Asymptotic Behavior of Heterogeneous TCP Flows and RED Gateway

Peerapol Tinnakornsrisuphap, *Member, IEEE,* and Richard J. La, *Member, IEEE*

***Abstract*—We introduce a stochastic model of a bottleneck ECN/RED gateway under a large number of heterogeneous TCP flows, i.e., flows with diverse round-trip delays and session dynamics. We investigate the asymptotic behavior of the system and show that as the number of flows becomes large, the buffer dynamics and aggregate traffic simplify and can be accurately described by simple stochastic recursions independent of the number of flows, resulting in a scalable model. Based on the Central Limit analysis in the paper, we identify the sources of fluctuations in queue size and describe the relationship between the system parameters such as the marking function and variance of queue size. A closed-form approximation for the mean queue size as a function of system parameters is provided from a simple steady-state analysis. Numerical examples are provided to validate our results.**

***Index Terms*—Congestion control, modeling, stochastic systems.**

I. Introduction

With the growing size and popularity of the Internet as a medium for exchanging information and conducting business, there has been growing interest in modeling and understanding Internet traffic. Accurate modeling of Internet traffic is also important from the perspective of deploying differentiated services since it is likely that best-effort traffic will comprise a significant portion of the Internet traffic in the foreseeable future.

Today's Internet traffic consists of many heterogeneous traffic sources, the majority of which utilize the Transmission Control Protocol (TCP) congestion control mechanism [8]. Some applications, such as File Transfer Protocol (FTP), are relatively long-lived, while others are typically short-lived, e.g., Web browsing. Characterizing and modeling TCP traffic yield an understanding of the interactions between the transport layer (TCP) and the network layer.

To this end, researchers have proposed a number of different models, from detailed single flow models to predict the throughput of a single flow as a function of round-trip delay and packet loss probability [1], [10], [12] to linearized and nonlinear fluid models motivated by a control theoretic viewpoint [6], [15]. As a result, the behavior of a single long-lived TCP flow is relatively well understood. However, despite these efforts, there remain several aspects of TCP that are not well understood, especially in the context of designing good active queue management (AQM) mechanisms that will interact with many TCP flows.

The introduction of AQM mechanisms adds additional complexity to accurate modeling of the interaction of TCP flows with the network layer. The problem is further compounded by the heterogeneous round-trip delays of flows. In addition, much of modeling emphasis in the past was placed on understanding the behavior of long-lived TCP flows [12] and the role of session dynamics (connection arrivals and departures) has been largely ignored in modeling the interaction of TCP flows with the network layer, i.e., drop-tail gateways and AQM mechanisms.

Accurate modeling of individual TCP flows requires modeling of complex dynamics rising from the additive-increase/multiplicative-decrease (AIMD) mechanism of TCP protocol, session dynamics, and heterogeneous round-trip delays in conjunction with the underlying network layer. As the size of state space explodes with the number of sessions, this represents a major obstacle to modeling the interaction of many TCP flows in a realistic setting. For the same reason, even numerical experiments become computationally prohibitive, and fail to provide an insight into the complex dynamics.

The existing literature on TCP traffic modeling usually skirts these major obstacles by relying on ad hoc assumptions, which causes the model to be accurate only in certain regimes. Hollot *et al.* model short-lived TCP flows as exponential pulses, i.e., time-shifted, increasing exponential functions of limited durations, whose interarrival times are exponentially distributed, i.e., Poisson process [7]. The statistics of these exponential pulses can be characterized through a time-reversal of a well-known class of processes called *shot-noise processes*. This model assumes that the short-lived flows last only a few round-trip times and do not experience packet drops or marks, thereby implicitly assuming that congestion level is relatively low. Furthermore, flows are always in either congestion avoidance (long-lived connections) or slow start (short-lived connections), and do not transition from one to the other. In other words, the session dynamics, where connections arrive and leave the network after transfers are completed, are not explicitly modeled. A similar approach to modeling short-lived flows is also taken in [11].

At the other end of the spectrum, Kherani and Kumar [9] suggest that as the capacity of a bottleneck link serving TCP flows with *homogeneous* round-trip times (RTTs) becomes very small, this queue can be accurately described as a processor sharing queue. When the capacity is large, however, this processor sharing model becomes less accurate as newly arrived TCP flows cannot fully utilize their allocated bandwidth. In fact,

Manuscript received September 13, 2004; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor S. Low.

P. Tinnakornsrisuphap is with the Systems Engineering Group, Corporate Research and Development, Qualcomm, Inc., San Diego, CA 92121 USA (e-mail: peerapol@qualcomm.com).

R. J. La is with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742 USA (e-mail: hyongla@isr.umd.edu).

Digital Object Identifier 10.1109/TNET.2005.863453

in the large capacity regime these short-lived flows may terminate even before they can increase their transmission rates to fully utilize their share of bandwidth.

The shortcomings of these models suggest a need for a *unified* model that is accurate in *all* regimes, instead of being restricted to a specific regime. Recently, there has been an increasing interest in *macroscale* modeling of TCP flows [2], [16], [20]. Unlike in microscale models where each individual TCP flow is modeled in detail, macroscale models can be developed by systematically applying limit theorems to derive limiting traffic models. Since the number of connections that share a bottleneck link inside a network is likely to be large, especially in the core network or on a transcontinental link, such a macroscale model promises a potential to provide an accurate and yet scalable model without having to introduce any ad hoc assumptions.

In this paper, we present a stochastic model of a RED gateway [5] serving many TCP flows. While we only consider RED for the AQM mechanism since it is the first and most widely deployed AQM mechanism, many of the conclusions drawn from the analytical results are also applicable to many probabilistic AQMs. The model explicitly incorporates: 1) complete packet-level operations of TCP; 2) a probabilistic packet marking mechanism in RED; 3) heterogeneous RTTs of TCP flows; and 4) session-layer dynamics, i.e., connection arrivals and departures. Using this detailed model, we provide various asymptotics and Central Limit analysis. Our results reveal several interesting behaviors of the network dynamics, which are summarized in Section II. A preliminary steady-state result is also presented, and illustrates how the distribution of file sizes and round-trip delays of TCP flows affect the steady-state marking probability at the RED gateway.

This paper is organized as follows. An overview of the results in the paper is presented in Section II. The model and the dynamics of network, transport, and session layers are described in Section III. The asymptotic results are presented in Section IV, followed by a steady-state analysis in Section V. Section VI gives a comparison with previously proposed models. Central limit theorem complementing the asymptotic results is presented in Section VII. Numerical examples and NS-2 simulation results are provided in Section VIII. We conclude in Section IX.

Some words on the notation in use: Equivalence in law or in distribution between random variables (rvs) is denoted by $=_{st}$. The indicator function of an event A is given by $\mathbf{1}[A]$, and we use $\xrightarrow{P}_n$ (resp. $\Longrightarrow_n$) to denote convergence in probability (resp. weak convergence or convergence in distribution) with n going to infinity. For scalars a and b we write $a \vee b = \max(a,b)$, $a \wedge b = \min(a,b)$, and $[a]^+ = \max\{0,a\}$. We write $X^{(N)}$ to indicate the explicit dependence of the quantity X on the number N of sessions. An expectation of a rv X with a distribution function F is given by either $\mathbf{E}[X]$ or $\mathbf{E}[F]$. For simplicity, we introduce the notation $\mathbf{1}_X[x]$ and $\mathbf{P}_X[x]$ for $\mathbf{1}[X = x]$ and $\mathbf{P}[X = x]$, respectively.

II. Contributions

In this section, we summarize the main results obtained using our detailed stochastic model of TCP flows with a RED gateway. This model is described in detail in Section III.

A. Law of Large Numbers

We prove that the queue size per session and the per-session traffic arrival during a period converge to deterministic processes as the number of sessions increases. We refer to this result as a Law of Large Numbers (LLN).

Here the deterministic processes represent the average or expected behavior of the system. We demonstrate that the recursion of the deterministic process depends only on the capacity of the bottleneck link and the *expected* traffic arrival. Moreover, in a homogeneous RTT setting, this average traffic arrival rate (with a large number of flows) is closely related to the expected arrival rate of a single flow utilizing the same TCP congestion control mechanism. In other words, the deterministic process behaves like a suitably scaled single TCP flow interacting with RED mechanism similar to the deterministic model used in a control theoretic analysis [6]. Therefore, this result provides a justification for the use of a deterministic feedback system model for studying the stability of RED mechanism with a large number of TCP flows. This limiting model is also shown to agree with the previously proposed models in [7], [9] and [11] in their respective regimes, i.e., when the capacity is very large or small.

In addition, our results show that the sessions become asymptotically independent as the number of sessions becomes large, suggesting that the RED gateway does alleviate the synchronization problem among the connections observed with drop-tail gateways.

B. Steady-State Analysis

A simple steady-state analysis is also presented, and a closed-form approximation for the steady-state queue size is derived as a function of the file size and session idle period distributions. We demonstrate that the steady-state queue size can be approximated using only their first-order statistics.

C. Central Limit Theorem

While the deterministic processes from the LLN capture the expected behavior of the system, they cannot capture the random fluctuations caused by the probabilistic nature of RED. Therefore, we complement the LLN result with a Central Limit theorem (CLT), which yields a distribution of the queue deviation from its mean derived from the deterministic process. Therefore, as the number of flows becomes large, the RED queue dynamics can be accurately approximated by a sum of a deterministic process and a stochastic process.

We also provide a formula for a distributional recursion of the queue size. Combined with the LLN, this formula can be used for network provisioning/dimensioning. Further, a closer inspection of the CLT results reveals that the sources of queue size fluctuation can be decomposed into: 1) protocol structure; 2) fluctuation in the feedback information; 3) binary nature of the feedback information; and 4) variation in the RTTs and file sizes. To the best of our knowledge, this is the first result of its kind that reveals the sources of *random* fluctuations due to the probabilistic marking mechanism as opposed to the deterministic fluctuations in the fluid models. These fluctuations

cannot be captured in a model without detailed packet-level operations.

Characterizing the sources of queue fluctuations is important for two reasons. First, although the stability of a RED system with TCP flows can be studied using a control theoretic approach, the detailed performance of TCP flows also depends on many other factors including the delay jitter experienced by the packets, which is mostly determined by the fluctuations in the queue sizes at the bottleneck links. Furthermore, large queue fluctuations could lead to frequent empty queues and, hence, result in wasted resources. Second, identifying the sources of queue fluctuations will enable us to isolate each source and study their relative contributions to the overall fluctuations. This will help network engineers better understand the interaction of the RED mechanism with TCP and use this insight to design a more suitable AQM mechanism.

III. Network Model

For each $N \in \{1,2,3,\cdots\}$, let $\mathcal{N} = \{1,\cdots,N\}$ be the set of sessions that share a bottleneck RED gateway. We assume that time is slotted into contiguous timeslots. Here the RTTs of TCP connections are approximated as integer multiples of timeslots, where a timeslot is a common divisor of the RTTs of TCP flows. For our analysis, we model three layers of dynamics—network, transport, and session layers—which interact with each other through mechanisms that will be specified shortly. At the lowest level, the network is simplified to be a single bottleneck router with an ECN/RED marking mechanism controlling the congestion level. The traffic injected into the network by a connection is controlled by a TCP congestion control mechanism at the transport layer, which reacts to the marks from the network. A TCP connection is initiated by a session when a file transfer request arrives. A session can be either active or idle. When a session is active, a file or an object is being transferred by a TCP connection. A busy period of a session is defined to be the period from the time when the session receives a file to transfer till the time at which the TCP connection is torn down after completion of file transfer and the session goes idle. The duration of an idle period is random and represents the idle time between consecutive file transfers. When a new file/object to be transferred arrives, the session becomes active again and sets up a new TCP connection. We now give detailed descriptions of the model for each layer and the interaction of these three layers.

A. Heterogeneous Round-Trip Times

As mentioned earlier, we approximate the RTTs of TCP connections as integer multiples of timeslots, and any congestion control actions by TCP flows, i.e., window size updates, occur at the end of the round trip. The RTT of an active flow i at time t is denoted by $D_i^{(N)}(t) \in \mathcal{D} := \{2,\cdots,D_{\max}\}$ for some integer $D_{\max} \geq 2$.[1] We use $\beta_i^{(N)}(t+1)$ to denote the number of timeslots that have elapsed since the last action by an *active* flow i. Then, $\beta_i^{(N)}(t)$ evolves according to

$$\beta_i^{(N)}(t+1) = \left(1+\beta_i^{(N)}(t)\mathbf{1}\left[\beta_i^{(N)}(t) < D_i^{(N)}(t)\right]\right) \times \mathbf{1}\left[X_i^{(N)}(t) > 0\right] \quad (1)$$

where $X_i^{(N)}(t)$ is the remaining workload (in packets) of session i at the beginning of timeslot $[t,t+1)$. The rv $X_i^{(N)}(t)$ is greater than zero only if session i is active in timeslot $[t,t+1)$, and, hence, the last indicator function is one only if the session is active. This will be explained further in the next subsection.

Given $i \in \mathcal{N}$, $a, b \in \mathbb{R}$ and $t = 0,1,\ldots$, we define

$$G_{i,t+1}(a,b) := a \cdot \mathbf{1}\left[\beta_i^{(N)}(t+1) < D_i^{(N)}(t+1)\right] + b \cdot \mathbf{1}\left[\beta_i^{(N)}(t+1) \geq D_i^{(N)}(t+1)\right] \quad (2)$$

in order to simplify our notation later.

Given collections of $\mathbb{R}$-valued rvs $\{Y_i(t), t = 0,1,\ldots\}$, and $\{Y_{i,new}(t), t = 0,1,\ldots\}$, we see that

$$\begin{aligned} Y_i(t+1) &= G_{i,t+1}\left(Y_i(t), Y_{i,new}(t+1)\right) \\ &= \begin{cases} Y_{i,new}(t+1), & \beta_i^{(N)}(t+1) \geq D_i^{(N)}(t+1) \\ Y_i(t), & \text{otherwise.} \end{cases} \end{aligned} \quad (3)$$

In other words, the value of $Y_i(t+1)$ is updated to $Y_{i,new}(t+1)$ only at the end of the round trip. Otherwise, the value of $Y_i(t+1)$ remains to be $Y_i(t)$ since no action will be taken before the end of the round trip.

B. Session Dynamics

Each session i in $\mathcal{N}$ is either active or idle. An idle session at the beginning of timeslot $[t,t+1)$ has no packet to transmit in that timeslot. An idle session in timeslot $[t,t+1)$ becomes active at the beginning of timeslot $[t+1,t+2)$ with probability P_{ar}, where $P_{ar} \in (0,1)$, independently of past events. In other words, the duration of an idle period is *geometrically* distributed with parameter P_{ar} (hence, with mean $1/P_{ar}$). This attempts to capture the dynamics of connection arrivals, where the interarrival times are reported to be exponentially distributed [14].[2] Let $\{U_i(t), i \in \mathcal{N}; t = 0,1,\cdots\}$ be a collection of independent and identically distributed (i.i.d.) rvs uniformly distributed on [0, 1], and let $\mathbf{1}[U_i(t+1) \leq P_{ar}]$ be the indicator function of the event that a new file/object arrives in the timeslot $[t+1,t+2)$ for an idle session i.

Let $\{F_i(t), i \in \mathcal{N}; t = 0,1,\cdots\}$ be a collection of i.i.d. positive, integer-valued rvs distributed according to a general probability mass function (pmf) $F : \{1,2,\ldots\} \to [0,1]$. The workload of a new connection initiated by session i that becomes active at the beginning of timeslot $[t,t+1)$ is given by $F_i(t)$. This workload represents the *total* number of TCP segments[3] the connection will have to transmit before it is torn down. Thus, if

[1] Although $\mathcal{D}$ does not include one in our model, it can be included at the price of more cumbersome proofs. Moreover, this does not cause any loss of generality of the model.

[2] Recall that an exponential rv X with parameter α can be approximated by $\lceil X \rceil$, which is a geometric rv with parameter $p = 1 - e^{-\alpha}$.

[3] In this model, each TCP segment is transmitted as a separate packet.

a given TCP connection is used to transfer more than one object, this workload variable $F_i(t)$ represents the total number of TCP segments brought in by all these objects. The duration of a connection is defined to be the amount of time that elapses from the moment a new file request for the connection arrives, which initiates the three-way handshake in our model, until the time the connection is torn down at the end of timeslot in which the last packet of the file is transmitted.

Recall that $X_i^{(N)}(t)$ denotes the remaining workload (expressed in packets) of connection i at the beginning of timeslot $[t, t+1)$.[4] If session i is idle in timeslot $[t, t+1)$, then $X_i^{(N)}(t) = 0$. The session dynamics described in this section is summarized through the following recursion:

$$X_i^{(N)}(t+1) = \mathbf{1}\left[X_i^{(N)}(t) > 0\right]\left(X_i^{(N)}(t) - A_i^{(N)}(t)\right) + \mathbf{1}\left[X_i^{(N)}(t) = 0\right]\mathbf{1}\left[U_i(t+1) \le P_{ar}\right]F_i(t+1) \quad (4)$$

where $A_i^{(N)}(t)$ denotes the number of packets injected into the network by connection i at the beginning of timeslot $[t, t+1)$. This is regulated by TCP and will be explained in the next subsection.

When a new connection is set up by a session, its RTT is randomly selected. Under this assumption, the RTT of session i in timeslot $[t+1, t+2)$ is given by

$$D_i^{(n)}(t+1) = D_i^{(N)}(t)\mathbf{1}\left[X_i^{(N)}(t) > 0\right] + \mathbf{1}\left[X_i^{(N)}(t) = 0\right]\mathbf{1}\left[U_i(t+1) < P_{ar}\right]D_{i,new}(t+1) \quad (5)$$

where the $\mathcal{D}$-valued rvs $\{D_{i,new}(t+1), t = 1, 2, \ldots\}$ are i.i.d. rvs that determine the RTT of newly arrived connections. The bound on the maximum RTT does not constrain the model because actual TCP flows also cannot have RTTs larger than the maximum timeout value.

C. TCP Dynamics

For each $i \in \mathcal{N}$, let $W_i^{(N)}(t)$ be an integer-valued rv that encodes the congestion window size (in packets) at the beginning of timeslot $[t, t+1)$. We assume that the rv $W_i^{(N)}(t)$ has a range $\{0, 1, \cdots, W_{\max}\}$ where $W_{\max} \ge 2$ is a finite integer representing the receiver advertised window size of the TCP connection. The congestion window size of an idle session is assumed to be zero. When an idle session i becomes active at the beginning of timeslot $[t+1, t+2)$, the congestion window size of the new TCP connection is set to one at the beginning of timeslot $[t + D_i^{(N)}(t+1) + 1, t + D_i^{(N)}(t+1) + 2)$, where $D_i^{(N)}(t+1)$ is the RTT of the newly established connection determined by $D_{i,new}(t+1)$ in (5), and the TCP sender transmits one packet. This models one round-trip delay for the three-way handshake of TCP. We now describe how the congestion window sizes of active connections evolve.

Each TCP sender transmits as many of the remaining data packets as allowed by its congestion window only at the end of the round trip. We simplify the packet transmission of a connection during a round trip and assume that all packets from the connection arrive in a single timeslot, rather than being spread out throughout the round trip. Such simplification can be justified by the following reasons:

i) In the Internet, most of the packet arrivals at a bottleneck are usually compressed together due to the "ACK compression" phenomenon [22], which leads to bursty arrivals at the bottlenecks. Hence, modeling the packet arrivals over an RTT as a batch arrival in a single timeslot tends to be more accurate than modeling them as smooth arrivals throughout an RTT.

ii) Aggregating a round-trip worth of packet arrivals into a single timeslot will result in burstier traffic from each connection. This will cause queue dynamics to fluctuate more than having a smooth arrival pattern. Therefore, the queue fluctuation in this model will provide an upper bound on the actual queue fluctuations with smoother packet arrival patterns.

iii) The information used for control action at the RED gateways is the *average* queue size. Since the time constant of the exponential averaging mechanism of RED is expected to be larger than the RTTs of TCP connections for the network stability [15], the difference in the average queue size due to our bursty packet arrivals will be smoothed out. As a result, the control behavior in both cases should not differ significantly.

The number of packets transmitted by connection i at the beginning of timeslot $[t, t+1)$, denoted by $A_i^{(N)}(t)$, is given by

$$A_i^{(N)}(t) = \min\left(W_i^{(N)}(t), X_i^{(N)}(t)\right) \times \mathbf{1}\left[\beta_i^{(N)}(t) \ge D_i^{(N)}(t)\right] \quad (6)$$

where the indicator function is one only at the end of a round trip, and, hence, a connection transmits once per RTT.

A TCP connection operates in one of two different modes, namely *slow start* (SS) and *congestion avoidance* (CA). A new TCP connection starts in SS. While in SS, the congestion window size doubles every RTT until one or more packets are marked. If a mark is received, then the congestion window size is halved and TCP switches to CA. The congestion window size is limited by the receiver advertised window size $W_{\max}$. Hence, a connection in SS that is due to update its congestion window size in timeslot $[t+1, t+2)$ will update the congestion window size according to

$$W_{i,SS}^{(N)}(t+1) = \min\left(2W_i^{(N)}(t) \vee 1, W_{\max}\right) M_i^{(N)}(t+1) + \left\lceil \frac{W_i^{(N)}(t)}{2} \right\rceil \left(1 - M_i^{(N)}(t+1)\right) \quad (7)$$

where $M_i^{(N)}(t+1)$ is an indicator function of the event that no marks have been received in the round trip preceding the timeslot $[t, t+1)$, i.e., $M_i^{(N)}(t+1) = 0$ if at least one packet is marked in the previous round trip and $M_i^{(N)}(t+1) = 1$ otherwise. The marking mechanism and evolution of $M_i^{(N)}(t)$ will be explained in more detail in Section III-D.

In CA, the congestion window size in the next round trip is increased by one packet if no marks are received during the

[4] We refer to a TCP connection of an active session i by connection i when there is no confusion.

current round trip, and if one or more packets are marked the congestion window is reduced by half. When a connection in CA is due to update its congestion window size in timeslot $[t+1, t+2)$, its congestion window size will be updated to

$$W_{i,CA}^{(N)}(t+1) = \min\left(W_i^{(N)}(t)+1, W_{\max}\right) M_i^{(N)}(t+1) + \left\lceil \frac{W_i^{(N)}(t)}{2} \right\rceil \left(1 - M_i^{(N)}(t+1)\right). \quad (8)$$

Since we only update the congestion window size at the end of each round trip, we use the short representation defined in (2) to retain the value of the congestion window size until the end of a round trip at which time it is updated. If connection i is in SS in timeslot $[t, t+1)$, its congestion window size $\hat{W}_{i,SS}^{(N)}(t+1)$ in timeslot $[t+1, t+2)$ will be given by

$$\hat{W}_{i,SS}^{(N)}(t+1) = G_{i,t+1}\left[W_i^{(N)}(t), W_{i,SS}^{(N)}(t+1)\right] \quad (9)$$

where $W_{i,SS}^{(N)}(t+1)$ is given in (7), if the connection remains active in timeslot $[t+1, t+2)$. Similarly, if connection i is in CA in timeslot $[t, t+1)$, its congestion window $\hat{W}_{i,CA}^{(N)}(t+1)$ in timeslot $[t+1, t+2)$ will be

$$\hat{W}_{i,CA}^{(N)}(t+1) = G_{i,t+1}\left[W_i^{(N)}(t), W_{i,CA}^{(N)}(t+1)\right] \quad (10)$$

where $W_{i,CA}^{(N)}(t+1)$ is given in (8).

Let the $\{0, 1\}$-valued rvs $\{S_i^{(N)}(t), i \in \mathcal{N}\}$ encode the state of TCP connections, with the interpretation that $S_i^{(N)}(t) = 0$ (resp. $S_i^{(N)}(t) = 1$) if connection i is in CA (resp. in SS) at the beginning of timeslot $[t, t+1)$. Combining (9) and (10), we see that the complete recursion of the congestion window size described in this section can be written as

$$W_i^{(N)}(t+1) = \mathbf{1}\left[X_i^{(N)}(t) - A_i^{(N)}(t) > 0\right] \times \left(S_i^{(N)}(t)\hat{W}_{i,SS}^{(N)}(t+1) + \left(1 - S_i^{(N)}(t)\right)\hat{W}_{i,CA}^{(N)}(t+1)\right). \quad (11)$$

The first indicator function in (11) resets the congestion window size to zero when session i runs out of data to transmit and returns to an idle state.

Finally, the evolution of $\{S_i^{(N)}(t), t = 0, 1, \ldots\}$ is given by

$$S_i^{(N)}(t+1) = \mathbf{1}\left[X_i^{(N)}(t) \le W_i^{(N)}(t)\right] + \mathbf{1}\left[X_i^{(N)}(t) > W_i^{(N)}(t)\right] S_i^{(N)}(t) M_i^{(N)}(t+1). \quad (12)$$

This equation can be interpreted as follows. Connection i is in SS in timeslot $[t+1, t+2)$ if either: 1) there is no packet left to transmit (so the connection resets) at the beginning of the next timeslot or 2) the connection was active and in SS in timeslot $[t, t+1)$ and received no mark in the timeslot. Equation (12) assumes that a new TCP connection in SS is ready to be set up one timeslot after the previous connection is torn down upon finishing its workload. It also implicitly assumes that the SS/CA state is updated in the next timeslot following transmission. However, the window size is updated one RTT after transmission according to (11) using the appropriate SS/CA state as in the correct operation of TCP.

D. Network Dynamics

In this subsection, we explain how packets are marked to provide the congestion notification to the active TCP connections. The capacity of the bottleneck link is $N \cdot C$ packets/timeslot for some positive constant C. The buffer size is assumed to be infinite so that no packets are dropped due to buffer overflow. Thus, congestion control is achieved solely through the random marking algorithm of the RED gateway.

Let $Q^{(N)}(t)$ denote the number of packets queued in the buffer at the beginning of timeslot $[t, t+1)$. Connection i injects $A_i^{(N)}(t)$ packets into the network, and they are put in the buffer at the beginning of timeslot $[t, t+1)$. Let the rv

$$A^{(N)}(t) := \sum_{i=1}^{N} A_i^{(N)}(t) \quad (13)$$

denote the aggregate number of packets offered to the network by all N sessions at the beginning of timeslot $[t, t+1)$. Hence, $Q^{(N)}(t) + A^{(N)}(t)$ packets are available for transmission during that timeslot. Since the bottleneck link has a capacity of NC packets/timeslot, $\left[Q^{(N)}(t) + A^{(N)}(t) - NC\right]^+$ packets will not be served during timeslot $[t, t+1)$ and will remain in the buffer, and their transmission is deferred to subsequent timeslots. The number of packets in the buffer at the beginning of timeslot $[t+1, t+2)$, $Q^{(N)}(t+1)$, is therefore given by

$$\begin{aligned} Q^{(N)}(t+1) &= \left[Q^{(N)}(t) - NC + A^{(N)}(t)\right]^+ \\ &= \left[Q^{(N)}(t) - NC + \sum_{i=1}^{N} \min\left(W_i^{(N)}(t), X_i^{(N)}(t)\right) \times \mathbf{1}\left[\beta_i^{(N)}(t) \ge D_i^{(N)}(t)\right]\right]^+ \end{aligned} \quad (14)$$

where the last equality makes use of (13) and (6).

The average queue size $\hat{Q}^{(N)}(t)$ is computed utilizing an exponentially weighted moving average (EWMA) filter with parameter $0 < \alpha \le 1$. This parameter α determines the time constant of the averaging mechanism. The average queue size $\hat{Q}^{(N)}(t+1)$ at the beginning of timeslot $[t+1, t+2)$ is given by

$$\hat{Q}^{(N)}(t+1) = (1-\alpha)\hat{Q}^{(N)}(t) + \alpha Q^{(N)}(t+1).$$

Each incoming packet into the router in timeslot $[t, t+1)$ is marked with a probability $f^{(N)}(\hat{Q}^{(N)}(t))$, which is a function of the average queue length at the beginning of the timeslot $[t, t+1)$. We represent this event using $\{0, 1\}$-valued rvs $M_{i,j}^{(N)}(t+1)$, $j = 1, \ldots, A_i^{(N)}(t)$, with the interpretation that $M_{i,j}^{(N)}(t+1) = 0$ (resp. $M_{i,j}^{(N)}(t+1) = 1$) if the jth packet from connection i is marked (resp. not marked) in the RED buffer. To do so we introduce a collection of i.i.d. [0, 1]-*uniform* rvs $\{V_i(t+1), V_{i,j}(t+1), i, j = 1, \cdots; t = 0, 1, \cdots\}$ that are assumed to be independent of other rvs. The process by which packets are marked is

as follows. For each $i \in \mathcal{N}$ and $j = 1, 2, \ldots$, we define the marking rvs

$$M_{i,j}^{(N)}(t+1) = \mathbf{1}\left[V_{i,j}(t+1) > f^{(N)}\left(\hat{Q}^{(N)}(t)\right)\right].$$

The indicator function of the event that no packets from connection i in timeslot $[t, t+1)$ are marked can then be written as

$$M_{i,new}^{(N)}(t+1) = \begin{cases} \prod_{j=1}^{A_i^{(N)}(t)} M_{i,j}^{(N)}(t+1), & A_i^{(N)}(t) \geq 1 \\ 1, & A_i^{(N)}(t) = 0. \end{cases}$$

While this information is available in the next timeslot, it is used one RTT later to update the congestion window size according to (11). Using the short representation in (2), the rv $M_i^{(N)}(t+1)$ evolves according to

$$M_i^{(N)}(t+1) = G_{i,t}\left(M_i^{(N)}(t), M_{i,new}^{(N)}(t+1)\right).$$

Notice that $t+1$ in the subscript of $G_{i,t+1}(\cdot)$ in (2) is replaced by t to delay the update of rv $M_i^{(N)}(t+1)$ so that the congestion window size (given by (9) and (10)) evolves based on the markings in the previous round trip. For example, if $W_i^{(N)}(t)$ is updated in timeslot $[t, t+1)$ then $M_i^{(N)}(t+1)$ is updated in timeslot $[t+1, t+2)$ and its value will be used in the timeslot $[t+D_i^{(N)}(t), t+D_i^{(N)}(t)+1)$ to determine the new congestion window size.

IV. The Asymptotics

The first main result of this paper consists of the asymptotics for the normalized queue size as the number of sessions becomes large. For convenience, we denote the vector of state variables for session i in timeslot $[t, t+1)$ by

$$\mathbf{Y}_i^{(N)}(t) := \left(W_i^{(N)}(t), X_i^{(N)}(t), S_i^{(N)}(t), D_i^{(N)}(t), \beta_i^{(N)}(t), M_i^{(N)}(t)\right).$$

The rv $\mathbf{Y}_i^{(N)}(t)$ takes a value in the discrete set

$$\mathcal{Y} := \{0, 1, \ldots, W_{\max}\} \times \{0, 1, \ldots\} \times \{0, 1\} \times \{0, 2, 3, \ldots, D_{\max}\} \times \{0, 1, \ldots, D_{\max}\} \times \{0, 1\}.$$

This result is discussed under the following Assumptions (A1) and (A2):

(A1) There exists a continuous function $f : \mathbb{R}_+ \to [0, 1]$ such that for each $N = 1, 2, \ldots$,

$$f^{(N)}(x) = f(N^{-1}x), \qquad x \geq 0.$$

(A2) For each $N = 1, 2, \ldots$, and $i = 1, \ldots, N$, the initial conditions of rvs in (1), (4), (5), (11), (12), and (14) are given by

$$Q^{(N)}(0) = W_i^{(N)}(0) = X_i^{(N)}(0) = \beta_i^{(N)}(0) = D_i^{(N)}(0) = 0$$

and

$$S_i^{(N)}(0) = M_i^{(N)}(0) = 1.$$

Assumption (A1) is a structural condition. Since we are interested in the dynamics when there exists N flows in the system, f is just a surrogate function representing the average contribution that each flow has on the marking probability. On the other hand, (A2) is made essentially for technical convenience as it implies that for each $N = 1, 2, \ldots$, and $t = 0, 1, \ldots$, the rvs $\mathbf{Y}_1^{(N)}(t), \ldots, \mathbf{Y}_N^{(N)}(t)$ are *exchangeable*. Assumption (A2) can be omitted but at the expense of a more cumbersome discussion.

Theorem 1: Assume that (A1) and (A2) hold. Then, for each $N = 1, 2, \ldots$, and $t = 0, 1, \ldots$, there exist (nonrandom) constants $q(t)$ and $\hat{q}(t)$ and a $\mathcal{Y}$-valued random vector

$$\mathbf{Y}(t) = (W(t), X(t), S(t), D(t), \beta(t), M(t))$$

such that the following holds:

(i) *The following convergences take place:*

$$\frac{Q^{(N)}(t)}{N} \xrightarrow{P}_N q(t), \quad \frac{\hat{Q}^{(N)}(t)}{N} \xrightarrow{P}_N \hat{q}(t)$$

and

$$\mathbf{Y}_1^{(N)}(t) \Longrightarrow_N \mathbf{Y}(t).$$

(ii) *For any bounded function $g : \mathbb{N}_+^6 \to \mathbb{R}$, we have*

$$\frac{1}{N}\sum_{i=1}^{N} g\left(\mathbf{Y}_i^{(N)}(t)\right) \xrightarrow{P}_N \mathbf{E}\left[g\left(\mathbf{Y}(t)\right)\right]. \tag{15}$$

(iii) *For any integer $I = 1, 2, \ldots$, the random vectors $\{\mathbf{Y}_i^{(N)}(t), i = 1, \ldots, I\}$ become asymptotically independent as N becomes large, with*

$$\lim_{N\to\infty} \mathbf{P}\left[\mathbf{Y}_i^{(N)}(t) = \mathbf{y}_i, i = 1, \cdots, I\right] = \prod_{i=1}^{I} \mathbf{P}\left[\mathbf{Y}(t) = \mathbf{y}_i\right] \tag{16}$$

for any $\mathbf{y}_i \in \mathcal{Y}$, $i = 1, \ldots, I$.

In addition, with initial conditions $q(0) = W(0) = X(0) = D(0) = \beta(0) = 0$ and $S(0) = M(0) = 1$, we have the following recursion:

$$q(t+1) = \left(q(t) - C + \mathbf{E}\left[A(t)\right]\right)^+ \tag{17}$$

and

$$\hat{q}(t+1) = (1-\alpha)\hat{q}(t) + \alpha q(t+1) \tag{18}$$

where

$$A(t) = \min\left(W(t), X(t)\right)\mathbf{1}\left[\beta(t) \geq D(t)\right].$$

The distribution of $\mathbf{Y}(t)$, $t = 0, 1, \ldots$, can be calculated recursively starting with $t = 0$.

A proof of Theorem 1 and the complete distributional recursion of $\mathbf{Y}(t)$ are provided in [17].

From Claim (ii) in Theorem 1, it is straightforward to see that

$$\frac{1}{N}\sum_{i=1}^{N} A_i^{(N)}(t) \xrightarrow{P}_N \mathbf{E}\left[A(t)\right] = \mathbf{E}\left[\min\left(W(t), X(t)\right)\mathbf{1}\left[\beta(t) \geq D(t)\right]\right] \tag{19}$$

as the congestion window size is always bounded by $W_{\max}$.

A. Discussion

Theorem 1 tells us that, for large N, the queue size at time t, $Q^{(N)}(t)$, can be approximated by $Nq(t)$ with $q(t)$ determined via a simple deterministic recursion that is independent of the number of sessions. The offered traffic into the network during the timeslot, $A^{(N)}(t)$, can also be approximated by $N \cdot \mathbf{E}[A(t)]$. These approximations become more accurate as the number of sessions N becomes large, and the computational complexity does not depend on N. The limiting model is therefore "scalable" as it does not suffer from the explosion of state space, nor does it require any ad hoc assumptions.

Claim (iii) in Theorem 1 also suggests that the dependency among the sessions becomes negligible with a large number of sessions, and, hence, RED breaks the global synchronization when the number of sessions is large, which is one of the design goals of RED [5].

The Weak Law of Large Numbers also provides justification for the analysis of the TCP/RED interaction through a deterministic feedback system when there exists a large number of flows. More specifically, the recursion of the asymptotic (or average) queue, i.e., $q(t)$, depends only on the capacity of the queue and the *expected* amount of traffic injected into the network in each timeslot, i.e., $\mathbf{E}[A(t)]$. This is similar to many of the existing models which analyze the dynamics of TCP and AQM mechanisms as deterministic feedback systems with delay(s). If we simplify the model to consider only persistent TCP flows (as in [20]), the recursion for the limiting number of packets injected into the network is also closely related to the recursion of a single traffic flow, i.e., at any time they both have the same conditional expectation given the same state in the previous timeslot and marking/dropping probability. This resembles the deterministic model used for stability analysis of the RED mechanism with TCP flows [6].

Although the sequence $\{(q(t), \hat{q}(t), \mathbf{Y}(t)), t = 0, 1, \ldots\}$ is a time-homogeneous Markov chain, we do not address the convergence of the Markov chain to steady-state as $t \to \infty$, for complications arise due the fact that the first two components are degenerate (i.e., deterministic). However, if the system is configured so that the limiting queue is asymptotically stable in the sense that $q(t) \longrightarrow_t q^\star$ for some $q^\star > 0$, then clearly $\hat{q}(t) \longrightarrow_t q^\star$, and consequently, the marking probability converges to that corresponding to $q^\star$. Once the marking probability converges, it is easy to show that $\mathbf{Y}(t)$ can be modeled as an aperiodic, irreducible, and positively recurrent Markov chain, and, hence, $\mathbf{Y}(t) \Longrightarrow_t \mathbf{Y}^\star$ for some $\mathcal{Y}$-valued rv $\mathbf{Y}^\star$.

V. Steady-State Regime

Using the results from the previous section, we carry out a simple steady-state analysis. In order to facilitate our further analysis, we introduce the following assumptions.

(A3) The sequence $\{(q(t), \hat{q}(t), \mathbf{Y}(t)), t = 0, 1, \ldots\}$ admits a steady-state in the sense that $(q(t), \hat{q}(t), \mathbf{Y}(t)) \Longrightarrow_t (q^\star, q^\star, \mathbf{Y}^\star)$ for some triple $(q^\star, q^\star, \mathbf{Y}^\star)$ where $q^\star$ is a constant and $\mathbf{Y}^\star = (W^\star, X^\star, S^\star, D^\star, \beta^\star, M^\star)$ is a $\mathcal{Y}$-valued rv.

(A4) We assume that $\mathbf{E}[F] \gg \mathbf{E}[W^\star]$, where F is the workload distribution.

(A5) We assume that when an active connection finishes its last packet transmission, it waits an additional RTT before resetting its window size to zero.[5]

It is easily seen that Assumption (A3) immediately implies that the steady-state marking probability is $f(q^\star)$. We want to find the steady-state queue level $q^\star$ as a fixed-point solution to

$$\begin{aligned} q^\star &= (q^\star - C + \mathbf{E}[A^\star])^+ \\ &= (q^\star - C + \mathbf{E}[\min(W^\star, X^\star)\mathbf{1}[\beta^\star \geq d^\star]])^+ . \end{aligned}$$

Since the window size and the workload are both zero when a session is idle, we have

$$\begin{aligned} \mathbf{E}[A^\star] &= \mathbf{P}[\text{active}]\mathbf{E}[A^\star|\text{active}] \\ &= \mathbf{P}[\text{active}]\mathbf{E}[\min(W^\star, X^\star)|\text{active}] \\ &\quad \times \mathbf{P}[\beta^\star \geq D^\star|\text{active}] \end{aligned} \tag{20}$$

where the last equality follows from the following result. Its proof is given in [17].

Lemma 1: Assuming (A1)–(A5), the rvs $\min(W^\star, X^\star)$ and $\mathbf{1}[\beta^\star \geq D^\star]$ are conditionally independent given that the connection is active.

The probability $\mathbf{P}[\text{active}]$ that a session is active in steady-state is given by

$$\mathbf{P}[\text{active}] = \frac{\mathbf{E}[\text{connection duration}]}{\mathbf{E}[\text{connection duration}] + \mathbf{E}[\text{idle period}]}$$

by elementary arguments from renewal theory.

First, we can rewrite

$$\begin{aligned} &\mathbf{P}[\beta^\star \geq D^\star|\text{active}] \\ &\quad = \sum_{d_i} \mathbf{P}[\beta^\star \geq d_i|\text{active}, D^\star = d_i]\mathbf{P}[D^\star = d_i|\text{active}]. \end{aligned}$$

Conditioning on the event that $D^\star = d_i$, it is easy to see that

$$\mathbf{P}[D^\star = d_i|\text{active}] = \frac{d_i \mathbf{P}[D = d_i]}{\sum_{d_j \in \mathcal{D}} d_j \mathbf{P}[D = d_j]}.$$

Our original model outlined in Section III does not assume (A5), and, hence, the above equality does not hold. However, this discrepancy will cause only a marginal difference under (A4).

Assumption (A4) suggests that a connection typically lasts many RTTs. Consequently, we have

$$\mathbf{P}[\beta^\star \geq d_i|\text{active}, D^\star = d_i] \approx \frac{1}{d_i}.$$

Therefore,

$$\begin{aligned} \mathbf{P}[\beta^\star \geq D^\star|\text{active}] &\approx \sum_{d_i \in \mathcal{D}} \frac{\mathbf{P}[D = d_i]}{\sum_{d_j \in \mathcal{D}} d_j \mathbf{P}[D = d_j]} \\ &= \frac{1}{\mathbf{E}[D]}. \end{aligned}$$

[5]Although this assumption alters our model slightly, the results presented in this paper can be proved essentially in the same way. In addition, it will have only a marginal effect under Assumption (A4) that $\mathbf{E}[F] \gg \mathbf{E}[W^\star]$. Here, we assume that the connection is active until the end of the last round trip, and the definition of the duration of a connection is appropriately modified to include the last round-trip time from the definition in Section III-B.

Let F_{ar} be the initial workload with distribution F and D be the RTT of a connection. The conditional expected value of a connection duration given its initial workload size and round-trip delay depends on the detailed dynamics of rv $\mathbf{Y}(t)$, and is difficult to model accurately. Instead, we approximate it as

$$\mathbf{E}[\text{connection duration}|F_{ar}=x, D=d] = \frac{x}{T(d, f(q^\star))}$$

where $T(d, f(q^\star))$ is the mean throughput of a TCP connection (in packets/timeslot) with RTT of d and packet marking probability of $f(q^\star)$. Since the *average* number of round trips required to complete a transfer is large under Assumption (A4), this is a reasonable assumption. We validate this assumption using numerical studies in [18]. Here we approximate the average throughput of a TCP connection by the well-known throughput formula

$$T(d, f(q^\star)) \approx \frac{K}{d\sqrt{f(q^\star)}}$$

where K is some constant in the interval [1, 8/3] (see [20] for analysis confirming this throughput formula for the model with persistent flows).

This implies that

$$\mathbf{E}[\text{connection duration}|F_{ar}=x, D=d] \approx \frac{xd\sqrt{f(q^\star)}}{K}.$$

Since the initial workload and the RTT are independent, we have $\mathbf{E}[\text{connection duration}] \approx (\mathbf{E}[F_{ar}]\mathbf{E}[D]\sqrt{f(q^\star)}/K)$. Therefore,

$$\mathbf{P}[\text{active}] \approx \frac{\mathbf{E}[F_{ar}]\mathbf{E}[D]\sqrt{f(q^\star)}}{\mathbf{E}[F_{ar}]\mathbf{E}[D]\sqrt{f(q^\star)} + K/P_{ar}}$$

Finally, in order to compute (20), we need to calculate $\mathbf{E}[\min(W^\star, X^\star)|\text{active}]$, which can be approximated by $\mathbf{E}[W^\star|\text{active}]$ under Assumption (A4), i.e., $\mathbf{E}[F] \gg \mathbf{E}[W^\star]$:

$$\begin{aligned}
&\mathbf{E}[W^\star|\text{active}] \\
&\quad = \sum_{d_i \in \mathcal{D}} \mathbf{E}[W^\star|D^\star = d_i, \text{active}]\mathbf{P}[D^\star = d_i|\text{active}] \\
&\quad = \sum_{d_i \in \mathcal{D}} \frac{K}{\sqrt{f(q^\star)}} \frac{d_i \mathbf{P}[D=d_i]}{\sum_{d_j \in \mathcal{D}} d_j \mathbf{P}[D=d_j]} \\
&\quad = \frac{K}{\sqrt{f(q^\star)}} \qquad (21)
\end{aligned}$$

where the second equality follows from

$$\mathbf{E}\left[\frac{W^\star}{D^\star}|D^\star = d_i, \text{active}\right] = T(d_i, f(q^\star)).$$

Combining (20) and (21), we get

$$\mathbf{E}[A^\star] \approx \frac{K\mathbf{E}[F_{ar}]}{\mathbf{E}[F_{ar}]\mathbf{E}[D]\sqrt{f(q^\star)} + K/P_{ar}}.$$

If $0 < f(q^\star) < 1$, it is necessary that $C = \mathbf{E}[A^\star]$. After some simple algebra, we can show

$$f(q^\star) \approx \frac{K^2}{\mathbf{E}[D]^2}\left(\frac{1}{C} - \frac{1}{P_{ar}\mathbf{E}[F_{ar}]}\right)^2.$$

If f is invertible, then

$$q^\star \approx f^{-1}\left(\frac{K^2}{\mathbf{E}[D]^2}\left(\frac{1}{C} - \frac{1}{P_{ar}\mathbf{E}[F_{ar}]}\right)^2\right). \qquad (22)$$

Note that we can approximate the steady-state marking probability and the average queue size using only the mean values of the round-trip delay and incoming workload distributions. Numerical examples validating the analysis are given in [18]. This simple formula can be used as a guideline on how to design the feedback probability function to control the queue size at the steady-state, given the system parameters.

VI. Comparison to Other Models

We briefly consider the resulting model from Theorem 1 when C is either very large or very small under the following assumption.

(A1c) The marking function $f : \mathbb{R} \to [0,1]$ in Assumption (A1) is monotonically increasing with $f(0) = 0$ and $\lim_{x\to\infty} f(x) = 1$.

First, suppose that $C \to \infty$. In this case, it is easy to see that $\lim_{C\to\infty} q(t) = 0$ for all $t = 0, 1, \ldots$, and, hence, the marking probability per flow also converges to zero from (A1c) for all t. Therefore, each incoming flow will always operate in SS, and the resulting input traffic into the network can be approximated by the superposition of flows that arrive according to a Poisson process, each of which brings a random amount of workload and doubles its window size every round trip. Thus, the aggregate input traffic is similar to the time-reversed shot-noise processes, in agreement with [7] and [11].

On the other hand, if $C \simeq 0$, the queue will start building up, whence $\lim_{t\to\infty} q(t) = \infty$. Thus, for large t, all TCP flows (including incoming TCP flows) will experience marking probability close to one from Assumption (A1c). This implies that the congestion window size converges to one and each connection can transmit only one packet per round trip. Since the bottleneck router will transmit packets nonselectively, any active flow will receive roughly equal throughput and, hence, the queue behavior approaches that of processor sharing as claimed in [9], assuming identical RTTs among all flows.

VII. Central Limit Theorem

In this section we complement the Law of Large Numbers results in Theorem 1 with a Central Limit theorem (CLT). The analysis is carried out under the same model in Section III. However, we need to strengthen Assumption (A1) and introduce an additional assumption.

(A1b) Assumption (A1) holds with mapping $f : \mathbb{R}_+ \to [0,1]$, which is continuously differentiable, i.e., its derivative $f' : \mathbb{R}_+ \to \mathbb{R}$ exists and is continuous.

(A6) The workload of a new TCP connection is bounded, i.e., there exists an integer $X_{\max}$ such that for $i = 1, \ldots, N$ and $t = 0, 1, \ldots, F_i^{(N)}(t) \in \{1, \cdots, X_{\max}\}$.[6]

Fix $t = 0, 1, \ldots$. The following quantity plays a crucial role in the analysis:

$$K(t) := C - q(t) - \mathbf{E}\left[A(t)\right]. \tag{23}$$

We can interpret $K(t)$ as the asymptotic residual capacity per session in the timeslot $[t, t+1)$. Now we define a collection of rvs that is integral to the analysis. For each $N = 1, 2, 3, \ldots$, and $\mathbf{y} = (w, x, s, d, b, m) \in \mathcal{Y}$, let

$$L_0^{(N)}(t) = \frac{Q^{(N)}(t)}{N} - q(t)$$
$$\hat{L}_0^{(N)}(t) = \frac{\hat{Q}^{(N)}(t)}{N} - \hat{q}(t)$$

and

$$L_{\mathbf{y}}(t) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}_{\mathbf{Y}_i^{(N)}(t)}[\mathbf{y}] - \mathbf{P}_{\mathbf{Y}(t)}[\mathbf{y}].$$

Theorem 2: Assume (A1b), (A2), and (A6) hold. Then, for each $t = 0, 1, \ldots$, there exists an $\mathbb{R}^{|\mathcal{Y}|+2}$-valued rv $\mathbf{L}(t) = (L_0(t), \hat{L}_0(t), L_{\mathbf{y}}(t), \mathbf{y} \in \mathcal{Y})$ such that the convergence

$$\sqrt{N}\left(L_0^{(N)}(t), \hat{L}_0^{(N)}(t), L_{\mathbf{y}}^{(N)}(t), \mathbf{y} \in \mathcal{Y}\right) \Longrightarrow_N \mathbf{L}(t) \tag{24}$$

holds. Moreover, the distributional recursions

$$L_0(t+1) =_{st} \begin{cases} 0, & K(t) > 0 \\ L_0(t) + \bar{L}(t), & K(t) < 0 \\ \left(L_0(t) + \bar{L}(t)\right)^+, & K(t) = 0 \end{cases} \tag{25}$$

and

$$\hat{L}_0(t+1) =_{st} (1-\alpha)\hat{L}_0(t) + \alpha L_0(t+1) \tag{26}$$

hold, where

$$\bar{L}(t) = \sum_{\mathbf{y} \in \mathcal{Y}} \min(w, x)\mathbf{1}[b \geq d] L_{\mathbf{y}}(t). \tag{27}$$

The distribution of the rvs $L_{\mathbf{y}}(t)$, $\mathbf{y} \in \mathcal{Y}$, $t = 0, 1, \ldots$, can be calculated recursively starting with $t = 0$.

Finally, for any $t = 1, 2, \ldots$, the rv $L_0(t+1)$ is Gaussian if $K(s) \neq 0$ for all $s \leq t$.[7]

A proof of Theorem 2 and the complete distributional recursion of $L_{\mathbf{y}}(t)$ are provided in [17].

From the last statement of Theorem 2, a necessary condition for $L_0(t)$ not to be Gaussian is $K(s) = 0$ for some $s < t$. This is, however, a technical artifact of the limiting regime as in practice it is unlikely that the real-valued residual capacity would attain the *exact* value of zero. Therefore, the distribution of the RED buffer at any fixed time t for large N can be well approximated by a Gaussian rv with a mean of $N \cdot q(t)$ (from Theorem 1) in practice.

A. Discussion

While Theorem 1 suggests that as the number of sessions becomes large, the aggregate queue behavior can be well approximated by a deterministic process, Theorem 2 tells us how the fluctuations in the queue and average queue sizes behave around the expected behavior given by the deterministic process. These two results combined give us an accurate model of the queue dynamics when the number of sessions is large, which is of the form

$$Q^{(N)}(t) \simeq Nq(t) + \sqrt{N}L_0(t)$$

where the calculations of $q(t)$ and $L_0(t)$ are independent of the number of sessions.

The CLT analysis also reveals the sources of (random) fluctuations in the queue size. As a byproduct of the proof of Theorem 2, the distributional recursion of $L_{\mathbf{y}}(t)$, $\mathbf{y} \in \mathcal{Y}$, $t = 0, 1, \ldots$, is derived in the proof of Theorem 2 [17]. Combining this with (27), we find that the fluctuation in the traffic arrival $\bar{L}(t+1)$ consists of four components:[8]

(i) Fluctuation caused by the discrepancy between the feedback information from RED to TCP sources $f^{(N)}(\hat{Q}^{(N)}(t))$ and the limiting feedback information $f(\hat{q}(t))$: This uncertainty in feedback information can be explained by the following lemma (also known as the *Delta Method* [21]). First define

$$\gamma(t) := f\left(\hat{q}(t)\right) \quad \text{and} \quad \gamma^{(N)}(t) := f\left(\frac{\hat{Q}^{(N)}(t)}{N}\right).$$

Lemma 2 *If $f : \mathbb{R}_+ \to [0,1]$ is differentiable with a continuous derivative at $x = \hat{q}(t)$, then the convergence $\sqrt{N}((\hat{Q}^{(N)}(t)/N) - \hat{q}(t)) \Longrightarrow_N \hat{L}_0(t)$ implies $\sqrt{N}(\gamma^{(N)}(t) - \gamma(t)) \Longrightarrow_N f'(\hat{q}(t))\hat{L}_0(t)$.*

Note that as the slope of the feedback function increases, the magnitude of fluctuation due to this component increases as well. This verifies the observation that the magnitude of queue size oscillation at RED gateways increases with the slope of marking probability function of RED mechanism [20].

(ii) Binary nature of feedback information: A TCP source updates its congestion window size based on whether a transmitted packet has been marked or not. This binary nature of feedback information poses limited feedback information granularity, and causes a fluctuation in the

[6]The limit $X_{\max}$ can be lifted with a more complicated analysis. However, such a restriction is still necessary for the numerical calculation of the CLT on a computer.

[7]Here we interpret a constant as a Gaussian rv with variance of zero.

[8]The rv $\bar{L}(t+1)$ represents the fluctuation in packet arrivals and appears in the fluctuation of the queue size $L_0(t+2)$ (if $K(t+1) \leq 0$) through the recursion in (25).

queue size. This fluctuation can be well approximated by a Gaussian rv.

(iii) Fluctuation caused by the arrival of new TCP connections and the random idle periods and workload: The larger the workload and idle period variances are, the larger the magnitude of this fluctuation is. This part of the fluctuation can also be described by a Gaussian rv.

(iv) Fluctuation caused by the structure of protocols: The structure of the protocols determines how the rvs discussed in (i)–(iii) at time t are combined and propagate to time $t+1$. The resulting rv captures the overall fluctuation in the arriving traffic (and subsequently in the queue).

Components (ii) and (iv) are due to the protocols and cannot be mitigated without a major modification to the protocols. Component (iii) depends on users behavior, and, hence, is beyond the control of the network. Thus, network designers can only manipulate the slope of the packet marking function to control oscillation of queue size. Although reducing the slope of the marking function can decrease the magnitude of fluctuation, it also increases the average queue size as suggested by (22).

If the protocol suite (i.e., TCP and ECN/RED) can be modified, then we can further reduce the magnitude of queue fluctuation caused by coarse feedback granularity [component (ii)]. One simple scheme to improve the feedback information granularity is to increase the number of feedback information bits in the ECN mechanism. Given that the improved feedback information is properly utilized, the magnitude of queue fluctuation will be reduced. Multi-level ECN (MECN) [4] is an example of such a scheme. It results in reduced oscillations in the queue size at the expense of an increased signaling overhead.

VIII. Simulations

In this section, we provide numerical examples to validate the results presented in Section VII. Using these examples, we will investigate the relative contribution from various sources [components (i)–(iv) in Section VII-A] to the random fluctuations in the queue size. Simulation results for the Law of Large Numbers established in Section IV are provided in [18].

We have shown in Section VII that the random fluctuations in the queue size can be approximated by Gaussian rvs when the number of flows N is large. In other words, for any $t = 0, 1, \ldots$, and large N

$$\frac{Q^{(N)}(t)}{N} - q(t) \approx \frac{L_0(t)}{\sqrt{N}}$$

for some zero-mean Gaussian rvs $L_0(t)$. This result tells us that the sample standard deviation (SD) of the normalized queue fluctuation at steady-state should decrease with the rate $1/\sqrt{N}$ for large values of N. This will be demonstrated using both Monte Carlo simulations of the model described in Section III and NS-2 simulations.

While Theorem II identifies different sources of the random queue fluctuations, it does not provide a clear quantitative picture on the relative contributions of these components as the numerical recursions rising from Theorem II are complicated. Therefore, we resort to numerical studies for the answers. Quantifying the relative contributions will help us better understand the causes of (unexpected) fluctuations in the queue size and design better AQM mechanisms.

In this section, we carry out the simulation under various settings and compare the sample SDs of the normalized queue. We first run the simulation with persistent TCP flows (without session dynamics, i.e., the workload size is infinite), and then compare the SD of the normalized queue size with the results from simulations with session dynamics where the workload brought in by a connection is given by a geometric distribution. Then, we vary the slope of the marking function to study the impact of the slope on the magnitude of the queue oscillation—component (i) in Section VII-A. Finally, we run the simulation with different workload distributions to understand their effects on the queue oscillations. Note that components (ii) and (iv) are parts of the protocol and cannot be isolated without introducing new protocols, and, hence, their investigation is beyond a scope of this study.

A. Monte Carlo Simulation of the Model

This subsection presents numerical results from the Monte Carlo simulations of the model outlined in Section III. The system and control parameters are set as follows. The marking functions are chosen to be

$$f^{(N)}(x) = \begin{cases} 0, & \text{if } x < 0 \\ \frac{x}{Q_{\max}^{(N)}}, & \text{if } 0 \le x \le Q_{\max}^{(N)} \\ 1, & \text{otherwise} \end{cases}$$

for some $Q_{\max}^{(N)}$. We choose $Q_{\max}^{(N)}$ to be $100N$, $75N$, and $125N$ to change the slope of the marking function in order to investigate the effects of the Delta method.

The receiver advertised window size $W_{\max}$ is set to 64. The exponential averaging parameter α is set to 1. The random round-trip delay $H_i(t)$ of a new connection is uniformly distributed on the set $\mathcal{H} = \{2, \ldots, 16\}$. We take the average of 10 independent simulation runs; each run consists of 10 000 timeslots with the variables evolving from their initial values according to the dynamics outlined in Section III.

In all of our simulations, we have selected the system and control parameters so that the queue behavior is stable and the normalized queue size oscillates around a small neighborhood of steady-state mean. We discard the first 1000 timeslots to remove the transient period and consider the last 9000 timeslots. In our simulations, the queue behavior appears to reach steady-state in the first 1000 timeslots. We compute the steady-state SD of the normalized queue size by averaging the sample queue size SDs of 10 simulation runs.

1) Effects of Session Dynamics: For the system with session dynamics, the normalized capacity per flow C is set to be 0.6 packets/timeslot. The workload distribution F is geometric with $p = 1/400$, i.e., $\mathbf{E}[F] = 400$ packets. The idle periods of sessions are geometrically distributed with a mean of 100 timeslots. Under these parameters, the normalized queue size at steady-state $q^\star$ is 5.1 packets/user for $Q_{\max}^{(N)} = 100N$.

In order to quantify the effects of the session dynamics on the random queue fluctuations, we also simulate the system

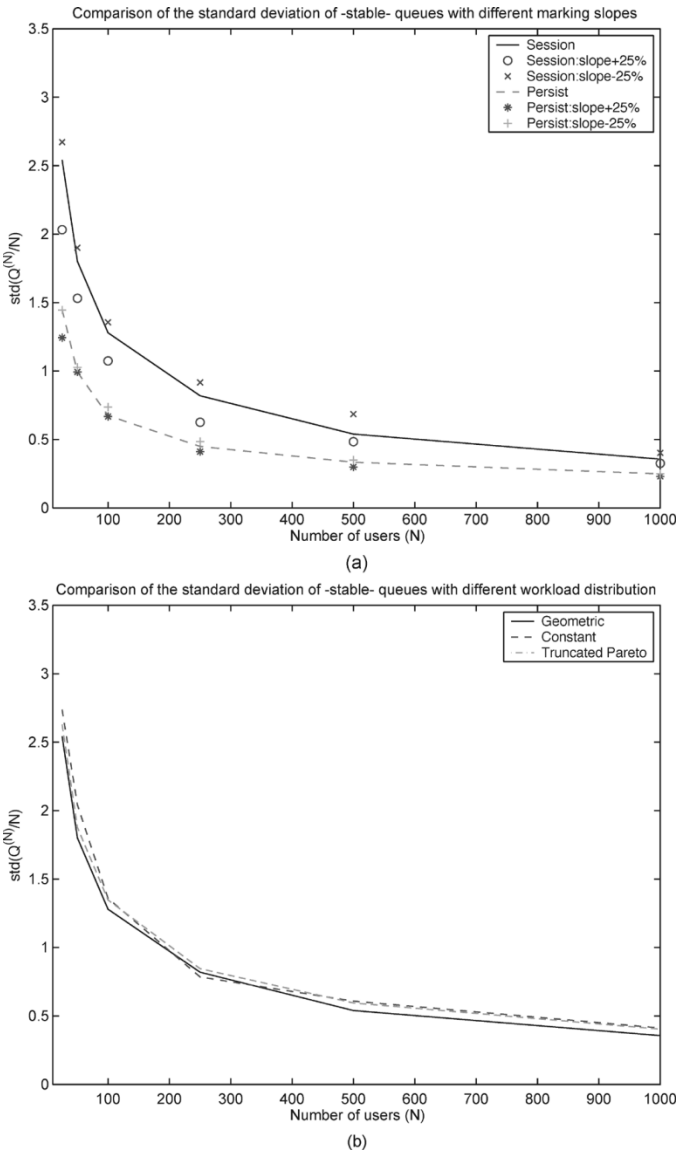

Fig. 1. Sample standard deviations of the normalized queue size from Monte Carlo simulations of the model. (a) Persistent flows versus session dynamics with different marking function slopes. (b) Different workload distributions.

with only persistent flows, i.e., flows with infinite workload. If the same parameter settings used with session dynamics are adopted, then the steady-state queue size becomes larger since all users are always active with persistent flows. As we are interested in the magnitude of queue oscillations (more specifically, the SD of the queue oscillations), we need to maintain the same steady-state queue size to make a fair comparison with and without session dynamics. To this end, we adopt the same parameter settings used with session dynamics and increase the normalized capacity per user C to 1.0 packet/user. This yields the same steady-state queue size of 5.1 packets/user for $Q_{\max}^{(N)} = 100N$.

Fig. 1(a) shows the simulation results comparing the sample SDs between these two setups. As expected from Theorem 2, the standard deviation decreases as $1/\sqrt{N}$ for large values of N in all of the simulations. It is clear from Fig. 1(a) that a major contribution to the random queue fluctuations comes from the session dynamics. Furthermore, as suggested earlier from the Delta method, the magnitude of the queue fluctuations increases with the slope of the marking function. While the changes in the slope of the marking function have minimal effects on the SD in the system with only persistent flows (as long as the queue is stable), the SD of random queue fluctuations becomes much more sensitive to the slope of the marking function with session dynamics.

2) Effects of Different Workload Distributions: In this subsection, we investigate the effects of different workload distributions on the magnitude of random queue fluctuations. To this end we simulate the system with session dynamics under the same settings except for the workload distribution F. We have set F to be: 1) truncated Pareto and 2) constant workload in addition to the geometric distribution used in the previous subsection. We keep $\mathbf{E}[F]$ fixed at 400 packets for all distributions, and, hence, the steady-state queue size remains at 5.1 packets/user as the steady-state queue size depends only on the mean workload as shown in (22). For truncated Pareto distribution,[9] the shape parameter is set to be 1.7 and the location parameter, i.e., the minimum size of the workload, is set to be 250 packets. For constant workload, a new connection has a fixed workload of 400 packets.

Fig. 1(b) shows the simulation results comparing the sample SDs with these workload distributions. Our results suggest that the workload distribution does not significantly alter the magnitude of random queue fluctuations.

B. NS-2 Simulations

In this subsection, we verify the observations in the numerical examples in the previous subsection using a more realistic event-driven NS-2 simulator. In the simulations, we first gradually vary the number of sessions N from 25 to 1000, and observe the SD of the normalized queue fluctuations under similar setups as in the previous subsection.

The parameters used in the simulation with session dynamics are given as follows. The bottleneck link capacity is $0.20 \cdot N$ Mb/s, and the buffer size is set to $200N$ packets. The RED gateway with ECN option is configured with the marking function similar to the ones in Section VIII-A. Packet size is fixed to 1000 bytes, and the receiver advertised window size $W_{\max}$ is set to 64 packets. The exponential averaging weight of the RED gateway is configured to be $0.05/N$ in order to have a similar time constant regardless of the number of users. A new connection generates a workload according to either a geometric, truncated Pareto, or constant distribution with the same mean of 400 packets. The rest of the parameters are set to similar values as in Section VIII-A. An idle period of a user between two consecutive connections is exponentially distributed with a mean of 5 seconds. A connection terminates when it runs out of data to transfer. We also enable the *drop_front_*

[9]For Pareto distribution, the cumulative distribution function is $F(x) = 1 - (k/x)^{\alpha}$, $x \geq k$, where $\alpha > 0$ is the shape parameter and $k > 0$ is the location parameter, i.e., the minimum value of the rv with such distribution. With the shape parameter between $(1, 2)$, the distribution is heavy-tailed, i.e., it has a finite mean but infinite variance. The truncated Pareto distribution limits the maximum value of the distribution and is often used to model the workload distribution of the objects transferred in the Internet for some $\alpha \in (1, 2)$ [3]. One can also specify the truncated Pareto distribution by specifying the shape, location, and the mean value.

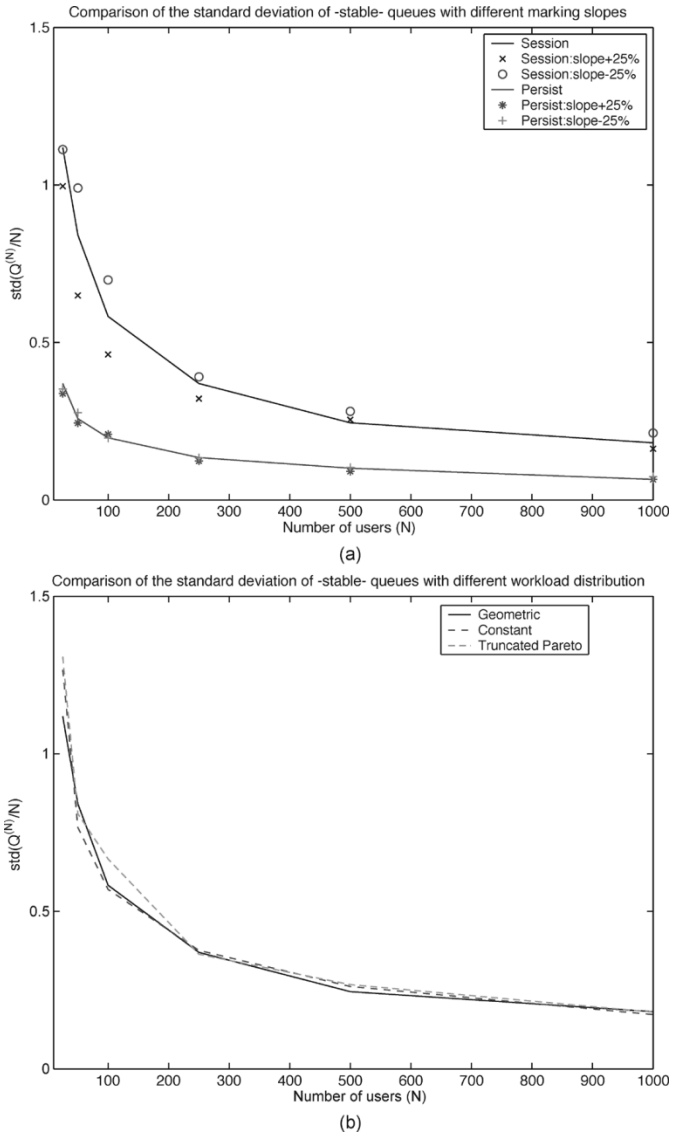

Fig. 2. Sample standard deviations of the normalized queue size from NS-2 simulations. (a) Persistent flows versus session dynamics with different marking function slopes. (b) Different workload distributions.

option of the RED mechanism so that the RED gateway marks the packet at the front of the queue rather than the packet that has just arrived. This reduces the feedback delay to the TCP senders. The round-trip propagation delays of the users are randomly selected uniformly from [50, 150] ms, with a mean of 100 ms. Under these parameters, the queue appears stable and the steady-state queue size is around 5 packets/user for $Q_{\max}^{(N)} = 100N$.

For the system with persistent TCP flows of infinite traffic, the same parameter values are adopted again, except for the capacity being scaled up to $0.24 \cdot N$ Mb/s to maintain the same normalized steady-state queue size. In all of the simulations the queue is sampled every 100 ms, and the simulation time is 1000 seconds. We conduct 10 independent simulation runs for each setup. The sample SD is calculated by taking the average of 10 runs after discarding the first 100 seconds from each run, which is sufficient for the queue size to settle down to near steady-state.

Fig. 2 shows the NS-2 simulation results. It is clear that the numerical results follow a similar trend as in the Monte Carlo simulations of the model in the previous subsection. This suggests that our stochastic model captures the essential qualitative behavior of queue dynamics of an ECN/RED gateway. The SDs of the stochastic model from the Monte Carlo simulations are somewhat larger than those from the NS-2 simulation. This is partially an artifact of the discrete-time model as all packet transmissions from a connection in a RTT take place in a single timeslot in our model (as assumed in Section III). Furthermore, all the flows are synchronized at the timeslot level (at the beginning of each timeslot), whereas in the NS-2 simulations the flows are not synchronized with respect to any time scale and react to the marks from the RED gateway in an asynchronous manner (hence, the aggregate traffic is smoother).

C. Discussion

There are several observations one can make from both Monte Carlo and NS-2 simulations. First, it is clear that the arrivals and departures of connections (or simply session dynamics) are a major contributor to the random queue fluctuations. This is in part due to the aggressive behavior of the slow-start mechanism of TCP that causes bursty traffic arrival, combined with the fact that RED is not aimed at controlling short bursty flows by design [5], [7]. Second, the magnitude of the random queue fluctuations is *not* sensitive to the slope of the marking function. However, the *stability* of a system with TCP flows and RED gateway(s) is shown to be relatively sensitive to the slope of the system [6], [15], and, hence, a proper selection of the marking function is important from the viewpoint of maintaining a stable system. Third, oscillations caused by components (i), (ii), and (iv) are also significant. Hence, it may be possible to considerably reduce the queue oscillations by reducing these components through more fine-grained feedback information and/or a modification of TCP protocol.

Finally, the magnitudes of random fluctuations are not sensitive to the workload distribution. This observation may be somewhat surprising at first as there are reports and analytical results suggesting that heavy-tailed file size distributions (such as Pareto distribution) are a cause of the self-similar behavior of Internet traffic, leading to the heavy-tailed distribution of a bottleneck queue size (see [13] for examples). We note, however, that such observations are collected before AQM mechanisms such as RED are being widely-adopted, and the analytical results suggesting the heavy-tailed distribution of queue size are carried out in the context of queue with infinite buffer and no interaction of TCP with an AQM mechanism. Our observation can be explained by the following arguments. In a stable TCP/AQM network, the AQM mechanism attempts to maintain a queue size around the stable operating point by regulating the TCP traffic. Hence, as long as there is sufficient TCP traffic, a stable network with an AQM mechanism will be able to successfully regulate the incoming traffic and, hence, the dynamics of the queue will not be sensitive to the workload distribution.

IX. Conclusion

In this paper, we have developed a scalable model of a RED gateway under a large number of TCP flows. We have demonstrated several interesting behaviors of the limiting

model. These results are further strengthened by our CLT results. They provide us with a scalable tool that can be utilized for network dimensioning without suffering from the curse of dimensionality. Furthermore, their proofs provide valuable insights into the queue behavior and help us design better AQM mechanisms. Our model is shown to be consistent with other previously proposed models in their respective regime. A formula for computing the average queue size in steady-state as a function of system parameters is derived and validated through numerical examples. The approach taken in this paper is extended to a generic probabilistic AQM mechanism and a congestion control mechanism in [19].

This research complements the control-theoretic studies of TCP/RED dynamics which derive sufficient conditions for the stability of the system that will ensure that the user rates and queue size asymptotically settle to equilibrium values. Under these sufficient conditions, Assumption (A3), i.e., $(q(t), \hat{q}(t), \mathbf{Y}(t)) \Longrightarrow_t (q^\star, q^\star, \mathbf{Y}^\star)$, is reasonable. Therefore, the average behavior of such a complex stochastic feedback system at the equilibrium can be described using the LLNs, CLT, and the steady-state analysis when there is a large number of flows and the control parameters are set properly using the sufficient conditions for stability from the control-theoretic analyzes.

ACKNOWLEDGMENT

The authors would like to thank Prof. A. M. Makowski and Prof. S. H. Low for their helpful discussions and comments.

REFERENCES

[1] E. Altman, K. Avrachenkov, and C. Barakat, "TCP in presence of bursty losses," in *Proc. ACM SIGMETRICS*, Santa Clara, CA, 2000, pp. 124–133.

[2] F. Baccelli, D. R. McDonald, and J. Reynier, "A mean-field model for multiple TCP connections through a buffer implementing RED," INRIA, Sophia Antipolis, France, Tech. Rep., Apr. 2002.

[3] M. E. Crovella and A. Bestavros, "Self-similarity in World Wide Web traffic: evidence and possible causes," in *Proc. ACM SIGMETRICS*, Philadelphia, PA, 1996, pp. 160–169.

[4] A. Durresi, M. Sridharan, C. Liu, M. Goyal, and R. Jain, "Multilevel early congestion notification," in *Proc. 5th World Multiconf. Systemics, Cybernetics and Informatics*, Orlando, FL, Jul. 2001, pp. 12–17.

[5] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE Trans. Netw.*, vol. 1, no. 4, pp. 397–413, Aug. 1993.

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

[7] C. V. Hollot, Y. Liu, V. Misra, and D. Towsley, "Unresponsive flows and AQM performance," in *Proc. IEEE INFOCOM*, Apr. 2003, pp. 85–95.

[8] V. Jacobson, "Congestion avoidance and control," in *Proc. ACM SIGCOMM*, Aug. 1988, pp. 314–332.

[9] A. Kherani and A. Kumar, "Stochastic models for throughput analysis of randomly arriving elastic flows in the Internet," in *Proc. IEEE INFOCOM*, 2002, pp. 1014–1023.

[10] M. Mathis, J. Semske, J. Mahdavi, and T. Ott, "The macroscopic behavior of TCP congestion avoidance algorithm," *Comput. Commun. Rev.*, vol. 27, no. 3, pp. 67–82, Jul. 1997.

[11] M. Mellia, I. Stoica, and H. Zhang, "TCP model for short lived flows," *IEEE Commun. Lett.*, vol. 6, no. 2, pp. 85–87, Feb. 2002.

[12] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP Reno performance: a simple model and its empirical validation," *IEEE/ACM Trans. Netw.*, vol. 8, no. 2, pp. 133–145, Apr. 2000.

[13] K. Park and W. Willinger, Eds., *Self-Similar Network Traffic and Performance Evaluation*. New York: Wiley, 2000.

[14] V. Paxson and S. Floyd, "Wide area traffic: the failure of Poisson modeling," *IEEE/ACM Trans. Netw.*, vol. 3, no. 3, pp. 226–244, Jun. 1995.

[15] P. Ranjan, E. H. Abed, and R. J. La, "Nonlinear instabilities in TCP-RED," *IEEE/ACM Trans. Netw.*, vol. 12, no. 6, pp. 1079–1092, Dec. 2004.

[16] S. Shakkottai and R. Srikant, "How good are deterministic fluid models of Internet congestion control?," in *Proc. IEEE INFOCOM*, Jun. 2002, pp. 497–505.

[17] P. Tinnakornsrisuphap and R. J. La, "Asymptotic behavior of heterogeneous TCP flows and RED gateways," Inst. Syst. Res., Univ. Maryland, College Park, MD, Tech. Rep., 2003.

[18] P. Tinnakornsrisuphap and R. J. La, "Limiting model of ECN/RED under a large number of heterogeneous TCP flows," Inst. Syst. Res., Univ. Maryland, College Park, MD, Tech. Rep., 2003.

[19] P. Tinnakornsrisuphap and R. J. La, "Characterization of queue fluctuations in probabilistic AQM mechanisms," *ACM SIGMETRICS Perform. Eval. Rev.*, pp. 283–294, Jun. 2004.

[20] P. Tinnakornsrisuphap and A. M. Makowski, "Limit behavior of ECN/RED gateways under a large number of TCP flows," in *Proc. IEEE INFOCOM*, Apr. 2003, pp. 873–883.

[21] A. W. van der Vaart, *Asymptotic Statistics*. Cambridge, U.K.: Cambridge Univ. Press, 1998.

[22] L. Zhang, S. Shenker, and D. Clark, "Observations on the dynamics of a congestion control algorithm: the effects of two-way traffic," in *Proc. ACM SIGCOMM*, Sep. 1991, pp. 133–145.

Peerapol Tinnakornsrisuphap (S'98–M'05) received the B.Eng. degree from Chulalongkorn University, Thailand, the M.S. degree from the University of Wisconsin-Madison, and the Ph.D. degree from the University of Maryland, College Park, in 1998, 2000, and 2004, respectively, all in electrical engineering.

He is a Senior Systems Engineer in Corporate Research and Development, Qualcomm, Inc., San Diego, CA. His research interests include resource allocation and congestion control in both wireline and wireless networks, end-to-end application performance, and queueing theory.

Richard J. La (S'98–M'01) received the B.S.E.E. degree from the University of Maryland, College Park, in 1994, and the M.S. and Ph.D. degrees in electrical engineering from the University of California at Berkeley in 1997 and 2000, respectively.

From 2000 to 2001, he was a Senior Engineer in the Mathematics of Communication Networks group at Motorola. Since August 2001, he has been on the faculty of the Department of Electrical and Computer Engineering, University of Maryland, College Park. His research interests include resource allocation in communication networks and application of game theory.

Dr. La is a recipient of a National Science Foundation (NSF) CAREER Award.