Bandwidth Allocation under End-to-End Percentile Delay Bounds

Bushra Anjum[†], Harry Perros[†], Xenia Mountrouidou[‡], Kimon Kontovasilis[*]

[†] North Carolina State University,
Computer Science Department,
Raleigh, NC 27695, USA
Email: banjum@ncsu.edu, hp@csc.ncsu.edu

[‡] College of William and Mary,
Computer Science Department
Williamsburg, VA 23187, USA
Email: xmount@cs.wm.edu

[*] National Center for Scientific Research "Demokritos",
Institute of Informatics and Telecommunications, Greece
Email: kkont@iit.demokritos.gr

Abstract

We describe an efficient and accurate approximation method for calculating the bandwidth that should be allocated on each link along the path of a point-to-point MPLS connection, so that the end-to-end delay D is less than or equal to a given value T with a probability γ, that is, $P(D \leq T) = \gamma$. We model a connection by a tandem queueing network of infinite capacity queues. The arrival process of packets to the connection is assumed to be bursty and correlated and it is depicted by a two-stage Markov Modulated Poisson Process. The service times are exponentially distributed. The proposed method uses only the first queue of the tandem queueing network to construct an upper and lower bound of the required bandwidth so that $P(D \leq T) = \gamma$. Subsequently, we estimate the required bandwidth using a simple interpolation function between the two bounds. Extensive comparisons with simulation showed that the results obtained have an average relative error of 1.25%.

1. Introduction

Bandwidth allocation typically involves reserving part of the transmission rate of the output port of each router along the path of a connection for the traffic flow associated with the connection. The problem of allocating bandwidth under quality of service (QoS) restrictions has been analyzed extensively in the literature. Most of the proposed schemes calculate the necessary bandwidth so that the packet loss is bounded. The equivalent bandwidth, proposed originally for ATM networks, see Perros [1], is a good example of such a scheme. A survey of various call admission algorithms can be found in Wright [2].

Bandwidth calculation schemes based on packet loss do not provide guarantees for the end-to-end delay. In this paper, we are concerned with the calculation of the bandwidth to be allocated at the output port of each router along the path of a connection in an MPLS-enabled IP network, so that the end-to-end delay D is less than or equal to a given target delay value T with a probability γ, i.e., $P(D \leq T) = \gamma$. We assume that the path of an LSP, i.e., an MPLS connection, has already been calculated.

In order to calculate the bandwidth so that the end-to-end delay D is less than or equal to a given value T with probability γ, we first have to calculate the end-to-end delay. A connection is typically represented by a series of queues forming a tandem queueing network, with each queue representing the output port of a router along the path. Consequently, the calculation of the end-to-end delay has been identified with the problem of calculating the end-to-end delay in a tandem queueing network. This problem has been addressed in the literature under a variety of assumptions. A common approach is to calculate a bound on the end-to-end delay. Charara and Scharbarg et al [3] proposed three different methods for calculating an upper bound on the end-to-end delay using Network Calculus. Koij et al [4] obtained an upper bound on the end-to-end delay for the real-time CBR traffic, and Goyal et al [5] calculated an upper bound on the end-to-end delay assuming a guaranteed rate scheduling algorithm, such as a virtual clock and generalized processor sharing.

An exact numerical expression of the end-to-end delay in a tandem queueing network can also be obtained by calculating its Laplace transform and then inverting it numerically to obtain delay percentiles. This approach was used by Xiong and Perros,[6] within the context of resource optimization of web services using a tandem queueing network with Poisson arrivals and exponentially distributed service times. Also, Vleeschauwer et al [7] obtained the moment genrating function of the end-to-end delay of N statistically independent M/G/1 nodes which was then inverted numerically in order to compute delay percentiles. They also proposed two less complex and highly accurate approximations for percentile calculation, assuming that the individual queuing delays have an exponential tail distribution.

Yeung and Lehoczky [8] used a Brownian process to study a two-stage queueing network with customers having deadlines (constant and uniformly distributed) and calculated bounds for two different scheduling disciplines, early-deadline first and FIFO. Fractional Brownian Motion (FBM) was used by Lelarge et al [9] to show that the end-to-end delay of a tagged flow in a tandem queueing network, and more generally in a tree network, is completely dominated by the queue with the maximal Hurst parameter.

Finally, Feng and Chang [10] used single-node decomposition to analyze a tandem queueing network with general service times where the arrival process to the first queue consisted of multiple heterogeneous MMPPs. They proposed two approximation schemes for calculating the mean end-to-end delay for a single (tagged) MMPP stream. Central to the proposed schemes is the calculation of the departure process of the tagged MMPP stream by an MMPP2. Specifically, the authors calculate the first three moments of the inter-departure time and the lag 1 autocorrelation of the successive inter-departure times, which are matched to those of an MMPP2 in order to obtain the four parameters that define an MMPP2. (more details on MMPP2 are presented in section 2)

In this paper, we consider a tandem queueing network with exponentially distributed service times. The service rate at each queue represents the bandwidth μ allocated on each output port. Consequently, all service rates are the same and equal to μ. The arrival process of packets is assumed to be bursty and correlated and it is modeled by a two-stage Markov Modulated Poisson Processes (MMPP2). Using this queueing network, we calculate approximately the bandwidth μ to be allocated at each output port so that the end-to-end delay D is less than or equal to a given target delay value T with a probability γ, i.e., $P(D \leq T) = \gamma$.

The MMPP2 is a popular traffic model, because it is capable of capturing burstiness and autocorrelation characteristics commonly present in network traffic, (see e.g., Jiang and Dovrolis [11], and Figueiredo et al [12]) while satisfying a reduced complexity, compared to other processes, such as MAP. It has been used extensively in the literature to capture the correlation characteristics of multimedia sources in broadband integrated services digital networks (see e.g., Adas [15] and Kuehn [16]). Also, appropriately constructed Markov models using MMPP-2 appear to be a viable modeling tool also in the context of modeling long-range dependent traffic over several time scales [17, 18, 19].

This tandem queueing network can be easily analyzed using a single-node decomposition by using the following two results: the moment-matching method proposed by Feng and Chang [11] and a result due to Jean-Marie et al [12] who showed that the waiting time distribution in a single MMPP*n*/M/1 queue is *n*-stage hyper-exponential (H_n). Specifically, the first queue is analyzed as an MMPP2/M/1 queue in isolation from the rest of the network. The departure process is characterized approximately by an MMPP2 using the moment-matching scheme, which permits us to analyze the second queue as an MMPP2/M/1 queue as well, and so on until the last queue is analyzed. The result of this decomposition is that the waiting time in each queue is characterized by a 2-stage hyper-exponential (H_2). Thus the distribution of the end-to-end delay D can be obtained by combining these H_2 distributions into a single phase-type distribution, from which we can calculate the delay percentiles and also the bandwidth that guarantees that $P(D \le T) = \gamma$, where T and γ are given, as described in section 3. We implemented this single-node decomposition and compared it extensively with simulation. It turns out that this approach does not have a good accuracy because the departure process from an MMPP2/M/1 queue is not MMPP.

In view of this, we propose the following alternative method. We first construct an upper and a lower bound on a given percentile of D, from which we obtain bounds of the bandwidth such that $P(D \le T) = \gamma$, for given T and γ. These two bounds are then combined using an interpolation function to obtain an accurate estimate of the bandwidth. The main contribution of this paper is that the upper and lower bounds are constructed by analyzing only the first queue of the tandem queueing network. Thus, the proposed scheme provides results with a lower computational complexity, compared to the single node decomposition approach. Validation tests against simulation for a variety of input parameters showed that the proposed scheme is very accurate with an average relative error of 1.25%.

The paper is organized as follows. In section 2 we describe the queueing network under study, and in section 3 we briefly describe the single-node decomposition algorithm mentioned above. In section 4, we present the lower and upper bounds on the bandwidth and the interpolation function between the two bounds. Section 5 provides numerical results validating the proposed scheme, and the conclusions are given in section 6.

2. The queueing network under study

The queueing network under study is an open tandem network consisting of N infinite-capacity queues as shown in Figure 1.

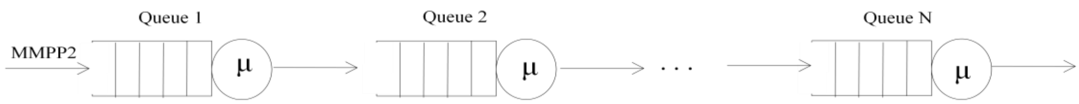

Figure 1. The tandem queueing network under study

The arrival process is a two-state MMPP (MMPP2) characterized by the 2x2 matrices:

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}, Q = \begin{bmatrix} -\sigma_1 & \sigma_1 \\ \sigma_2 & -\sigma_2 \end{bmatrix}$$

where λ_i, and $1/\sigma_i$ is the rate of the Poisson arrivals and the mean value of the exponentially distributed sojourn time in state i=1,2, respectively. The steady state probability vector w of the Markov process associated with the infinitesimal generator Q is:

$$w = [w_1 \quad w_2] = \begin{bmatrix} \frac{\sigma_2}{\sigma_1 + \sigma_2} & \frac{\sigma_1}{\sigma_1 + \sigma_2} \end{bmatrix}$$

The mean arrival rate is:

$$\bar{\lambda} = \lambda_1 \frac{\sigma_2}{\sigma_1 + \sigma_2} + \lambda_2 \frac{\sigma_1}{\sigma_1 + \sigma_2}$$

The main advantage of MMPP2 as a traffic model is that it can depict the burstiness and the autocorrelation of the successive inter-arrival times. The lag 1 autocorrelation function ρ is:

$$\rho = \frac{P(\Lambda - Q)^2 \Lambda (I - eP)(\Lambda - Q)^2 \Lambda e}{(2P(\Lambda - Q)^{-3} \Lambda e) - (P(\Lambda - Q)^{-2} \Lambda e)^2}$$

where

$$P = \frac{1}{w\begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix}} w\Lambda ,$$

I is the 2x2 identity matrix and e is a column vector containing ones. The lag 1 autocorrelation ρ of the MMPP2 takes values in (0, 0.5), see Casale et al [14]. The burstiness of the arrival process is measured by the squared coefficient of variation c^2 of the inter-arrival times, and is as follows:

$$c^2 = \frac{(P(\Lambda - Q)^{-2}\Lambda e) - (2P(\Lambda - Q)^{-3}\Lambda e)^2}{(2P(\Lambda - Q)^{-3}\Lambda e)^2}$$

The queueing network shown in figure 1 models the delays in an MPLS connection over an IP network. Each queue represents the queueing encountered by the packets of the connection at the output port of each router along the path of the connection. The service time at each queue is exponentially distributed with the same rate μ, which represents the bandwidth allocated to the connection. The propagation delay between the routers is not included in the model since it is fixed.

The queueing network is used to calculate the probability density function of the one-way end-to-end delay D of the connection. Using this density function, we can obtain any percentile of the end-to-end delay distribution. The bandwidth μ for which D is less than or equal to a given target delay T with a probability γ can be found using a simple linear search as described below.

3. The decomposition algorithm

In this section we briefly summarize the single-node decomposition algorithm for analyzing the queueing network under study. As stated in section 1, the queueing network under study is analyzed using a simple nodal decomposition. Specifically, we consider the first queue in isolation as an MMPP2/M/1 queue, and we calculate the pdf of the H_2 delay in the queue (including the service time) using the results by Jean-Marie et al [13]. We also approximate its departure process by an MMPP2 using the matching-moment method by Feng and Chang [10] to estimate parameters λ_1, λ_2, σ_1, and σ_2 (The results from [10] and [13] were applied without any modification). This departure process becomes the arrival process to the next queue, which is analyzed as an MMPP2/M/1 queue. We proceed in this fashion until all queues are analyzed. The pdf of the end-to-end delay D is obtained by combining the N individual H_2 distributions into the phase-type distribution shown in figure 2. The pdf of the phase-type distribution with arrival state vector α and the transition rate matrix S, is as follows:

$$f_D(x)=\alpha e^{Sx} S_0$$

where,

$\alpha = (1,0,\ldots,0)$

$S_0=-Se$

e =[1,1,…,1]^{T}

$$S = \begin{pmatrix} -\beta_{11} & 0 & \alpha_2\beta_{11} & (1-\alpha_2)\beta_{11} & \dots & 0 & 0 \\ 0 & -\beta_{12} & \alpha_2\beta_{12} & (1-\alpha_2)\beta_{12} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -\beta_{N1} & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & -\beta_{N2} \end{pmatrix}$$

The corresponding cumulative density function (cdf) of D is $F_D(x)=1-\alpha e^{Sx}e$, from which any given percentile of the end-to-end delay can be calculated.

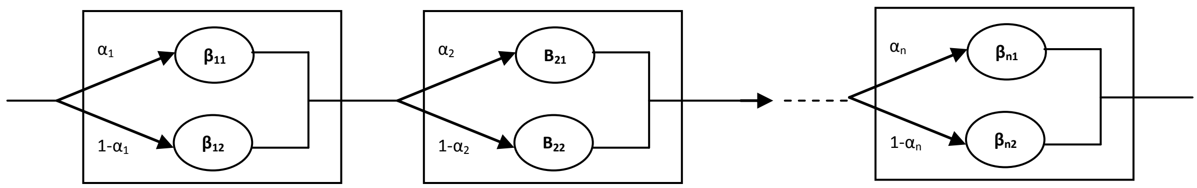

Figure 2. The phase type distribution of the end-to-end delay

The bandwidth that should be allocated at each output port, i.e., the value of μ, so that P($D \leq T$) = γ for given T and γ, is calculated using a simple linear search algorithm. We initialize the service rate at each queue μ to a random value, and then we calculate the γ percentile of D. If it is less than the target value T, then we have over-provisioned the initial bandwidth, therefore we need to reduce μ by a small step δ. If it is higher, then the initial bandwidth is under-provisioned and we need to increase μ by a small step δ. We iterate in this fashion until the absolute value of the difference between T and the γ percentile of D is less ε, where ε was set to 0.01.

Extensive comparisons of the decomposition algorithm against simulation results showed that the percentile of the end-to-end delay calculated by the decomposition algorithm, is a lower bound of the true percentile as estimated by simulation. As shown in table 1, section 5, this is due to the fact that the algorithm underestimates significantly the correlation of the departure process from each queue. As a result, the bandwidth for a specific delay percentile is also underestimated. In view of this, we propose an alternative method based on an upper and a lower bound of the end-to-end delay.

4. Bandwidth estimation based on bounds

In this section, we describe a novel method for calculating the bandwidth approximately which does not require the analysis of the entire queueing network. The method is based on an upper and lower bound of the cdf of the end-to-end delay. These bounds are easy to calculate, as they are constructed by analyzing only the first queue. Using these bounds, we obtain an upper and a lower bound on the bandwidth which satisfies a given percentile of the end-to-end delay, that is, $P(D \le T) = \gamma$ for given T and γ. The required bandwidth is then obtained by interpolating between these two bounds.

An upper bound on the cdf of the end-to-end delay D is readily provided by the cdf of the delay D_1 in the first queue. Since D is made up of the delay in the first queue as well as in the other N-1 queues, the probability that a customer waits more than t in the first queue is less than or equal to the probability that the end-to-end delay D of a customer is more than t, i.e., $P(D_1 > t) \le P(D > t)$ for all t. (Stated differently, the tail of the end-to-end delay D is "higher" than the tail of D_1.) Consequently, we have that $P(D_1 \le t) \ge P(D \le t)$ for all t. In addition, we have observed empirically that when the c^2 of the arrival process to the first queue is high, then $P(D_1 < t) \rightarrow P(D < t)$, as $t \rightarrow \infty$.

A lower bound can also be calculated based on the pdf of the delay D_1 of the first queue, by assuming that the arrival process to each queue i, i=2,3,...,N, of the tandem queueing network is the same as the arrival process to the first queue. Let D' be the resulting end-to-end delay, where $D' = D_1 + D_1 + ... + D_1$. D' has a phase-type representation similar to the one in figure 2 except that all the H_2 distributions are identical to that of the first queue. We have observed empirically that in this case, the tail of D' is "higher" than that of D, i.e., $P(D > t) \le P(D' > t)$ for all t, which means that $P(D \le t) \ge P(D' \le t)$ for all t. This is due to the burstiness of the traffic, as measured by the c^2 of the inter-arrival time, which is reduced as the traffic goes through the queues of the tandem queueing network. This is mainly because of the smoothing effect of the service at each queue. This drop in the c^2 values is particularly pronounced in the first queues, and specifically from queue 1 to 2, when the offered traffic to queue 1 has a high c^2.

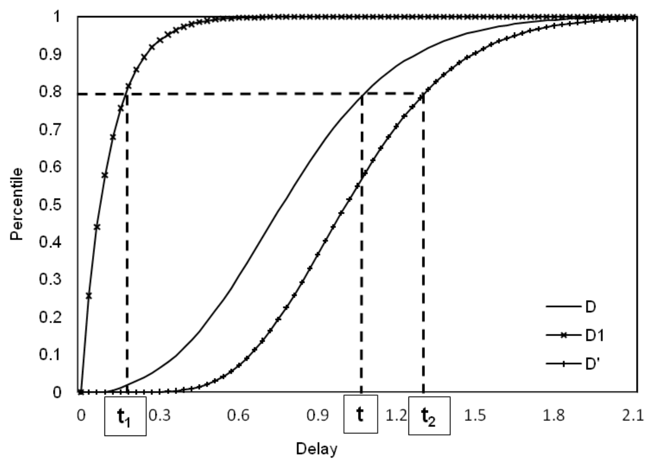

Figure 3. The cdf of D, D_1, and D'

Let t_1, t, and t_2, be the delay values such that $P(D_1 \leq t_1) = \gamma$, $P(D \leq t) = \gamma$, and $P(D' \leq t_2) = \gamma$. Then, since $P(D_1 \leq t) \geq P(D \leq t) \geq P(D' \leq t)$ as $t \rightarrow \infty$, we have that $t_1 \leq t \leq t_2$. In other words, the upper (lower) bound on the cdf of D gives a lower (upper) bound on the delay for a given percentile γ. This can be seen pictorially in figure 3. The cdf graphs for D, D_1, and D' were obtained for a tandem queueing network with 10 queues. The parameters of the MMPP2 arrival process were: λ_1=1, λ_2=52.7046, σ_1=0.3 and σ_2=0.4, which result to a mean arrival rate is 23.16 and a c^2 of the inter-arrival time 20. The service rate for each node μ was set to 60. We observe that for γ=0.8, t_1 = 0.16, t = 1.09 and t_2 = 1.32 and therefore $t_1 \leq t \leq t_2$.

So far we have established that for fixed μ, the delay values t_1, t and t_2 are such that $t_1 \leq t \leq t_2$. Now, let μ_1, μ and μ_2 be the service rates such that $P(D_1 \leq T) = \gamma$, $P(D \leq T) = \gamma$, and $P(D' \leq T) = \gamma$ respectively for a given T and γ. Then, we have that $\mu_1 \leq \mu \leq \mu_2$. That is, using the upper (lower) bound on the cdf of D, we obtain a lower (upper) bound on the bandwidth. This can be easily shown as follows. Let us assume that the value of μ for which $P(D \leq T) = \gamma$ is known, and let us call it μ_{init}. Then, using μ_{init} we calculate the pdf of the delay D_1 in the first queue. (We recall that this is an H_2 distribution which can be computed analytically using the results from Jean-Marie [13].) Subsequently, we calculate the delay value τ_1 such that $P(D_1 \leq \tau_1) = \gamma$ and compare it to T. As discussed above, for the same percentile γ, we have that $t_1 \leq t \leq t_2$, and consequently $\tau_1 \leq$ T. If $T - \tau_1 > 0.01$ we decrease the value of μ_{init} by a small step δ, following the simple search algorithm described in section 3, and recalculate τ_1. The decrease in the value of μ_{init} results in an increase in the delay value τ_1 for the given γ. We iterate on μ_{init} until we obtain a value μ_1 such that the condition T - τ_1< 0.01 is satisfied. Clearly $\mu_1 \leq \mu$. Following similar arguments with the upper bound, we can show that the value μ_2 for which $\tau_2 - T$< 0.01 is such that $\mu_2 \geq \mu$. Hence we have that $\mu_1 \leq \mu \leq \mu_2$

An example of these two bandwidth bounds is given in figure 4. The graphs were obtained for a tandem queueing network with 10 queues. The parameters of the MMPP2 arrival process are: λ_1=1, λ_2=132.349, σ_1=0.3 and σ_2=0.4, which result to a mean arrival rate is 57.3, a c^2 of the inter-arrival time 50, and a lag 1 autocorrelation of 0.376. For reference, the plots also include the bandwidth as calculated by simulation for the said network. (This was obtained using the same simple iterative procedure, but at each step the delay value t for a given γ was estimaed by simulation.). As $T \rightarrow 0$, the upper and lower bandwidth bound approach infinity. This follows from the fact that as the delay requirements become more stringent, more bandwidth is required to meet them, thus leading to the limiting case of infinite bandwidth. On the other hand, as T increases, the bandwidth bounds decrease and they eventually tend to the mean arrival rate. Notice that the bandwidth requirement cannot fall below the mean arrival rate otherwise the system will become unstable. Hence the two bounds intersect again on the x axis corresponding to a very high value of T. This required bandwidth is the minimum bandwidth threshold required to keep the system stable, i.e., to keep the utilization of the first queue (and consequently of all the other queues) less than 1. We can see this in figure 4 that the lower bound tends to the mean arrival rate of 57.3 as the delay T increases. (Please note that this behavior of the bounds and the simulated results holds for any percentile of the delay).

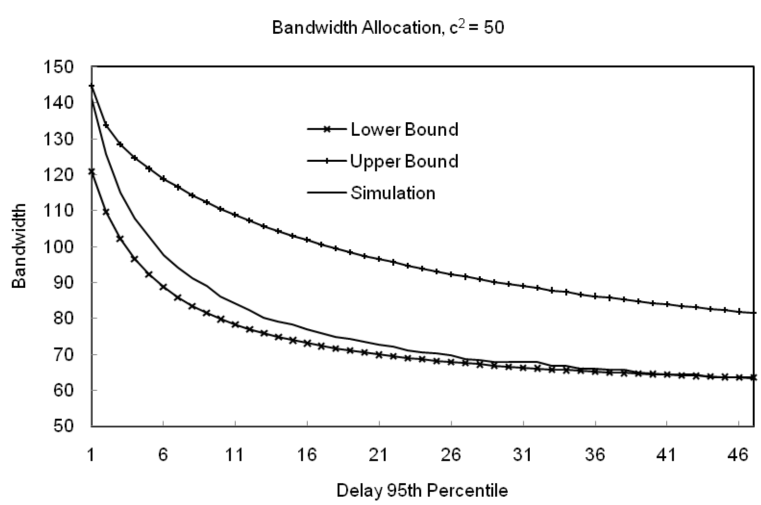

Figure 4. The upper and lower bounds of the bandwidth

The simulation curve showing the exact required bandwidth lies between the two bounds and it also tends to infinity as T approaches 0 and to the average arrival rate as T tends to infinity. In general, as T decreases, the simulation curve approaches the upper bound, and as T increases, the simulation approaches and eventually coincides with the lower bound. In addition, as c^2 decreases, the simulation curve becomes closer and eventually coincides with the upper bound, and as c^2 increases, the simulation curve approaches and eventually coincides with the lower bound. This behavior is consistent with the discussion above on the two bounds of the delay percentile and it can be seen in the graphs of figure 5. Based on the above observations, we propose the following interpolation function $f_I(x)$:

$$f_I(x) = \mu_1(x) + \big(\mu_2(x) - \mu_1(x)\big)e^{-zlog(c)}, z = \max\left(\frac{1}{N}, \log(x+1) - \frac{1}{x}\right)$$

where $\mu_1(x)$ and $\mu_2(x)$ are the bandwidth values calculated from the lower and upper bounds, N is the number of queues and c is the coefficient of variation. It should be mentioned here that this is one of the possible interpolation functions that combine the upper and the lower bounds. This particular version was selected after evaluating various possible alternatives on the basis of resulting in least relative errors. It can be seen directly that it has three very desirable properties:

1. For any given number of nodes N, the factor z becomes larger as x grows, thus the combination formula tends to the single node approximation. This is desirable, as we know that asymptotically as $x \to \infty$, both bounds and the exact end-to-end delay should tend to this single node asymptotic.
2. For the smaller values of x (where the asymptotic regime of large x just discussed isn't in effect), z decreases as the number of nodes N increases, thus the combination formula deviates from the single node bound towards the other bound, to reflect the effect of subsequent nodes on the end to end delay. Again this is desirable behavior.
3. As the burstiness of the traffic at the first node increases (i.e., c is larger), the single node bound comes in effect from smaller values of x and/or N. Again this is desirable, because with burstier traffic most of the end to end delay is due to the first node. After that, the traffic is smoothed out and the subsequent N-1 nodes (which have the same μ as the first one) do not have a significant impact.

We compared the interpolation function with simulation results extensively, and the average error was 1.25% whereas for the decomposition algorithm was as high as 6.3%. Verification results and graphs are presented in the next section.

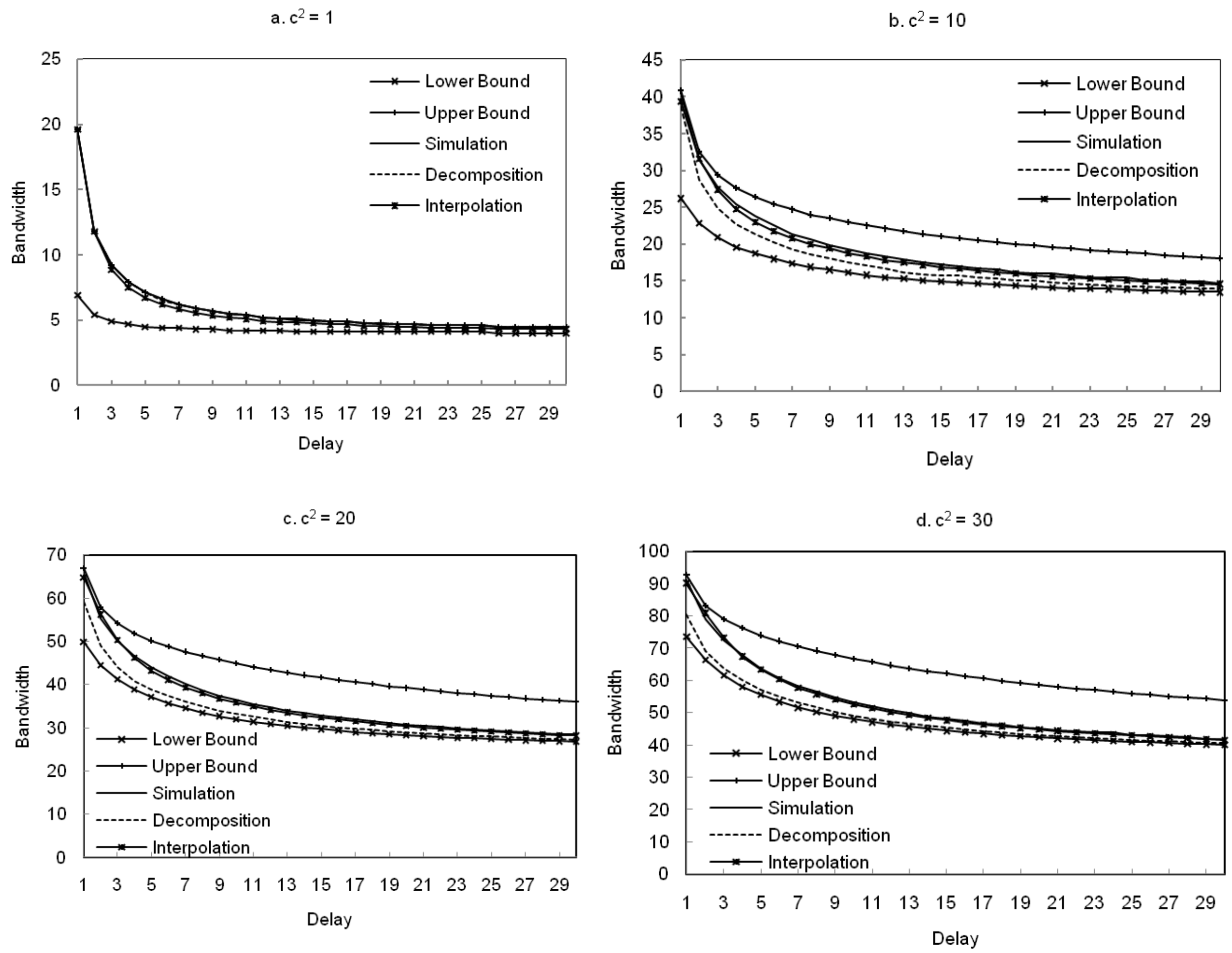

Figure 5. Bandwidth results for a tandem network with 10 queues

5. Validation

In this section, we compare the proposed analytic method based on the interpolation of the upper and lower bounds against simulation. We also include numerical results using the decomposition method. The simulation program was implemented in Java and it simulates the queueing network under study. The results obtained were based on a sample between 1 million and 10 million packets, depending upon the value of the squared coefficient of variation c^2 of the arrival process. The confidence intervals for the end-to-end delay percentile were computed using the batch means method. They are not shown in the figures since they are insignificant.

In Figure 5, we plot the bandwidth calculated using the upper and lower bounds, the interpolation method, the decomposition method, and simulation. The bandwidth was computed so that P(delay $\leq T$) = 0.95, where T varies from 1 to 30. Four different graphs are given, each for a different value of c^2 (c^2=1, 10, 20, 30). The tandem network under study consists of 10 queues and the arrival process to the first queue was obtained by fixing three parameters of the MMPP2 process, i.e., λ_1=1, σ_1=0.3, and σ_2=0.4, and calculating the fourth parameter λ_2.from c^2 using the c^2 formula given in section 2.

The decomposition algorithm gives good results for low values of c^2. However, as c^2 increases, the decomposition results deviate from the simulation results and almost coincide with the lower bound. For a given c^2 the bandwidth estimated using the decomposition algorithm tends to the lower bound as T increases and eventually coincides with it. In table 1, we report the squared-coefficient of variation values for the departure process from queues 2, 5 and 10, based on the results for c^2 = 50 (for the arrival process). We note that the trend of the deviations of the squared-coefficient of variation of the departure process is consistent with the trend of deviations on the estimated bandwidth.

T	**Queue 2**		**Queue 5**		**Queue 10**	
	Simulation	**Decomp**	**Simulation**	**Decomp**	**Simulation**	**Decomp**
1	50.2435	28.00154	47.9517	27.866885	45.1816	27.704413
10	29.2861	16.60698	25.0926	14.78435	21.844	14.539953
20	20.1104	11.853817	16.1835	9.854284	13.5943	9.777885
30	13.0486	9.279408	9.9354	7.506598	7.75653	7.478823

Table 1. Squared-coefficient of variation of the departure process.

Similar results to those presented in figure 5 have been obtained for tandem queueing networks of 5 and 15 queues. The trends and conclusions are similar to those in figure 5. For presentation purposes we only give two graphs in figures 6 and 7, where we compare the interpolation scheme with simulation. In both cases, the parameters λ_1, σ_1, and σ_2 of the arrival process are the same as above and λ_2 is calculated from the expression of the c^2. Figure 6 gives results for c^2=30, and the bandwidth was calculated so that P(delay$\leq$20) = γ, where γ is the 90th, 95th and 99th percentile. Figure 7 gives results for c^2=30, and the bandwidth was calculated so that P(delay≤T) = 0.95, where T=5,20,30.

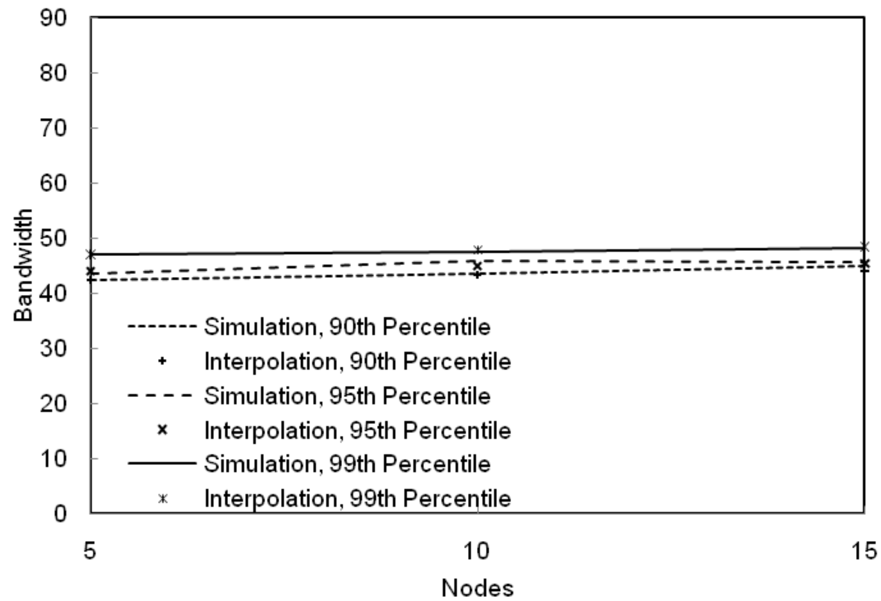

Figure 6. Interpolation function vs simulation for various delay percentiles

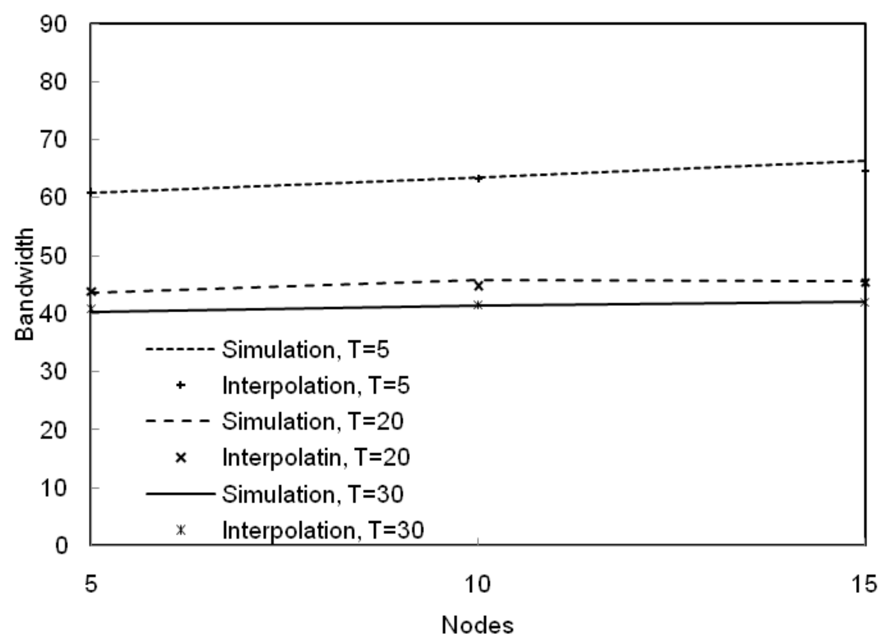

Figure 7. Interpolation function vs simulation for various values of the delay T

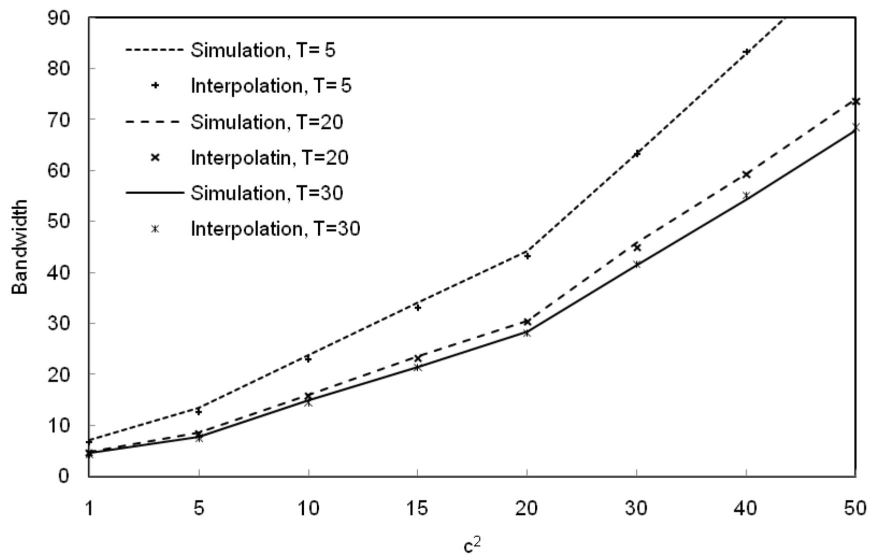

Figure 8. Interpolation function vs simulation for various c^2 values

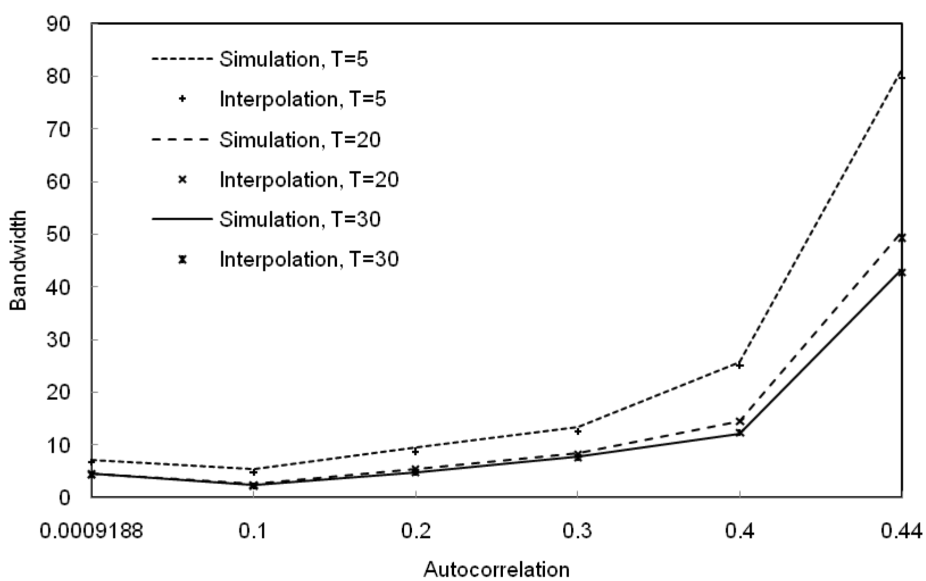

Figure 9. Interpolation function vs simulation for various ρ values.

Finally, in figures 8 and 9 we compare the interpolation scheme with simulation as a function of c^2 and the lag 1 autocorrelation ρ respectively. In both figures, we give the bandwidth obtained by the interpolation scheme and simulation for a queueing tandem network with 10 queues, so that P(delay $\leq T$) = 0.95, where T = 5, 20, 30. The parameters λ_1, σ_1, and σ_2 of the arrival process are the same as above and λ_2 is calculated from the expression of the c^2 in the case of figure 8, and as a function of ρ (given in section 2) in the case of figure 9. (Recall that the lag 1 autocorrelation of an MMPP2 takes values in (0, 0.5), see Casale et al [14].) We observe that in both graphs the interpolation results track the simulation results very well.

As can be seen from the above results the interpolation method has a good accuracy. In table 2, we summarize the relative errors calculated for all the results obtained using the interpolation method and the decomposition method. The maximum relative error observed was 5.17% for the interpolation method and 13.833% for the decomposition method. The average relative error observed was 1.25% for the interpolation method and 6.29% for the decomposition method

Relative Error	**Interpolation**	**Decomposition**
Maximum	5.1676%	13.833%
Minimum	0%	0%
Average	1.252%	6.287%

Table 2. Relative Errors

We conclude this section, by presenting an example where the arrival process is an Interrupted Poisson Process (IPP). This process is obtained from the MMPP-2 by setting the arrival rate in one of the two stages to zero. As a result, the process consists of alternating ON and OFF states. Let λ be the rate of arrivals during the ON state, and let $1/\alpha$ and $1/\beta$ be the mean value of the exponentially distributed duration of the ON and OFF state respectively. The IPP can be used to model the traffic stream generated by a Voice over IP (VoIP) call. VoIP calls follow a similar ON and OFF representation but with deterministic arrivals. That is, the packet inter-arrival time during the ON period is constant, equal to the packetization delay of the voice codec. Simulation results (not reported here) have shown that assuming exponentially distributed inter-arrival times during the ON period, as is the case of the IPP, with a mean equal to the packetization delay, gives a very tight upper bound on the percentile delay obtained assuming constant inter-arrival times during the ON period. Resultantly, the bandwidth calculated to satisfy a delay percentile with an IPP arrival process also provides a tight upper bound on the bandwidth required by a VoIP stream to satisfy a similar end-to-end percentile delay.

The parameter values chosen for the IPP stream are set equal to the parameter values of a VoIP call. A VoIP has, on average, 400 ms of ON period, 600 ms of OFF period [1], and the number of packets generated per second during the ON period is 50. A voice payload of 160 bytes is carried in a separate IP packet with a 40-byte IP/UDP/RTP header [20]. Thus, we have α=2.5, β=1.667 and λ=50. We consider a similar network as before, consisting of 10 nodes, and calculate the bandwidth required to satisfy the 95th percentile delay, where the 95th percentile delay varies from 20 ms to 200 ms. This particular range of delays was selected keeping in mind the recommended bounds on the one-way delay for a voice conversation [20]

The results obtained for a 10-node tandem queueing network are shown in figure 10. Again the interpolation gives good accuracy with a maximum relative error 5.2% and an average relative error 2.16%. Let us now consider a VoIP call between two end-points with a 30 ms propagation delay. In order for the 95th percentile of the total end-to-end delay to be 150 ms, the 95th percentile of the sum of the queueing delays has to be 130 ms. From figure 10, we see that this can be achieved by allocating a bandwidth to each queue equal to 254Kbps.

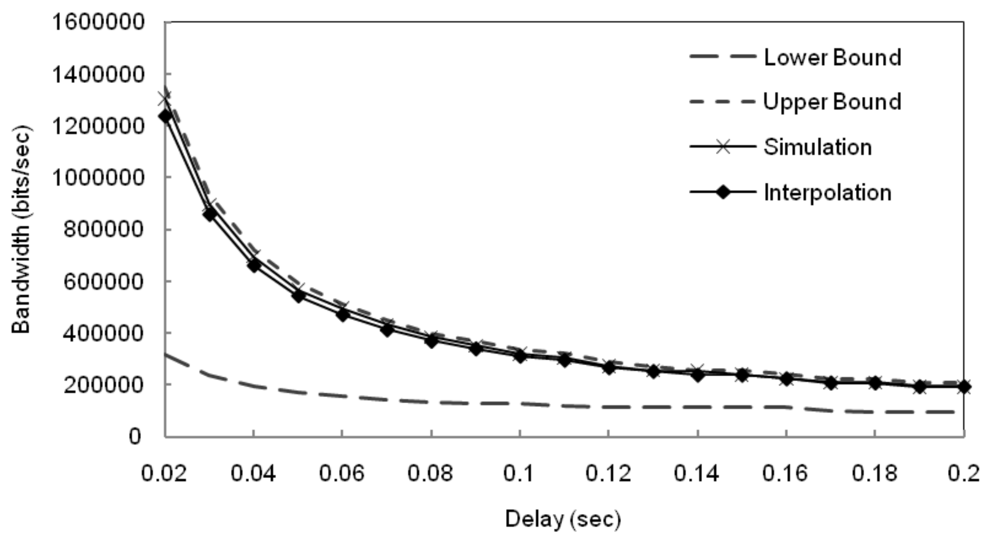

Figure 10. Bandwidth results for a tandem network with 10 queues for a single IPP.

6. Conclusions

In this paper, we propose a simple and efficient scheme for estimating the bandwidth that should be allocated on each link of an MPLS connection in an IP network so that the end-to-end delay D is bounded statistically. That is, D is less than or equal to a given value T with a probability γ. The arrival process is assumed to be bursty and correlated and it is depicted by an MMPP2, and the service times are exponentially distributed. Of interest is that this scheme requires the calculation of an upper and lower bound of the bandwidth by analyzing only the first queue. Extensive comparisons with simulation shows that the proposed method has an average relative error of 1.25%. We also implemented a decomposition algorithm using known results from which the bandwidth can also be estimated. This decomposition algorithm is a fairly common method for analyzing queueing networks that do not have product-form solution. We show that this decomposition does not have a good accuracy in addition to being more CPU intensive than our proposed scheme.

References

[1] H. G. Perros, Connection-Oriented Networks: SONET/SDH, ATM, MPLS, and OPTICAL NETWORKS. Wiley, 2005.

[2] S. Wright., "Admission Control in Multi-Service IP Networks: A Tutorial," *IEEE Comm. Surveys*, vol. 9, no. 2, pp. 72–78, 2007.

[3] H. Charara, J. Scharbarg, J. Ermont, C. Fraboul, "Methods for Bounding End-to-End Delays on an AFDX Network", ECRTS '06.

[4] R. E. Kooij, O. Osterbo and J.C. Wal, "Calculating End-to-End Queuing Delay for Real-Time Services on an IP Network", Art-QoS 2003, LNCS 2698, pp. 115-126, 2003.

[5] P. Goyal, S. S. Lam and H. M. Vin, "Determining End-to-End Delay Bounds In Heterogeneous Networks", Proceedings of the 5th International Workshop on Network and Operating System Support for Digital Audio and Video, 1995

[6] K. Xiong and H. Perros, "Resource Optimization Subject to a Percentile Response Time SLA for enterprise computing," in *IEEE Globecom*, 2006.

[7] D. Vleeschauwer, M.J.C. Buchli, A Van Moffaert and R.E. Kooij, "End-to-End Queuing Delay Assessment in Multi-service IP Networks", Journal of Statistical Computation and Simulation, Vol. 72(10), 2002, pp 803-824.

[8] S.-N. Yeung and J. Lehoczky. End-to-end Delay Analysis for Real-Time Networks. In Proceedings of 22nd IEEE Real-Time Systems Symposium (RTSS 2001)

[9] M. Lelarge, Z. Liu, and C. Xia, Asymptotic Tail Distribution of End-to-End Delay in Networks of Queues with Self-Similar Cross Traffic, INFOCOM 2004.

[10] H. W. Feng and J. F. Chang, "Connection-wise End-to-End Performance Analysis of Queueing Networks with MMPP Inputs," *Performance Evaluation Journal*, vol. 43, pp. 39–62, 2001.

[11] H. Jiang and C. Dovrolis, "Why is the Internet Bursty in Short Time Scales?", ACM SIGMETRICS Performance Evaluation Review, Vol 33, pp: 241-252 (2005).

[12] D. Figueiredo, B. Liu, V. Misra, and D. Towsley, "On the autocorrelation structure of the TCP traffic", Computer Networks, Vol 40, pp: 339-361 (2002).

[13] A. Jean-Marie, Z. Lieu, P. Nain and D. Towsley, "Computational Aspects of the Workload Distribution in the MMPP/G1/1 Queue", *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 5, 1998.

[14] G. Casale, E. Z. Zhang, E. Smirni, “Trace data characterization and fitting for Markov modeling”, Performance Evaluation 67(2) , February 2010, pp. 61-79.
[15] A. Adas, Traffic models in broadband networks, IEEE Commun. Mag. 35 (7) (1997) pp 82–89.
[16] P.J. Kuehn, Remainder on queueing theory for ATM networks, Telecommun. Systems 5 (1996) pp 1–24.
[17] A.T. Andersen, B.F. Nielsen, A Markovian approach for modeling packet traffic with long-range dependence, IEEE JSAC 16 (5) (1998) pp 719–732.
[18] A. Horváth, M. Telek, Markovian modeling of real data traffic: Heuristic phase type and MAP fitting of heavy tailed and fractal like samples, in: Performance Evaluation of Complex Systems: Techniques and Tools, IFIP Performance 2002, in: LNCS Tutorial Series, vol. 2459, 2002, pp. 405–434.
[19] A. Riska, E. Riedel, Long-range dependence at the disk drive level, in: Proc. of 3rd Conf. on Quantitative Evaluation of Systems (QEST), IEEE Press, 2006,pp. 41–50.
[20] Harry G. Perros, "Connection-Oriented Networks: SONET/SDH, ATM, MPLS and Optical Networks", Wiley; 2005