In *Proceedings of the 2003 ACM SIGMETRICS*, 2003.

APPENDIX A
BOUNDS ON GAIN FOR NON-NEGLIGIBLE QUEUES

For the general case of $\alpha_n \geq 0$, we will prove a weaker form of Lemma 1 which is still sufficient to prove Theorem 2.

Lemma 3: Let

$$\tilde{F}(\theta) = \min(0, \cos(\theta)) + \min\left(0, -\frac{\sin(\theta)}{\theta}\right). \tag{33}$$

If $\gamma \leq (\pi/2)^2$ and

$$\sum_{n=1}^{N} \frac{\mu_n}{\tau_n} \tilde{F}(\omega\tau_n) + \frac{1}{\gamma\hat{\tau}} > 0 \tag{34}$$

then $L(j\omega) \in H(\omega)$. Moreover, $\tilde{F}_{\min} := \min_{\theta \geq 0}(\tilde{F}(\theta)) = F_{\min}$.

The function $\tilde{F}(\theta)$ is shown in Figure 11, along with its two constituent terms.

Proof: By (12), a sufficient condition for $L(j\omega) \in H(\omega)$ is

$$\arg\left(1 + \sum_{n=1}^{N} \gamma\mu_n \frac{j\omega + 1/\tau_n}{j\omega + 1/\hat{\tau_n}} \frac{\tau_n - q}{j\omega\tau_n^2 + \gamma q} e^{-j\omega\tau_n}\right) - \arg(j + \omega\hat{\tau}) \in (-\pi, 0). \tag{35}$$

Using (19), this is equivalent to

$$\arg\left(\frac{j\omega + \frac{1}{\hat{\tau}}}{\gamma} + \sum_{n=1}^{N} \frac{\mu_n}{\tau_n} \frac{(j\omega + \frac{1}{\tau_n})(\tau_n - q)}{j\omega\tau_n + \gamma q/\tau_n} e^{-j\omega\tau_n}\right) \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right). \tag{36}$$

Thus it is sufficient that the real part of the left hand side be strictly greater than 0 for all ω:

$$\frac{1}{\gamma\hat{\tau}} + \sum_{n=1}^{N} \frac{\mu_n}{\tau_n} \frac{\tau_n - q}{(\omega\tau_n)^2 + (\gamma q/\tau_n)^2} \operatorname{Re}\left(\left(j\omega + \frac{1}{\tau_n}\right)\left(\frac{\gamma q}{\tau_n} - j\omega\tau_n\right) e^{-j\omega\tau_n}\right) > 0 \tag{37}$$

or equivalently

$$\sum_{n=1}^{N} \frac{\mu_n}{\tau_n}\left(1 - \frac{b_n}{\gamma}\right)\left(\frac{a_n^2 + b_n}{a_n^2 + b_n^2}\cos(a_n) - \frac{a_n^2 - a_n^2 b_n}{a_n^2 + b_n^2}\frac{\sin(a_n)}{a_n}\right) + \frac{1}{\gamma\hat{\tau}} > 0 \tag{38}$$

where $a_n = \omega\tau_n$ and $b_n = \gamma q/\tau_n$.

Because $q/\tau_n \leq 1$, it is sufficient that

$$\sum_{n=1}^{N} \frac{\mu_n}{\tau_n}\left(A_n \min(0, \cos(a_n)) + B_n \min\left(0, -\frac{\sin(a_n)}{a_n}\right)\right) + \frac{1}{\gamma\hat{\tau}} > 0 \tag{39}$$

where $A_n = (1 - b_n/\gamma)(a_n^2 + b_n)/(a_n^2 + b_n^2)$ and $B_n = (1 - b_n/\gamma)(a_n^2 - a_n^2 b_n)/(a_n^2 + b_n^2)$.

Note that $B_n \in [0, 1]$. Also, $A_n \geq 0$ and

$$A_n \leq \frac{a_n^2 + b_n(1 - (a_n^2 + b_n)/\gamma)}{a_n^2 + b_n^2},$$

giving $A_n \leq 1$ if $1 - (a_n^2 + b_n)/\gamma \leq b_n$, which is true if $a_n > \pi/2$, since $\gamma \leq (\pi/2)^2$. Thus $A_n \in [0, 1]$ when $\cos(a_n) \leq 0$. Since each term in (39) is non-positive, it is bounded below by $(\mu_n/\tau_n)\tilde{F}(\omega\tau_n)$ and the first part of the lemma is established.

It remains to show $\tilde{F}_{\min} = F_{\min}$. Now $\tilde{F}_{\min} \leq F_{\min} < -1$. For $\tilde{F}_{\min} < -1$, the minimum must occur when both $\cos(a_n)$ and $\sin(a_n)/a_n$ are negative as both are bounded in magnitude by 1. When that occurs, $F(a_n) = \tilde{F}(a_n)$, giving $\tilde{F}_{\min} \geq F_{\min}$, and hence the result. ■

APPENDIX B
DEGENERATE CONTINUOUS RTT DISTRIBUTION

To see that the right hand side of (29) approaches $-F_{\min}$ as $\beta \to 1$ from above, let $a = k - 1$ and note that

$$\begin{aligned} \frac{\sin(\omega)k - \sin(\omega k)}{\omega k \log(k)} &= \frac{\sin(\omega) + a\sin(\omega) - \sin(\omega + \omega a)}{\omega(1 + a)\log(1 + a)} \\ &= \frac{\omega a[-\sin'(\omega) + O(a)] + a\sin(\omega)}{\omega a + O(a^2)} \\ &\to -\cos(\omega) + \frac{\sin(\omega)}{\omega} = -F(\omega). \end{aligned}$$