Les Cahiers du GERAD ISSN: 0711–2440

Design of QoS Aware IP Network Supporting Services with Average Delay Requirement

S. Srivastava, A. Girard,
B. Sansò

G–2006–13

March 2006

Design of QoS Aware IP Networks Supporting Services with Average Delay Requirement

Shekhar Srivastava
GERAD
HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada, H3T 2A7
shekhar.srivastava@gerad.ca

André Girard
GERAD and INRS-EMT
Place Bonaventure
800, rue de la Gauchetière ouest, Suite 6900
Montréal (Québec) Canada, H5A 1K6
andre@emt.inrs.ca

Brunilde Sansò
GERAD and École Polytechnique de Montréal
C.P. 6079, Succ. Centre-ville
Montréal (Québec) Canada, H3C 3A7
brunilde.sanso@polymtl.ca

March 2006

Les Cahiers du GERAD

G–2006–13

Abstract

In this paper, we study the problem of determining the capacity requirements for applications that require QoS guarantees. We consider three kinds of source models; Poisson based, On-Off based and leaky bucket filter based, and present explicit expressions capturing the required capacity. We compare the required capacity for average delay for these models with the required capacity for absolute delay and comment on the differential. Further, using simulation results, we study the average and variance of the observed delay for voice and video sources and compare the three models. We found that On-Off and leaky bucket models are very powerful and ensure that the actual delays are less than the required delay and that the variance remains acceptable. These results seem to indicate that it may be possible to dimension systems based on average QoS requirements and still get adequate performance for other requirements such as jitter. This would then provide us with computational tools to dimension networks efficiently.

Résumé

Nous voulons déterminer le débit nécessaire pour garantir aux applications qui en ont besoin des qualités de service adéquates. Nous étudions trois modèles de source : Poisson, *On-Off* et contrôlée par un *leaky bucket*. Nous donnons des expressions explicites pour le débit requis. Nous comparons ensuite ces débits pour un délai moyen donné à celle qui serait nécessaire pour une contrainte sur le délai maximum des paquets et nous discutons de la différence entre les deux. À l'aide de simulations, nous étudions la variance du délai pour des sources de trafic de voix et de vidéo en comparant les trois modèles. Nous montrons que les modèles *On-Off* et *leaky bucket* sont suffisamment robustes pour garantir que le délai et la variance demeurent en dessous des bornes supérieures admissibles. Ces résultats montrent qu'on pourrait dimensionner des systèmes sur la base d'un critère de délai moyen et obtenir quand même une qualité de service suffisante en terme des autres paramètres comme la gigue. Ceci nous permettra de développer des outils de dimensionnement de réseaux efficaces.

Acknowledgments: We would like to thank C. Dovrolis from Georgia Tech. for providing the simulation code for PDD scheduling.

1 Introduction

The Internet of today has grown from an experimental academic medium to the common omnipresent platform for connecting diverse networks and computers into a common world-wide-web. Such a deep penetration of internet was only possible due to the simplicity and pervasiveness of the Internet Protocol (IP). IP provided best-effort performance and connectivity to users, which was sufficient for most applications and users.

At the same time, other technologies such as Asynchronous Transport Mode (ATM), Frame Relay (FR), etc. were proposed for supporting quality-of-service (QoS) based applications. These networks operated in a private domain and their services had to be bought at a premium price from Internet Service Providers (ISP). These networks had a very controlled behavior and traffic was well monitored to ensure low delays, almost no packet loss and negligible delay variations. Companies used these services to support applications such as virtual meetings, inventory management/regulation, etc. The ISPs provided source-destination based virtual leased lines (VLL) or multiple locations based virtual private networks (VPN) as services to the customers. The service level agreements (SLA) [16] were designed exclusively for each customer or group of users based on their requirements and objectives.

Increased pressure for competitive pricing has led to re-considering the separate network approach (best-effort and QoS based). ISPs have started exploring the possibility of moving all the applications on a common IP network with the hope of realizing massive cost savings. However, there are some issues. Such QoS centric IP networks will have to carry service classes with very stringent delay requirement (say voice) along side classes with no delay restriction (say email). One way of achieving this is to ensure that the delays of all the classes should be less than the most stringent class. This could potentially translate into massive over-provisioning and would have undermined the potential of QoS based IP networks [8].

If we want to avoid over-provisioning, we must be able to dimension networks is a systematic way. This in turn means that we need methods to dimension networks to ascertain QoS of various service classes. Unfortunately, at the present time, there are very few such models and the ones that we have are based either exclusively on a maximum delay approach or are based on simplistic average delay requirement. In practice, the users who need guaranteed average delay performance, would still want low jitter and losses for some real-time applications such as voice, video, etc. In this context, dimensioning just for average delay may not give adequate performance as far as the other QoS measures (jitter, etc.) are concerned. Falling back on absolute delay based models will lead to gross over-provisioning.

We present a framework, accompanied models and performance results for providing services with average delay bounds. We investigate the extent of over-provisioning induced by deterministic delay bounds and also evaluate the impact of average delay based provisioning upon other QoS measures such as jitter, etc. We explore if other QoS measures can also be taken into account while dimensioning based on only average delay requirement. We focus on a single link only, and consider multiple links/network wide design issues in future extensions.

Internet Engineering Task Force (IETF) has standardized the Differentiated Services architecture (DiffServ) for large scale deployment of IP networks with QoS support [13]. They have provided three types of service to packets: expedited forwarding (EF) [11], assured forwarding (AF) [10] and best-effort. The applications requiring absolute delay bound are mapped to EF class. For providing the average delay bound, we use the AF class. We propose to use the proportional delay differentiation (PDD) model of Dovrolis et.al. [4, 6] for providing different delays to the subclasses within the AF class. The PDD based approach is unique in its simplicity and tractability. Recently, many real time applications have been successfully mapped to delay and loss differentiation parameters of the PDD subclasses [15, 17, 18]. In this paper, we consider various source models such as Poisson, on-off and leaky bucket constrained to determine the amount of bandwidth required to guarantee that all the sources achieve their average case requirement. Using analytical results and simulations, we also demonstrate that many real-time applications such as voice and video can be effectively supported via services guaranteeing average delays only. We also compare the bandwidth requirements for such an AF class vis-a-vis equivalent EF class and comment on the magnitude and order of bandwidth differential. Such a difference could be effectively translated into providing AF services at a much more economical price as compared to services of EF class.

In this paper, we focus on delay as the main performance measure. We claim that the other important measure, the delay jitter, could be controlled using playout buffers assuming that the delay encountered is small enough. For non-elastic sources such as UDP-based interactive services having small delays, losses might be totally avoidable. Observe that only when the delays are high and queues in the routers build up, do we have losses. Since the sources are non-elastic, losses are not coupled with throughput.

The rest of the paper is organized as follows. In Section 2, we provide the framework/architecture for providing average and absolute delay based services in an IP based QoS network. In Section 3, we discuss the models for absolute delay bounds and the required capacity. In Section 4, we discuss the models for average delay bounds and the required capacity. In Section 5, we present numerical results derived using the explicit expressions developed in the previous section. Further, in Section 6, we present simulation

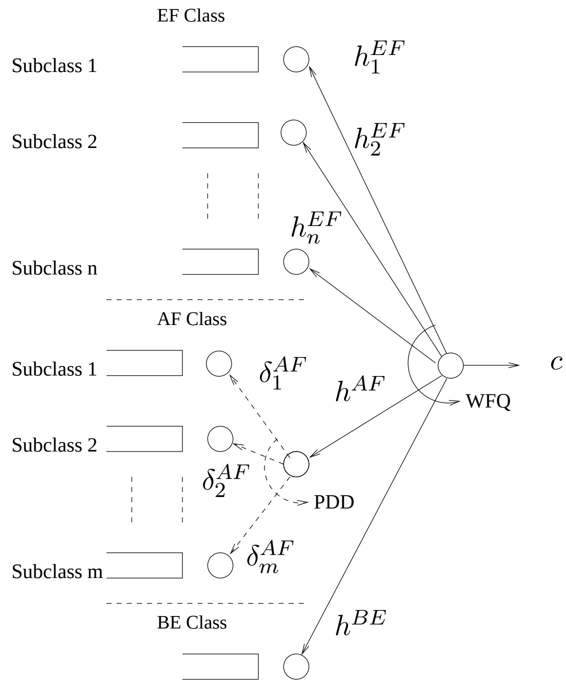

Figure 1: Forwarding interface of router A

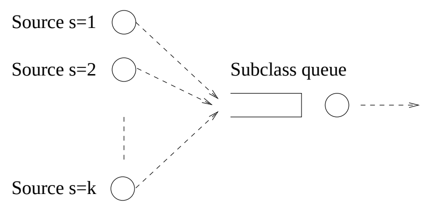

Figure 2: Mapping between sources and subclasses at node A

results to compare the performance of various source models for average case dimensioning. We provide summary and discuss future work in Section 7.

2 Core Router Architecture

In this section, we discuss the architectural details of a forwarding interface of a core IP router which provides absolute and average case guarantees for the sources. For this purpose and towards discussions in later sections, consider multiple sources which want to send their traffic from node A to node B connected by a direct link ℓ.

Consider now the ingress (A) and egress (B) routers. Assume that the concerned forwarding interface on node A to node B has been configured for n EF subclasses and m AF subclasses. At the interface, each source is mapped to EF or AF class based on the provisions of the customer's SLA (absolute or average delay). The mapping to subclass (such as i) within the class is based on the source application running at the source (voice, video, etc.). The demand pair (A-B) has to support a set of sources $s \in \mathcal{S}$. In Figure 1, we present the architecture of the forwarding interface of IP router A, supporting DiffServ. Let the capacity of the direct link connecting router A to B be c. We assume that the bandwidth is distributed amongst the n EF subclasses, AF class and BE class using a weighted fair queueing scheduler (WFQ) where the vector $\mathbf{h}$ determines the weights used in scheduling. This ensures that each EF subclass on the link gets no less bandwidth than c_i^{EF}, where

$$c_i^{EF} = \frac{h_i^{EF}\ c}{h^{BE} + h^{AF} + \sum_{i=1}^{n} h_i^{EF}}. \tag{1a}$$

Here, c_i^{EF} is the minimum bandwidth required for subclass i in order to provide the target delay D_i^{EF} to the sources belonging to the class. Similarly, h^{AF} captures the weight for the AF class which translates into minimal bandwidth of

$$c^{AF} = \frac{h^{AF}\ c}{h^{BE} + h^{AF} + \sum_{i=1}^{n} h_i^{EF}}. \tag{1b}$$

c^{AF} is the total bandwidth available to the AF class such that can be shared between the m subclasses. The bandwidth c^{BE} available to the BE class can be computed as

$$c^{BE} = \frac{h^{BE}\ c}{h^{BE} + h^{AF} + \sum_{i=1}^{n} h_i^{EF}}, \tag{1c}$$

In Figure 2, we show how sources are mapped onto each subclass (EF or AF) at the ingress edge router. Let the set of sources belonging to the ith EF subclass be $\mathcal{S}_i^{EF}$, then every source $s \in \mathcal{S}_i^{EF}$ has a *absolute* delay requirement $D_s \geq D_i^{EF}$. Similarly, for the ith AF subclass, sources $s \in \mathcal{S}_i^{AF}$ have an *average* delay requirement such that $D_i^{AF} \leq D_s$. Source s belonging to AF or EF class has an average arrival rate of r_s. When generated by an On-Off source model, it has a peak rate of R_s and the on period of average length I_s. Such a source can be effectively shaped by a leaky bucket filter of parameter (σ_s, ρ_s), where ρ_s is the average arrival rate and σ_s is the maximum allowed burst length of the LB filter. In order to ensure low losses, it is advisable to have $\rho_s > 1.1 r_s$ and high value of σ_s.

Observe that sometimes the bandwidth allocated to the AF class needs to be shared between the m subclasses such that each subclass meets its target delay requirement. This is done by using PDD scheduling [6] between the subclasses where the value of parameter δ_i^{AF} determines the extent of differentiation as discussed in Section 4. Furthermore, each AF subclass can have an end-to-end delay requirement or for the concerned hop. Providing hop-by-hop delay allows greater flexibility and options of better mapping the sources to subclasses. This is outside the scope of current paper and is currently under study.

3 Providing Absolute Delay Bound

In this section, we discuss the amount of capacity required to provide an absolute delay guarantee to sources belonging to the EF class. The capacity is a function of the characteristics of the sources and their delay requirement. Such a problem is sometimes referred to as the *equivalent capacity* problem (see [12], and references therein). Absolute bounds on delays can be obtained for sources which are shaped by a *leaky bucket* (LB) filter. Behavior of such shaped sources has been extensively studied in the literature. Cruz studied such shaped sources in isolation, when multiplexed and on an end-to-end basis using fluid flow models [1–3]. As discussed before, we consider sources whose traffic is shaped by a LB of parameter (σ_s, ρ_s). The value of the maximum delay D_s that can be incurred by any packet can be determined based on the nature of the application connected to the source s. First we consider a subclass i and we define

$$\rho_i^{EF} = \sum_{s \in \mathcal{S}_i^{EF}} \rho_s, \qquad \sigma_i^{EF} = \sum_{s \in \mathcal{S}_i^{EF}} \sigma_s.$$

Then the maximum backlogged traffic from A to B for EF subclass i [14] will be

$$Q_i^{EF} \leq \sigma_i^{EF}.$$

Observe that direct addition of burst lengths could be a conservative approach. However, it is necessary in order to guarantee deterministic delays to each individual source. Furthermore, we have

$$c_i^{EF} = \max \left\{ \rho_i^{EF}, \frac{\sigma_i^{EF}}{D_i^{EF}} \right\}. \tag{2}$$

The required minimal capacity can be ensured to the subclass i by adjusting the weights based on the equations (1). The above presented capacity requirement is only used for comparison with the required capacity for AF classes which provide average delay guarantee.

4 Providing Average Delay Bound

In this section, we determine the minimal bandwidth c^{AF} required by the AF class in order to ensure that the average delay for each AF subclass meets or exceeds its required average delay D_i^{AF}. Recall that AF class only provides average delay guarantee to its sources.

Such a problem was originally considered by Dovrolis et. al. in [5]. They accounted for the average arrival rates of the sources but did not account for their burstiness and used simulation to arrive at the required capacity. The burstiness of the sources impacted the derived capacity in an indirect way. The approach is simple yet effective. For each AF subclass, define the target average delay as d_i^{AF}. Then, we know that

$$D_i^{AF} \geq d_i^{AF}. \tag{3}$$

To ascertain this, consider an imaginary queue which is being fed by the sources belonging to the AF class. The packets are serviced at the rate c^{AF} on a first-come-first-serve (FCFS) basis. Let q^{AF} denote the average length of such a queue. The aggregate arrival rate will be

$$r^{AF} = \sum_{i=1}^{m} \sum_{s \in \mathcal{S}_i^{AF}} r_s.$$

Recall that r_s is the average arrival rate for source $s \in \mathcal{S}$, and here we assume that $r^{AF} < c^{AF}$. Then, the required capacity c^{AF} is such that the imaginary queue has queue length,

$$q^{AF} \leq \sum_{i=1}^{m} \sum_{s \in \mathcal{S}_i^{AF}} r_s d_i^{AF}. \tag{4}$$

Alternately, the average waiting time for the imaginary queue should be

$$d^{AF} \leq \frac{1}{r^{AF}} \sum_{i=1}^{m} \sum_{s \in \mathcal{S}_i^{AF}} r_s d_i^{AF}. \tag{5}$$

It will ensure that each source belonging to subclass i will have an average delay of d_i^{AF} or less.

The approach argues that if the number of packets in the imaginary queue conforms to the condition (Eq. 4) then in the real queue, PDD scheduler can distribute the available capacity amongst the contending AF subclasses such that each one of them conforms to the desired average delay on short as well as long time scales. The PDD scheduler requires the parameter δ^{AF} which can be computed as follows [5]. Without loss of generality, we assume that subclass m has the maximum delay requirement, then

$$\delta_i^{AF} = \frac{d_i^{AF}}{d_m^{AF}}, i = 1, 2, .., m-1, \text{ and } \delta_m^{AF} = 1. \tag{6}$$

We then have to determine the capacity c^{AF} required to achieve an average queue length of q^{AF}. For a given value of average queue length, characteristics of sources impact the amount of capacity required. The capacity was determined using simulations in [5]. The approach provides good estimates for required capacity but has limited utility towards network dimensioning. Due to the use of simulation, it would be hard to incorporate in an overall network design problem. This would greatly curtail its usefulness towards the goal of designing QoS-based IP networks.

In this paper, we refine on the approach and use it towards the goal of determining the value of c^{AF}. In some cases, we *do* have computational models for the imaginary queue. We can then use them in the Dovrolis framework and this gives us a fast computational technique for estimating delays. We apply this approach to Poisson, On-Off and LB controlled sources. For each of the three scenarios, we develop the expressions for average queue length and use it to compute the required capacity.

4.1 Poisson Sources

We first consider that each source generates packets with an exponential inter arrival time. Such systems are fairly well studied in the literature. We know that each source s has the average arrival rate of r_s, i.e., its inter-arrival times are exponential with a mean of $1/r_s$. The required capacity is referred to as c_P^{AF}. Using the $M/M/1$ queue length formula, we have

$$d^{AF} = \frac{\left(r^{AF}/c_P^{AF}\right)^2}{r^{AF}\left(1 - r^{AF}/c_P^{AF}\right)}.$$

The equation can be rearranged to get,

$$\left(c_P^{AF}\right)^2 - r^{AF} c_P^{AF} - \frac{r^{AF}}{d^{AF}} = 0.$$

Then the required capacity will be

$$c_P^{AF} = \frac{r^{AF} + \sqrt{(r^{AF})^2 - 4r^{AF}/d^{AF}}}{2}. \tag{7}$$

The Poisson based model was also considered in [5] and is presented here for comparison.

4.2 On-Off Sources

Next we consider sources which have a two-state, On-Off behavior. Such models are sometimes used to characterize voice or video sources. As mentioned before, each source has an average rate of r_s, peak rate of R_s and average on-period of I_s. We develop upon the work presented in [7]. The required capacity, referred to as c_{OO}^{AF} and the average delay d^{AF} are related by

$$r^{AF} d^{AF} = \frac{1}{\left(c_{OO}^{AF} - r^{AF}\right)} \sum_{i=1}^{m} \sum_{s \in \mathcal{S}_i^{AF}} \left[(R_s - r_s)\left(R_s - c_{OO}^{AF} + r^{AF} - r_s\right) \frac{r_s I_s}{R_s} \right].$$

Upon solving, we get

$$c_{OO}^{AF} = \frac{\left(r^{AF}\right)^2 d^{AF} + \displaystyle\sum_{i=1}^{m} \sum_{s \in \mathcal{S}_i^{AF}} (R_s - r_s)(R_s - r_i + r^{AF}) \frac{r_s I_s}{R_s}}{r^{AF} d^{AF} + \displaystyle\sum_{i=1}^{m} \sum_{s \in \mathcal{S}_i^{AF}} (R_s - r_s) \frac{r_s I_s}{R_s}}. \tag{8}$$

4.3 Shaped Sources

Now consider the scenario where each source is policed by a leaky bucket with parameters ρ_s and σ_s. For this we consider the results presented in [9]. They have shown the following result.

Theorem 1 *The average delay for a queue serving at rate c to multiplexed stream of sources (s=1,2,..,S) policed by leaky bucket (ρ_s, σ_s) is*

$$\bar{d} = \frac{\sum_{s=1}^{S} \sigma_s \rho_s}{2c\left(c - \sum_{s=1}^{S} \rho_s\right)}. \tag{9}$$

For ease of presentation, define the average regulated arrival rate for the AF class as

$$\rho^{AF} = \sum_{i=1}^{m} \sum_{s \in \mathcal{S}_i^{AF}} \rho_s,$$

and the regulated arrival rate weighted burstiness for the AF class as

$$\gamma^{AF} = \sum_{i=1}^{m} \sum_{s \in \mathcal{S}_i^{AF}} \rho_s \sigma_s.$$

Then the value of capacity, referred to as c_{LB}^{AF} should satisfy the relation

$$\left(c_{LB}^{AF}\right)^2 - c_{LB}^{AF} \rho^{AF} - \frac{\gamma^{AF}}{2\, d^{AF}} = 0, \tag{10}$$

where d^{AF} was expressed in eq. (5). Solving the quadratic equation and discarding the value which is smaller than ρ^{AF}, we get

$$c_{LB}^{AF} = \frac{1}{2}\left(\rho^{AF} + \sqrt{\left(\rho^{AF}\right)^2 + \frac{4\,\gamma^{AF}}{2\, d^{AF}}}\right). \tag{11}$$

Observe that the required capacity increases with burstiness weighted with average arrival rate and decreases with increasing target delay.

5 Numerical Results

In this section, we present numerical results comparing the required capacity for providing average delay guarantee for Poisson, On-Off and shaped sources. We also present for comparison the results for absolute delay guarantees. We consider two types of on-off sources, voice and video.

5.1 Voice Application

We model a voice source as a two state on-off source where it generates packets with a deterministic inter-arrival time of 15 msecs in the on-state. On-periods are exponential with rate 2.5 and off-period are also exponential with a rate 1.67. This leads to an average rate $r = 25.632$ Kbps, $R = 64$ Kbps and $I = 2.5$. Each source can be policed by a leaky bucket with parameters $\rho = 28$ kbits and $\sigma = 192$ kbits which incurs losses of less than 0.1%.

5.2 Video Application

The video source is also modeled as a two state on-off source. During each burst, the source generates 184.4 packets per second, each packet of size 1000 bytes for an packet inter-arrival time of 5.4 msecs during the active period. The length of the active period is exponentially distributed with an average of 0.23 secs. This produces an average rate $r = 1.08$ Mbps and peak rate $R = 1.475$ Mbps. Each application can be policed by a leaky bucket with parameter $\rho = 1200$ kbits and $\sigma = 360$ kbits which causes fairly low losses.

Now, we compare the values of c for absolute delay requirement on one hand, and average delay for Poisson, On-Off and leaky bucket shaped sources on the other. To normalize the voice and video sources, we compute the ratio of capacity to the average arrival rate (c/r), referred to as *overprovisioning*. The amount of overprovisioning is dependent upon the source models and the delay requirements. To ensure that the link is not overloaded and that the queues do not build up excessively, we impose the condition that the value of capacity is always greater than 1.05 time average arrival rate. Furthermore, we consider two variations, Application based Partitioning and Application based Sharing.

5.3 Application based Partitioning

Here, we reserve separate bandwidth for each subclass. In other words, different applications such as voice and video, do not share bandwidth. However, sources belonging to each subclass share the capacity. The scenario could be useful for service providers who wish to guard applications from each other by isolation. As an outcome, benefits of multiplexing between applications can not be yielded. Here, c_1/r and c_2/r represent the overprovisioning required to guarantee abolute delay for voice and video sources. c_{LB}/r, c_{OO}/r and c_P/r refer to overprovisioning required when dimensioning for average delays using leaky bucket based model, using on-off based model and Poisson based model, respectively. In this regard, we present the results for voice sources in Figure 3 and that of video sources in Figure 4. We see that the requirement of absolute delay causes an overprovisioning as high as 1000 for voice traffic and more than 8 for video traffic. Guaranteeing average delay reduces the overprovisioning to the range 2–11 in voice sources and less than 1.2 in video

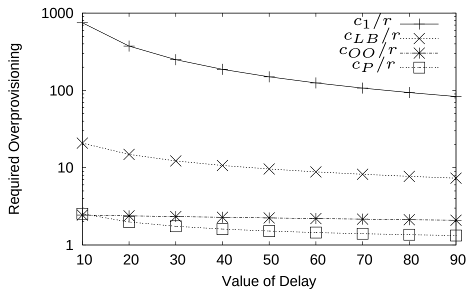

Figure 3: Required overprovisioning for single Voice Source

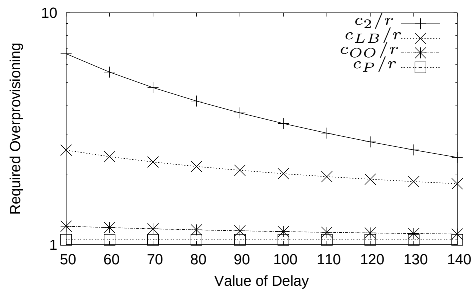

Figure 4: Required overprovisioning for single Video Source

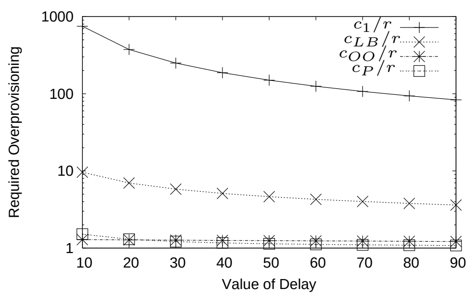

Figure 5: Required overprovisioning for 5 Voice Sources

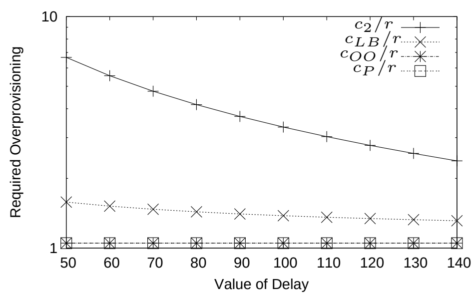

Figure 6: Required overprovisioning for 5 Video Sources

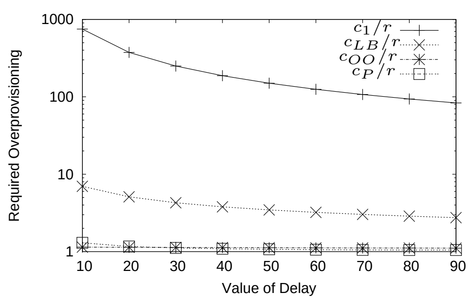

Figure 7: Required overprovisioning for 10 Voice Sources

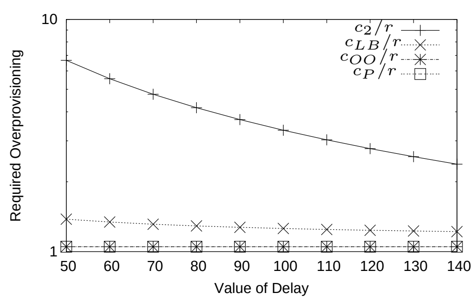

Figure 8: Required overprovisioning for 10 Video Sources

sources. Particularly, design with Poisson models requires less than twice the arrival rate for target delays in the range 10–100 ms. For video, minor overprovisioning is sufficient to ensure the average delay of 50 to 100 ms.

In Figures 5-8, we present the required overprovisioning for 5 and 10 voice and video sources. Observe that the benefits of multiplexing between the sources of the same subclass further help in decreasing the overprovisioning for the average delay scenario. For absolute delay, the values are same. Interestingly, for 5 and 10 voice sources requiring 10 ms of average delay (Figure 5 and Figure 7), the on-off based model requires marginally less capacity than the Poisson based model. This can be ascribed to the fact that for on-off based model, the packets are generated at regular deterministic intervals during an active period and hence multiplexed on-off sources can lead to a smoother traffic than Poisson.

5.4 Application based Sharing

The previous scenario did not require scheduling between the voice and video sources since they are not sharing the capacity. Now, we consider such a sharing. For the absolute delay requirement, sharing is still not possible although WFQ ensures that capacity unused by other classes is made available to active classes, but no guarantee can be provided. Therefore, we do not present results for absolute delay requirement. When considering average delay requirements, in the previous sections we have discussed that the bandwidth could be shared between the subclasses using the PDD scheduling based on the parameter δ^{AF} (see eq. 6). We now consider that the voice sources are allocated to AF subclass 1 and video sources are mapped to AF subclass 2. In Figures 9–11, we present the overprovisioning required to support both the AF subclasses 1 and 2, each having one, five and ten sources. Here also c_{LB}/r, c_{OO}/r and c_P/r refer to overprovisioning required when dimensioning for average delays using leaky bucket based model, using on-off based model and Poisson based model, respectively.

Note that for single voice and video source, design using leaky bucket filter requires more capacity than the on-off source based design model whereas for five and ten voice and video sources, on-off based model requires more capacity. This can be attributed to the way these two models derive the benefit of multiplexing. The LB based model better accounts for the multiplexing gain as compared to the On-Off based model. These interactions are subjects for further work.

6 Simulation Results

The design models that we have examined are all based on an average delay QoS requirement. In practice, however, average delay is not sufficient for many real-time applications such as voice or video where jitter must also be taken into account. We currently don't have design models that can take jitter into account so we need to evaluate whether the jitter remains acceptable in a system designed with an average delay method.

In this section, we present simulation results in order to study the delays encountered by the individual voice and video sources under various provisioning scenarios and compare them with the required delays for voice and video, respectively. We used ns-2 to conduct simulations. We only simulated the AF subclasses where multiple sources send packets to each class, and packets of each subclass are served in the order of their arrival while sharing bandwidth between the subclasses using PDD scheduling. The parameter for the PDD scheduling are determined based on the discussions in section 2. The simulation model for AF class is shown in Figure 12. We simulate a voice source using a two state on-off model where it generates packets with a deterministic inter-arrival time of 15 msecs in the on-state. On-periods are exponential with rate 2.5 and off-periods are also exponential with a rate 1.67. Each packet is of size 120 bytes. The video source is modeled using deterministic batch arrivals with batch inter-arrival time of 33 msecs. The number of packets in a batch are geometrically distributed with an average of 5 packets. In each

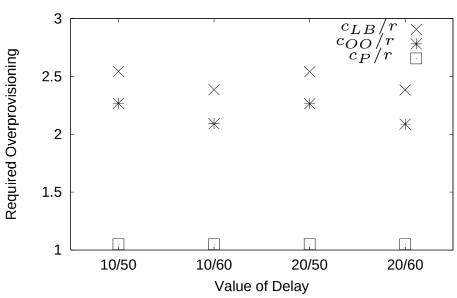

Figure 9: Required overprovisioning for single sources

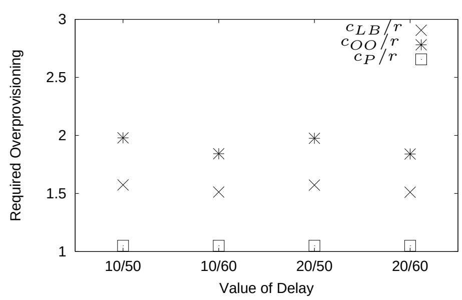

Figure 10: Required overprovisioning for 5 sources

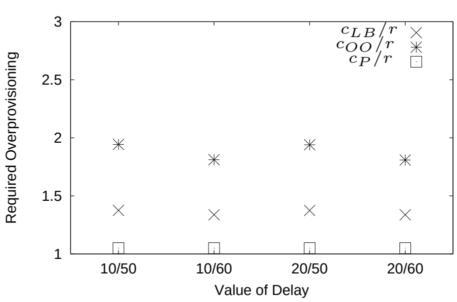

Figure 11: Required overprovisioning for 10 sources

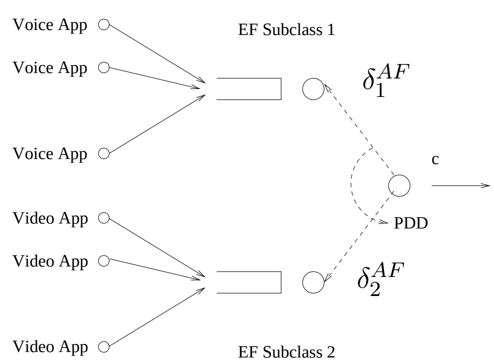

Figure 12: Simulation Model for AF class

burst, the last packet has size distributed as $uniform(0,1000)$ bytes. All other packets have 1000 bytes.

Here also c_{LB}/r, c_{OO}/r and c_P/r refer to overprovisioning required when dimensioning for average delays using leaky bucket based model, using on-off based model and Poisson based model, respectively. Observe that the capacity computed using these models along with PDD based scheduling were presented in Figures 9–11 for single, five and ten voice and video sources. Now we use that capacity for the simulation and compare in Figures 13–18 the delays for single, five and ten voice and video sources. We have plotted the observed mean delay and error bars corresponding to twice the sample standard deviation for voice and video sources. We also present a horizontal line showing the required average delay for each source.

Note that for the Poisson-based capacity model with single sources, the actual mean delay is many times the target delay, both for voice and video. Moreover, some voice packets can have a delay as high as 400 ms and will be useless at the receiver. For video also, packets can have delays as much as 1 sec. Such a capacity planning is not very useful and could lead to unsatisfied customers. When we multiplex 5 or 10 voice and video sources, the average delays get closer to the target delays and for 10 sources, they are even acceptable for both voice and video. However, there is still a large variance in the observed delays and voice packets could still have as high as 40 ms and video as high as 100 ms. Note that such high delays could be tolerable if they affect only a small number of packets.

Next, we consider on-off and leaky bucket based design models. Observe that both the approaches provide acceptable delays, average as well as average along with two times standard deviation. The values are smaller than the required delays and hence a significant fraction of packets belonging to voice and video sources will encounter less than required delays. These models remain consistent for single, five or ten sources and provide acceptable performance to individual sources. Note that the leaky bucket-based model provides delays which are less than the target for both voice and video, although it requires lesser capacity than the on-off-based models. Observe that not only the delays are acceptable but also the variance is quite small.

Based on these results, it can be argued that leaky bucket-based model could be used to determine required capacity for a source requesting an average delay QoS. When allocating capacity for a small number of sources, it can achieve the multiplexing gain and provides minimal capacity to meet the required delays.

7 Summary and Future Work

In this paper, we consider applications that do not require absolute delay guarantee but for whome the average delay requirement will be satisfactory. This could be due to high

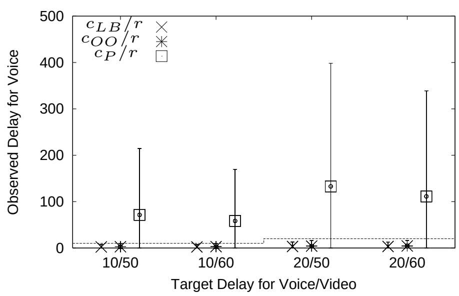

Figure 13: Delay for single Voice source

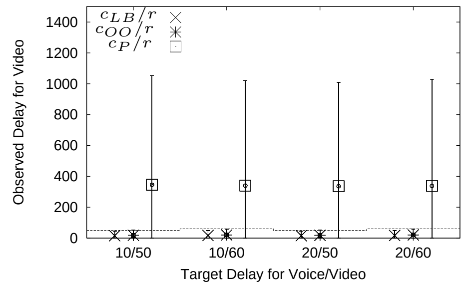

Figure 14: Delay for single Video source

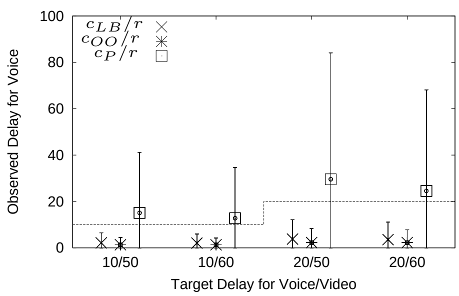

Figure 15: Delay for five Voice source

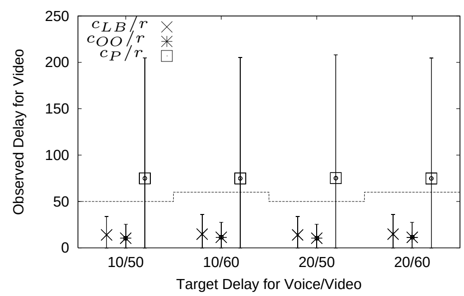

Figure 16: Delay for five Video source

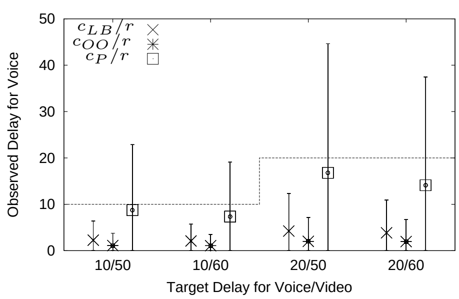

Figure 17: Delay for ten Voice source

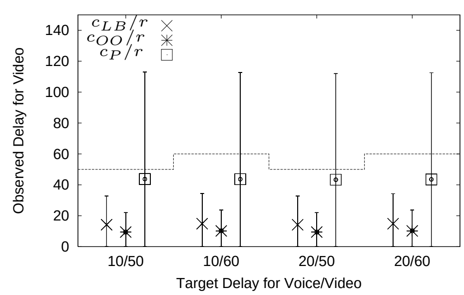

Figure 18: Delay for ten Video source

relative cost of absolute delay guarantee. For average delay guarantee, we consider AF service class of the differentiated services based architecture for QoS aware IP networks. Three kinds of models were accounted for: Poisson, On-Off and leaky bucket based. We present closed form expressions to determine the capacity required to ensure average delay to each service class. Using numerical results, we compare the required capacity for the three models with the capacity required to guarantee absolute delays for voice and video applications. It was observed that absolute delays require many orders of more capacity than the average delay models. We then use these capacity values to simulate a typical link and present results demonstrating the delays encountered by voice and video sources for these capacity models. It was also found that leaky bucket based model is suitable for classes with few sources. However, for networks with high number of individual sources, Poisson based models can also be used successfully.

We are in the process of incorporating other source structures such as three state, long range dependent, etc. into the design models. We are also in the process of extending the analysis to multi link and/or end-to-end network based models.

References

[1] R. L. Cruz. A calculus for network delay, Part I: Network elements in isolation. *IEEE Trans. Information Theory*, 37:114–131, 1991.

[2] R. L. Cruz. A calculus for network delay, Part II: Network analysis. *IEEE Trans. Information Theory*, 37:132–141, 1991.

[3] R. L. Cruz and H. Liu. End-to-end queueing delay in ATM networks. *Journal of High Speed Networks*, 3(4):413–427, 1994.

[4] C. Dovrolis and P. Ramanathan. A Case for Relative Differentiated Services and the Proportional Differentiation Model. *IEEE Network*, pages 2–10, Sept/Oct 1999.

[5] C. Dovrolis and P. Ramanathan. Dynamic class selection and class provisioning in proportional differentiated services. *Computer Communications*, 26:204–221, 2003.

[6] C. Dovrolis, D. Stiliadis, and P. Ramanathan. Proportinal Differentiated Services: Delay Differentiation and Packet Scheduling. *IEEE/ACM Transactions on Networking*, 10(1), 2002.

[7] H. Dupuis and B. Hajek. Simple Formulas for Multiplexing Delay for Independent Regenerative Sources. In *Proc. INFOCOM'93*, pages 28–35, 1993.

[8] C. Filsfils and J. Evans. Deploying Diffserv in Backbone Networks for Tight SLA Control. *IEEE Internet Computing*, pages 58–65, Jan/Feb 2005.

[9] F. Guillemin, N. Likhanov, R. Mazumdar, and C. Rosenberg. Extremal traffic and bounds for the mean delay of multiplexed regulated traffic streams. In *Proc. Infocom 2002*, June 2002.

[10] J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski. Assured forwarding PHB group. IETF RFC 2597, June 1999.

[11] V. Jacobson, K. Nichols, and K. Poduri. An expedite forwarding PHB. IETF RFC 2598, June 1999.

[12] F. Kelly. Notes on Effective Bandwidths. In F. Kelly, S. Zachary, and I. Ziedins, editors, *Stochastic Networks: Theory and Applications*, volume 4 of *Royal Statistical Society Lecture Notes Series*, pages 141–168. Oxford University Press, 1996.

[13] K. Nichols, S. Blake, F. Baker, and D. Black. RFC 2474: Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers. IETF, dec 1998.

[14] A. K. Parekh and R. G. Gallager. A generalized processor sharing approach to flow control in integrated service networks: the multiple node case. *IEEE/ACM Transactions on Networking*, 2:137–150, 1994.

[15] W. Tan and A. Zakhor. Packet classification schemes for streaming MPEG video delay and loss differentiated networks. In *Proc. Packet Video Workshop'01*, April 2001.

[16] K. White. RFC 2758: Definitions of Managed Objects for Service Level Agreements Performance Monitoring. IETF, feb 2001.

[17] L. Zhao, , J. Shin, J. Kim, and C.-C. J. Kuo. MPEG-4 FGS Video Streaming with Constant Quality Rate Control and Differentiated Forwarding. In *Proc. Video Communications and Image Processing*, pages 230–241, 2002.

[18] L. Zhao, J. Kim, and C.-C. J. Kuo. Rate Adaptation and error control for scalable video streaming over Diffserv Networks. In *Proc. Symposium on Electronic Imaging*, pages 20–25, 2002.