

Simplification of Network Analysis in Large-Bandwidth Systems

Do Young Eun and Ness B. Shroff
Purdue University
West Lafayette, IN 47907-1285, U.S.A.
{eun, shroff}@ecn.purdue.edu

Abstract—In this paper, we show that significant simplicities can arise in the analysis of a network when link capacities are large enough to carry many flows. In particular, we prove that, when an upstream queue serves a large number of regulated traffic sources, the queue-length of the downstream queue converges almost surely to the queue-length of a simplified queueing system (single queue) obtained by removing the upstream queue. We provide similar results (convergence of the queue-length in distribution) for general (including non-regulated) traffic arrivals. In both cases, the convergence of the overflow probability is uniform and at least exponentially fast. Through an extensive numerical investigation, we demonstrate several aspects and implications of our results in simplifying network analysis.

I. INTRODUCTION

THE internet has already undergone a tremendous increase both in network capacity and in the number of end-users, and this trend is expected to continue for the next several years. Further, these end-users are becoming increasingly sophisticated and demand high-bandwidth, low-delay network services at affordable prices. The conflicting requirements of maintaining a high level of network utilization (for affordable prices or high revenue), while at the same time keeping network congestion under check (for ensuring a good level of quality of service), make it imperative to understand at a fundamental level how to design and control next generation networks.

The issues outlined above appear to be daunting within the confines of traditional stochastic and queueing techniques. However, we will show that the fact that a large number of traffic flows will be supported on the network can actually be exploited to help obtain results for predicting performance and allocating network resources.

The analysis of a queueing network is a well-known difficult problem except in special cases, such as Markovian queueing networks, for which product-form solutions are available [1], [2]. The difficulty in analysis is primarily because the traffic processes lose their original statistical characteristics as they traverse through the network. Given a myriad of sophisticated techniques developed for analyzing a single queue, there has been some recent work to introduce the notion of decomposition or decoupling in a network to make it possible to analyze

a network using those techniques [3], [4], [5]. However, due to the large deviations framework used in these works, the queue dynamics in the network are described only through the rate functions of traffic flows.

In this paper we present results that, from a queueing analysis perspective, allow us to virtually “ignore” a node that is capable of serving a large number of traffic flows. This property enables us to develop a simple decomposition method to simplify the analysis of a network with multiple queues. We first consider the case when the input traffic to the network is regulated (e.g., with a leaky-bucket, etc.), i.e., there exists a maximal arrival pattern in the worst case. Under this assumption, we prove that, as the number of flows (or the capacity) at the upstream queue increases, the queue-length at the downstream queue converges almost surely to the queue-length of a single queue that is obtained by removing the upstream queue from (or decomposing) the original system. For general (including non-regulated) traffic, we provide a result that shows that the overflow probability at the downstream queue converges uniformly to that of a single queue by removing the upstream queue. However, for this case, the proofs are quite technical and long, hence we do not provide them here. In both cases, the convergence of the overflow probability of the original downstream queue to that of the decomposed system happens uniformly and at least exponentially fast. Hence, if internal nodes in a network are capable of serving many flows, we can remove these nodes from consideration and the queueing behavior of other network nodes remains largely the same. In this way, we can, in many cases, simplify the analysis of a queueing network into that of a single queue, for which many analytical techniques are available in the literature. We also provide a number of numerical simulations that apply our decomposition method for different network configurations using different traffic models and actual input traces. Our numerical results confirm that removing nodes with large capacity will not change the performance of the network, e.g., the overflow probability at the downstream node or the end-to-end delay distribution for a particular traffic flow.

The rest of the paper is structured as follows. In Section II, we present some background material and formulate our problem. In Section III-A, we prove almost sure convergence, and also provide a result on the speed of convergence. In Section III-B, we discuss several different aspect of our

This work has been partially supported by the National Science Foundation through the Special Projects Award ANI-0099137 and the Indiana 21st Century Research and Technology Award 122000634.

theoretical results, and then briefly explore how to calculate the error, or the rate of convergence in Section III-C. Similar, but weaker convergence results for general (including non-regulated traffic) arrivals are presented in Section III-D. Next, in Section IV, we show numerical results to verify our simplifying methods under different network conditions. Finally in Section V, we conclude.

II. PRELIMINARY RESULTS AND PROBLEM DESCRIPTION

In this section, we present some preliminary results. We will first summarize the many-sources-asymptotic results from the literature, particularly an upper bound, which will serve as a starting point toward the development of our main results later in this paper. We then present our model and formulate our problem in detail.

A. Many-sources-asymptotic upper bound

Consider the queue shown in Figure 1. In this figure, for flow i , $A_i(s, t)$ and $D_i^N(s, t)$ represent the amount of traffic arrival and departure, respectively, during a time interval $[s, t)$ ($s, t \in \mathbb{Z}$). The server capacity of the queue is NC and $q^N(t)$ denotes the queue-length at time $t \in \mathbb{Z}$.

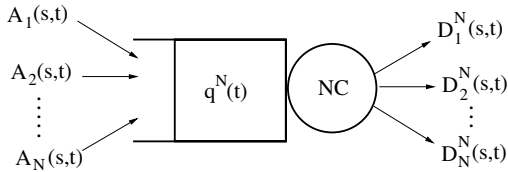


Fig. 1. Example of a queue fed by many sources

Throughout the paper, for simplicity of exposition, we assume that $A_i(s, t)$, $i = 1, 2, \dots, N$, are *i.i.d.* with stationary increments, unless otherwise noted. However, our results can easily be extended for non-*i.i.d.* arrivals [6]. For stability, we require that $\mathbb{E}\{A_i(-t, 0)\}/t := \lambda < C$. Then, assuming that the system starts at $-\infty$, the steady-state queue-length at time t can be expressed as

$$q^N(t) := \sup_{s \leq t} \left[\sum_{i=1}^N A_i(s, t) - CN(t-s) \right]. \quad (1)$$

Note that from the stationary increments property of $A_i(s, t)$, the distribution of $q^N(t)$ does not depend on t . We define

$$J_t(b) := \sup_{\theta} \left[\theta(Ct + b) - \log \mathbb{E}\{e^{\theta A_1(0, t)}\} \right]. \quad (2)$$

In this setup, we state the following many-sources-asymptotic upper bound from the work in [7].

Proposition 1: [many-sources-asymptotic upper bound] Suppose that

$$\liminf_{t \rightarrow \infty} J_t(0) / \log t > 0. \quad (3)$$

Then, we have, for any t ,

$$\limsup_{N \rightarrow \infty} \frac{1}{N} \log \mathbb{P}\{q^N(t) > Nb\} \leq -I(b), \quad b \geq 0, \quad (4)$$

where $I(b)$ is given by

$$I(b) := \inf_{t > 0} \sup_{\theta} \left[\theta(Ct + b) - \log \mathbb{E}\{e^{\theta A_1(0, t)}\} \right]. \quad (5)$$

□

The assumption (3) is shown to be more general than the one used in [8] in that (3) holds even for on-off sources with heavy-tailed on-time distributions (see [7]). For the classical treatment of the many-sources-asymptotic results, we refer to the papers [8], [9].

B. Problem description and model assumptions

We consider a two-stage queueing system shown in Figure 2. In this figure, the upstream queue (with queue-length $q^N(t)$) represents a node that is capable of serving a large number of traffic flows in a network, while the downstream queue (with queue-length $Q_I^N(t)$) could be a node with only a small capacity (for example, an output port of an edge router).

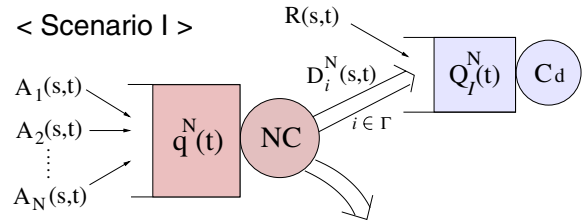


Fig. 2. Queueing network: Scenario I

< Scenario II >

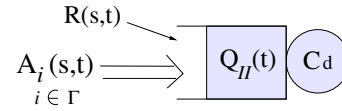


Fig. 3. Scenario II: a simplified version of Scenario I

In Scenario I shown in Figure 2, among the N flows, a *fixed subset** (not dependent on N) of the flows i ($i \in \Gamma$) after being served at the first (upstream) queue arrives to the downstream queue with interfering traffic $R(s, t)$, while the rest of flows depart the queueing system. We can thus write the steady-state queue-length at the downstream node as

$$Q_I^N(0) := \sup_{t \geq 0} \left[\sum_{i \in \Gamma} D_i^N(-t, 0) + R(-t, 0) - C_d t \right].$$

We are then interested in estimating the steady-state overflow probability $\mathbb{P}\{Q_I^N(0) > x\}$ for a given buffer level x . In order to do that, we consider a simple single-stage queueing system shown in Figure 3, a simplified version of the original two-stage queueing system in Figure 2. In Scenario II, the queue has the same interfering traffic $R(s, t)$ and the same service capacity C_d as that of Scenario I, except that the traffic arrival

*In general, Γ does not have to be a fixed subset for the results to hold. However, the results are more meaningful in the case when Γ constitutes a fixed set of flows (see Remark 1 in Section III-A).

of interest to the queue is now $A_i(s, t)$ instead of $D_i^N(s, t)$. Specifically, we write the steady-state queue-length in Scenario II as

$$Q_{II}(0) := \sup_{t \geq 0} \left[\sum_{i \in \Gamma} A_i(-t, 0) + R(-t, 0) - C_d t \right].$$

Thus, we obtain Scenario II if we remove the upstream queue in Scenario I (the queue with large capacity or bandwidth). Note that $Q_{II}(0)$ does not depend on N , while $Q_I^N(0)$ does. In the next section, we will relate aspects of the queue-length ($Q_{II}(0)$) in the simplified system to that of the downstream queue in the original system ($Q_I^N(0)$), when N is large.

III. MAIN RESULTS

A. Regulated traffic

We first assume that the input traffic arrivals $A_i(s, t)$ and the interfering traffic $R(s, t)$ are *regulated*; i.e., there exists a function $A^*(t)$ and $R^*(t)$ such that $A_i(s, s+t) \leq A^*(t)$ and $R(s, s+t) \leq R^*(t)$ for all $s, t > 0$. In this case, we can assume that, without loss of generality, the functions $A^*(t)$ and $R^*(t)$ are non-decreasing and subadditive [10]. Thus, $A^*(t)/t$ converges to its minimum value, and so does $R^*(t)/t$ (see Lemma 6.1.11 in [11]). Let a^* be this minimum value, i.e., we define

$$a^* := \lim_{t \rightarrow \infty} \frac{A^*(t)}{t} = \inf_{t > 0} \frac{A^*(t)}{t}. \quad (6)$$

We also define

$$r^* := \lim_{t \rightarrow \infty} \frac{R^*(t)}{t} = \inf_{t > 0} \frac{R^*(t)}{t}.$$

We will need the following assumption:

(A1): Let a^* and r^* be defined as above, and $|\Gamma|$ be the cardinality of the set Γ . Then, $a^* < C$ and $a^*|\Gamma| + r^* < C_d$.

The inequality conditions in (A1) were posed in order to exclude trivial cases. To see this, if $a^* \geq C$ and the traffic arrives according to the worst case $A^*(t)$, then we have from (6) that $A_i(-t, 0) \geq Ct$ for all t , and thus $q^N(0)$ in (1) grows without bound for any fixed N . The most common example of the bounding function $A^*(t)$ in the literature is the *dual leaky-bucket* type of regulator (see [12], [13]) with $A^*(t) = \min\{Pt, \gamma t + \sigma\}$, where $\gamma < C$.

We are now ready to provide our main result.

Theorem 1: Under assumption (A1), we have

$$\lim_{N \rightarrow \infty} Q_I^N(0) = Q_{II}(0), \quad \text{almost surely.} \quad (7)$$

Furthermore, the speed of convergence of $\mathbb{P}\{Q_I^N(0) > x\}$ to $\mathbb{P}\{Q_{II}(0) > x\}$ is uniformly at least exponentially fast in the sense that

$$\limsup_{N \rightarrow \infty} \frac{1}{N} \log \left(\sup_{x \geq 0} \left| \mathbb{P}\{Q_I^N(0) > x\} - \mathbb{P}\{Q_{II}(0) > x\} \right| \right) \leq -I(0), \quad (8)$$

where x is the buffer level and $I(0)$ is from (5). \square

Proof: We first observe from assumption (A1) that there exists a $t_1 > 0$ such that, for all i ,

$$A_i(s, s+t) < Ct \quad \text{for all } s \text{ and } t \geq t_1. \quad (9)$$

Similarly, there exists a t_2 such that $\sum_{i \in \Gamma} A_i(-t, 0) + R(-t, 0) < C_d t$ for $t \geq t_2$, since $\sum_{i \in \Gamma} A_i(-t, 0) + R(-t, 0) \leq |\Gamma|A^*(t) + R^*(t)$. Thus, we have

$$\begin{aligned} Q_{II}(0) &= \sup_{t \geq 0} \left[\sum_{i \in \Gamma} A_i(-t, 0) + R(-t, 0) - C_d t \right] \\ &= \sup_{0 \leq t \leq t^*} \left[\sum_{i \in \Gamma} A_i(-t, 0) + R(-t, 0) - C_d t \right] \end{aligned} \quad (10)$$

for any $t^* \geq t_2$, since the inside of the bracket becomes negative for $t > t_2$.

We let $q_i^N(t)$ denote the workload in $q^N(t)$ due to flow i . Then, the following claim is straightforward to show from the regularity assumption.

Claim: $q_i^N(-t) \leq A^*(t_1)$ for all i and t , where t_1 is given by the relation (9).

Proof of Claim: We note that the maximum busy period of the queue $q^N(t)$ is bounded by t_1 . To see this, suppose that there exists a busy period longer than t_1 . Then, at the end of this busy period, the total amount of traffic arrival should be larger than the capacity NCt_1 , otherwise, the busy period ends earlier. So we have $\sum_{i=1}^N A_i(s, s+t_1) \geq NCt_1$ for some s , but this contradicts (9). Now, note that the workload due to the flow $i \in \Gamma$ at time $-t$ should be bounded by the case that all the traffic from flow i has been accumulated during the maximum busy period t_1 . Hence, we have $q_i^N(-t) \leq A_i(-t - t_1, -t)$ and from the definition of $A^*(t)$, $A_i(-t - t_1, -t) \leq A^*(t_1)$. Thus, the claim follows. \square

Note that from the above claim, for all t , we have

$$\begin{aligned} D_i^N(-t, 0) &= A_i(-t, 0) + q_i^N(-t) - q_i^N(0) \\ &\leq A^*(t) + A^*(t_1). \end{aligned}$$

This implies that the departure traffic $D_i^N(-t, 0)$ is also bounded by some function $D^*(t) = A^*(t) + A^*(t_1)$ and $\lim_{t \rightarrow \infty} D^*(t)/t = a^*$. Hence, as in (10), there exists a $t_3 > 0$ such that

$$\begin{aligned} Q_I^N(0) &= \sup_{t \geq 0} \left[\sum_{i \in \Gamma} D_i(-t, 0) + R(-t, 0) - C_d t \right] \\ &= \sup_{0 \leq t \leq t^*} \left[\sum_{i \in \Gamma} D_i(-t, 0) + R(-t, 0) - C_d t \right] \end{aligned} \quad (11)$$

for any $t^* \geq t_3$.

Now choose $t_0 = \max\{t_2, t_3\}$. Then, from (10) and (11),

we have, for any $N > 0$,

$$\begin{aligned}
& \left| Q_I^N(0) - Q_{II}(0) \right| \\
&= \left| \sup_{t \geq 0} \left[\sum_{i \in \Gamma} D_i^N(-t, 0) + R(-t, 0) - C_d t \right] \right. \\
&\quad \left. - \sup_{t \geq 0} \left[\sum_{i \in \Gamma} A_i(-t, 0) + R(-t, 0) - C_d t \right] \right| \\
&= \left| \sup_{0 \leq t \leq t_0} \left[\sum_{i \in \Gamma} D_i^N(-t, 0) + R(-t, 0) - C_d t \right] \right. \\
&\quad \left. - \sup_{0 \leq t \leq t_0} \left[\sum_{i \in \Gamma} A_i(-t, 0) + R(-t, 0) - C_d t \right] \right| \\
&\leq \sup_{0 \leq t \leq t_0} \left| \sum_{i \in \Gamma} D_i^N(-t, 0) - \sum_{i \in \Gamma} A_i(-t, 0) \right| \\
&= \sup_{0 \leq t \leq t_0} \left| \sum_{i \in \Gamma} q_i^N(-t) - \sum_{i \in \Gamma} q_i^N(0) \right| \\
&\leq \sup_{0 \leq t \leq t_0} \sum_{i \in \Gamma} q_i^N(-t) \\
&\leq \sup_{0 \leq t \leq t_0} q^N(-t). \tag{12}
\end{aligned}$$

Since the distribution of $q^N(t)$ is independent of t , we have

$$\begin{aligned}
& \mathbb{P} \left\{ |Q_I^N(0) - Q_{II}(0)| > 0 \right\} \leq \mathbb{P} \left\{ \sup_{0 \leq t \leq t_0} q^N(-t) > 0 \right\} \\
&\leq \sum_{t=0}^{t_0} \mathbb{P} \{ q^N(-t) > 0 \} = (t_0 + 1) \mathbb{P} \{ q^N(0) > 0 \}. \tag{13}
\end{aligned}$$

Since assumption (A1) clearly satisfies (3), taking log, dividing by N , and then taking the lim sup on both sides of (13) yields

$$\begin{aligned}
& \limsup_{N \rightarrow \infty} \frac{1}{N} \log \mathbb{P} \left\{ |Q_I^N(0) - Q_{II}(0)| > 0 \right\} \\
&\leq \limsup_{N \rightarrow \infty} \frac{1}{N} \log(t_0 + 1) \mathbb{P} \{ q^N(0) > 0 \} \\
&= \limsup_{N \rightarrow \infty} \frac{1}{N} \log \mathbb{P} \{ q^N(0) > 0 \} \\
&\leq -I(0), \tag{14}
\end{aligned}$$

where the last inequality comes from (4).

Finally, for $t \in \mathbb{Z}_+$, it has been shown that [8], [9]

$$\begin{aligned}
I(0) &:= \inf_{t > 0} \sup_{\theta} [\theta C t - \log \mathbb{E} \{ e^{\theta A_1(0,t)} \}] \\
&= \sup_{\theta} [\theta C - \log \mathbb{E} \{ e^{\theta A_1(0,1)} \}]. \tag{15}
\end{aligned}$$

From the stability condition, i.e., $C > \mathbb{E} \{ A_1(0,1) \} = \lambda$, and the convexity of the function $\log \mathbb{E} \{ e^{\theta A_1(0,1)} \}$ in θ , $I(0)$ in (15) is always positive. Thus, from (14), we can write

$$\begin{aligned}
& \sum_{N=1}^{\infty} \mathbb{P} \{ |Q_I^N(0) - Q_{II}(0)| > 0 \} \\
&\leq \sum_{N=1}^{\infty} \exp(-NI(0) + o(N)) < \infty.
\end{aligned}$$

Hence, (7) follows from the Borel-Cantelli lemma, and (8) follows by noting that, for any $x > 0$,

$$\begin{aligned}
& \left| \mathbb{P} \{ Q_I^N(0) > x \} - \mathbb{P} \{ Q_{II}(0) > x \} \right| \\
&\leq \mathbb{P} \{ |Q_I^N(0) - Q_{II}(0)| > 0 \}.
\end{aligned}$$

This completes the proof of Theorem 1. \blacksquare

Remark 1: In Theorem 1, our primary emphasis is on the case that Γ is fixed, for which $\mathbb{P} \{ Q_I^N(0) > x \}$ does not go to zero as N increases. However, our results remain unchanged if $|\Gamma|$, the number of flows feeding the downstream queue, also increases as N increases provided that the capacity of the downstream queue C_d is also scaled to ensure stability. For example, if the interfering traffic $R(s, t) = 0$ and $\Gamma = \{1, 2, \dots, N\}$, we note that

$$\begin{aligned}
& \mathbb{P} \{ Q_I^N(0) > 0 \} \leq \mathbb{P} \{ Q_{II}(0) > 0 \} \\
&\quad + \left| \mathbb{P} \{ Q_I^N(0) > 0 \} - \mathbb{P} \{ Q_{II}(0) > 0 \} \right|. \tag{16}
\end{aligned}$$

The first term of the RHS of (16) decreases to zero exponentially fast with rate $I(0) > 0$, and so does the second term from (8). Following the same approach as in the proof of Theorem 1, we see that the queue-length at the downstream queue also decreases to zero almost surely. However, this case is less interesting because both $Q_I^N(0)$ and $Q_{II}(0)$ go to zero almost surely.

In the proof of Theorem 1, the fact that $I(0)$ is positive plays a crucial role in establishing the convergence of $Q_I^N(t)$ to $Q_{II}(t)$. If we set $b = 0$ in (4), we see that $q^N(t)$, where the aggregation takes place, also decreases to zero almost surely by the Borel-Cantelli lemma. Hence, the departure traffic flows are more likely to be identical to arrival traffic flows, as the system size increases. An interesting interpretation of these observations is the following: *If congestion occurs in a network, the node under congestion will either be unstable or carry a small number of traffic flows at high utilization.*

Although the convergence of $\mathbb{P} \{ q^N(0) > 0 \}$ to zero is necessary to prove Theorem 1, we will show in Section IV that removing nodes and thus simplifying network analysis still remains in effect even if the actual value of $\mathbb{P} \{ q^N(0) > 0 \}$ is not very small.

B. Simplifying network analysis

Note that relation (8) in Theorem 1 allows us to write

$$\begin{aligned}
& \sup_{x \geq 0} \left| \mathbb{P} \{ Q_I^N(0) > x \} - \mathbb{P} \{ Q_{II}(0) > x \} \right| \\
&\leq \exp(-NI(0) + o(N)).
\end{aligned}$$

Thus, for any target QoS ($\mathbb{P} \{ Q_I^N(0) > x \}$) in the original two-stage queueing system, we can replace it by a simpler system in which the first node has been removed and the maximum error is less than $\exp(-NI + o(N))$, which decreases exponentially fast. If we extend this idea to an entire network, we can say that it is safe to remove nodes that have the capacity to carry a large number of flows from consideration.

This simplification greatly reduces the burden of analyzing a queueing network, and helps simplify the difficult end-to-end QoS estimation problem into potentially a simpler single-stage queueing problem.

From a measurement point of view this type of decomposition could also be useful. At one extreme, if one is able to access any point inside a network to obtain a sample path of a traffic flow, we can then simply measure the traffic and compute the QoS at the node in hand (for example, we could employ the “measurement-analytic” approach in [14]). However, measurements in the interior of a network could be costly or even practically infeasible. To avoid this problem, we could use the decomposition idea (via Theorem 1) and measure the traffic only at the network periphery to make QoS predictions within the network.

Another feature of our simplifying approach is that we do not have to maintain any per-flow information at the interior nodes with large capacity, even if each flow requires stringent end-to-end QoS. Once we remove these nodes to simplify network analysis, they become invisible to us. In other words, inside a network, we do not really care where traffic flows are coming from and heading to, as long as all the nodes are kept stable. In Section IV, we will verify via simulations that our simplifying method works well under a variety of different network configurations and different traffic types.

C. Estimating the constant $I(0)$

In this section, we briefly examine how we can estimate the error $e^{-NI(0)}$, and how small it can be. As we see in (15), the constant $I(0)$ depends on the distribution of the traffic flow. Consequently, it is not obvious how one would be able to estimate the constant $I(0)$. Recall that this error term is an upper bound on the *maximum* difference in the overflow probability between two scenarios. Thus, the actual error for a specific buffer level could be significantly smaller than the maximum error. Nevertheless, it turns out that we are able to find a lower bound on the constant $I(0)$ (hence, an upper bound on the error term $e^{-NI(0)}$ as a first-order approximation) in terms of the mean and the peak rate of the traffic that are relatively easy to estimate.

Let P be the peak rate of a traffic flow $A_i(s, t)$. Observe that

$$\begin{aligned} I(0) &:= \inf_{t>0} \sup_{\theta} [\theta C t - \log \mathbb{E}\{e^{\theta A_1(0,t)}\}] \\ &= \inf_{t>0} \sup_{\theta} [\theta C - \log \mathbb{E}\{e^{\theta \frac{A_1(0,t)}{t}}\}]. \end{aligned}$$

Since $0 \leq \frac{A_1(0,t)}{t} \leq Pt$ with $\mathbb{E}\{\frac{A_1(0,t)}{t}\} = \lambda$, using Corollary 2.4.5 in [11], we have

$$\mathbb{E}\{e^{\theta \frac{A_1(0,t)}{t}}\} \leq \frac{\lambda}{P} e^{\theta P} + (1 - \frac{\lambda}{P}). \quad (17)$$

Thus, we get

$$\begin{aligned} &\sup_{\theta} [\theta C - \log \mathbb{E}\{e^{\theta \frac{A_1(0,t)}{t}}\}] \\ &\geq \sup_{\theta} [\theta C - \log(\frac{\lambda}{P} e^{\theta P} + (1 - \frac{\lambda}{P}))] = \sup_{\theta} h(\theta), \end{aligned}$$

where

$$h(\theta) := \theta C - \log(\frac{\lambda}{P} e^{\theta P} + (1 - \frac{\lambda}{P})).$$

It is easy to see that $h(\cdot)$ is a concave function and $h'(0) = C - \lambda > 0$. Thus, by direct calculation, we have

$$I(0) \geq \frac{C}{P} \log(\frac{C}{\lambda}) + (1 - \frac{C}{P}) \log(\frac{P - C}{P - \lambda}), \quad (18)$$

where $\lambda < C < P$.

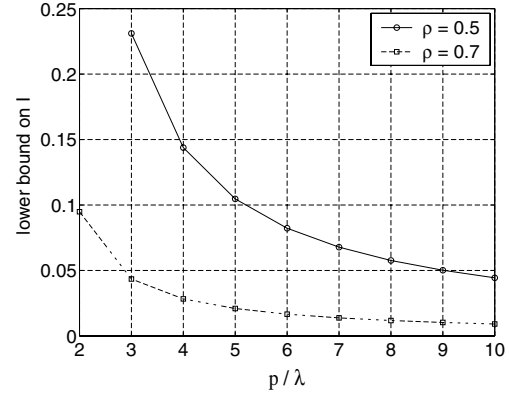


Fig. 4. RHS of (18) as a function of P/λ for two different utilizations

Figure 4 shows the right hand side of (18) as a function of the ratio between the peak rate and the mean rate (P/λ) for two different utilizations. For example, if $P = 5\lambda$ and $\rho = \lambda/C = (N\lambda)/(NC) = 0.5$, then Figure 4 shows that $I \geq 0.1$. Hence, if there are 200 such sources being multiplexed, then

$$\exp(-NI + o(N)) \leq \exp(-200 \times 0.1 + o(200)) \approx 2 \times 10^{-9}.$$

Thus, the maximum error on the overflow probability at the downstream queue incurred by traversing the queue with 200 such flows is at most 2×10^{-9} approximately. Note that the equality in (17) corresponds to the special case that the traffic rate is either the peak rate with probability λ/P or zero with probability $1 - \lambda/P$. If we have more information on the marginal distribution of the traffic, e.g., the variance, we can then find a sharper bound on the constant $I(0)$.

D. General (non-regulated) traffic

So far, we have assumed that the traffic flows of interest are regulated. For non-regulated traffic arrivals, i.e., there is no bounding function $A^*(t)$, things are more complicated. In this case, the difference between $Q_I^N(0)$ and $Q_{II}(0)$ now depends on the entire past history of $q^N(t)$, i.e., $\sup_{t \geq 0} q^N(-t)$, rather than $\sup_{0 \leq t \leq t_0} q^N(-t)$ as we see in (12) for regulated traffic arrivals. However, we are still able to show that the distribution of $Q_I^N(t)$ converges uniformly to that of $Q_{II}(t)$, and that the convergence happens at least exponentially fast. Moreover, this holds also for non-*i.i.d.* traffic arrivals. However, the proofs of these results are quite technical and beyond the scope of this paper. Hence, we simply state the results here and provide a reference for the proofs.

Theorem 2: [General traffic arrival case from [6]] Under appropriate (yet very general) assumptions[†], we have

$$\lim_{N \rightarrow \infty} |\mathbb{P}\{Q_I^N(0) > x\} - \mathbb{P}\{Q_{II}(0) > x\}| = 0,$$

uniformly in $x > 0$. In addition, there exists a positive constant J^* ($0 < J^* < I(0)$) such that

$$\limsup_{N \rightarrow \infty} \frac{1}{N} \log \left(\sup_{x \geq 0} |\mathbb{P}\{Q_I^N(0) > x\} - \mathbb{P}\{Q_{II}(0) > x\}| \right) \leq -J^*.$$

□

Note that, since Theorem 2 also holds for non-regulated traffic arrivals, we are unable to prove the almost sure convergence as in (7) of Theorem 1. However, we have convergence in distribution and show that the speed of convergence is still exponential, albeit with a rate little slower than $I(0)$. We note that in this case, our simplifying approach still remains intact. The overflow probability at a downstream queue can be replaced with that of a single queue, in a manner similar to the regulated traffic arrivals. For more details on the required assumptions and proofs, see [6].

IV. NUMERICAL RESULTS

In this section, we conduct an extensive set of simulations to numerically test the performance of our simplifying approach for different network scenarios. To represent a realistic system as closely as possible, our simulator is a packet-based (versus fluid), event-driven one, and consists of a number of queues with arbitrary input traffic arrivals we designate. We are able to monitor every packet anywhere in the network, each of which is associated with its packet size and time stamp. The QoS metrics we consider are the overflow probability at a particular node we are monitoring, and the end-to-end delay for a particular traffic flow. To be specific, we record queue sizes of a node just before each packet enters the node, and then take average (over the number of packets) to estimate the overflow probability at that node. Similarly, we measure the end-to-end delay per packet for the traffic flow we are monitoring, and then take the average. For calculating the end-to-end delay of each packet, we only consider the queueing delay at each node and ignore the propagation delay, since the propagation delay is a constant value.

A. Two-stage queueing network

1) *Voice traffic sources:* We first apply our simplifying approach to a simple two-stage queueing network, as shown in Figure 2. The arrival traffic to the first queue consists of N multiplexed voice traffic sources. For a single voice traffic, we use an on-off Markov Modulated Deterministic (MMD) model. This is a continuous-time two-state Markov chain, which generates traffic at a constant rate (64 Kbps) while in the “on” state, and no traffic while in the “off” state. The sojourn time

[†]e.g., the usual many-sources asymptotic assumptions and the finite moment of a queue with single input. See [6] for details.

in the “on” and “off” states are exponentially distributed with mean 352 msec and 650 msec, respectively [15]. To simulate the constant rate during the on state using our simulator, we generate a series of packets with constant interarrival time where each packet has a fixed size of 53 bytes[‡].

We fix the number of voice traffic flows feeding the downstream queue as 10, i.e., $|\Gamma| = 10$. However, we vary the number of voice traffic flows at the upstream queue (N) while keeping the utilization there the same, and then, we vary the utilization at the upstream queue while keeping N the same. These experiments allow us to investigate the effect of different levels of aggregation and utilization at the upstream queue on the behavior of the downstream queue. For interfering traffic at the downstream queue, $R(s, t)$, we generate fractional Gaussian noise sequence with Hurst parameter $H = 0.8$, such that its mean is chosen to be equal to that of 10 multiplexed voice traffic flows. We choose the capacity of the downstream queue such that its utilization is 0.8.

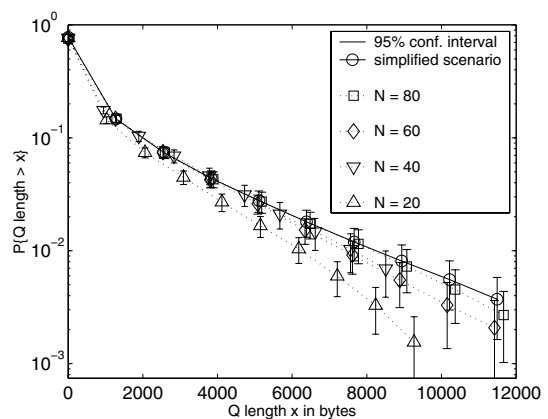


Fig. 5. Buffer overflow probability at the downstream queue for different N at the upstream queue. ρ is fixed to 0.8.

Figure 5 shows the different overflow probabilities at the downstream queue for different values of N at the upstream queue. The utilization, ρ , at the upstream queue remains $\rho = NC/N\lambda = 0.8$, where $\lambda = 22.48$ Kbps is the mean traffic rate of a single voice traffic flow. We ran many independent experiments, and here report their average with 95% confidence interval for each point in the figure. As N , the number of aggregated traffic flows, increases, we see that the buffer overflow probability at the downstream queue approaches to that of a single queue obtained by removing the upstream queue (simplified scenario). In particular, when N is equal to 80, the curve is nearly indistinguishable from the simplified scenario case.

We now fix the number of multiplexed voice traffic flows N at $N = 60$, and then vary the utilization (i.e., the capacity) at the upstream queue. As Figure 6 clearly shows, we observe that removing the upstream nodes hardly affects the queueing performance at the downstream node, even at high utilizations.

[‡]The packet size has been chosen arbitrarily and any other packet size would suffice.

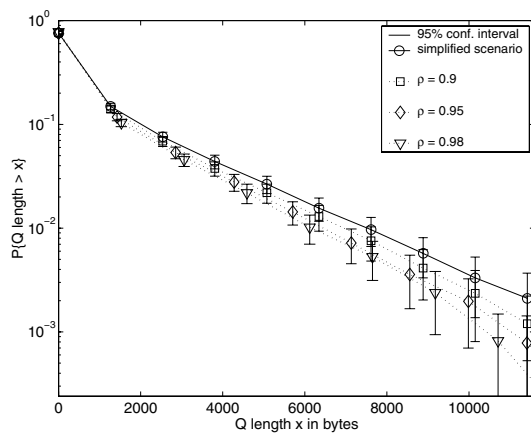
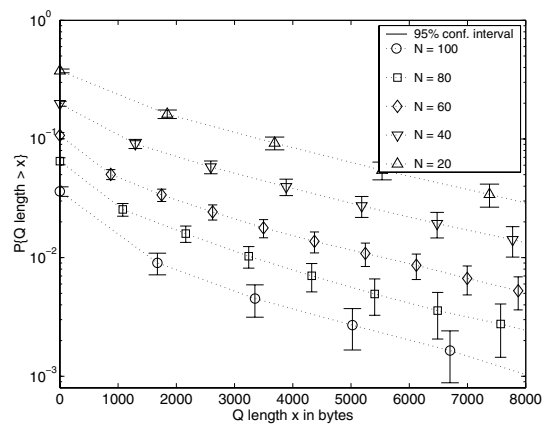


Fig. 6. Buffer overflow probability at the downstream queue for different utilization ρ at the upstream queue. N is fixed to 60.

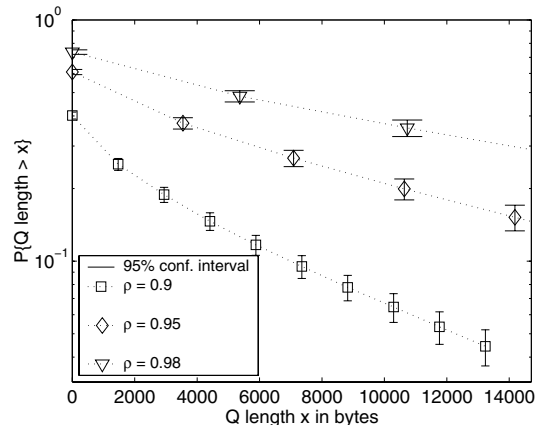
From the proof of Theorem 1, we have seen that as N increases, the queue-length at the upstream queue decreases exponentially fast. Figure 7 shows different overflow probability curves at the upstream node for (a) different N and (b) different utilizations ρ . Clearly, as N increases, the overflow probability decreases over all buffer levels. However, if we carefully compare Figure 7 to Figure 5 or Figure 6 for the same buffer levels, it turns out that the overflow probability at the upstream queue is not so small, and sometimes even larger than that of the downstream queue. Thus, although we used the fact that $q^N(t)$ decreases to zero in order to prove our theorem, it does not appear to be necessary for the decomposition to work well.

2) *Ethernet (LAN) traffic trace*: In this section, we repeat the same set of simulations as in Section IV-A.1, except that we now use an actual Ethernet (LAN) trace[§] instead of a traffic model. The Ethernet trace consists of a sequence of arrival times and packet sizes. To obtain each multiplexed Ethernet traffic, we generate different traffic traces from the original one with random offset (random starting points), and then superpose these traces. Since Ethernet traffic is known to be much burstier than the voice traffic [16], the question is whether we can still remove the upstream node in this case[¶]. Figures 8 and 9 show the overflow probability at the downstream queue under different scenarios. As N increases and/or ρ decreases, the curve becomes closer to the case of the simplified scenario, but the “convergence” happens slower than in the voice traffic case, due to the high level of burstiness of the Ethernet traffic.

Figure 10 presents the overflow probability at the upstream queue with N multiplexed Ethernet traffic arrivals. As in the voice traffic case, the overflow probability decreases as N increases. However, in contrast to Figure 7 (a), the probability that the buffer is non-empty stays around a fixed point as N



(a) different N



(b) different utilization ρ

Fig. 7. Buffer overflow probability at the upstream queue

varies, while for the voice traffic case it also decreases as N increases (see Figure 7 (a)). Remember that $\mathbb{P}\{q^N(t) > 0\}$ decreases to zero if $I(0) > 0$. Also, the inequality in (18) tells us that $I(0) > 0$ whenever there exists a peak rate (even for continuous time traffic models). Since there is a peak rate (“on” state rate) for a *single* voice traffic, Figure 7 (a) is to be expected. However, for the Ethernet traffic trace, there is no pre-determined peak rate and thus $\mathbb{P}\{q^N(t) > 0\}$ does not go to zero in the range of N shown in Figure 10. Nonetheless, we point out that removing the upstream node still works well for most cases as shown in Figures 8 and 9. We also observed similar results for other traffic arrivals, e.g, MPEG encoded video trace, Auto-Regressive models, etc, which we do not present here for space considerations.

We next investigate how different utilizations and levels of aggregation quantitatively affect the actual error of the overflow probability between two scenarios. For a given two-stage queueing network, we test if the following relation

$$|\mathbb{P}\{Q_I^N(0) > x\} - \mathbb{P}\{Q_{II}(0) > x\}| \leq K \mathbb{P}\{Q_{II}(0) > x\} \quad (19)$$

holds for all buffer level x . Here, the constant K corresponds to some error margin around the target QoS. If (19) holds for $K = 0.1$, it means that the error between the two scenarios

[§]trace file can be obtained at “<http://ita.ee.lbl.gov/traces/BC-Oct89Ext.TL.Z>”

[¶]Although Ethernet trace was in fact obtained from a closed-loop network, it is simply used as one of ‘bursty’ traffic sources for simulating open-loop queueing networks in this section.

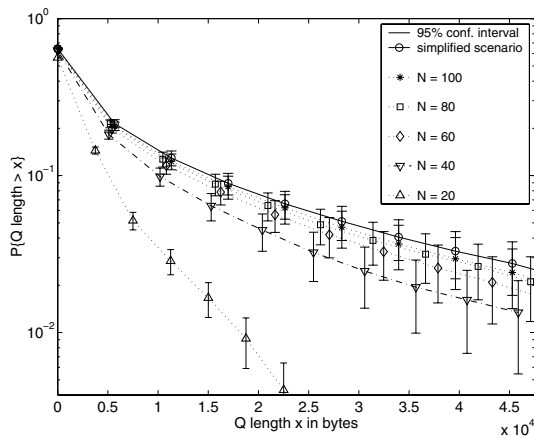


Fig. 8. Buffer overflow probability at the downstream queue for different N . Utilization at the upstream queue is fixed to $\rho = 0.7$

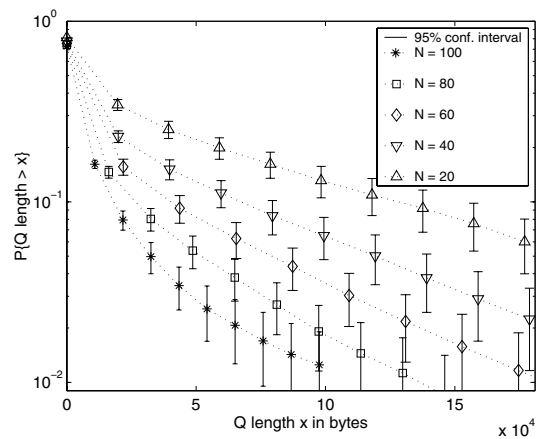


Fig. 10. Buffer overflow probability at the upstream queue for different N

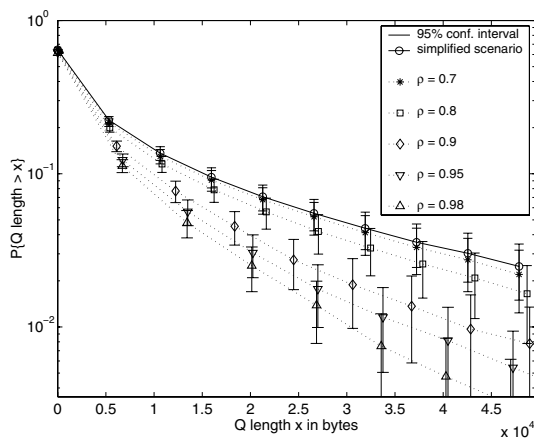


Fig. 9. Buffer overflow probability at the downstream queue for different utilization ρ . N is fixed to 60.

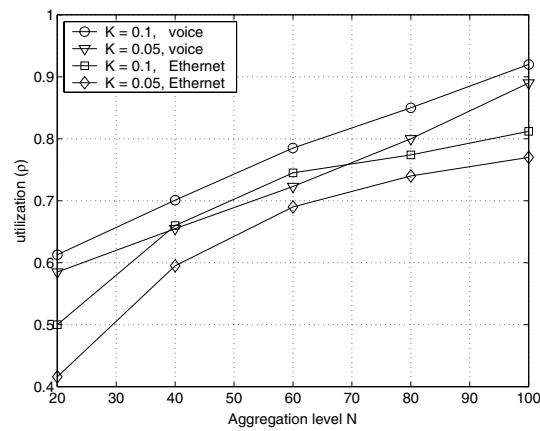


Fig. 11. Relation between N and ρ for given error constraint (19) for voice and Ethernet traffic

differs by only a 10% margin. For each traffic arrival and each N , we gradually decrease the utilization ρ , i.e., increases the capacity at the upstream queue, until (19) holds for *all* buffer levels used in the simulation. We record the largest utilization value that satisfies (19) for all x . Figure 11 depicts the results of these tests for voice and Ethernet traffic. As the aggregation level (N) increases, the maximum possible utilization for (19) to hold also increases. For example, in Figure 11, for the Ethernet traffic case when $N = 60$ (recall that only 10 flows feed the downstream queue), we see that about 70% (or less) utilization at the upstream queue guarantees that the error is less than 5% of the overflow probability of the simplified scenario, regardless of the buffer level x .

B. Multi-stage queueing network

In this section, we apply our simplifying techniques to a multi-stage queueing network. Consider the queueing network shown in Figure 12. In this figure, we assume that each node represents a queue with a certain capacity, where the numbers in parentheses stand for the capacity or the bandwidth of each node in Mbps. Nodes 5, 6, and 7 correspond to high-speed

links (e.g., core routers), while the other nodes correspond to lower capacity links (e.g., edge routers) serving only a small fraction of flows. Traffic types and their routes are summarized in Table I.

TABLE I
DESCRIPTION OF TRAFFIC FLOWS IN FIGURE 12

aggregate flow	traffic type	routes
S1	10 Ethernet	1 - 5 - 7 - 4 - 8
S2	100 voice or 6 MPEG video traffic	5 - 7
S3	10 fGn	1 - 5 - 6 - 2
S4	5 MMPP	6 - 7 - 4 - 9

The number of multiplexed traffic flows is chosen such that the mean rates of the traffic flows S1, S3, and S4 are approximately the same, while the mean rate of S2 is about 20 times larger than the others. The resulting utilization of each node varies between 0.7 and 0.9 for each case reported here. We then remove node 5 from the original scenario shown in Figure 12 and record the overflow probability at node 7. Figure 13 shows the resulting simplified scenario. Contrary

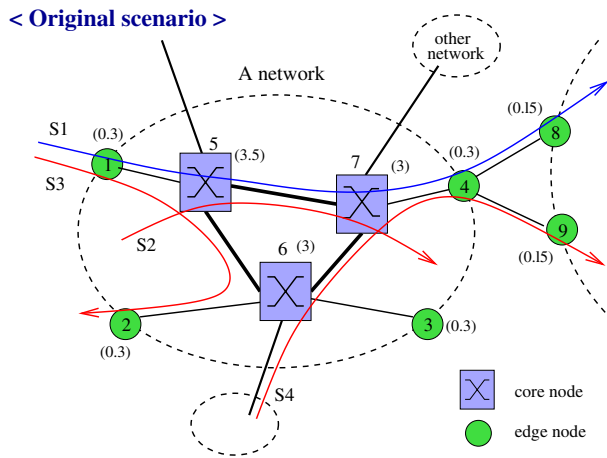


Fig. 12. Multi-stage queuing network. The numbers in parentheses represent capacity at the nodes in Mbps.

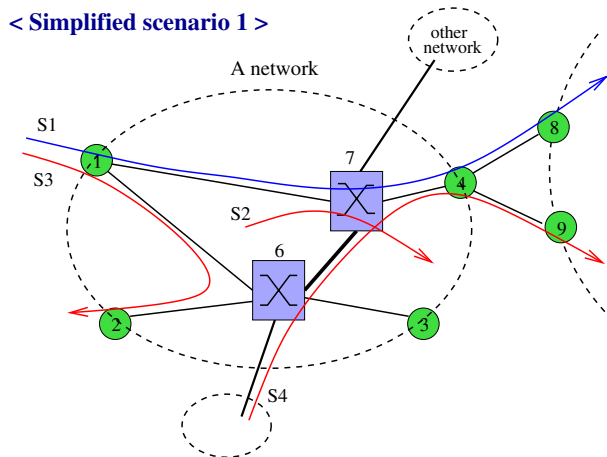
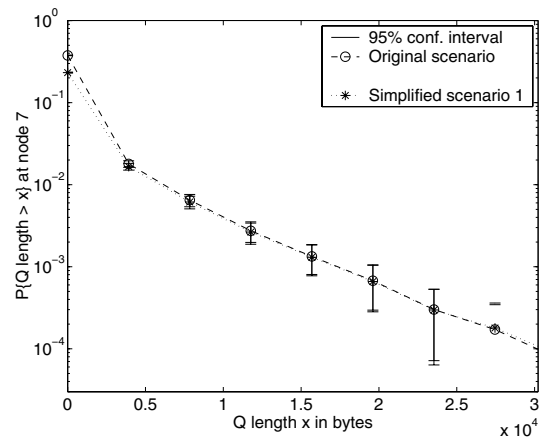


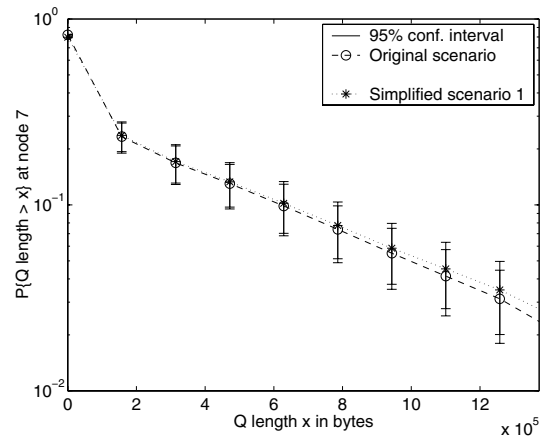
Fig. 13. Simplified version of Figure 12 after removing node 5. We are interested in the overflow probability at node 7

to the previous two-stage queuing network, in which the capacity of the downstream queue is much smaller than the upstream queue, note here that the capacity of node 7 is also large and comparable to that of node 5. Figure 14 shows the overflow probability at node 7 for the original scenario and the simplified scenario 1, while the interfering traffic S2 constitutes 100 multiplexed voice traffic flows for Figure 14(a) and 6 multiplexed MPEG video traffic flows for Figure 14(b). Since the capacity of node 5 is large, the figures show that removing node 5 does not affect the overflow probability at the downstream node in either case.

Figures 14(a)-(b) also have several points worth mentioning. Since most traffic flows at node 5 arrive to node 7, we expect from Theorem 1 (or Theorem 2) that the overflow probability at node 7 itself goes to zero, as the number of multiplexed flows increases at the previous node. Therefore, when S2 constitutes 100 multiplexed voice traffic flows, the overflow probability at node 7 is very small. In this regard, Figure 14(a) implies that removing the upstream queue with large capacity works well even for estimating small overflow probability at



(a) S2 is voice traffic



(b) S2 is MPEG video traffic

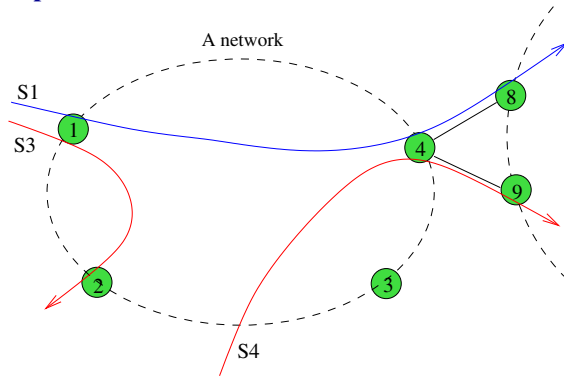
Fig. 14. Buffer overflow probability at node 7 for different scenarios

the downstream queue. Now, when S2 corresponds to 6 multiplexed MPEG video traffic flows, the overflow probability at node 7 is quite large, as we see in Figure 14(b), due to the burstiness of the video traffic. In this case, even though the capacity at node 5 is large, the number of multiplexed flows at the node is small, and the load at node 5 is quite high (approximately 0.76, from the simulation), we can still remove node 5 to estimate the overflow probability at the downstream node without incurring much error.

We will next remove multiple nodes from the original network and then monitor the queuing behavior at node 8 and the end-to-end delay for flow S1 (Ethernet traffic). Figure 15 shows two different simplified scenarios we now consider. First, we remove nodes 5, 6 and 7, which have much larger capacity than the others, to obtain the simplified scenario 2. We further remove every node except the last one for flow S1 to obtain the simplified scenario 3. (Note that nodes 1 and 4 have just twice the capacity as the node 8.) Since we are interested in flow S1, flow S2 acts as a multitude of interfering traffic flows inside the network.

Figures 16 and 17 show the overflow probability at node 8 and the end-to-end delay distribution for flow S1 when the

< Simplified scenario 2 >



< Simplified scenario 3 >

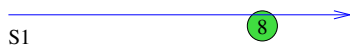


Fig. 15. Simplified scenario 2 and Simplified scenario 3 of Figure 12 after removing multiple nodes

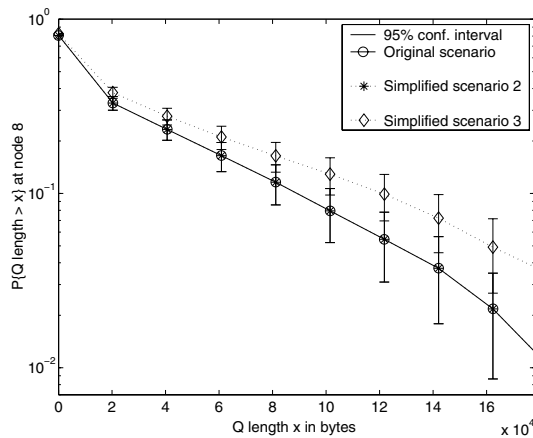


Fig. 16. Buffer overflow probability at node 8 for different scenarios. S2 is 100 multiplexed voice traffic.

crossing traffic S2 corresponds to 100 multiplexed voice traffic flows. In both figures, the curves for simplified scenario 2 are almost identical to the original scenario cases. In other words, removing nodes with large capacity has no effect on the performance of the queue at node 8 and the end-to-end delay for flow S1. This is because flow S1 comprises only a very small portion (about 5%) of total traffic flows feeding nodes 5 and 7. Further, we note that flow S2 (voice traffic), which dominates the traffic at nodes 5 and 7, is much smoother than flow S1. Hence, flow S1 virtually is not altered at all as it goes across nodes 5 and 7. If we also remove nodes 1 and 4 (edge nodes) and thus only node 8 remains, the queueing behavior at node 8 is a little different from (but still a reasonable approximation to) the original scenario, as can be seen in Figure 16. Obviously, the capacity of nodes 1 or 4 is comparable to that of node 8, and thus removing these nodes clearly affects the traffic flow S1. However, the end-to-end delay does not change greatly even if we remove

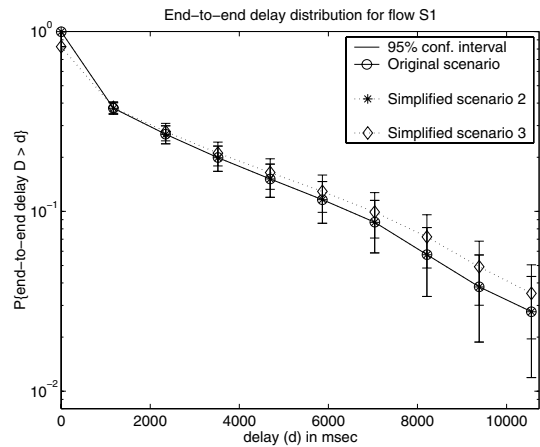


Fig. 17. End-to-end delay distribution of flow S1 for different scenarios. S2 is 100 multiplexed voice traffic.

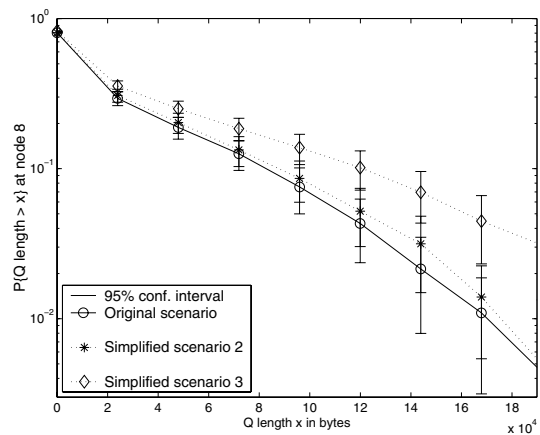


Fig. 18. Buffer overflow probability at node 8 for different scenarios. S2 is 6 multiplexed MPEG video traffic.

all the nodes except node 8 (as shown in Figure 17). This implies that, in the original scenario, most of the end-to-end delay for flow S1 is due to the delay at node 8. Note that at nodes 1 and 4, flow S1 is multiplexed with other traffic flows that are somewhat less burstier than flow S1, while there is only flow S1 itself, which is quite bursty, at node 8. Since the capacity of node 8 is smaller than any other nodes, it appears that packets from flow S1 suffer the largest delay at node 8.

Figures 18 and 19 show similar curves to Figures 16 and 17, except that the interfering traffic S2 is now 6 multiplexed MPEG video traffic. Observe that removing nodes 5, 6, and 7 changes the overflow probability at node 8 a little, while the end-to-end delay still remains unchanged. Since MPEG video traffic is significantly burstier than the voice traffic, removing nodes 5–7 contributes to some extent to a small change (compared to Figure 16) in the overflow probability curve. However, note that the capacity of nodes 5 and 7 is at least ten times larger than that of nodes 1, 4 or 8, and that the delay is inversely proportional to the capacity for the same queue length. As a result, this large capacity absorbs that small difference in the overflow probabilities and makes the delay

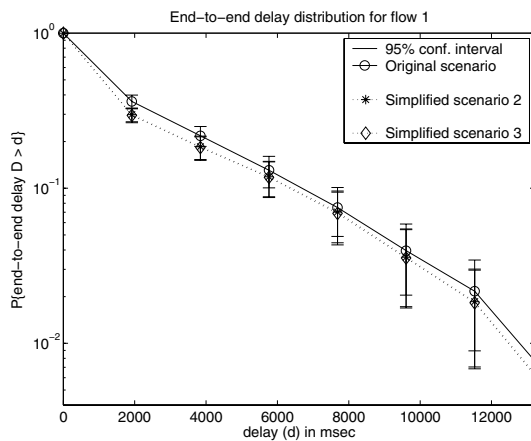


Fig. 19. End-to-end delay distribution of flow S1 for different scenarios. S2 is 6 multiplexed MPEG video traffic.

distributions almost the same. Now note that the simplified scenario 3 results in a somewhat larger (but still tolerable) error in estimating the overflow probability curve at node 8 (see Figure 18). However, the end-to-end delay distribution remains largely unaffected, as shown in Figure 19.

Through these and other simulations, we have observed the following: (1) For those nodes with capacity large enough to serve a large number of flows (without a few flows dominating), decomposing the network by removing these nodes will not affect the overflow probability at other nodes in the network (nor the end-to-end delay distribution) regardless of the traffic characteristics. (2) Now consider the case when we decompose the network by removing nodes that can multiplex only a small number of flows (e.g., less than 50 flows) and/or have very high utilization. In this case, other factors, such as burstiness of the traffic flows and the interfering traffic characteristics, come into play in determining the size of the error caused by the decomposition. (3) When even a small to moderate number of flows are multiplexed at a node, the streams routed to the different downstream nodes are not significantly altered, as long as each routed stream does not constitute a large fraction of the multiplexed flows. This idea enables us to analyze the performance of any particular queue in the network by decomposing the entire network into a single queue, as was done for node 8 in the simplified scenario 3.

V. CONCLUSION

In this paper we have shown how to simplify a queueing network for large bandwidth systems. We proved that, under the regularity assumption of traffic arrivals, the queue-length at the downstream queue converges almost surely to that of a

single queue obtained by removing the upstream queue (as the capacity and the number of flows at the upstream queue are made large). For general (including non-regulated) input traffic, we provide a similar result, but with convergence of the queue-length in distribution instead of almost sure convergence. In both cases, the convergence of the overflow probability is uniform and at least exponentially fast. These results help us to simplify network analysis by mapping the original network into a simplified network in which all the nodes with large capacity have been eliminated. Through an extensive numerical investigation, we demonstrate several aspects and implications of our results in simplifying network analysis. We believe our analysis and numerical results shed insight into the understanding of the queueing behavior in a network, where a large number of flows are aggregated at various nodes.

REFERENCES

- [1] J. R. Jackson, "Networks of Waiting Lines," *Operations Research*, vol. 5, pp. 518–521, 1957.
- [2] J. F. C. Kingman, "Markov Population Processes," *Journal of Applied Probability*, vol. 6, pp. 1–18, 1969.
- [3] G. D. Veciana, C. Courcoubetis, and J. Walrand, "Decoupling bandwidths: A decomposition approach to resource management in networks," in *Proceedings of IEEE INFOCOM*, 1994.
- [4] D. Wischik, "The output of a switch, or, effective bandwidths for networks," *Queueing Systems*, vol. 32, pp. 383–396, 1999.
- [5] —, "Sample path large deviations for queues with many inputs," *Annals of Applied Probability*, 2000, to appear.
- [6] D. Y. Eun and N. B. Shroff, "The Impact of Aggregation in Simplifying Network Analysis," Purdue University, West Lafayette, IN, Tech. Rep., 2002.
- [7] N. Likhanov and R. Mazumdar, "Cell loss asymptotics for buffers fed with a large number of independent stationary sources," *Journal of Applied Probability*, vol. 36, no. 1, pp. 86–96, 1999.
- [8] D. D. Botvich and N. Duffield, "Large deviations, the shape of the loss curve, and economies of scale in large multiplexers," *Queueing Systems*, vol. 20, pp. 293–320, 1995.
- [9] C. Courcoubetis and R. Weber, "Buffer overflow asymptotics for a buffer handling many traffic sources," *Journal of Applied Probability*, vol. 33, pp. 886–903, 1996.
- [10] C.-S. Chang, *Performance Guarantees in Communications Networks*. Springer-Verlag, 2000.
- [11] A. Dembo and O. Zeitouni, *Large Deviations Techniques and Applications*. Springer-Verlag, 1998, 2nd Edition.
- [12] R. L. Cruz, "A Calculus for Network Delay, Part I: Network Elements in Isolation," *IEEE Transactions on Information Theory*, vol. 37, pp. 114–131, Jan. 1991.
- [13] —, "A Calculus for Network Delay, Part II: Network Analysis," *IEEE Transactions on Information Theory*, vol. 37, pp. 132–142, Jan. 1991.
- [14] D. Y. Eun and N. B. Shroff, "A Measurement-Analytic Framework for QoS Estimation Based on the Dominant Time Scale," in *Proceedings of IEEE INFOCOM*, Anchorage, AK, 2001.
- [15] I. W. Habib and T. N. Saadawi, "Multimedia Traffic Characteristics in Broadband Networks," *IEEE Commun. Mag.*, pp. 48–54, July 1992.
- [16] W. E. Leland, M. Taqqu, W. Willinger, and D. V. Wilson, "On the Self-Similar Nature of Ethernet Traffic (Extended Version)," *IEEE/ACM Transactions on Networking*, vol. 2, no. 1, pp. 1–15, Feb. 1994.