Providing Time of Service Guarantees in Video-On-Demand Servers*

Nabil J. Sarhan Chita R. Das
Department of Computer Science and Engineering
The Pennsylvania State University
University Park, PA 16802
Phone: (814) 865-0194
{sarhan,das}@cse.psu.edu

1. INTRODUCTION

Multimedia applications have become an essential part of the World Wide Web (WWW). These applications can be classified into three main classes: *video-on-demand* (VOD), *live streaming*, and *interactive real-time* (such as Internet telephony and video conferencing). The application of interest in this paper is VOD. Besides its popularity in the entertainment field, VOD has been of great importance in education and distant learning in particular. By contrast with broadcast-based systems such as cable TV, VOD servers enable customers to watch the movies they want at the times of their choosing and allow them to apply VCR-like operations.

Unfortunately, the requirements of the real-time playback and the high transfer rates highly constrain the throughput of a VOD server. Resource sharing techniques address this problem by servicing multiple requests from a common set of resources.

The achieved resource sharing depends greatly on how a VOD server schedules the waiting requests. A VOD server maintains a waiting queue for each movie to facilitate scheduling, selects one queue for service whenever the necessary resource become available, and services all requests in the selected queue together. Scheduling policies for VOD servers include *First Come First Serve* (FCFS), *Maximum Queue Length* (MQL), and *Maximum Factored Queue Length* (MFQL). FCFS selects the queue with the oldest request, whereas MQL selects the longest queue, and MFQL selects a queue based on both queue lengths and movie access frequencies. (A detailed investigation of these policies can be found in [1].) This paper focuses on VOD servers that employ batching as the primary resource sharing technique. Batching accumulates the requests for the same movies and services them together by utilizing the multicast facility. Although other resource sharing techniques may offer better resource sharing, they require higher bandwidth and buffer space at the client, reduce the quality of the initial parts of the presentations, or increase the overall cost of the server [1].

By providing time of service guarantees, a VOD server can enhance customer-perceived quality of service (QoS) and can influence customers to wait, thereby increasing its throughput. Unlike most other policies, FCFS is believed to provide hard time of service guarantees [2]. We found, however, that FCFS may violate these guarantees because not all customers continue to wait for services. We also found that FCFS is incapable of producing accurate time of service guarantees.

To address the shortcomings of FCFS, we propose a new scheduling policy, called *Next Schedule Time First* (NSTF) and demonstrate its effectiveness through simulation.

2. NEXT SCHEDULE TIME FIRST (NSTF)

The proposed NSTF policy assigns schedule times to incoming requests and guarantees that they will be serviced no later than and accurately as scheduled. In contrast with FCFS, NSTF schedules requests based on the assigned schedule times rather than the arrival times. In the absence of VCR-like operations, a VOD server knows exactly when resources will become available for servicing new requests because each running stream requires a fixed playback time. Hence, when a new request calls for the playback of a movie with no waiting requests, NSTF assigns that request a new schedule time that is equal to the closest un-assigned completion time of a running stream. If the new request, however, is for a movie that has already at least one waiting request, then NSTF assigns it the same schedule time assigned to the other waiting request(s) because all these requests can be serviced together. Applying VCR-like operations, which are typically supported by using contingency channels, leads to early completions and thus servicing some requests earlier than scheduled. When all customers waiting for the playback of a movie defect (i.e., cancel their requests), their schedule time become available and can be used by other customers.

We consider here the implementation of NSTF that assigns the freed schedule times to the waiting requests for a selected movie. All the requests in the selected movie must have waiting time guarantees beyond a certain threshold. If multiple eligible movies exist, this implementation selects the one with the largest number of waiting requests and thus has the advantage of combining the benefits of FCFS and MQL. This implementation notifies the requests with time of service guarantees beyond a certain threshold that they may be serviced earlier and provides all other requests with hard time of service guarantees.

3. PERFORMANCE EVALUATION

We analyze the effectiveness of NSTF through simulation. We consider five performance metrics: server throughput,

*This research was supported in part by NSF grants CCR-9900701, CCR-0098149, CCR-0208734, and EIA-0202007.

WWW2003, May 20–24, 2003, Budapest, Hungary.
ACM xxx.

the average request waiting time, the number of violations of time of service guarantees, the average deviation from the time of service guarantees, and unfairness (against unpopular movies). The first metric is the most important, and the second comes next in importance.

3.1 Simulation Platform and Workload Characteristics

We have developed a simulator for a VOD server that supports various scheduling policies. The simulation terminates after a steady state analysis with 95% confidence interval is guaranteed. Like most prior studies, we assume that the arrival of the requests to a VOD server follows a Poisson Process and that the accesses to the movies follow a Zipf-like distribution with a value of 0.271 for the skewness parameter. We study a VOD server with 120 two-hour long movies and examine the server at different loads by fixing the request arrival rate at 40 requests per second and varying the number of server channels from 500 to 1750. (A channel is a set of resources needed to deliver a multimedia stream.) We characterize the waiting tolerance of customers by two models. In *Model A*, the customers who receive time of service guarantees will wait for service if their waiting times will be less than 5 minutes; the waiting times of all other customers follow an exponential distribution with a mean of 5 minutes. *Model B* is used in [2] and is the same as Model A except that a truncated normal distribution with a mean of 5 minutes and a standard deviation of 1.67 minutes is used in place of the exponential distribution. Truncation excludes the waiting times that are negative or greater than 12 minutes. In both models, we assume that the customers who are expected to wait longer than 12 minutes will defect immediately.

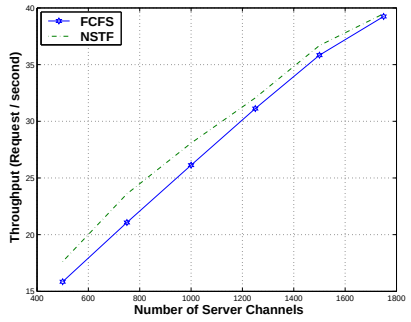
3.2 Main Results

The results demonstrate that FCFS may violate its time of service guarantees, but these violations happen very occasionally (especially for high server capacities) because FCFS tends to overestimate significantly these guarantees. In fact, overestimating these guarantees is the most critical problem of FCFS. The actual times of service with FCFS differ from the time of service guarantees by 20 seconds to more than 4.5 minutes on the average, whereas the average difference in the case of NSTF is within 0.1 second.

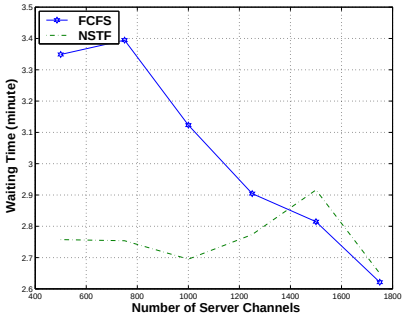
The results also show that NSTF achieves higher throughput than FCFS under both tolerance models. NSTF also generally leads to shorter waiting times when the tolerance follows Model B, but the waiting times are slightly shorter with FCFS when the tolerance follows Model A. As expected, FCFS is fairer under both models. Figure 1 compares FCFS and NSTF in terms of the achieved throughput and waiting times when the tolerance follows Model B.

When compared with policies that do not provide time of service guarantees (MQL and MFQL), NSTF generally leads to the highest throughput under model A. It also achieves the highest throughput under Model B but only when the overall request turn-away percentage is less than 20, which is the most likely operating region as much larger turn-away percentages would be unacceptable. MQL and MFQL, however, perform better in terms of the waiting times as expected.

4. CONCLUSIONS



(a) In Throughput



(b) In Waiting Time

Figure 1: Comparing NSTF with FCFS (Model B)

In this paper, we have proposed a new scheduling policy, called *Next Schedule Time First* (NSTF), which assigns schedule times to incoming requests and performs scheduling based on these schedule times. We have demonstrated the effectiveness of NSTF through simulation. The results can be summarized as follows. (1) NSTF always meets the time of service guarantees and produces very accurate schedule times. The average deviation of the actual times of service from the schedule times are within 0.1 second. In contrast, FCFS may violate its time of service guarantees, and these guarantees differ from the actual times of service by 20 seconds to more than 4.5 minutes on the average! (2) NSTF also achieves higher throughput and, in certain situations, shorter waiting times than FCFS. (3) By motivating customers to wait, NSTF outperforms MQL and MFQL (both of which cannot provide time of service guarantees) in terms of throughput for one of the studied models of waiting tolerance. For the other model, NSTF also achieves the highest throughput but only within the most likely operating region of the server. NSTF is also fairer than MQL and MFQL for high server capacities. As expected, NSTF leads to longer waiting times than MQL and MFQL as a reasonable price for the enhanced throughput and QoS.

We conclude that NSTF not only can provide hard time of service guarantees and very accurate schedule times, but also can deliver outstanding performance benefits.

5. REFERENCES

[1] Nabil J. Sarhan and Chita R. Das. A Simulation-Based Analysis of Scheduling Policies for Multimedia Servers. To appear in *Proceedings of the 36th Annual Simulation Symposium*, March 2003.

[2] A. K. Tsiolis and M. K. Vernon. Group-Guaranteed Channel Capacity in Multimedia Storage Servers. In *Proceedings of the ACM SIGMETRICS Conference on Measurements and Modeling of Computer Systems*, pages 285-297, June 1997.