Scheduling Video Programs in Near Video-on-Demand Systems *

Emmanuel L. Abram-Profeta and Kang G. Shin

Real-Time Computing Laboratory
Department of Electrical Engineering and Computer Science
The University of Michigan
Ann Arbor, Michigan 48109-2122
{*abram,kgshin*}@*eecs.umich.edu*

Abstract

This paper presents an analytical (in contrast to commonly-used simulations) approach to program scheduling in near video-on-demand (NVoD) systems. NVoD servers batch customers' requests by sourcing the same material at certain intervals called *phase offsets*. The proposed approach to analytical modeling integrates both customers' and service-provider's views to account for the tradeoff between system throughput and customers' partial patience. We first determine the optimal scheduling of movies of different popularities for maximum throughput and the lowest average phase offset. Next, we deal with quasi video-on-demand (QVoD) systems, in which programs are scheduled based on a threshold on the number of pending requests. The throughput is found to be usually greater in QVoD than in NVoD, except for the extreme case of nonstationary request arrivals. This observation is then used to improve throughput without compromising customers' QoS in terms of average phase offset and the corresponding dispersion.

Index Terms — Near video-on-demand (NVoD), quasi video-on-demand (QVoD), partially patient customers, batching, video server throughput.

1 Introduction

The need to batch requests for the same movie title together in a video-on-demand (VoD) system has long been recognized for scalability and immediate deployment [10]. Reduction of per-customer system cost and improvement of system scalability can both be achieved by delaying the VoD server's response to customers' requests made during the same *batching interval* and hence enabling the server to multicast the requested video. In this paper we investigate a batching strategy which sources the same material at equally-spaced intervals, called *phase offsets*. This kind of VoD service, in which subscribers who order a particular movie to start within a specific time window are grouped together, is termed "Near-VoD" (NVoD) [2, 11].

As mentioned in [2], the main advantage of NVoD systems over other batching policies is that, by keeping the batching interval nearly constant per movie title, it is possible to provide customers with *limited and scalable VCR capability*. It is usually recognized that full support for continuous interactive functions in a multicast VoD system can only be achieved by dedicating a channel per customer, thereby seriously limiting system scalability. In NVoD systems, on the other hand, limited continuity in VCR actions can be provided by caching a small amount of video data (e.g., 5 minutes' worth of video) in a buffer located close to the client, for instance, in the customers' premise equipment (CPE). This buffer can then be accessed without removing the customer from the multicast group. Moreover, staggered phase offsets support for discontinuous VCR actions is provided in NVoD by allowing customers to specify the length of video they want to skip, possibly in integer multiples of the phase offset duration. In this case, the NVoD server will reassign the customer to the multicast group whose play-out point is the closest to that requested by the customer. Multicast group membership may also change in a quasi-continuous fashion when customers attempt to access video data outside the CPE buffer as a result of a continuous VCR action; in such a case, the NVoD server will assign customers to an adjacent multicast group. Support for intermittent VCR actions can thus be provided in NVoD without limiting system scalability.

The number of concurrent channels supported by a VoD server is usually limited by the underlying storage capacity and organization. This concurrent channel capacity has to be shared among different movie titles of heterogeneous popularities in the system. We define a *schedule* of video programs in NVoD as the assignment of phase offsets to each movie title. For a concurrent channel capacity of s, a schedule corresponds to a partition $[s_1, \cdots, s_N]$, $\sum_{m=1}^{N} s_m = s$ such that the phase offsets are given by $[\frac{L}{s_1}, \cdots, \frac{L}{s_N}]$, where L is the movie length. The number of phase offsets allocated to each movie title depends on its popularity. For instance, in the case of a rarely-requested movie, the NVoD server will probably allocate a very few channels to it so that more channels may be allocated to popular titles, and therefore, more customers may be served. Customers' willingness to wait for service will also impose constraints on the maximum acceptable phase offset, and hence on the channel allocation. In this paper, we present heuristics to determine schedules that optimize NVoD server objectives, such

*The work reported in this paper was supported in part by the National Science Foundation under Grant MIP-9203895 and the Office of Naval Research under Grant N00014-94-1-0229.

as maximum throughput, defined as the average number of customers served per movie transmission, and minimum phase offset, indicating customers' QoS.

The paper is organized as follows. We first outline how NVoD can be implemented in practice. Next, we analytically derive expressions for the throughput under general conditions of customers' patience and request rates. Based on our analysis, we then present and compare various heuristics to determine optimal schedules. After relaxing the assumption of constant phase offsets, we will show that the throughput of an NVoD server can be improved by using a threshold-based admission control of customers' requests. The cost for such an improvement is that the functionality of discontinuous VCR actions can only be provided *in an average sense*. However, we will show that a dramatic improvement in throughput can be achieved for a reasonably low dispersion of the phase offsets.

2 Implementation of NVoD

Multicast groups in NVoD can actually be formed by having multiple CPEs listen to the same channel. Assuming a frequency division multiplexed system such as in the already-available CATV, a multicast group is identified by a particular channel, and a customer can join the group by tuning to the appropriate frequency [14]. Thus, considering possible CATV support for multimedia services delivery to subscribers, NVoD is a good candidate for future implementation. In order to source the same material at equally-spaced intervals, the NVoD server needs to retrieve video data from a source device with multi-readout capability, so that multiple requests may share the same buffer and read-write head. A technology particularly adapted to the interleaved retrieval of videos in NVoD, is coarse-grained striping on disk arrays (e.g., RAID-5) [6, 13], which interleaves each access to several disks, so one disk may be read while seeking others. This form of data striping reduces the access time and results in a lower-cost VoD server compared to other forms of storage.

In order to understand how limited support for VCR actions is provided in NVoD, one can discriminate two levels of interactivity according to the continuity experienced by customers [2]. *Continuous* interactive functions allow a customer to fully control the duration of interaction, whereas *discontinuous* interactive functions can only be specified for durations that are integer multiples of a predetermined time increment, which, in NVoD, corresponds to a phase offset. Support for continuous interactive functions in NVoD is provided by caching a limited amount of video data close to the user, for instance, in a CPE buffer, so that the user may access it with a very low latency during interaction [2]. Discontinuous actions happen in two possible scenarios. First, customers may suddenly exceed CPE buffer capacity while performing a continuous action, e.g., rewinding for too long. In that case, the NVoD server will simply transfer the customer to the multicast group corresponding to an adjacent phase offset. Second, customers may directly specify the length of video they want to skip, in which case the NVoD server will determine the multicast group whose playout point is the closest to that requested by the customer. Note that support for both continuous and discontinuous interactions in NVoD is independent. Continuous actions are taken care of by the CPE buffer, while the general operation of the NVoD server is to assign the customer to another multicast group when needed.

Let's consider the general operation of a CPE buffer during a continuous VCR action. Video frames are received in a synchronous fashion, and those frames already displayed ("past frames") are kept within the buffer, for reverse search and rewind capability. Initially, the playout point will correspond to the most recently-received video frame. Upon pause, stop, fast reverse or rewind within the CPE buffer, the playout point will change. The CPE buffer manager can then attempt to keep the playout point as close to the middle of the CPE buffer as possible, so there should always be past frames available for reverse search and unplayed frames, available for pause or fast search. This would be a natural choice if "backward" interactions (rewind and reverse search) are as likely as "forward" interactions (fast forward and fast search). If interactive access is dominated by "backward" interactions, the playout point should rather correspond to the most recently-received video frame or GOP. Note that the only mechanism available to the CPE buffer for controlling the playout point is to discard old frames at a rate slower or faster than the arrival rate of frames from the server. Efficient CPE buffer management should ultimately discard past frames in such a way that a large number of VCR actions can be satisfied without the NVoD server's support. Both customers' interaction latency and load on the NVoD server will then be minimized. Efficient CPE buffer management is still an open issue, as it is highly dependent on customers' behavior.

In general, the size of the CPE buffer will be subject to affordability constraints. As an example, 5 minutes of MPEG-1 compressed video at 1.5 Mbps represent approximately 56.25 MB, which, stored in DRAM for fast access, should cost around $200 in 1997. On the other hand, the phase offset is adjusted by the NVoD server depending on the movie popularities, so more channels are allocated to the most frequently-requested movie titles. Thus, an important distinction has to be made between movie titles, depending on the relative size of the CPE buffer compared to the phase offset. The NVoD server will probably allocate a very few channels (e.g., $2-3$) to a rarely-requested movie, and the phase offset will be much larger than what the CPE buffer can hold. In this case, support for both continuous and discontinuous VCR actions will be extremely limited, and it is safe to assume that group changes will not occur unless expressly requested by customers. For more popular movies, the CPE buffer should be large enough to hold a phase offset's worth of frames so that a group change can possibly be performed in a continuous fashion. Even in this case, the CPE buffer cannot always guarantee a smooth transition between adjacent multicast groups. To illustrate this fact, let's consider a viewer who initiates a fast search shortly after the beginning of a movie. As no future frame will be available in the CPE buffer, the multicast group change will cause the viewer to experience a jump in phase offset. If the CPE buffer attempts to keep the playout point in the middle of the buffer, whereas the play function can be resumed as soon as the the first frame from the new multicast group is fetched, other interactions such as reverse search, pause, stop and slow motion will become fully functional again only once half a phase offset has been fetched. In summary, whether the phase offset is larger or smaller than the CPE buffer capacity, continuous VCR actions can only be provided intermittently and without any guarantee on the discontinuities experienced by customers (even with proper CPE buffer management). Thus, the assumption of a constant phase offset need not hold to guarantee QoS in VCR actions. We will use this observation in Section 6 to show that it is possible to provide the same granularity of

$$P_{M/M/\infty}(t,i,n) = e^{-\frac{\lambda_m}{\alpha}(1-e^{-\alpha t})} \cdot \sum_{k=0}^{n} \left\{ \frac{\left(\frac{\lambda_m}{\alpha}\right)^{n-k}}{(n-k)!} (1-e^{-\alpha t})^{n-2k+i} e^{-k\alpha t} \binom{i}{k} \right\} \quad (1)$$

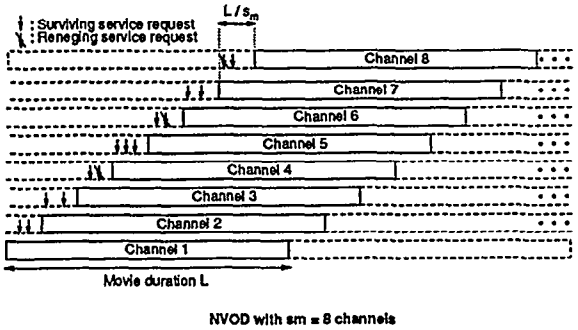

Figure 1: NVoD service for $s_m = 8$.

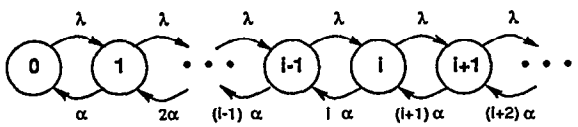

Figure 2: Transition diagram for the patience birth-death process.

discontinuity in VCR actions, as measured by the average phase offset, while increasing the NVoD server throughput by serving more customers.

3 Performance Analysis of NVoD

Analytical modeling of batching systems should capture the tradeoffs generally observed in resource sharing systems between customers' and service provider's point of views and conflicting objectives. In this section, we derive the NVoD throughput for a single movie title as a function of the length of a phase offset. As we shall see, the throughput is also a measure of the customers' defection rate.

3.1 Stationary Arrivals

We assume that a total of s concurrent channels is available to the NVoD server, and that these s channels have to be partitioned among N movie titles available to the customers. Thus, s_m channels will be allocated to movie m and the corresponding phase offset is $\frac{L}{s_m}$, where L is the movie length (assumed the same for all movie titles). Popularities can be expressed as a vector of access probabilities $[p_1, p_2, \cdots, p_N]$, $\sum_{m=1}^{N} p_m = 1$, where N is the number of different movie titles [11]. If the total request arrivals are a Poisson process with parameter λ, the request arrivals at movie title m are Poisson with parameter $\lambda_m = p_m\lambda$. To be realistic, we also consider partially patient customers, who agree to wait τ time units or more with the probability $p_w(\tau, \overline{\tau}) = e^{-\frac{\tau}{\overline{\tau}}}$, where $\overline{\tau}$ is the average time customers agree to wait. In general, the patience rate $\alpha = \frac{1}{\overline{\tau}}$ can be assumed independent of the requested movie title. The NVoD service is depicted in Figure 1.

A similar system was studied by the authors of [8], under the assumption that the number of requests "accumulated" between two consecutive starts of the service is the number of customers who agreed to wait *exactly* $\frac{L}{s_m}$ units of time. In other words, upon arrival, customers are asked if they are willing to wait $\frac{L}{s_m}$, and admitted into the system only if they agreed; had they been admitted, they would have actually waited much less, depending on the time they arrived within the reservation interval before the service phase offset. Our formulation is less restrictive and more realistic than that of [8], in that customers are allowed to make reservations and renege spontaneously. Another way to look at our system is that upon arrival, customers are asked if they are willing to wait for the *exact amount of time* τ *which separates them from the service interval*, and they accept with a probability $p_w(\tau, \overline{\tau})$.

For the calculation of the number of customers waiting between two consecutive services, we consider the *transient* analysis of the $M/M/\infty$ "self-service" queueing system in Figure 2, with arrival rate λ_m and self-service with a negative exponential distribution at rate $\alpha = \frac{1}{\overline{\tau}}$. This system is analyzed in [4], deriving (i) the probability, $P_{M/M/\infty}(t,i,n)$ in Eq. (1), that there are n customers in the $M/M/\infty$ system at time t given there were i customers at time 0, and (ii) the mean number in the system at time t (Eq. (2)):

$$L_{M/M/\infty}(t,i) = \frac{\lambda_m}{\alpha}(1-e^{-\alpha t}) + ie^{-\alpha t}. \quad (2)$$

The server throughput for movie m per L time units — the average number of service requests granted during L time units — is then obtained by applying Eq. (2) to each of s_m phase offsets of movie title m:

$$T_m = s_m \frac{\lambda_m}{\alpha}(1-e^{-\alpha\frac{L}{s_m}}). \quad (3)$$

Let $\rho = \frac{\lambda_m L}{s_m}$ denote the traffic intensity. Then, the throughput variations are linear in ρ for a fixed s_m.

The average loss rate L_{rm} of customers for movie title m due to lack of patience is given by the difference between the average number of arrivals and the number of customers served within $\frac{L}{s_m}$ time units:

$$L_{rm} = \lambda_m\left(1 - \frac{1-e^{-\alpha\frac{L}{s_m}}}{\alpha\frac{L}{s_m}}\right). \quad (4)$$

Note that minimizing the average loss rate maximizes the system throughput. Also, the average defection rate $\frac{L_{rm}}{\lambda_m}$ — the ratio of reneging customers to the total number of customers — depends only on the customers' reneging behavior expressed by α, not on the traffic intensity.

Once admitted, customers move from one multicast group to the next when they request VCR actions that can only be satisfied in a discontinuous fashion. Even though it is possible that no new service requests for a particular time slot were placed in the previous $\frac{L}{s_m}$ time units, requests for that particular time slot may be made later on. Thus, the VoD server has to be *work-conserving*, by restarting service every $\frac{L}{s_m}$ units of time. We want to evaluate the cost of

providing work-conserving service, and hence, discontinuous VCR actions capability, or *utilization* of the NVoD server. This can be done by comparing a work-conserving NVoD server with a non-work-conserving one in which, if no new requests for movie m are placed within this interval preceding a particular channel, that channel will be idle for time of duration L. Note that the average throughput is the same in both systems if all parameters are kept the same. The reason for this is that the throughput indicates the number of *new requests* granted every L time units. When no request was made in the previous $\frac{L}{s_m}$ time units, the throughput is unaffected by restarting a channel. Let a *cycle* be the time between two consecutive service starts of a channel, then the cycle in a work-conserving NVoD system is simply the length L of a movie. The cost of providing work-conserving service can be calculated by comparing the cycle lengths of work-conserving and non-work-conserving systems. If they are about the same, then the utilization of the work-conserving NVoD server is very high. On the other hand, a much shorter cycle in the NVoD system supporting discontinuous VCR actions implies that, for the same number of channels, more bandwidth is used to achieve the same throughput as in the non-work-conserving system, thus resulting in a low utilization. An indicator of the extra cost due to low utilization is

$$C_{VCR} = \frac{\sum_{m=1}^{N} s_m \overline{\tau}_m}{sL}, \tag{5}$$

where $\overline{\tau}_m$ is the average cycle in the non-work-conserving system.

Now, let's calculate $\overline{\tau}_m$ for the non-work-conserving NVoD system. If we consider one channel for movie m in isolation, the service restarts if at least one request survived during the last $\frac{L}{s_m}$ segment. Hence, with probability $1 - P_{M/M/\infty}(\frac{L}{s_m}, 0, 0)$ the service restarts, and with probability $P_{M/M/\infty}(\frac{L}{s_m}, 0, 0) = exp(-\frac{\lambda_m}{\alpha}(1 - e^{-\alpha\frac{L}{s_m}}))$ the system becomes idle. If $P_{M/M/\infty}(\frac{L}{s_m}, 0, 0)$ is replaced by $P_{0,m}$, we have:

$$\begin{aligned}
\overline{\tau}_m &= (1 - P_{0,m})L + P_{0,m}(1 - P_{0,m})2L + \\
&\quad P_{0,m}^2(1 - P_{0,m})3L + \cdots \\
&= (1 - P_{0,m})L\left[\frac{\partial}{\partial P_{0,m}}\left(\sum_{k=0}^{\infty} P_{0,m}^{k+1}\right)\right] \\
&= (1 - P_{0,m})L\frac{\partial}{\partial P_{0,m}}\left(\frac{P_{0,m}}{1 - P_{0,m}}\right) \\
&= \frac{L}{1 - P_{0,m}}.
\end{aligned} \tag{6}$$

The cost C_{VCR} in Eq. (5) is now fully determined, and C_{VCR} converges to 1 as the arrival rate of requests increases and as the patience factor increases. This observation is consistent with the fact that in both cases, the likelihood of requests' survival from one phase offset to the next increases, thus reducing the number of idle periods.

3.2 Nonstationary Arrivals

In real NVoD systems, request arrivals are usually nonstationary. Variations in the request rate can be observed on a daily basis, between "prime time" (e.g., circa 8 p.m.) and "off-hours" (e.g., early morning). On a larger time scale (e.g., one week), changes in movie popularities due, for example, to new releases, or to loss of customers' interest in current titles over time, may also cause changes in request rates. To account for nonstationary request rates, the $M/M/\infty$ patience queue in Section 3.1 must be replaced by the $M_t/M/\infty$ system, analyzed in [5, 7].

Fortunately, if we choose an arrival rate function carefully, it is possible to analytically determine the number of surviving customers at the end of a phase offset, which in fact corresponds to *the number of busy servers* of $M_t/M/\infty$ at the end of a phase offset. We assume that the aggregated total request rate for all movie titles (and correspondingly for each movie title m) is sinusoidal:

$$\begin{aligned}
\lambda(t) &= \overline{\lambda} + A\sin(\gamma t) \\
\lambda_m(t) &= p_m\overline{\lambda} + p_m A\sin(\gamma t) \\
&= \overline{\lambda}_m + A_m\sin(\gamma t),
\end{aligned}$$

where $\overline{\lambda}$ is the daily average arrival rate, $A(> 0)$ is the amplitude, $\gamma = \frac{2\pi}{T}$, T being a 24-hour period, and p_m the popularity of movie title m. More general arbitrary models of nonstationarity have been proposed in [3]. Most of these models usually comprise successive intervals with approximately constant request arrival rates over an extended period of time (e.g., 1 hour). To the best of our knowledge, there are no published realistic (or empirically verified as such) models of customers generating nonstationary requests in a VoD system. Thus, we had to choose an arbitrary model which should, ideally, reproduce a realistic demand on the NVoD server while being computationally tractable. The sinusoidal rate is a convenient and representative model that captures the customers' cyclic behavior; most of the approximations presented below can also be generalized to a wide range of periodic functions.

As in Section 3.1, the main idea in the calculation of NVoD throughput is to model successive phase offsets for one movie title m with an $M_t/M/\infty$ queueing system that gets reset and restarted every $\frac{L}{s_m}$ time units. Eq. (17) of Appendix A shows how to calculate the number of surviving requests $L_{M_t/M/\infty}(\frac{L}{s_m}, t_0)$ at the end of a reservation interval of length $\frac{L}{s_m}$ starting at time t_0. In order to express the NVoD throughput, we have to consider all phase offsets during a 24-hour period, or more generally, during a period equal to $LCM(T, \frac{L}{j_m})$ if the number of phase offsets within T is not integer (i.e., a day is not divisible by the movie length). The NVoD throughput for movie title m is therefore:

$$T_m = \frac{\sum_{i=0}^{\mathcal{T}_m - 1} L_{M_t/M/\infty}((i+1)\frac{L}{j_m}, i\frac{L}{s_m})}{\mathcal{T}_m} s_m. \tag{7}$$

where $\mathcal{T}_m = \frac{LCM(T, \frac{L}{s_m})}{\frac{L}{s_m}}$. Finally, we can express the average loss rate of customers for movie title m as given in Eq. (8). As noticed in [7], if $\lambda(t)$ is a general, not necessarily periodic, function, the analysis of the $M_t/G/\infty$ system can be inferred from the sinusoidal case by combining periodic overlap and Fourier decomposition of $\lambda(t)$. These techniques and their restrictions are elaborated on in [7]. Our formulation of throughput and loss rate can thus be adapted easily to a wide range of nonstationary arrival rate.

Similarly to Section 3.1, in order to evaluate the cost of providing discontinuous VCR actions, C_{VCR} given in Eq. (5), we need to calculate the average cycle in the non-work-conserving NVoD system. The calculation of the average

$$L_{rm} = \frac{1}{T_m} \cdot \sum_{i=0}^{\mathcal{T}_m - 1} \left\{ \int_{i\frac{L}{s_m}}^{(i+1)\frac{L}{s_m}} \lambda_m(t)dt - L_{M_t/M/\infty}((i+1)\frac{L}{s_m}, i\frac{L}{s_m}) \right\}. \quad (8)$$

cycle is simplified by the fact that the number of customers surviving at the end of a phase offset is known to have Poisson distribution. But the average of this distribution varies from epoch to epoch. If we consider a particular phase offset $i < LCM(T, \frac{L}{s_m})$ for movie m, service will restart with probability $1-P_{0,m,i} = 1-\exp(-L_{M_t/M/\infty}((i+1)\frac{L}{s_m}, i\frac{L}{s_m}))$. Hence, for this particular phase offset, the cycle specific to that particular phase offset can be calculated by the following recursive formula:

$$\begin{aligned} \overline{\tau}_{m,i} &= (1-P_{0,m,i})L + P_{0,m,i}(1-P_{0,m,i+1})2L + \\ &\quad P_{0,i}P_{0,m,i+1}(1-P_{0,m,i+2})3L + \cdots \\ &= L\sum_{k=i}^{\infty}\left[k(1-P_{0,m,k})\prod_{l=i}^{k-1}P_{0,m,l}\right], \end{aligned} \quad (9)$$

with $\prod_{l=i}^{k-1} = 1$ if $k = i$. We found empirically that the product terms beyond the first few terms of the summation are insignificant, and $P_{0,m,i}$'s between adjacent phase offsets are similar. After approximations and calculations similar to those leading to Eq. (6), we obtain:

$$\overline{\tau}_{m,i} \approx \frac{L}{1-\exp(-L_{M_t/M/\infty}((i+1)\frac{L}{s_m}, i\frac{L}{s_m}))}. \quad (10)$$

The average cycle length can finally be approximated by:

$$\overline{\tau}_m = \frac{\sum_{i=0}^{\mathcal{T}_m-1} \frac{L}{1-\exp(-L_{M_t/M/\infty}((i+1)\frac{L}{s_m}, i\frac{L}{s_m}))}}{\mathcal{T}_m}. \quad (11)$$

4 Optimal Scheduling in NVoD

4.1 Problem Statement

We now use the throughput calculations in Section 3 to determine partitions of the NVoD channel capacity s so that the throughput of the server averaged over all movie titles may be maximized.

Problem NVoD T-OPT: Given s concurrent channels, $\overline{s} = [s_1, \cdots, s_N]$ partition of the channels among the N different movie titles, λ the aggregated request rate, $\mu = \frac{1}{L}$ the constant service rate, $[p_1, \cdots, p_N]$ the movie popularity vector, $\lambda_m = p_m\lambda$ the arrival rate, and $T_m(s_m)$ the throughput corresponding to movie title m, determine

$$\max_{\overline{s}} \sum_{m=1}^{N} T_m(s_m) \quad \text{such that} \quad \sum_{m=1}^{N} \frac{s_m}{s} = 1$$

and

$$s_m > 0, \qquad m = 1, \cdots, N.$$

Problem NVoD T-OPT can be further refined. For instance, field studies can determine a range of phase offsets "acceptable" or tolerable to customers, in terms of service latency, discontinuity of VCR actions, and affordability of the CPE buffer. In this case, the objective of the NVoD server is to achieve maximum throughput *while satisfying constraints on the maximum and minimum allowable phase offsets for each movie title*, in order to provide a service that is "appealing" to the customer. Another objective could be to operate under the lowest C_{VCR} possible. In this case, the objective function can be replaced by $\min_{\overline{s}} C_{VCR}$.

Since the objective function of Problem NVoD T-OPT is convex, and the first constraint is linear, NVoD T-OPT is a discrete convex separable resource allocation problem, and the optimal partition $[s_1, \cdots, s_N]$, denoted by T-OPT, can be found by using integer programming techniques in at most s steps, based on a method initially presented in [12]. In the special case of infinitely patient customers ($\alpha = 0$), all customers get served within a phase offset. In this case, the NVoD throughput is constant for any partition $\overline{s}$, and one can use another criterion to determine the partition, such as minimizing the average phase offset. The average phase offset is an indicator of the average waiting time experienced by customers. In the general case of $\alpha > 0$, however, minimizing the average phase offset $\sum_{m=1}^{N} p_m \frac{L}{s_m}$ conflicts with maximization of throughput, yielding a separate allocation vector $\overline{s} = [s_1, \cdots, s_N]$ (denoted by EW-OPT) whose determination can be done in s steps similarly to that of T-OPT.

The following four allocation policies are considered to evaluate the T-OPT performance while varying the number of channels, traffic intensities and patience factors.

1. The heuristic NVoD T-OPT which determines the partition T-OPT by maximizing the throughput of the NVoD server.
2. A proportional allocation policy, which assigns each movie the number of channels in proportion to its popularity determining a partition T-PROP.
3. The heuristic which allocates the number of channels proportional to the square root of popularity, because it was found in [1] that batching customers' requests according to the maximum factored queue length (MFQL)[1] leads to minimal customers' latency in case of infinitely patient customers. In case of discrete pre-determined phase offsets, even though T-SQRT may not be optimal, it might offer an interesting tradeoff between throughput and average phase offset. The corresponding partition is denoted by T-SQRT.
4. The allocation policy which minimizes the average phase offset. The resulting partition is denoted by EW-OPT.

4.2 Simulation Results

We chose to compare the throughput (given by Eq. (3)), the average phase offset $\overline{EW}$, and the dispersion D — defined as the coefficient of variation of the phase offsets — of the various above-mentioned policies:

$$D = \frac{\sqrt{\sum_{m=1}^{N} p_m \left(\frac{L}{s_m}\right)^2 - \overline{EW}^2}}{\overline{EW}}, \quad (12)$$

[1] for a particular movie title, the queue length divided by the square root of the title popularity

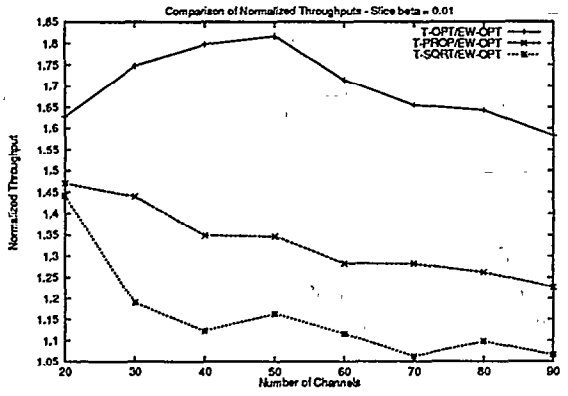

Figure 3: Normalized throughput for $\beta = 0.01$.

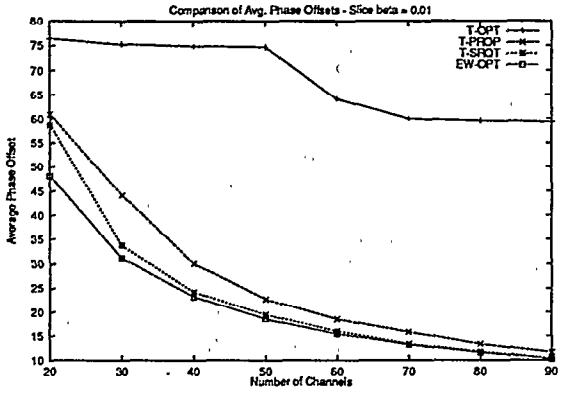

Figure 4: Average phase offset for $\beta = 0.01$.

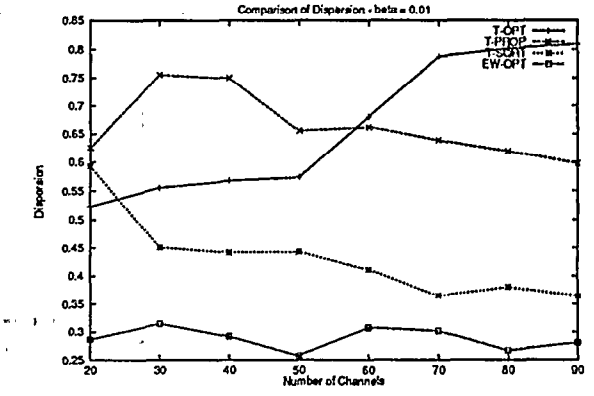

Figure 5: Dispersion for $\beta = 0.01$.

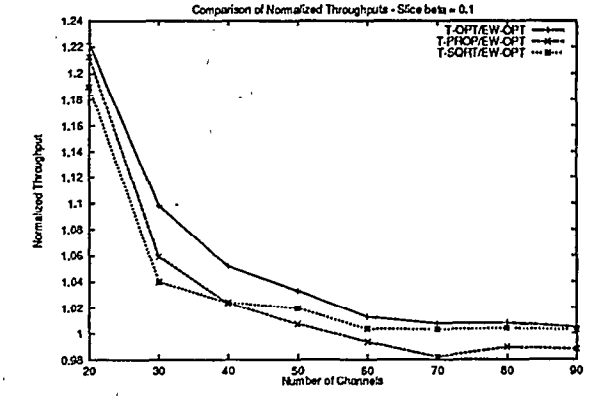

Figure 6: Normalized throughput for $\beta = 0.1$.

with $\overline{EW} = \sum_{m=1}^{N} p_m \frac{L}{s_m}$. Lower values of the dispersion D indicate more homogeneous allocation policies which lessen variations in the customers' waiting times. We adopt the Zipf's law [1, 6, 15] as the stationary model of movie popularity. In a Zipf-like distribution, we have:

$$p_m = \frac{f_m}{\sum_{j=1}^{N} f_j} \tag{13}$$

where $f_m = \frac{1}{m^{1-\theta}}$, $m = 1, \cdots, N$, and θ is added to specify the skew. A value of $\theta = 0.271$ is known to closely match the popularities generally observed by video store rentals [1].

The throughput given by Eq. (3) in the stationary case, and that by Eq. (7) in the nonstationary case, are quasi-linear in traffic intensity. An important consequence of this property, confirmed by our simulations, is that once an optimum allocation $\overline{s}$ for T-OPT or EW-OPT has been determined for an arbitrary traffic intensity, *it will not change for any other value of traffic intensity*, under the condition that the total number of channels s and the patience factor, defined as $\beta = \frac{\overline{r}}{L}$, are kept constant. By combining both throughput linearity and allocation invariance with traffic intensity, one can now observe that the ratio of throughputs of any two partitions chosen among T-OPT, EW-OPT, T-PROP and T-SQRT is independent of traffic intensity. Consequently, in order to evaluate different allocation policies for an arbitrary channel capacity s and patience factor β, it is enough to simply assume an arbitrary traffic intensity. Then, different partitions can be compared with respect to their *relative* throughputs, *normalized with the throughput corresponding to an arbitrary partition*. For this effect, we selected EW-OPT to be the normalizing partition, since it corresponds to the lowest throughput, as we shall see. Note that the allocation invariance property also implies that the average phase offsets and the corresponding dispersions of each partition are independent of traffic intensity.

We measured the variations of the normalized throughput, the average phase offset and the corresponding dispersion in various cases of channel capacities shared among 10 movie titles. Two values of the patience factor are considered: (i) $\beta = 0.01$ corresponding to customers who are willing to wait 1 minute on average, thus *very impatient*; (ii) $\beta = 0.1$ corresponding to moderately patient customers, willing to wait for 10 minutes on average. (Note that this definition of customers' behavior is arbitrary and used only for an illustrative purpose.)

T-OPT provides an upper bound on the maximum achievable throughput, and EW-OPT a lower bound on the minimum average phase offset, and the corresponding dispersion. Intuitively, for very impatient customers, T-OPT will tend to allocate all channels to the most popular movie title, thereby increasing dispersion and average phase offset. Figures 3, 4 and 5 clearly demonstrate this pronounced tradeoff between throughput and dispersion for very impatient customers. In such situations, T-PROP appears to be a good compromise.

For more patient customers ($\beta = 0.1$), Figures 6, 7 and 8 indicate that the throughput is less sensitive to the choice of an allocation policy, as the distinction between policies is less clearcut. T-SQRT then represents a good tradeoff

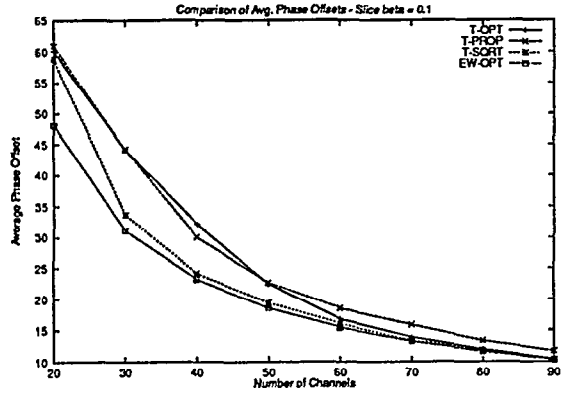

Figure 7: Average phase offset for $\beta = 0.1$.

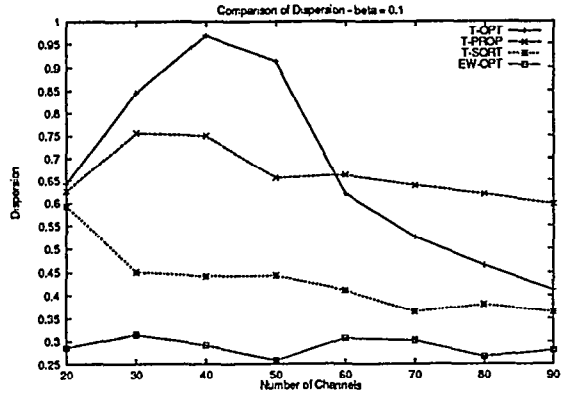

Figure 8: Dispersion for $\beta = 0.1$.

among throughput, average phase offset and dispersion. To summarize our simulation results, the choice of a allocation heuristic by the NVoD server will depend on the number of channels available, customers' behavior expressed by β, and finally, on such performance parameters as throughput, average phase offset, dispersion, or a tradeoff among all of them. We found that the latter case can be achieved with simple heuristics such as T-PROP for impatient customers, and T-SQRT for moderately to very patient customers.

Finally, we compare T-OPT, T-PROP, T-SQRT and EW-OPT with respect to C_{VCR} given in Eq. (5), which is an indicator of the additional bandwidth needed to provide work-conserving service and discontinuous VCR actions support. A low C_{VCR} value represents allocations for which work-conserving scheduling of channels is less costly than non-work-conserving scheduling. Figure 9 shows the simulation results for a request arrival rate $\lambda = 5.0$. (Consistent results were obtained in other experiments with different arrival rates.) For very impatient customers ($\beta = 0.01$), T-OPT exhibits the highest system utilization, since it assigns most of the channel capacity to the most popular movie, which accounts for most of C_{VCR}. The difference between work-conserving and non-work-conserving cycle length is then reduced. For similar reasons, EW-OPT performs worst as it tends to assign channels uniformly. As we kept increasing the patience factor ($\beta \geq 0.01$), T-PROP yielded the best utilization, followed by T-SQRT. This serendipitous result indicates that it is a sensible decision to choose the heuristics that provide a good tradeoff among throughput, average phase offset and dispersion.

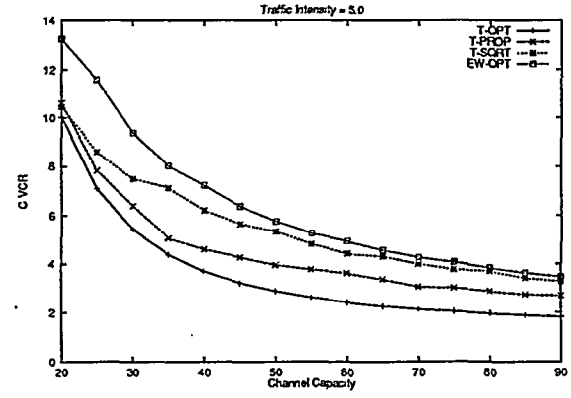

Figure 9: Comparison of C_{VCR} for $\beta = 0.01$.

5 Threshold-Based NVoD

We present in this section an intuitively-appealing alternative to fixed-length phase offsets. Suppose service is provided only when no less than K_m requests are "accumulated" for movie title m and a channel is available. If a request is placed while all servers are busy, the customer has to wait for a server to become available, and also until the desired number of requests is accumulated. If, by the end of the movie length L, there are at least K_m customers waiting then the service restarts; else the service is delayed until the number of requests reaches exactly K_m. We call such class of VoD systems *threshold-based NVoD*, or *Quasi-VoD* (QVoD). We want to compare the performance of QVoD to that of NVoD.

5.1 Performance Analysis

Similarly to the approach taken in Section 3, our first analytical step is to study the throughput of a QVoD server by focusing on the channels allocated to one particular movie title. For tractability, we will restrict our analysis to the case where a single stream is allocated to a particular movie title m, i.e., $s_m = 1$, and then generalize the results to any number of servers.

Assuming partially patient customers and stationary arrivals, at the end of the service interval L, the service resumes with probability

$$1 - \sum_{i=0}^{K_m-1} P_{M/M/\infty}(L, 0, i),$$

where $P_{M/M/\infty}(L, 0, i)$ is given by Eq. (1). With probability $\sum_{i=0}^{K_m-1} P_{M/M/\infty}(L, 0, i)$ the system becomes idle, waiting until the number of waiting customers reaches K_m. As noticed in [8], if the number of surviving customers at the end of the service interval is k, the length of the idle interval will be the *first-passage time* of the patience birth-death process in Figure 2 from state k to state K_m:

$$t_{k,K_m} = \sum_{i=k}^{K_m-1} t_{i,i+1},$$

where t_{N_1,N_2} is the mean first-passage time from state N_1

$$T_m = \frac{L}{L + \sum_{k=0}^{K_m-1} t_{k,K_m} \mathcal{P}_k} \cdot \left\{ K_m \sum_{k=0}^{K_m-1} \mathcal{P}_k + (\mathcal{L} - \sum_{k=0}^{K_m-1} k\mathcal{P}_k)(1 - \sum_{k=0}^{K_m-1} \mathcal{P}_k) \right\}. \quad (14)$$

to state N_2, given by the following recursive formula [8, 9]:

$$t_{i,i+1} = \frac{\sum_{k=0}^{i} \frac{\left(\frac{\lambda_m}{\alpha}\right)^k}{k!}}{\lambda_m \frac{\left(\frac{\lambda_m}{\alpha}\right)^i}{i!}}.$$

The mean idle time can now be computed as:

$$\overline{\tau}_{im} = \sum_{k=0}^{K_m-1} t_{k,K_m} P_{M/M/\infty}(L, 0, k).$$

Having expressed the mean idle time, the average cycle duration is $\overline{\tau}_{cm} = \overline{\tau}_{im} + L$ and the average number of requests served per cycle is (with $\mathcal{P}_k = P_{M/M/\infty}(L, 0, k)$ and $\mathcal{L} = L_{M/M/\infty}(L, 0) = \frac{\lambda_m}{\alpha}(1 - e^{-\alpha L})$):

$$\begin{aligned} N_{cm} &= (1 - \sum_{k=0}^{K_m-1} \mathcal{P}_k) \sum_{k=K_m}^{\infty} k\mathcal{P}_k + K_m \sum_{k=0}^{K_m-1} \mathcal{P}_k \\ &= (\mathcal{L} - \sum_{k=0}^{K_m-1} k\mathcal{P}_k) + \\ &\quad (K_m - \mathcal{L} + \sum_{k=0}^{K_m-1} k\mathcal{P}_k) \sum_{k=0}^{K_m-1} \mathcal{P}_k. \end{aligned}$$

Finally, the throughput for movie title m, defined as the ratio of N_{cm} to $\overline{\tau}_{cm}$, can be expressed as in Eq. (14). The average loss rate, due to impatient customers who leave the queue without receiving service, is simply given by $\lambda_m - \frac{N_{cm}}{\overline{\tau}_{cm}}$.

For an arbitrary traffic intensity, small threshold values K_m lead to a sub-optimal throughput due to under-collected service requests, while large values of K_m may cause losses of customers due to long waits. Thus, in the general case of $s_m \geq 1$, there is an optimal value of the threshold K_m^{opt} which maximizes the QVoD server throughput for movie title m. Figure 10 illustrates this phenomenon for 100 channels allocated to movie title m. Our simulation results also indicate that K_m^{opt} plays a critical role in achieving the lowest request defection rate. This observation is particularly important if the traffic intensity and customers' patience vary with time (e.g., in a 24-hour cycle). In such situations, the QVoD server may have to change the value of K_m dynamically. In the next subsection we evaluate the effectiveness of such an adaptive approach in a nonstationary environment.

5.2 Throughput Comparison QVoD vs. NVoD

Relaxing the constant-phase-offset constraint in NVoD by using QVoD is justifiable if the resulting throughput gain is significant. Thus, we now compare the throughput for one movie title in both NVoD and QVoD systems. A complete comparison requires evaluation of both stationary and nonstationary traffic intensities. In the stationary case, Figure 11 represents the ratio of the QVoD throughput to that

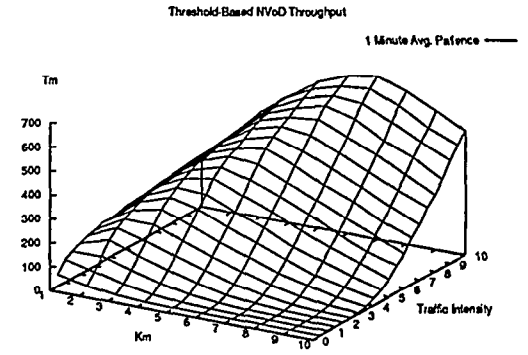

Figure 10: QVoD throughput for 100 channels and $\beta = 0.01$.

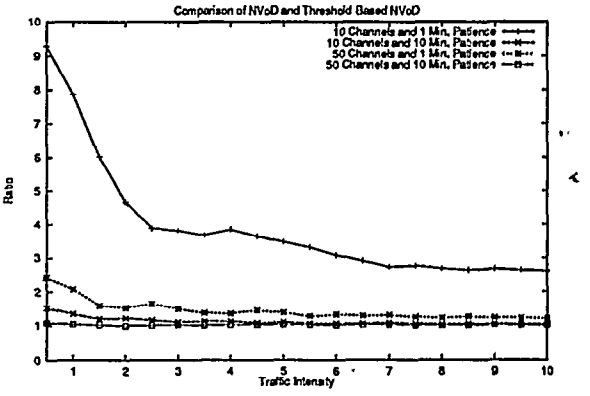

Figure 11: Throughput ratio for stationary arrival rates.

of NVoD, for four different combinations of the channel capacity s_m allocated to movie m, and of the patience factor β. For each traffic-intensity value, the QVoD throughput corresponds to the optimal threshold K_m^{opt}. Our results indicate a higher throughput in QVoD systems, although the difference between NVoD and QVoD diminishes as both customers' patience and channel capacity increase. This trend can be explained by the fact that, when the average customers' patience is comparable to, or greater than, half an NVoD phase offset (e.g., 1 minute for 50 channels), the NVoD throughput will be pretty high, hence lessening any relative improvement from using QVoD.

QVoD appears superior to NVoD if the optimum K_m^{opt} is used. However, our simulation results indicate a sharp decrease in QVoD performance when non-optimal values of K_m are used for a particular traffic intensity. This raises questions regarding the applicability of QVoD in case of nonstationary request arrivals. We also noticed that the defection rate in QVoD is very sensitive to variations of traffic intensity, the patience factor and K_m. In the NVoD system, on the other hand, defection rates depend only on the customers' patience.

There are two policies that a QVoD server can adopt in

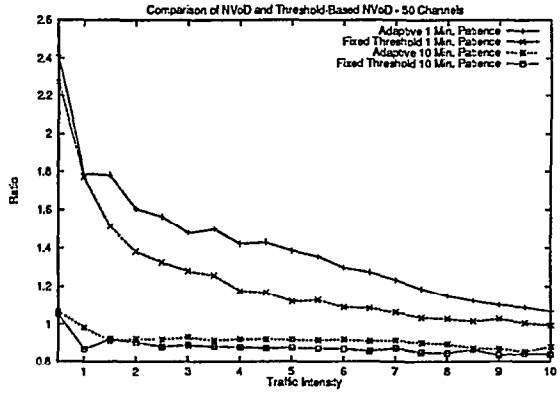

Figure 12: Throughput ratio for nonstationary arrival rates, 50 channels, and $RA = 0.9$.

a nonstationary environment. First, the threshold K_m can be dynamically adapted by choosing K_m^{opt} for the instantaneous arrival rate. Alternatively, the QVoD server can choose the fixed threshold which maximizes the throughput averaged over a 24-hour period. We used simulations to evaluate the ratio of the QVoD throughput in each approach to that of NVoD, for different values of the patience factor. We assumed sinusoidal arrival rates of relative amplitude $RA = 0.9$, which represent an extreme case of nonstationarity. The NVoD throughput was calculated from Eq. (7), whereas the QVoD throughput was obtained through recursive simulations. The simulation results in Figure 12 for $s_m = 50$ show that, as the number of channels and the patience factor increase, QVoD becomes less attractive for both choices of the threshold, adaptive or fixed. This conclusion confirms that NVoD should not, in general, be dismissed in favor of QVoD for nonstationary arrival rates. Also, the similarity of performance between adaptive and fixed threshold QVoD policies indicates that threshold-based scheduling of videos is not well adapted to continuously-changing load conditions. Finally, until a closed-form equation is found for key QVoD performance variables such as throughput or average latency as functions of K_m and s_m, recursive simulations must be used to determine K_m^{opt}.

6 "QVoD-Enhanced" NVoD

As seen in Section 2, even with proper CPE buffer management and constant phase offsets, continuity in VCR actions can only be provided intermittently and without any guarantee on the discontinuities experienced by customers. Thus, it is intuitively appealing to relax the assumption of constant phase offsets in NVoD for a higher throughput, as long as the same *average granularity of discontinuity in VCR actions*, as measured by the average phase offset, is provided to customers. We show in this section that by using QVoD over a partition $[s_1, \cdots, s_N]$ of the capacity s among the N movie titles *initially determined for NVoD*, one can achieve a much higher throughput while providing support for discontinuous VCR actions comparable to that of NVoD in an average sense. We will restrict our analysis to stationary request rates, although it is valid in most nonstationary cases.

Suppose the NVoD server will initially determine a partition $[s_1, \cdots, s_N]$ by optimizing an arbitrary pre-determined objective. This objective may be to maximize throughput, minimize the average phase offset, or to make a tradeoff between throughput and average phase offset. We examine how NVoD performance will be affected by switching to QVoD based on the same $[s_1, \cdots, s_N]$, and the corresponding vector of optimal thresholds $\overline{K}^{opt} = [K_1^{opt}, \cdots, K_N^{opt}]$. According to the results obtained thus far, we can make performance improvement for moderately stationary arrival rates, impatient customers and a relatively small number of channels allocated to each movie title.

We approach the problem by using QVoD in conjunction with NVoD in three experimental steps: (1) First, we have to select arbitrary NVoD partitions $[s_1, \cdots, s_N]$. Since we are interested in customers' QoS, we choose NVoD EW-OPT for minimization of the phase offset, and NVoD PROP or NVoD SQRT for the tradeoff between throughput and phase offset, depending on the value of the patience factor. These partitions were presented in Section 4.1; (2) Next, we evaluate performance by switching from NVoD to QVoD; (3) Finally, we compare the performance of QVoD with that of NVoD whose partition $[s_1, \cdots, s_N]$ corresponds to NVoD T-OPT, presented in Section 4.1. NVoD T-OPT is used as an indicator of the maximum throughput achievable with a NVoD server.

In summary, we compared the following five systems.

1. An NVoD server with $[s_1, \cdots, s_N]$ minimizing the average NVoD phase offset; this configuration is called NVoD EW-OPT.
2. A QVoD server with the same channel allocation vector $[s_1, \cdots, s_N]$ defined by NVoD EW-OPT, used in conjunction with $\overline{K}^{opt}$; this configuration is called QVoD EW-OPT.
3. An NVoD server with the partition $[s_1, \cdots, s_N]$ making an acceptable tradeoff among throughput, phase offset and fairness. In Section 4.2, allocating channels proportionally to the popularities (NVoD T-PROP) is shown to be a good candidate for very impatient customers. For moderately to very patient customers, allocation in proportion to the square root of the popularities (NVoD T-SQRT) is preferable. Thus, we consider the NVoD T-PROP partition for $\beta = 0.01$ and NVoD T-SQRT for $\beta = 0.1$.
4. A QVoD server with the same partition $[s_1, \cdots, s_N]$ determined by NVoD T-PROP for $\beta = 0.01$ and NVoD T-SQRT for $\beta = 0.1$, used in conjunction with $\overline{K}^{opt}$. These configurations are called QVoD T-PROP and QVoD T-SQRT.
5. An NVoD server with $[s_1, \cdots, s_N]$ maximizing the NVoD throughput; this configuration is called NVoD T-OPT.

Figures 13, 14 and 15 show the simulation results for 100 channels partitioned among 10 movie titles of 100 minutes each, and accessed by very impatient customers ($\beta = 0.01$). We measured throughput, average phase offset and dispersion. The improvement in throughput by using a QVoD server (configurations QVoD T-PROP and QVoD EW-OPT) is dramatic, with a relatively minor effect on the average phase offset and the corresponding dispersion. This is particularly noticeable for large traffic intensities. For very low traffic intensities ($\rho < 1.0$), QVoD EW-OPT is preferable to QVoD T-PROP, since it achieves a comparable throughput for lower phase offsets and dispersion. In summary, for very impatient customers, the configuration QVoD EW-OPT is the best choice, as it improves upon the maximum throughput achievable with NVoD (configuration NVoD T-OPT), for an average phase offset comparable to that of the NVoD configuration NVoD T-PROP, and the

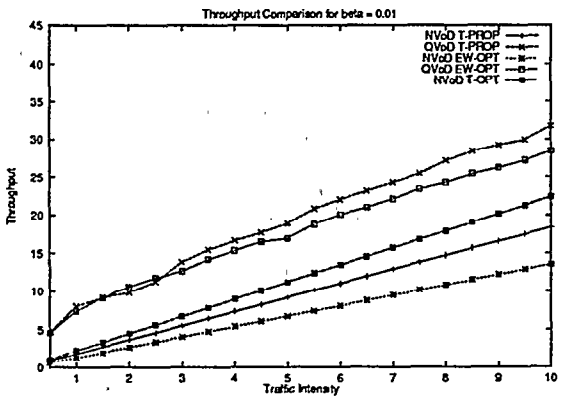

Figure 13: Comparison of NVoD and "QVoD-enhanced" NVoD: throughput for $\beta = 0.01$.

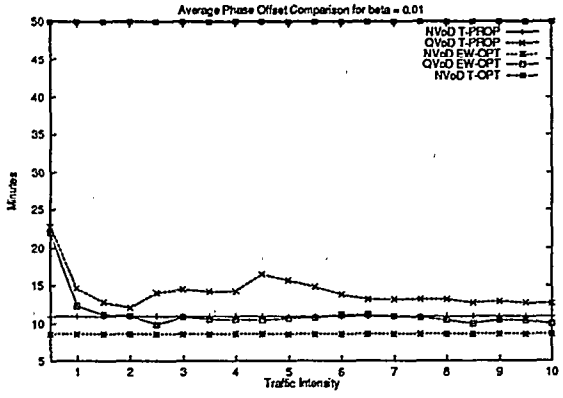

Figure 14: Comparison of NVoD and "QVoD-enhanced" NVoD: average phase offset for $\beta = 0.01$.

corresponding dispersion comparable to that of NVoD EW-OPT. For more patient customers ($\beta = 0.1$) though, we found in separate experiments that the marginal improvement in throughput by replacing NVoD by QVoD is not worth the significant increase in the average phase offset and dispersion. Finally, similar conclusions were made in the case of nonstationary arrival rates, for which QVoD can be used for very impatient customers.

7 Conclusion

In this paper, we presented an analytical (as opposed to commonly-used simulation) approach to program scheduling in NVoD systems. Both customers' and service provider's points of views are integrated in the approach in order to account for the tradeoff between system throughput and customers' partial patience. We first derived a closed-form expression for the NVoD server throughput, defined as the number of customers' requests served within two consecutive offerings of a movie over one channel. Our analysis of NVoD extends the work in [8], in which customers' requests are granted only if they agree to wait for exactly one phase offset. We then determined the optimal schedule of movies of different popularities for maximum throughput and the lowest average phase offset. We extended these results to the case of nonstationary request arrivals. In practice, the choice of a scheduling algorithm will depend only on the number of channels available, customers' patience, and on

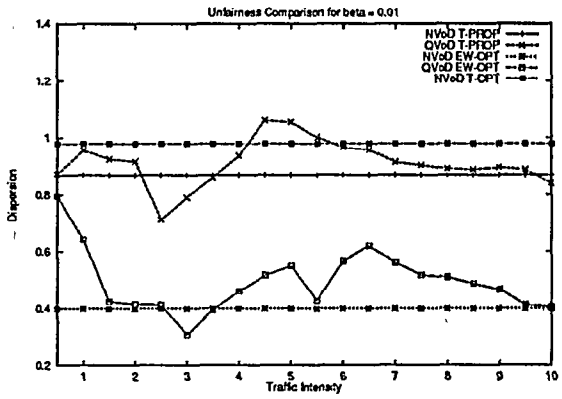

Figure 15: Comparison of NVoD and "QVoD-enhanced" NVoD: dispersion for $\beta = 0.01$.

the performance variable which is most valued by the NVoD service provider, i.e., throughput, average phase offset, fairness, or a tradeoff among all of these. We found that in the latter case, simple heuristics such as the allocation of channels in proportion to movie popularities yield good results. Next, we analyzed a QVoD system, which schedules channels based on a threshold of requests. The throughput is found to be usually greater in QVoD than in NVoD, except for the extreme case of nonstationary request arrivals. This last observation is used to improve throughput without compromising customers' QoS, by using QVoD in conjunction with NVoD.

A Analysis of the $M_t/M/\infty$ Patience Queue

Here we calculate the average number of customers waiting for service at the end of a reservation period, which corresponds to the phase offset before receiving service. The main $M_t/G/\infty$ result, of which $M_t/M/\infty$ is a special case, is thanks to Palm and Khintchine. This result, presented in [7], states that the number of busy servers at time t, which is, in the patience queue, the actual number of customers waiting for service, has a Poisson distribution, and is therefore fully specified by its average. In our case of exponential patience (or service rate), the *average* number of busy servers in $M_t/M/\infty$ can be found from the following differential equation:

$$\dot{L}_{M_t/M/\infty}(t,t_0) + \alpha L_{M_t/M/\infty}(t,t_0) = \lambda_m(t) \qquad (15)$$

$$L_{M_t/M/\infty}(t,t_0) = 0 \qquad t \le t_0, \qquad (16)$$

where

$$\dot{L}_{M_t/M/\infty}(t,t_0) = \frac{d}{dt}\left(L_{M_t/M/\infty}(t,t_0)\right).$$

We added Eq. (16) to represent the initial conditions of the $M_t/M/\infty$ system in case of NVoD, which states that the patience queue restarts empty at the beginning of each phase offset. After some calculations, we find that for the sinusoidal arrival rate of Section 3.2, the general solution of Eq. (15) is given by:

$$L_{M_t/M/\infty}(t,t_0) = \frac{\overline{\lambda}_m}{\alpha} + \left(\frac{A_m}{\alpha+\gamma^2}\right)\left\{\sin(\gamma t) - \frac{\gamma}{\alpha}\cos(\gamma t)\right\} + \left(\frac{\gamma}{\alpha}\cdot\frac{A_m}{\alpha+\gamma^2} - \frac{\overline{\lambda}_m}{\alpha}\right)e^{-\alpha(t-t_0)}. \qquad (17)$$

References

[1] C. C. Aggarwal, J. L. Wolf and P. S. Yu, "On Optimal Batching Policies for Video-on-Demand Storage Servers," *Proc. ACM Multimedia'96*, pp. 253-258, 1996.

[2] K. C. Almeroth and M. H. Ammar, "The Use of Multicast Delivery to Provide a Scalable and Interactive Video-on-Demand Service," *IEEE Journal on Selected Areas in Communications*, Vol. 14, No. 6, pp. 1110-1122, August 1996.

[3] K. C. Almeroth, A. Dan, D. Sitaram, and W. H. Tetzlaff, "Long Term Resource Allocation in Video Delivery Systems," *Proc. IEEE INFOCOM'97*, April 1997.

[4] B. D. Bunday, *An Introduction to Queueing Theory*, Arnold, 1996.

[5] M. J. Carillo, "Extensions of Palm's Theorem: A Review," *Management Science*, Vol. 37, No. 6, pp. 739-744, June 1991.

[6] A. L. Chervenak, D. A. Patterson, R. H. Katz, "Choosing the Best Storage System for Video Services," *ACM Multimedia'95*, pp. 109-119, 1995.

[7] S. G. Eick, W. A. Massey, W. Whitt, "$M_t/G/\infty$ Queues with Sinusoidal Arrival Rates," *Management Science*, Vol. 39, No. 2, pp. 241-252, February 1993.

[8] A. D. Gelman, S. Halfin, "Analysis of Resource Sharing Information Providing Services," *Proc. IEEE GLOBECOM'90*, pp. 312-316, December 1990.

[9] D. P. Heyman, M. J. Sobel, *Stochastic Models in Operations Research*, McGraw-Hill, 1982.

[10] C. N. Judice, E. J. Addeo, M. I. Eiger, H. L. Lemberg, "Video on Demand: A Wideband Service or Myth?," *Proc. of ICC'86*, pp. 1735-1739, June 1986.

[11] T. D. C. Little, D. Venkatesh, "Popularity-based assignment of movies to storage devices in a video-on-demand system," *Multimedia Systems*, Vol. 2, No. 6, pp. 280-287, 1995.

[12] G. Meempat and M. K. Sundareshan, "Optimal Channel Allocation Policies for Access Control of Circuit-Switched Traffic in ISDN Environments," *IEEE Transactions on Communications*, Vol. 41, No. 2, pp. 338-350, February 1993.

[13] B. Özden, R. Rastogi, A. Silberschatz, "Disk Striping in Video Server Environment," *Proc. ACM Multimedia'96*, pp. 580-589, 1996.

[14] N. Terada, H. Ishii, T. Tachi, Y. Okumura, H. Kotera, "An MPEG2-Based Digital CATV and VOD System using ATM-PON Architecture," *Proc. ACM Multimedia'96*, pp. 522-531, 1996.

[15] G. Zipf, *Human Behaviour and the Principle of Least Effort*, Addison-Wesley, 1949.