Stationary Behavior of TCP/AQM with Many Flows Under Aggressive Packet Marking

Do Young Eun
Department of Electrical and Computer Engineering
North Carolina State University
Raleigh, NC 27695
Email: dyeun@ncsu.edu

Xinbing Wang
Department of Electrical and Computer Engineering
North Carolina State University
Raleigh, NC 27695
Email: xwang8@ncsu.edu

Abstract—We consider a TCP/AQM system shared by many flows under general packet marking schemes. Traditional approaches in the literature require that the marking function \( p^N(x) \) be scaled linearly in the number of flows \( N \), i.e., \( p^N(Nx) = p(x) \) for some function \( p \), and they all invariably fail to predict the system performance if the marking function is scaled more aggressively, i.e., \( p^N(N^\alpha x) = p(x) \) with \( \alpha \in (0, 1) \). In this paper, by noting that there are two different sources of randomness in packet arrivals to the queue, we develop a simple stationary model for a TCP/AQM system with \( N \) flows under the aggressive packet marking. Our main results show that, under any aggressive marking scale, the system always behaves nicely in the sense that the link utilization goes to 1 and the queueing delay decreases to zero as the system size \( N \) increases. We verify our results using ns-2 simulation under different AQM schemes, and show that the buffer size can be chosen much smaller than \( O(\sqrt{N}) \) as recently predicted in [1] without affecting all the key performance metrics.

I. INTRODUCTION

In the current Internet, Transmission Control Protocol (TCP) is responsible for carrying more than 90% of the bytes of total traffic generated in the network. In addition to providing reliable data transport, TCP with AQM (Active Queue Management) at routers along the path of a flow performs congestion control. Since its inception, TCP/AQM has suffered numerous modifications in order to improve various performance metrics, e.g., throughput, delay, scalability, etc., under different network environments [2], [3], [4], [5], [6], [7].

Another noticeable feature of the current Internet is that it becomes larger and larger in the number of users (or flows) and the size of capacity at the network core. For instance, a core router is typically capable of serving over 10,000 flows [8]. Thus, the performance implication of this large system could potentially be very different from those predicted by a simple model with single or several flows. If any design guideline based on small systems is blindly applied to the current large systems, it can result in severe performance degradation or huge waste of link bandwidths.

Motivated on this, there recently have been several results in the literature focusing on a TCP/AQM system with many flows. For example, in [9], [10], [11], the authors showed that the trajectory of transmission rate of each flow in the presence of random noise converges to a solution of a system without the noise (i.e., nonlinear delayed-differential equations), and analyzed the stability of the resulting system. In [12], [13], [14], using packet-based models in which each packet is marked with some probability, the authors derived some limit behaviors of the system as the number of users \( (N) \) and the size of the capacity \( (NC) \) increase proportionally. However, for most of these results in the literature, the marking function \( p^N(x) \in [0,1] \), or the probability of packet marked when the queue-length is \( x \), was assumed to be scaled linearly, i.e., for all \( N \) and \( x \), \( p^N(Nx) = p(x) \) for some function \( p \), where the superscript \( \cdot N \) in the marking function \( p^N(x) \) means that there are \( N \) flows with capacity \( NC \). Accordingly, \( Q^N(t) \) turns out to grow linearly with \( N \). This implies that, even under AQM schemes with ECN (Explicit Congestion Notification) marks [3], the buffer size still should be on the order of \( O(N) \), in accordance with the common “rule-of-thumb” for buffer sizing under a drop-tail policy – being proportional to the bandwidth-delay product, i.e., proportional to \( N \) to avoid under-utilization of the link. (See also [15] for recent results on this.)

Quite recently, in [1], using ns-2 simulation under different AQM schemes, and show that the buffer size can be chosen much smaller than \( O(\sqrt{N}) \) as recently predicted in [1] without affecting all the key performance metrics.
AQM policy is considered, and [16] for $\alpha = 0$, for which the authors showed that their queue-based marking scheme with an exponential marking function (of REM type) behaves as if it were a rate-based marking scheme. However, their approaches are carefully tailored to the specific scales considered in the papers (i.e., $\alpha = 0.5$ or 0) and not applicable to any other scales of our interest. We note that different marking scale simply means different parameter set-up at the routers, which can be easily configured as desired. Moreover, the scaling of marking functions, or AQM schemes in general, is directly related to the issues of network design over a long time scale. For example, if the number of connections (or customers) were to double, the a natural question to ask is: what percentages of additional capacities or buffer sizes are needed at routers to provide the same level of performance to each user? What about the AQM strategies? What is the cost-optimal way? In this regard, it is very important to investigate different ways of scaling the marking function and any possible impact it brings to the system.

In this paper, we will focus on the marking scale of $N^\alpha$ with $\alpha \in (0,1)$. (See Section II for detailed description.) For $\alpha = 1/2$, as mentioned above, it was already shown that drop-tail policy works well under a large number of flows. However, the scale of our interest ($0 < \alpha < 1$) in general has never been addressed in the literature. In fact, as we will illustrate in Section II, any model based only on the evolution of window sizes, or any fluid-type model that has been widely used in the literature, fails to capture the system behavior under the scale of our interest ($0 < \alpha < 1$). Our approach to this problem is to first notice that there are in fact two different sources of randomness that need to be considered. We point out that these two sources of randomness could be the reason for the nice behavior of the system under any scale. Then, by assuming that the system is in ‘steady-state’, we propose a simple, yet versatile model that is able to capture the key system dynamics when there are many flows. Specifically, we show that even under the ‘aggressive’ marking scale with $0 < \alpha < 1/2$ (more aggressive than $\sqrt{N}$), the system still works well in the sense that the utilization approaches to 1 and the queueing delay of any packet goes to zero, as the system size $N$ increases. In particular, our result also shows that, for any buffer size scaled by $N^{\alpha+\epsilon}$ where $\epsilon > 0$ is arbitrary, the probability of packet drop in the buffer goes to zero as $N$ increases. This implies that buffer size can be much smaller than $O(\sqrt{N})$ without affecting all the ‘good’ performance measures the system needs to possess.

The rest of the paper is organized as follows. In Section II, we describe our model and illustrate the two sources of randomness. In Section III, under a steady-state assumption, we provide a simple model and present our main result with its implication to network design. Then, we present our simulation results in Section IV using ns-2 [17] to verify the results in Section III. Finally, we conclude in Section V.

II. Preliminaries

A. Scaling the Link: Basic Model

We consider a link with capacity $NC$ shared by $N$ simultaneous TCP connections or flows, where each flow’s window size evolves according to the Additive-Increase-Multiplicative-Decrease (AIMD) rule and $Q^N(t)$ denotes the queue length at time $t$. (See Figure 1 for illustration.) The buffer size at that link (or router) is also a function of $N$ and here denoted by $B(N)$. The common rule-of-thumb in network buffer dimensioning is that the buffer size at bottleneck links should be chosen in proportion to the bandwidth-delay product [18]. (That is, $B(N) = B \cdot N$ for some constant $B$.) In our work, we will explore the possibility of maintaining much smaller buffer size, i.e., $B(N) \sim O(N^\gamma)$ with any $\gamma \in (0,1)$, without sacrificing the system performance.

![Fig. 1. Simplified link model](image1)

For AQM schemes, we mainly consider queue-based schemes (e.g., RED) equipped with ECN capability. The consequence of using ECN is that routers mark the packets to notify incipient congestion (based on current or history of queue size) and drop only if the buffer is physically full and there is no room to accept the incoming packets. Figure 2 shows a version of marking functions that will be considered in the paper. When a packet arrives to the queue at time $t$, this packet will be marked with probability $p^N(Q^N(t))$, where $Q^N(t)$ denotes the queue size at time $t$. Hence, if the queue size $Q^N(t)$ is less than $q_{\min}N^\alpha$, no packet is marked. If the queue size becomes larger than $q_{\max}N^\alpha$, then all the incoming packets are marked (not dropped!). The packet-drop happens only when the queue size goes above the actual buffer size $B(N)$.

![Fig. 2. Example of scaling for RED with $0 < \alpha < 1$](image2)

B. Random Packet Arrivals

In this section, we explain how the queue-length can possibly remain on the order of $O(N^\alpha)$ when the marking function is scaled with $0 < \alpha < 1$, as opposed to being all the way up
to \(O(N)\) and resulting in system collapse. The main reason is that, as we will point out, there are two different sources of randomness: (1) random window size evolution due to the random packet marking, and (2) random packet arrivals to the queue given the window sizes of all flows.

Consider again a link with capacity \(NC\) shared by \(N\) flows as in Figure 1. Let \(T\) be the round-trip time (RTT) for all \(N\) flows and assume that they are all in the congestion avoidance phase. Let \(W_i^N(k)\) \((i = 1, 2, \ldots, N)\) be the window size of flow \(i\) at \(t = kT\) and \(W^N(k)\) be the sum of all the window sizes at \(t = kT\), i.e., \(W^N(k) = \sum_{i=1}^{N} W_i^N(k)\).

First, suppose that \(W^N(0) < NCT\) and \(Q^N(0) = 0\). Then, under the packet-based models in [12], [13], [14] where \(\alpha = 1\), the queue-length \(Q^N(t)\) for \(0 \leq t \leq T\) will remain zero since the total number of packets arrived during \([0, T)\) is smaller than the capacity. Thus, no packet will be marked and all the flows will increase their window sizes by one. So, if \(W^N(0)\) is very close to \(NCT\) (but still less than \(NCT\)), it follows that \(W^N(1) \approx NCT + N\). In their framework, since both the buffer size and the marking function are scaled by \(N\), this additional number of packets will be absorbed by the buffer and some of them will be marked randomly. Hence, \(W^N(k)\) also evolves randomly based on the number of marked flows during the previous RTT.

If we take the same approach for any marking scale of \(N^\alpha\) with \(\alpha \in (0, 1)\) for large \(N\), we would have very different system behavior. Specifically, under the same scenario, we would have \(W^N(1) \approx NCT + N\). But, since \(N \gg N^\alpha\) for large \(N\), most of the incoming packets during \([T, 2T]\) will now be marked and thus we have \(W^N(2) \approx (CT + 1)N/2\). Then, the sum of window sizes keeps increasing by \(N\) until it becomes greater than \(NCT\) by \(O(N)\), which again makes all the flows drop their rate by half. In other words, even though each packet is marked randomly, the marking scale of \(N^\alpha\) kills the randomness and in some sense makes the system synchronized.

However, this type of behavior never happens in reality. Once all the window sizes of the flows are given at time \(t = kT\), each sender will put \(W_i^N(k)\) number of packets onto the network. But, the actual packet arrival pattern to the queue is still random in time due to small variations in their inter-arrival times, which are mainly caused by interaction with other flows at routers prior to the one under our interest, time-varying queue-length (thus, time-varying queueing delays), different packet lengths, or small difference in their RTTs. Thus, we have another source of randomness for packet arrivals to the queue. In other words, the router queue behavior is governed by the number of packet arrivals as well as their arrival times, both of which are random. Figure 3 illustrate the difference.

Consider again a link with capacity \(NC\) shared by \(N\) flows and assume that they are all in the congestion avoidance phase. Under the model used in [12], [13], [14], once \(W_i^N(k)\) are given, it cannot distinguish the two arrival patterns to the queue as shown in Figure 3. As depicted at the bottom of Figure 3 in which the packet arrives randomly, for \(t \in [0, T]\), there will be some marked packets even though the sum of window sizes is smaller than the capacity. In other words, due to the random packet arrivals to the queue, there are some flows that already begin to drop their rate by half, even when the total arrival rate (over one RTT) is much smaller than the capacity, analogous to a situation where we have non-zero queue-length distribution for any utilization less than one, as expected from the traditional queueing theory. In some sense, the senders are able to detect the aggressive scaling used for the marking (by receiving packet marks earlier), and start to adjust to that scale in advance. Hence, instead of \(N\) flows that keep increasing their window sizes by one (and in the end, all of them will drop by half), there are some flows dropping the rate much earlier, and this “cooperation” with the marking scale will eventually make the system well-behaved and exhibit stationary behavior.

III. MAIN RESULTS

A. Steady-State Model

In this section, we describe our model in detail. Our model is simple, yet powerful enough to give us all the crucial performance measures under the scale regime of our interest. Motivated on the observations in Section II-B, we assume that the system is in steady-state in the sense that the distribution of packet arrivals to the queue does not change in time. In other words, the actual packet arrival process to the queue is stationary, and so is the queue-length distribution. As before, we consider a single link shared by \(N\) persistent flows, whose window sizes evolve according to the AIMD rule. (See Figure 1.) Let \(T\) be the round-trip-time delay for all \(N\) flows.

We define \(W_i^N\) to be the window size of flow \(i\) in the steady-state. Let \(w_{\text{max}}\) \((w_{\text{max}} > CT)\) be the maximum window size for all flows, i.e., \(1 \leq W_i^N \leq w_{\text{max}}\) for all \(i\) and \(N\). We define the steady-state utilization of the queue as

\[
\rho(N) = \frac{\mathbb{E}[W^N(k)]}{NCT} = \frac{\mathbb{E}[\sum_{i=1}^{N} W_i^N(k)]}{NCT}. \tag{1}
\]

We will show in Section IV that the process \(W^N(k)/N\) as a function of \(k\) can be well modeled by a stationary process for large \(N\) under our scale, thus supporting our steady-state assumption. Then, clearly, the utilization does not depend on time, and becomes a function of \(N\).

In order to capture the random nature in packet arrivals, we assume that actual aggregate packet arrivals to the queue in
the steady-state can be modeled by a Poisson process with mean rate $\rho(N)NCT$. (See Figure 4 for illustration.) The Poisson assumption for the aggregate arrivals to the queue is reasonable as long as packet arrivals from each flow are independent or ‘jittered’ independently, due to the fact that the superposition of independent point processes under a suitable scaling converges weakly to a Poisson process [19].

Since the arrival is a Poisson process with mean rate $\rho(N)NCT$, we can now find the queue-length distribution by using the steady-state solution of a standard queueing model such as $M/D/1$ or $M/G/1$ as long as $\rho(N) < 1$. Clearly, we should have $\rho(N) < 1$ for any large $N$. To see this, suppose $\rho(N) \geq 1$. Then, the system becomes unstable and the queue-length will grow up to infinity (much larger than $O(N^\alpha)$). Thus, most of the packets will be marked and most of the flows will reduce their rates by half. This eventually makes the utilization $\rho(N)$ about half of the current one, and thus, the system is not in steady-state.

Let $t_i^j$ is the arrival time instant of the $j^{th}$ packet from flow $i$. This packet will be marked, independently of any other, with probability $p^N(Q^N(t_i^j))$. Due to the steady-state assumption, the distribution of $Q^N(t)$ does not depend on $t$. Further, between $j^{th}$ and $(j + 1)^{th}$ packets of flow $i$, there will be approximately $N - 1$ interleaving packets coming from other $N - 1$ flows. So, we can also assume that $Q^N(t_i^j)$ and $Q^N(t_i^j)$ for $j \neq l$ are independent. Hence, the probability that flow $i$ receives no marks during one RTT becomes

$$f(N) := E\left\{\left[1 - p^N(Q_{p(N)}(N))\right]^{W_i^N}\right\},$$  \hspace{1cm} (2)

where the expectation is taken over $W_i^N$ and $Q_{p(N)}$, and $Q_{p(N)}$ is the random variable whose distribution is given by the steady-state queue-length distribution of the $M/D/1$ queue with utilization $\rho(N)$. Intuitively, the expression in (2) as a function of $N$ should not be too close to 0, nor to 1. To see this, assume that it is arbitrary close to 0 for some large $N$. Then, most of $N$ flows will increase their window sizes by one and the utilization during the next RTT duration clearly become different from the current one, and this contradicts that the system is in steady-state. Similar arguments hold for the case if it is close to 1. Hence, we should have the following:

$$0 < \liminf_{N \to \infty} f(N) \leq \limsup_{N \to \infty} f(N) < 1,$$  \hspace{1cm} (3)

where $f(N)$ is defined in (2).

In order to proceed, we need the following assumptions:

(A1) The packet lengths are distributed according to i.i.d. exponential random variables with mean 1.

(A2) The marking function $p^N(\cdot)$ satisfies $p^N(N^\alpha x) = p(x)$ for some $0 < \alpha < 1$ and for some non-decreasing function $p : \mathbb{R}^+ \to [0,1]$. Further, there exist $0 < q_{\min} < q_{\max} \in \mathbb{R}$ such that $p(x) = 0$ for $x \leq q_{\min}$ and $p(x) = 1$ for $x \geq q_{\max}$.

Remark 1: Assumption (A1) is equivalent to saying that the queueing model under consideration is $M/M/1$. Conceptually, we can go without this assumption. However, the queue-length distribution for $M/D/1$ is so complicated and defies any tractable analysis. Instead, by using $M/M/1$, the analysis becomes much more feasible and we can obtain key performance metrics in terms of $N$.

Assumption (A2) says that the marking function is scaled with $O(N^\alpha)$ and it has the form of an RED in that there are lower and upper thresholds for marking, beyond which none (or all) of the incoming packets are marked. Note that we do not require the marking function $p(x)$ for $q_{\min} \leq x \leq q_{\max}$ to be continuous. See Figure 2 for a typical example of a marking function satisfying (A2).

We are now ready to present our main result.

**Theorem 1:** Suppose that (A1) and (A2) are satisfied. Then, we have

$$\lim_{N \to \infty} \rho(N) = 1.$$  \hspace{1cm} (4)

In addition, if we define the steady-state queue-length random variable by $Q_{p(N)}$, then, for any $\epsilon > 0$, we have

$$\lim_{N \to \infty} \frac{Q_{p(N)}}{N^\alpha + \epsilon} = 0 \quad \text{with probability 1.}$$  \hspace{1cm} (5)

**Proof:** From (3), there exist constants $a,b \in (0,1)$ such that

$$0 < a \leq f(N) \leq b < 1,$$  \hspace{1cm} (6)

for all sufficiently large $N$. Note that from $(1-p^N(Q_{p(N)})) \in [0,1]$ and $1 \leq W_i^N \leq w_{\max}$, we have

$$E\left\{1 - p^N(Q_{p(N)}(w_{\max}))\right\} \leq f(N) \leq E\left\{1 - p^N(Q_{p(N)})\right\} = 1 - E\left\{p^N(Q_{p(N)})\right\}.$$  

Further, since the function $(1-x)^{w_{\max}}$ is convex for $x \in [0,1]$, we have from Jensen’s inequality that

$$1 - E\left\{p^N(Q_{p(N)})\right\} \leq E\left\{(1 - p^N(Q_{p(N)}))^{w_{\max}}\right\}.$$

Combining the above two relations and (6), we obtain

$$0 < 1 - b^{1/w_{\max}} \leq E\left\{p^N(Q_{p(N)})\right\} \leq 1 - a < 1,$$  \hspace{1cm} (7)

where $a, b \in (0,1)$.

Now, since $p^N(\cdot)$ is non-decreasing, we have, from Assumption (A2)

$$\{x \geq q_{\max} N^\alpha\} \leq p^N(x) \leq 1\{x \geq q_{\min} N^\alpha\}$$

for any $x \geq 0$. Thus, by taking expectation, we get

$$E\left\{p^N(Q_{p(N)})\right\} \leq E\left\{\left[1 - p^N(Q_{p(N)})\right]^{w_{\max}}\right\} \leq E\left\{p^N(Q_{p(N)})\right\},$$  \hspace{1cm} (8)

\hspace{1cm} (8)
Since $Q_{\rho}(N)$ has the steady-state queue-length distribution of $M/M/1$ queue with utilization $\rho(N) \in (0, 1)$,
\[ P\{Q_{\rho}(N) \geq n\} = \rho(N)^n \tag{9} \]
for any $n$. Thus, from (7), (8), and (9), we have
\[ 0 < 1 - \frac{1}{\max^w} \leq \rho(N)^{\max^w} \quad \text{and} \quad \rho(N)^{\min^w} \leq 1 - a < 1, \tag{10} \]
for all sufficiently large $N$. Then, from $\rho(N) \in (0,1)$ and (10), we have (4).

For (5), note that for any given $\delta > 0$, we have
\[ P\{Q_{\rho}(N) > \delta N^{\alpha + \epsilon}\} = (\rho(N))^{\delta N^{\alpha + \epsilon}} \leq (1 - \delta) \frac{N^{\alpha + \epsilon}}{\min^w} = \kappa^w, \]
for all sufficiently large $N$, where $\kappa = (1 - \delta) \frac{\delta}{\min^w} \in (0, 1)$ and the inequality follows from (11). Note that $\sum N \kappa^w < \infty$ for any $\epsilon > 0$. So, we have $\frac{\delta}{\min^w}$ as desired, since $\delta > 0$ is arbitrary, (5) follows from the Borel-Cantelli Lemma [20]. This completes the proof of Theorem 1.

B. Aggressive Marking in TCP/AQM with Many Flows

From Theorem 1, we know that for a marking function scaled as $N^\alpha$ with any $\alpha \in (0,1)$, the steady-state utilization $\rho(N)$ converges to 1 as the system size $N$ increases. Further, our theorem also shows that the steady-state queue-length random variable ($Q_{\rho}(N)$) fluctuates not more than $O(N^\alpha)$, in the sense that $Q_{\rho}(N)/N^{\alpha + \epsilon}$ goes to zero almost surely for any $\epsilon > 0$. Hence, for any scale $0 < \alpha < 1$ and for large $N$, the system behaves nicely in a way that the queue-length is kept in the ‘middle’, i.e., no more than on the order of $O(N^\alpha)$ as desired, while maintaining non-zero queue-length almost always, thus high utilization.

Our results would have huge impact on how to design TCP/AQM in large systems. Traditionally, designing AQM at core routers has been an extremely difficult and delicate problem. Although there are a myriad of results in the literature trying to propose ‘optimal’ ways to design AQM parameters, there is still no general agreement on this. Results vary depending on the network size and configuration, among other factors. Perhaps, the only viable solution out there for large systems would be that the buffer size should be approximately equal to the bandwidth-delay product as described earlier. This ‘first-order’ rule of scaling has been the basis for most of research work on TCP/AQM for large systems [9], [10], [11], [12], [13], [14], except the recent result in [1] saying that buffer sizes of $O(\sqrt{N})$ will do the same job for large $N$. Obviously, this first-order guideline for network design is much more important and should precede all the other ‘second-order’ guidelines on (optimal) fine-tuning of the parameters. In this regard, Theorem 1 clearly provides a new first-order guideline for TCP/AQM design for large networks. It says that, for large systems, you can use virtually any (aggressive) scale for the marking function while keeping high utilization (see (4)). Moreover, if the buffer size $B(N)$ is chosen on the order of $O(N^{\alpha + \epsilon})$ for any arbitrarily small $\epsilon > 0$, the packet drop probability is guaranteed to go to zero (see (5)). As a by-product, we also see that the queueing delay of a packet ($Q_{\rho}(N)/NC$) decreases to zero almost surely as $N$ grows. Note that since $\alpha \in (0,1)$ can be chosen arbitrarily as well, the required buffer size $B(N)$ to maintain high link utilization and low packet-drop probability can be much smaller than $O(\sqrt{N})$, provided that we use a queue-based AQM with ECN marks. This implies that we can save huge cost for high-speed buffers even further than the case of $O(\sqrt{N})$. In addition, although Theorem 1 assumes a sort of RED-type marking function, we will also show in Section IV through simulations that the same observation can be made for a wide class of AQM schemes including REM [6], PI [21], and even the Drop-Tail policy.

IV. Simulation Results

In this section, we perform simulations* using nws-2 [17] to verify Theorem 1. Although our theorem assumes an RED type of marking, we show that our result can also be applied to other popular AQM schemes such as REM [6], PI [21] and Drop-Tail (DT).

A. Simulation Topology and Parameters

We consider a simulation topology as in Figure 1 with $N$ TCP flows and the link capacity $NC$. We assume that each sender has an infinite amount of data to send and we set the receiver window size large enough so that the number of packets to be transmitted is purely governed by the congestion window size of each sender. We set the propagation delay of each link (from sender to the router and from the router to the receiver) as 50 ms. This gives $T=200$ ms if the queueing delay is negligible. Each packet size is fixed to 500 bytes and the link capacity is $N \times 100$ Kbps. Thus, we have $C = 25$ packets/sec, and the size of ‘pipe’ per flow becomes $CT = 25 \times 0.2 = 5$ packets. We consider four AQM schemes: RED, REM, PI, and Drop-Tail (DT).

<table>
<thead>
<tr>
<th>AQM</th>
<th>Scaled Parameters</th>
<th>Unscaled Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>RED</td>
<td>$mth_{\min} = N^\alpha$</td>
<td>$P_{\max} = 0.1$</td>
</tr>
<tr>
<td></td>
<td>$max_{\min} = 5N^\alpha$</td>
<td>$q_{\text{weight}} = 1$</td>
</tr>
<tr>
<td></td>
<td>Buffer Size = 6$N^\alpha$</td>
<td></td>
</tr>
<tr>
<td>REM</td>
<td>$pbo = N^\alpha$</td>
<td>$a = 1.001$</td>
</tr>
<tr>
<td></td>
<td>Buffer Size = 6$N^\alpha$</td>
<td>$q_{\text{weight}} = 1$</td>
</tr>
<tr>
<td>PI</td>
<td>$q_{\text{red}} = N^\alpha$</td>
<td>$b = 0.00000182$</td>
</tr>
<tr>
<td></td>
<td>Buffer Size = 6$N^\alpha$</td>
<td>$b = 0.00000018$</td>
</tr>
</tbody>
</table>

For RED, the minimum and the maximum thresholds ($q_{\min}N^\alpha$ and $q_{\max}N^\alpha$ in Figure 2) are $1 \times N^\alpha$ and $5 \times N^\alpha$, respectively. The maximum dropping probability, $P_{\max}$, is set to 0.1 and the queue averaging factor, $q_{\text{weight}}$, is set to 1. (We have also run our simulation with other queue averaging factors, e.g., $q_{\text{weight}}=0.002$, and obtained similar results.) The buffer size is given by $B(N) = 6 \times N^\alpha$.

*All the simulation source fi les including OTCL fi les and C++ fi les are available at [22].
Figure 5 shows the average window size evolution ($W^N(k)/N$) for different $N$ under this RED setting with $\alpha = 0.2$. As $N$ increases, the fluctuation of the average window size becomes less variable and well-behaved. This supports our assumption in Section III that the process $W^N(k)$ is stationary in $k$. Further, as $N$ increases, the average window size is steadily around $\approx 2$ packets, implying that we have almost full utilization as expected from Theorem 1.

For REM and PI, we set the target queue size (“pbo” of REM and “$q_{ref}$” of PI) as $N^\alpha$. The buffer sizes for REM, PI, and DT are the same as in RED. For other parameters of REM and PI, we use the default values as specified in ns-2. For all the AQM schemes except DT, we use ECN marking instead of dropping.

### B. Numerical Results

Throughout the simulation, we consider queueing delay, link utilization, and the packet loss (drop) ratio as our main performance metrics. These three metrics readily determine all the other performance metrics, e.g., throughput or goodput, and thus show how well the system behaves in any circumstance. Here, we do not concern the delay-jitter because, as we will show, the queueing delay itself approaches to zero.

First, we fix $\alpha = 0.2$ and increase the number of flows and the link capacity. Figure 6 shows how the three performance metrics change as $N$ increases under the four AQM schemes mentioned earlier. For the delay and the utilization in our simulation, we calculate the average queueing delay over all the packets arriving to the queue, and the average utilization over the entire duration of our simulation after deleting initial slow-start phase, respectively. Clearly, we see that for all AQM schemes under consideration, the queueing delay decreases and the utilization increases, as $N$ increases. Further, the loss ratio remains very small throughout, giving only 4% even in the worst case (drop-tail). Note that since $\alpha = 0.2$, even if $N$ increases from 40 to 180 in Figure 6, the thresholds for marking, target queue size, and the buffer size (all in $O(N^\alpha)$) increases only marginally from $40^0.2 = 2.09$ to $180^0.2 = 2.82$. Accordingly, all the performance metrics change very little.

Next, we fix the number of flows ($N = 200$) and vary $\alpha$ from $\alpha = 1$ to $\alpha = 0.2$. Note that, in this case, the number of flows and the size of the capacity remain fixed.
Instead, the marking thresholds, target queue-size, and the buffer sizes now decrease from 200 to $200^{0.2} = 2.88$ (i.e., becomes more and more aggressive), which is a much more dramatic change than in the previous case. Figure 7 shows the three performance metrics as $\alpha$ decreases from 1 to 0.2, while $N = 200$ is fixed. We note that the queuing delay for all the AQM scheme decreases sharply. For example, for RED, the queuing delay decreases from 100ms to almost zero (less than 1ms). However, the link utilization does not suffer too much from the decrease of $\alpha$. We see that nearly 95% link utilization is still achieved when $\alpha$ is decreased to 0.2. This decrease in the link utilization is well expected since the buffer size becomes much smaller and more and more packets are marked. This implies that we may need larger $N$ in order to obtain the same performance when $\alpha$ decreases. Similarly, the packet loss ratio also increases a little with the decrease of $\alpha$. But, even the highest packet loss ratio is still less than 4%. Moreover, from Figures 6 and 7, we can see that all the AQM schemes produce about the same trend for the three performance metrics. (Drop-tail gives a little poorer performance, but the difference is very minor.) This implies that the performance of TCP/AQM is mainly dominated by the marking scale (i.e., $\alpha$), not by specific AQM schemes being used.

Finally, it should be pointed out that, with our scale of $N^{0.2}$, the buffer size is much smaller than any other case considered in the literature. For example, according to the rule-of-thumb, the buffer size should be at least equal to the bandwidth-delay product, i.e., $NCT = 200 \times 5 = 1000$ packets. While for $\alpha = 0.5$, the buffer size for all the AQM schemes is $6 \times N^{\alpha} = 84.8$ packets, and for $\alpha = 0.2$, it is only 17.3 packets. Hence, we can do even better than $O(\sqrt{N})$ for the buffer sizing, while always maintaining high link utilization.

V. Conclusion

In this paper, we have developed a simple stationary model for a TCP/AQM system with $N$ flows under aggressive packet marking. We have pointed out that any traditional model used in the literature may wrongfully predict the failure of the system under the scale of our interest, while the system in fact works well and never collapses. We notice that there is another source of randomness that needs to be considered and this plays a crucial role in making the system well-behaved under our scale. Then, under the steady-state assumption, we have proved that the marking function can be chosen much more aggressively without affecting all the key performance measures as long as $N$ is large enough. This further implies that the buffer size can be chosen much smaller than $O(\sqrt{N})$. Our simulation results using various AQM schemes also confirm that the entire system behavior is mainly governed by the choice of scale in the marking function, and not by the specific AQM schemes and their parameters. Our results thus provide a new rule-of-thumb in designing AQM with many flows and show that the system works well under virtually any choice of scales for the marking functions, as long as the number of flows and the size of the capacity are large.

REFERENCES