Network Service Selection for Distributed Multimedia Applications

Robert Simon, Arun Sood and Padmavathi Mundur
Department of Computer Science
George Mason University
Fairfax, VA 22030
{simon,asood, pmundur}@cs.gmu.edu

Abstract

Distributed multimedia applications require that the underlying communication and computational support systems provide real-time performance and reliability on an end-to-end basis. An important question in the development of system support for distributed multimedia is the type *of service offered to applications. This paper compares two commonly used network service types, fair queueing and deadline driven. We show that for a broad class of high-bandwidth distributed multimedia applications fair queueing networks outperform deadline-driven networks, in terms of network throughput, while still providing applications adequate end-to-end service. This result holds despite the fact that deadline-driven network services offer applications far greater flexibility in terms of control over end-to-end delivery delay.*

Keywords: Multimedia Systems, Interactive Multimedia

1 Introduction

The next generation of distributed multimedia applications will be supported by different combinations of high speed networks, high capacity storage devices, and powerful workstations and PCs. Taken together these various subsystems form distributed multimedia system (DMS) architectures. A DMS provides real-time support on an end-to-end basis for the computational and communication needs of distributed multimedia applications. These needs are expressed by applications in terms of Quality-of-Service (QoS) parameters, such as bandwidth or end-to-end delivery delay.

DMS architectures assure application-level QoS support by combining subsystem services on an end-to-end basis. A subsystem service level is defined as the range of possible values offered by a DMS for a particular QoS parameter, and the type of performance guarantee for the service offered, such as guaranteed, statistical or best-effort. A fundamental question in the design of DMS architectures is how to choose appropriate service levels. This decision involves a number of tradeoffs. For instance, a higher level or amount of a service from subsystem may be better from the point of view of the application, but may unnecessarily limit the throughput of a DMS architecture as a whole. The opposite is also true; high throughput in the system as a whole may mean lower QoS levels for the application.

This paper addresses the architectural problem of end-to-end network-level service selection. We focus on modeling and comparing two distinct network service regimes, fair queueing and deadline-driven, in the context of a DMS that uses admission control and traffic regulation in order to guarantee end-to-end QoS. We show that for two important classes of distributed multimedia applications – high bandwidth interactive multimedia and high-bandwidth Video-on-Demand (VoD) – fair queueing service can simultaneously provide a higher level of network throughput and an appropriate level of application QoS support, as compared to deadline-driven policy. This is in spite of the fact that deadline-driven service policies are considered more powerful and flexible than fair queueing polices, because they can provide a much higher level of control over the end-to-end delay experienced by the application. Our analysis helps simplify network service selection options by showing that the added complexity of deadline-driven scheduling is not always required.

The paper proceeds as follows: Section 2 describes the application and DMS model. Section 3 presents the details of the fair queueing and deadline-driven network. Section 4 discusses a numerical comparison of these two network policies in the context of their support for interactive and VoD applications. Finally, Section 5 concluding remarks.

2 DMS Architectures and Applications

The DMS architecture is modeled as consisting of three distinct subsystems: a real-time producer of synchronized multimedia data, a communication network, and a real-time data synchronized information consumer. Real-time requirements for multimedia data transfer require tight coordination between the producer, network and consumer. The DMS architecture guarantees that synchronization is maintained by enforcing resource allocation and admission control policies within the various subsystems. This means that resources must be available within each subsystem before a new request is admitted into the system.

Applications use the DMS by making service requests. If a new request goes into service it is referred to as a *multimedia channel*. The DMS takes the request and runs a series of admission control algorithms at each of the underlying subsystems. For instance, a client-server application such as VoD requires that the video server be able to provide synchronized and timely retrieval of video data. Within the communication network, admission control requires a routing *path* to be set up and resources reserved along that path. The required bandwidth must be guaranteed for a bounded delay requirement. DMS architectures use unified admission control algorithms that reject multimedia channel requests if any of these requirements cannot be met.

Each multimedia channel is specified by two parameters. The first, D, is the upper bound on the permissible end-to-end delay through the network. The second, R_{pl}, is the the required playback rate at the consumer. The application sends the multimedia channel specification to the DMS, which then runs the admission control algorithms at each of the subsystems. If all of the admission control tests are passed, then the DMS guarantees an end-to-end delivery bound of $\bar{D}$ and a guaranteed data rate of R_{res} delivered to the consumer. Synchronized, starvation-free service is provided by the DMS as long as the next two conditions are met:

Application Synchronization Conditions

1. $D \leq \bar{D}$
2. $R_{pl} \leq R_{res}$

Distributed interactive multimedia applications, such as tele-conferencing and tele-immersion, and distributed client-server multimedia applications, such as Video-on-Demand, have different possible values for $\bar{D}$ and R_{pl}. For instance, an interactive tele-conferencing application typically requires that the underlying system provide a $\bar{D}$ of no more than 150 ms. A delay bound beyond 150ms negatively affects the participants in the application. The R_{pl} value depends on the compression technqiue, and can range anywhere from 1.5 Mbps to more than 20 Mbps. We will see in the next section that different network service policies differ as to how $\bar{D}$ and R_{res} are set.

3 Network Services

Our targeted network environment is capable of providing deterministic guarantees for $\bar{D}$ and R_{res}. Deterministic QoS guarantees for synchronized, starvation-free service for each multimedia channel can be provided by one of a number of guaranteed service disciplines. Two of the most important types of guaranteed service disciplines are *Fair Queueing* and *Deadline-Driven*. These policies differ in how they provide end-to-end QoS. In particular, Fair Queueing policies induce a strong coupling between $\bar{D}$ and R_{res}, while Deadline-Driven policies allow these two parameters to be decoupled from each other.

Deterministic network service policies guarantee end-to-end QoS of by first inserting a *traffic regulator* between the producer and the network [1], [2]. The regulator is needed to police the traffic injected by each multimedia channel into the network. The regulator guarantees that over any arbitrary time interval $t \geq 0$, the number of bits appearing at its output is $min(M+pt, b+rt)$ where b, r, and p are design parameters of the regulator such that $p \geq r$ and $b \geq M$. The parameters b and r control burstiness of the flow into the network. p and M are hardware parameters based on the network interface to the server. The parameter b is often called the *Token Bucket* burst factor, since it can be used to characterize short term bursts of data while regulating long-term peak rates. This provides a high level of flexibility for modeling bursty multimedia data production.

3.1 WFQ Network Component

Fair queueing disciplines operate based upon delay and fairness principles only. Fair queueing algorithms are scheduling and multiplexing disciplines capable of providing end-to-end delay and bandwidth guarantees on a per connection basis. They work by assigning packet transmission schedules for all backlogged connections. A backlogged connection is a connection which has traffic at the server which is ready to transmit. Depending upon a connection's priority or *weight*, fair queueing algorithms provide the same normalized service to all backlogged connections within a small interval of time.

A realizable implementation of a Fair Queueing system is called *Weighted Fair Queueing* (WFQ). WFQ works by simulating idealized fluid flow system and serving packets based upon their transmission times in the idealized fluid system. When a (b r p M) regulator is used, WFQ guarantees an end-to-end delay bound $\bar{D}$ given by [2]:

$$\begin{aligned}
\bar{D} &= \frac{b-M}{R_{res}}\frac{p-R_{res}}{p-r} + \frac{M+C_{tot}}{R_{res}} + D_{tot} \quad \text{if } R_{res} < p \\
\bar{D} &= \frac{M+C_{tot}}{p} + D_{tot} \quad \text{otherwise}
\end{aligned}$$

Since R_{res} is almost always less than p, the maximum rate of the network interface card, we only consider the first case. Here r is equal to the reserved rate at the network,

$r = R_{res}$. After substituting for b and r and some simplification, the end-to-end delay is:

$$\bar{D} = \frac{b}{R_{res}} + \frac{C_{tot}}{R_{res}} + D_{tot}$$

where $C_{tot} = \sum_{i=1}^{J} C_i$ and $D_{tot} = \sum_{i=1}^{J} D_i$. The values for C_i and D_i depend on the Fair Queueing scheduling mechanism implemented at the network. Using WFQ, the maximum delay bound per packet for a flow with a guaranteed rate of R_{res} and routed on a path that has J links is given by [5]:

$$\bar{D} = \frac{b}{R_{res}} + \frac{(J-1)M}{R_{res}} + \sum_{i=1}^{J} \frac{M_{max}}{A_i} \tag{1}$$

where M is the maximum size of the packet for the flow, M_{max} is the maximum size of a packet permitted in the network, and A_i is the overall bandwidth on link i.

Using Equation (1), we now derive R_{res} corresponding to an input delay bound $\bar{D}$. This required minimum rate, $R_{res} \geq R_{pl}$, is used in the admission control test before admitting a new request on the network.

$$R_{res} = \frac{b + (J-1)M}{\bar{D} - \sum_{i=1}^{J} M_{max}/A_i} \tag{2}$$

Admission control tests for new multimedia channels are run on a link by link basis over the routing path. Suppose that link i has currently accepted K multimedia channels $1, 2, \ldots, K$. The i^{th} multimedia channel has a reserved rate of R^i_{res} computed using (2). If the *new* channel requests a rate of R^{new}_{res}, it will be admitted only if the following relation holds:

$\sum_{i=1}^{K} R^i_{res} + R^{new}_{res} \leq A_i$

3.2 Deadline-Driven Scheduling

A deadline-driven service policy provides an alternate mechanism for supporting multimedia channels synchronization needs. Each packet is assigned a deadline on each of the links in its routing path. At each link, packets are scheduled for transmission according to their deadlines. Deadline-driven scheduling policies are considered amongst the most powerful class of network scheduling policies because, unlike WFQ, they allow applications to decouple the specification of R_{res} from $\bar{D}$.

One deadline-oriented scheduling mechanism that is appropriate for network implementation is called Non-preemptive Earliest Deadline First scheduling (NEDF). NEDF works by assigning higher priority to packets from connections with smaller deadlines. The EDF policy was originally proposed by Liu and Layland [3]. In NEDF packet transmission is non-preemptive; that is, once a packet begins transmission that transmission is never interrupted. The next packet is scheduled for transmission once a packet in service is completely transmitted. At that time, the packet with the highest priority is picked to transmit next. NEDF is similar to many other proposed types of deadline service, such as Delay-Earliest Deadline First.

In order for a multimedia channel to use NEDF, the DMS must specify a set of per-link delay values. This can be done as follows: suppose a routing path has J links on it. If the application specified end-to-end delay bound is $\bar{D}$, then each link i has a per channel delay bound δ_i set as $\delta_i = \frac{\bar{D}}{J}$.

Let $\delta_{i,z}$ be the per link delay bound for multimedia channel z at link i. A set of K multimedia channels is said to be schedulable over a link i if, for all z, $1 \leq z \leq K$, the maximum delay experienced by any packet from channel z is no greater than $\delta_{i,z}$.

We now show the schedulability condition for multimedia channels in an NEDF environment. Let τ_i be the amount of time it takes to transmit one bit at link i. Let M be the maximum packet size in bits. Then the *maximum* time P_i to transmit one packet is given by $P_i = M \cdot \tau_i$. Consider the case of K multimedia channels accepted for service at link i. Suppose channel z, $1 \leq z \leq K$, has a transmission timing deadline of $\delta_{i,z}$ and a maximum number of packets over that time, which require a total time of $U(i,z)$ to transmit. We explain below how $U(i,z)$ is calculated for an arbitrary multimedia channel. Under the original EDF policy at any single point in time the packet from the multimedia channel with the closest deadline is selected for transmission. Under the original Liu and Layland formulation a sufficient condition to guarantee the schedulability of K multimedia channels is that the following holds:

$$\sum_{z=1}^{K} \frac{U(i,z)}{\delta_{i,z}} \leq 1 \tag{3}$$

The schedulability condition as expressed in Equation (3) assumes that packet transmission is preemptive. However, packet transmission in high-speed networks is typically non-preemptive. The policy of not preempting packet transmission may lead to a priority violation, whereby a higher priority packet's transmission is delayed by the transmission of lower priority packets. It is therefore necessary to derive a new schedulability condition for multimedia channel traffic. The new schedulability test combines this source of priority violation into a single equation, which is expressed below as a Theorem.

Theorem *Consider a set of K multimedia channels. Let P_i be the time required to transmit one packet of maximal size. Let multimedia channel z, $1 \leq z \leq K$, have a transmission* *timing deadline at link i of $\delta_{i,z}$ and a maximum number of packets over that time which require a total time of $U(i,z)$ to transmit. Without loss of generality, let $\delta_{i,1}$ be the multimedia channel with the highest priority. Then the K multimedia channels are schedulable at link i if the following condition holds:*

$$\sum_{z=1}^{K} \frac{U(i,z)}{\delta_{i,z}} + \frac{P_i}{\delta_{i,1}} \leq 1 \tag{4}$$

The proof of the general form of this theorem is contained in [4]. For space reasons, we omit the details of that proof. We simply note that the proof proceeds by introducing the idea of *shadow activities*, such that the behavior of the non-preemptive system can be modeled as a fully preemptive hard real-time scheduling system.

It now remains to show the value of $U(i,z)$. This is done by considering the maximum number of bits which can arrive from the channel z at link i over the time period $\delta_{i,z}$. The upper bound on the traffic arrival pattern is given by the value of the (b r p M) regulator. Assume that multimedia channel z has (b r p M) parameters of b_z and p_z. Then

$$U(i,z) = (b_z + p_z) \cdot \delta_{i,z} \cdot \tau_i \tag{5}$$

4 Numerical Analysis

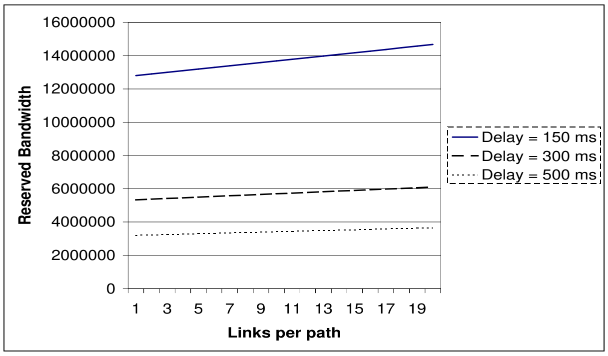

Figure 1. Highly bursty traffic, (b r p M) buffer size b of 1.6 Mbits

In this section we quantify the impact interactive and video-on-demand multimedia applications on WFQ and NEDF network components within the DMS architecture. We first consider three multimedia applications described solely by D, the end-to-end delay specifications. The first

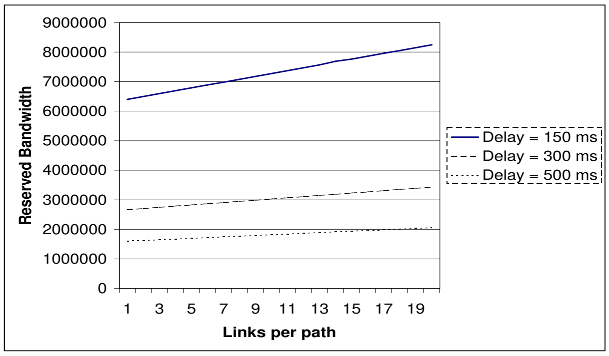

Figure 2. Bursty traffic, (b r p M) buffer size b of 800 Kbits

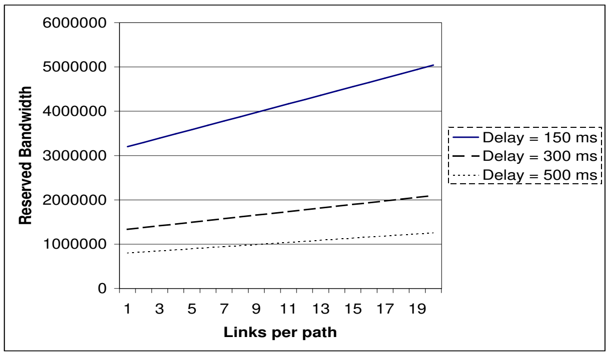

Figure 3. Moderately bursty traffic, (b r p M) buffer size b of 400 Kbits

application has a D value of 150 ms, corresponding to an interactive application. The second application has a D value of 300 ms, corresponding to a non-interactive, VoD application. The third application has a D value of 500 ms, also corresponding to a non-interactive VoD application with a higher permissible end-to-end delay value.

Since WFQ provides a strong coupling between $\bar{D}$ and R_{res}, our first set of comparisons is designed to judge the effect on the network of this coupling. In particular, we are interested in seeing if using WFQ results in a significant decrease in network throughput by forcing excessively high R_{res} values. A value of R_{res} is considered high if it is greater than the applications R_{pl} value. Multimedia aplications have wide range of possible R_{pl} values. For instance one way audio typically requires about 64 kbps, MPEG-1 compressed video needs 1.5 Mbps, while MPEG-2 video may require 15 Mbps and higher.

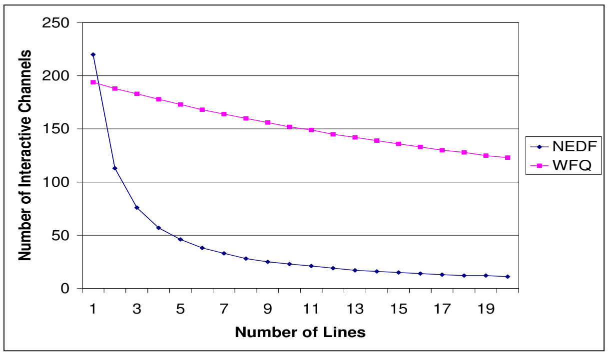

Figure 4. Comparison of the multimedia channel capacity of WFQ versus NEDF for an interactive application. Shown as a function of the number of links in the routing path.

Based on the above three applications we numerically determined the value of R_{res} as a function of the number of hops in the routing path and the (b r p M) parameter b. R_{res} is derived from Equation 2. The first result, shown in Figure 1, illustrates the effect of setting b to the extremely high burst value b of 1.5 Mbps. For an interactive application, the minimal network reserved value is between 12 and 14.5 Mbps, while for the two VoD-like applications the maximum network reserved value is around 4 Mbps. This result shows that for very bursty multimedia applications WFQ is a good service choice for high bandwidth applications, but forces a significant over-reservation of bandwidth resources for extremely bursty low bandwidth applications.

The next result, shown in Figure 2, illustrates the effect of setting b to a high burst value of 800 Kbits. For an interactive application, the minimal network reserved value is between 6.4 and 8.4 Mbits, while for the two VoD-like applications the values are no greater than around 3.4 Mbits. This result shows that as the burst level goes down WFQ becomes a better choice.

The third result, shown in Figure 3, illustrates the effect of setting b to a lower burst value of 400 Kbits. For an interactive application, the minimal network reserved value ranges between 3.2 and 5.1 Mbits, while for the two VoD-like applications the values are no greater than around 2.1 Mbits. Under this circumstance, WFQ is a resonable for most applications requiring an R_{pl} of over 1 Mbps.

The final result, shown in Figure 4 compares the number of multimedia channels for an interactive application requiring 1.5 Mbps with an end-to-end delay of 150 ms that can be supported over multi-hop routing paths for NEDF and WFQ. The value of b is set to 40 Kbits. It can be seen that for this low bandwidth application WFQ dramatically outperforms NEDF in terms of the number of channels supported, once the routing path exceeds one hop.

One of the reasons that NEDF in general performs so poorly is because as the number of hops increases the scheduling requirements at each hop drastically increase. In fact, the NEDF worst case analysis for the value of $U(i, z)$ in Equation 5 may be overly conservative. However, a more accurate traffic arrival model would require the introduction of an more complex type of regulator, both at the network edge and within the network. This may not be a realistic alternative for high-speed networks, which need to limit the amount of processing and protocol overhead per packet.

5 Conclusion

This paper has provided a quantitative comparison between two important network service classes, Weighted Fair Queueing and Non-preemptive Earliest Deadline First. We first showed that in a WFQ network the end-to-end delay bounds and the rate bounds are coupled. We then showed that these values are decoupled in an NEDF network and presented a basic schedulability result for NEDF networks. Next, we showed that, despite the flexibility of NEDF over WFQ, for many classes of distributed multimedia applications WFQ is entirely sufficient on an end-to-end basis, and in fact provides a higher throughput over NEDF. This basic result should simplify the design and selection of network service options in future distributed multimedia systems.

References

[1] R. Braden, D. Clark and S. Shenker, "Integrated Services in the Internet: an Overview," RFC 1633, 1994, work-in-progress.

[2] L. Georgiadis, R. Ruerin, V. Peris, and R. Rajan, "Efficient Support of Delay and Rate Guarantees in an Internet," *ACM SIGCOMM*, pp. 106 - 116, August, 1996.

[3] C. Liu and J. Layland, "Scheduling Algorithms for Multiprogramming in a Hard Real-time Environment," *JACM* 30(1), 1973.

[4] R. Simon, "Predictable Path Establishment in Multimedia Communication Network," to appear in *IEEE International Conference on Multimedia Computing Systems*, Florence, Italy, June 1999.

[5] A. Parekh and R. Gallager, "A Generalized Processor Sharing Approach to Flow Control – The Multiple Node Case," *IEEE Transactions on Networking*, vol. 2, no. 2, pp. 137 - 150, 1994.