INSTITUTO DE COMPUTAÇÃO
UNIVERSIDADE ESTADUAL DE CAMPINAS

Dimensioning the Capacity of True Video-on-Demand Servers

Nelson L. S. Fonseca *Hana K. Rubinsztejn*

Technical Report - IC-02-007 - Relatório Técnico

August - 2002 - Agosto

The contents of this report are the sole responsibility of the authors.
O conteúdo do presente relatório é de única responsabilidade dos autores.

Dimensioning the Capacity of True Video-on-Demand Servers

Nelson L.S. da Fonseca*[1] and Hana K. Rubinsztejn[2]

1 - State University of Campinas
Institute of Computing
P.O. Box 6176
13084-971 Campinas SP
Brazil
Phone: +55-19-37885878
Fax: +55-19-37885847
E-mail: nfonseca@ic.unicamp.br

2 - CPqD Telecom & IT Solutions
Rod Campinas / Mogi-Mirim, Km 118.5
P.O. Box 6070
13088-902 Campinas SP
Brazil
Phone: +55 19 3705-4404
E-mail: hana@cpqd.com.br

EDICS: 5-BEEP Multimedia Traffic Management

Abstract

The need to reduce the huge demand for bandwidth of video-on-demand services has led to the consideration of techniques based on multicast for the deployment of such services on a large scale. Interactiveness, a desirable feature of video services, includes the capacity to perform VCR operations. Whenever a viewer issues a VCR operation, his/her video stream becomes unsynchronized with that of his/her multicast group. The present paper introduces a novel approach for determining the number of video channels needed to support such interactiveness. Moreover, it investigates the performance of interactive systems with a reserved pool of channels for the support of VCR operations. Systems with batching and with both batching and piggybacking are analysed.

I) Introduction

Video-on-demand (VoD) engenders several areas of application, such as on-demand movies, digital libraries and distance learning. Interactiveness, which involves the capacity for performing VCR operations, such as pause, rewind (REW) and fast forward (FF), is a desirable feature of such services. In distance learning, for example, students may wish to pause the video during exhibition to take notes or to rewind the tape so they can revisit a specific section of a lecture and have a chance to understand it better. Such a system which allows VCR operations (true video-on-demand) should permit the performance of these operations at any time.

In order to maintain a continuous delivery of video streams, certain sources must be reserved, both at the network and at the server. A definite limit on the number of streams which a server is capable of supporting is usually established, either by I/O bandwidth limitations or CPU capacity. In a VoD system, the assignment of a separate video channel to each individual request would limit the number of active users to the number of channels available. However, a larger number of users in networks with multicast can be supported by the provision of a single video stream. Mechanisms based on multicast, such as piggybacking and batching [1], have thus been proposed for the reduction of the bandwidth demand of video services.

Piggybacking is based on the fact that viewers do not perceive any alteration in the display rate as long as this is maintained within 5% of the nominal rate [1]. In a VoD server with piggybacking, a request for viewing a video is immediately granted if a channel is available, but when another exhibition of the same video is in progress, the display rate of this initial stream is slowed down, while that of the recently admitted request is speeded up. When the faster stream catches up with the slower one, the two streams can be merged, thus releasing one of the channels.

In a VoD server with batching [2], requests for video exhibition are not granted as soon as they arrive, but rather are delayed somewhat so that all requests for the same video within a certain interval can be combined, and a single video channel is allocated to the whole batch of requests (users). On the one hand, batching increases the server throughput, but on the other, users may not be willing to wait for long periods of time, and may cancel their requests (reneging). It has been shown that a system using batching will admit a larger number of users than one with piggybacking [3]. Furthermore, adopting both batching and piggybacking increases the number of users admitted into the system even more [4].

When a user performs a VCR operation, his/her exhibition becomes unsychronized with the exhibition of the rest of his/her multicast group, and another video channel will be needed to support the newly unsynchronized stream. It is always desirable to have a channel available so that VCR operation requests can be granted immediately while simultaneously guaranteeing the continuity of exhibition. Moreover, the unsynchronized user will require the provision of a channel, at least until resynchronization with another stream is feasible, possibly until the end of the video exhibition. Thus, it is of paramount importance to determine the number of channels needed to support the bandwidth demands of VCR operations.

The main contribution of the present paper is the proposal of a novel dimensioning techniques for channel allocation in true-VoD systems. Systems with batching and those with both batching and piggybacking are investigated, as well as systems with and without a reserved pool of channels for handling VCR operations.

This paper is organized as follows. Section II provides a summary of previous work, while Section III describes the operation of an interactive VoD system. Section IV introduces the computation of the number of channels needed to support the VCR operation demands of a given population of users, and Section V verifies the accuracy of the approximation model. Section VI explains the simulation model used to evaluate the effectiveness of the method proposed in Section IV, while Section VII presents the numerical results; Section VIII reveals the conclusions drawn.

II) Previous Work

To maintain the pictorial quality of playback mode when performing VCR operations, it is necessary to have a larger bandwidth available than that required for playback mode. Reserving extra bandwidth for each user admitted into the system, however, leads to a waste of bandwidth. Dey-Sircar, Salehi, Kurose and Towsley [5] compared a system with deterministic guarantee, in which extra bandwidth is reserved upon user admission, to another with statistical guarantee, which provides the sharing of a pool of channels for the performance of VCR operations of all the users. These authors suggested two schemes for VoD systems with statistical guarantee, the first designed to cause the users either to wait or to release their channel when no bandwidth is available for performing a VCR operation, thus freeing channels for re-use by other requests, while the second avoids the wait or channel release, but at the expense of a reduced pictorial quality. They pointed out that a system with statistical guarantees is able to accept a larger number of users than one with deterministic guarantees.

Dan, Shahabuddin, Sitaram and Towsley [6] proposed a system with three pools of channels, one reserved for popular movies, one for channels allocated on-demand for non-popular movies, and the third for handling VCR operations (contingency channels). The number of channels in each of these pools is based on the long-term system load, while ignoring instantaneous fluctuations. Determining the dimensions of the number of channels in each pool involves an optimization problem which minimizes the probability of reneging. This minimization problem assumes that the waiting time can be determined by the distribution of the waiting time of an M/M/c queue. The joint use of batching and contingency channels can reduce the number of channels needed to support a fixed number of users, especially under high load conditions and for long pauses. Dan et al’s work provides static dimensioning of the contingency pool, as well as limits VCR operations to pausing. The proposal presented here is somewhat different from that of Dan et al. It involves a comparison of various systems with batching, some with contingency channels and others without, and a dynamic dimensioning of the contingency pool, involving not only pauses but also rewind and fast-forward operations.

III) Interactive VoD System

In the present paper, then, a VoD system with batching and another with both batching and piggybacking are studied. Requests for video exhibition are not necessarily granted immediately, but can be delayed to permit the formation of a batch of requests for the allocation of a channel in agreement with a specific batching policy. Moreover, in a system with both batching and piggybacking, *n* video streams can be merged, thus, releasing *n-1* video channels, according to the criteria adopted for piggybacking.

Each request for VCR operation causes the unsynchronization of the exhibition of this user with the exhibition of his/her multicast group and, consequently, the need for the allocation a specific channel. There are two procedures for dealing with this problem. The first is the reservation of a pool of contingency channels for handling VCR operations; whenever a batch of users is to be admitted into the system, the number of contingency channels is computed so that future VCR operation requests can be supported, and the batch is only accepted if a sufficient number of channels is available to handle potential VCR operations. If, however, a request for VCR operation is issued and there is no available channel in the contingency pool, a channel for regular playback can be allocated. The second option has no provision of a reserved pool of channels for the handling of unsynchronized streams, and each request for VCR operation competes with those for admitting new batches of users.

For both procedures, the allocated channel is occupied until the end of the exhibition, or, if possible, until merging with another stream. Moreover, if no channel is available to support a request for VCR operation, the request is refused and the user is forced to remain in his/her multicast group. An orthogonal approach would be to delay granting the request. This is not considered here, since it has not been proved to influence user perception of QoS. The bandwidth allocated for VCR operations is the same as for playback mode, i.e, when performing REW and FF, the pictorial quality is reduced.

IV) Dimensioning Number of Channels in Interactive VoD System

To assure proper functioning of an interactive VoD system, the size of the contingency pool should be dimensioned so that only the minimal number of requests for VCR operations will be rejected. The size of the pool varies as a function of the number of users admitted into the system. Therefore, whenever a batch of users is accepted into the system or leaves the system, the size of the contingency pool changes.

It is assumed that the arrival of a request for video exhibition follows a Poisson process, and that video titles are chosen according to a Zip distribution [7]. Not all users perform VCR operations, and only those who subscribe to interactive services are allowed to issue requests for such operations. The interarrival time of requests per user is exponentially distributed, as well as the duration of VCR operations, assumptions based on real data collected from an operational system [8].

System with Batching

In a system with batching, whenever a channel is allocated to an unsychronized stream, it is held until the end of the exhibition. Hence, dimensioning the size of the contingency pool is an easy task. Whenever a batch of n users is accepted into the system, $n \; x \; P_{u\_vcr}$ channels should be reserved for handling unsynchronized streams, where P_{u_vcr} is the probability of a user subscribing to interactive services.

System with Batching and Piggybacking

In a system with both batching and piggybacking, however, after the performance of a VCR operation, a video stream can be synchronized with another one, leading to the release of one of the video channels. The holding time of a contingency channel will be limited to the time required to perform the VCR operation plus the time required for resynchronization with another stream.

The number of users in the system between the arrivals of two batches or between the arrival of a batch and the end of an exhibition is fixed. Thus, the system can be modelled using a central server model, a specific closed queuing network model. In this model, the time users stay in playback mode corresponds to the service time of an infinite server [9]. After visiting the infinite server, users go to a queue with multiple servers, i.e., an M/M/c queue (a load dependent server in queuing network terminology). The service time of this queue corresponds to the holding time of a contingency channel. It includes the mean time required to perform VCR operations plus the time required to resynchronize with another stream, with weighting determined by the probability of the specific type of VCR operation (PAUSE, REW, and FF). The number of servers, *c*, is actually the size of the contingency pool. The key issue is to dimension *c* so that no user has to wait to be served. In this model, the failure to find an unoccupied server available corresponds to the rejection of a VCR operation in the real system. Therefore, *c* has to be computed so that the minimal number of requests will be rejected.

The size of the contingency pool could be determined by a queuing network algorithm, which is executed with a certain *c* value, and then verified to check the queue size of the load dependent server. This process proceeds until a value of *c* is found such that the mean queue size at the load dependent server is very small. However, the mean value analysis and the convolution algorithms [10] - [11], which are exact algorithm for closed queuing networks, present numerical instability when the queue size at the load dependent servers is very small. The normalized convolution algorithm [12] could be used to overcome this difficulty, but it is also unstable when the service time of the load dependent queue is a significant part of the cycle time (i.e. the time required to visit both the infinite server and the load dependent queue).

Given the numerical instability of exact algorithms for closed queuing networks, an open approximating model was adopted. Such a VoD system is modelled as if it where an Erlang B queue. The load dependent server in the closed queuing networks corresponds to an M/M/c queue with no waiting space in the open model, and requests are lost if no server is available. Although the arrival rate to the load dependent server in the closed queuing network algorithm is implicitly computed, in the open model, it must be approximated (Figure 1).

To determine the approximate arrival rate to the M/M/c queue, two users' states must be considered: playback and VCR. In the VCR state, the user holds a contingency channel, whereas in the playback state he/she is part of a multicast group. The mean arrival rate of VCR requests, i.e., the mean arrival rate to the M/M/c queue, is given by the mean number of users in the playback state who perform VCR operations times the rate of VCR requests per user:

$$\bar{\lambda} = N \ \lambda_{vcr} \ P_{playback}$$

where N is the number of users who perform VCR operations, i.e., the number of users admitted into the system who subscribe to interactive services,

λ_{vcr} is the rate of VCR requests per user, and

$P_{playback}$ is the probability of a user being in playback state.

The probability of being in VCR state, P_{vcr}, is the fraction of time a user holds a contingency channel during the whole exhibition. The duration of the exhibition is the original duration of the movie lengthened by the time spent performing the VCR operation. Moreover, the mean holding time of a contingency channel includes the mean duration of a VCR operation plus the mean resynchronization time. P_{vcr} is given by the following:

$$P_{vcr} = \frac{N_{vcr} \; D}{T + N_{vcr} \times t_{vcr}}$$

where N_{vcr} is the mean number of VCR operations performed by a user,

D is the mean holding time of a contingency channel per VCR operation,

t_{vcr} is the mean duration of a VCR operation, and

T is the original duration of the video.

The mean holding time of a contingency channel includes the mean time required for resynchronizing with another stream. To simplify the computation of *D*, it is assumed that the unsynchronized stream is eligible for merging only with its original stream, although in reality a stream can be merged with any stream exhibiting the same movie. The value of *D* is, thus, an upper bound for the true *D* value, since the unsynchronized stream may be merged with another stream which is closer to it than the original stream. The time of resynchronization depends on both the type, and duration of the VCR operation, as well as on the frame position at which the VCR operation was requested. Moreover, this position depends on the number of operations requested during the movie exhibition. *D* is, thus, given by the following:

$$D = \sum_{n=1}^{\infty} d(n) \;\; p(n)$$

where *d (n)* is the mean holding time of a contingency channel, given *n* operations per user during the video exhibition, and

p (n) is the probability of a user requesting *n* VCR operations during the movie exhibition.

p (n) and *d (n)* are expressed by the following equations:

$$p(n) = \frac{(\lambda_{vcr} \;\; T)^{n}}{n!} \; e^{(-\lambda_{vcr} \;\; T)}$$

$$d(n) = \frac{1}{n}\sum_{i=1}^{n}\sum_{op}\sum_{s=i+1}^{L-i} d_{op}(s)\ \lambda_{vcr}\ \frac{\lambda_{vcr}^{(i-1)}}{(i-1)!}\ e^{-\lambda_{vcr}}\ P_{op}$$

where λ_{vcr} is the rate of requests for VCR operation per user,

L is the video duration in frames,

$d_{op}\ (s)$ is the contingency channel holding time of an *op* operation which was issued at the *s*th frame,

P_{op} is the probability of the type of VCR operation (PAUSE, FF or REW), and

$\lambda_{vcr}\ \frac{\lambda_{vcr}^{(i-1)}}{(i-1)!}\ e^{-\lambda_{vcr}}$ is the probability that the *i*th operation be issued at the *s*th frame.

Each VCR operation has a different mean duration. Therefore, $d_{op}\ (s, t)$ should be computed as a function of the operation type:

$$d_{op}(s) = \int_{0}^{Max_{op}(s)} G_{op}(s,t)\ F_{op}(t)\qquad dt$$

where $Max_{op}\ (s)$ is the maximum duration of the operation *op* which occurred at the s^{th} frame,

$F_{op}\ (t)$ is the probability density of the duration of the operation *op*, and

$G_{op}\ (s, t)$ is the duration of the contingency channel holding time of a VCR operation which occurred at the s^{th} frame and lasted t seconds.

G_{op} is calculated as follows:

$$G_{Pause}(s,t) = \begin{cases} t + t \times A & if \quad t \times A \le \dfrac{L-s}{V_{max}} \\[2ex] t + \dfrac{L-s}{V_{max}} & if \quad t \times A > \dfrac{L-s}{V_{max}} \end{cases}$$

$$G_{Rew}(s,t) =$$

$$\begin{cases} t + (t + t \times R_{Rew}) \times A & if \;\; (t + t \times R_{Rew}) \times A \le \dfrac{L - (s - t \times V_{Rew})}{V_{max}} \\[2ex] t + \dfrac{L - (s - t \times V_{Rew})}{V_{max}} & if \;\; (t + t \times R_{Rew}) \times A > \dfrac{L - (s - t \times V_{Rew})}{V_{max}} \end{cases}$$

$$G_{FF}(s,t) =$$

$$\begin{cases} t + (t \times R_{FF} - t) \times A & if \;\; (t \times R_{FF} - t) \times A \le \dfrac{L - (s + t \times V_{FF})}{V_{max}} \\[2ex] t + \dfrac{L - (s + t \times V_{FF})}{V_{max}} & if \;\; (t \times R_{FF} - t) \times A > \dfrac{L - (s + t \times V_{FF})}{V_{max}} \end{cases}$$

where $A = \dfrac{V_{playback}}{V_{max} - V_{min}}$ is a constant which, when multiplied by the duration of the VCR operation, indicates the time needed to merge with the original stream,

V_{max} and V_{min} are the maximum and minimum display rates, respectively,

$V_{playback}$ is the normal display rate during playback,

$V_{Rew} = R_{Rew} \times V_{playback}$,

$V_{FF} = R_{FF} \times V_{playback}$, and

t is the duration of the VCR operation.

For instance $d_{Rew}\,(s, t)$ is given by:

$$d_{Rew}(s, t) =$$

$$(1 + A \cdot e)\int_{0}^{Min(L'/(A\ \ e - B), s/V_{Rew})} t \times \exp(\lambda_{Rew})\ \ dt \quad +$$

$$\int_{Min(L'/(A\ \ e - B), s/V_{Rew})}^{s/V_{Rew}} ((1 + B)\ \ t + L')\ \ \exp(\lambda_{Rew}) \quad dt$$

where $L' = \frac{L - s}{V_{Max}}$, $B = \frac{V_{Rew}}{V_{max}}$ and $e = 1 + R_{Rew}$

V) Accuracy of Approximation Model

To assess the accuracy of this approximation model, the estimated channel demand was compared to the results derived via simulation. Instead of simulating the whole system, this was limited to the arrival of request for VCR operations at the contingency pool. To vary the mean arrival rate of requests of VCR operations($\bar{\lambda}$), different values of N and λ_{vcr} were chosen, with those values of λ_{vcr} determined by the use of $\lambda_{\mathrm{vcr}} = N_{vcr} / T$, and variation of N in the range of 50 to 2000 and of N_{vcr} in the range of 1 to 5. Different values of $P_{playback}$ were obtained by changing the mean duration of VCR operations, i. e., the mean of $F_{op}(t)$. The estimated channel demand was compared to the mean and the maximum values obtained via simulation. For a fixed set of parameters values, the simulation experiments were run with different seeds for the random number generator.

Figure 2 illustrates the estimated number of demanded channels and the mean and maximum number of channels obtained via simulation, as a function of N, for different values of

N_{vcr}. The channel demand, as estimated by the approximation model, results in an upper bound of the mean number of demanded channels and a lower bound of the maximum number. The assumption that a video stream, which holds a contingency channel, can only merge with the stream associated with its original multicast group overestimates the mean holding time of contingency channels, and, consequently, leads to an inflated estimation of the required number of contingency channels.

N_{vcr} is the parameter which has the greatest influence on whether the estimated value is closer to the mean or the maximum. The higher N_{vcr} is, the closer the estimated value is to the maximum. This can be explained by the fact that the greater the number of visits to the contingency pool, the greater the chance of a request for VCR operation being issued at a frame that further merging with the original multicast stream will be impossible, thus leading to a longer mean holding time of a contingency channel.

VI) VoD System Model

The effectiveness of various interactive VoD systems has been evaluated using a specially-developed simulator. The performance parameters considered were the following: i) number of users admitted into the system, ii) probability of reneging (ratio between number of users who gave up watching a video and the total number of requests arriving to the system), and iii) percentage of denied VCR requests (rejections). Service providers aim at increasing the number of users admitted to the system, i.e., the number of granted requests for movie exhibition. The probability of reneging can be associate with the revenue lost due to the refusal of requests for video exhibition. The percentage of denied VCR operations reflects denials due to the lack of a channel to support them. These represent the breaking of the contract between the user and the service provider, and the frequency with which this occurs is a measure of the provided Quality of Service.

The arrival of a request to watch a movie follows a Poisson process with a mean rate of λ. Upon the arrival of a request, the video title involved is chosen according to a Zip distribution, and the request joins the queue for the chosen title. A Zip distribution accurately models user preferences, both at video rental stores and for operational video-on-demand systems [8]. Since not all users issue requests for VCR operations, when a batch of users is accepted into the system, the number of users in that group who will perform VCR operations is randomly determined, and the probability of a given user performing such an operation is expressed as p_{u_vcr}. The interarrival time of VCR operation requests is modelled by an exponential distribution, with the mean established by the duration of the video divided by the mean number of operations performed by a user (N_{vcr}.). The type of VCR operation is determined by P_{op} and the duration of this operation exponentially distributed, with a mean of $1 / \lambda_{\mathrm{op}}$, where *op* can be REW, FF, or PAUSE.

The batching policy used in the simulation experiment is the Look-Ahead-Maximize-Batch (LAMB) [3] and the piggybacking policy is the odd-even policy [1]. LAMB was chosen since it admits a larger number of users than any other batching policy. Under LAMB, scheduling points occur at the reneging time of users. LAMB maximizes the number of users admitted in a time window defined by the current scheduling time and the most distant reneging time of a user waiting to be served, taking into consideration all channel releases within this time window. When a decision is made to accept a new batch of users, the number of extra channels needed to support the demand of VCR operations brought about by this new batch is computed using the approach developed in Section IV, and the availability of this number of channels is verified. If insufficient channels are available to support the new batch, it is not admitted into the system at that time. Otherwise, it is accepted and the size of the contingency pool is increased by the number of necessary channels.

The odd-even piggybacking policy used merges streams by pairs [1]. Although other piggybacking policies such as Snapshot [13] and S2 [14] produce a lower number of displayed frames, both of these assume a Poisson arrival process. Such an assumption is not valid in a system with batching, since the Poissonic pattern of arrival is modified by the temporary holding of requests [4]. Any feasible merges are pursued in the simulation experiments, i.e., not only merges with the original streams, but also those with any other possible stream.

VII) Numerical Examples

The performance of the VoD system was simulated by the use of discrete event simulation. Evaluation used the replication method, considering intervals with a 99% level of confidence. The number of replications for each point on the curves displayed in this section was such that the width of the confidence intervals was at most 8% of the mean value, although these intervals are not displayed on the graphs for the sake of clarity in visual interpretation. VoD systems with both batching and with a combination of batching and piggybacking were studied, and the effectiveness of the adoption of a pool of contingency channels was also investigated.

Simulation experiments were conducted for various different system configurations under varying degrees of interactiveness and under different conditions of load. The results are reported as a function of the server capacity, i.e., the number of channels required for delivering video streams. The impact of server size (number of stored videos) on server performance was also assessed. For a fixed load, the rate of request for VCR operations was varied by modifying the degree of interactiveness, i.e., the percentage of users performing VCR operations, p_{u_vcr}. Target values for the probability of reneging, as well as for the percentage of denied VCR requests, were set at 1%, i.e., the number of channels in a VoD system was expected to be sufficient to limit these performance parameter values.

Results are shown for small servers with 100 stored movies of 2 hours of duration each and for large servers with 500 stored movies, as well as for low loads of 10 requests per minute and high ones of 60 requests per minute. Results are displayed for an average of 2 VCR operations per user was considered, with the probability of a pause request being 0.5 (with mean duration of 5 minutes) and that of both REW and FF being 0.25 (with mean duration of 30 seconds).

VoD System with Batching

Figures 3, 4 and 5 show the number of users admitted into the system, the probability of reneging, and the percentage of denied VCR operations, respectively, all as a function of the server capacity for different degrees of interactiveness for a small server. In all figures shown in this paper, WC and U denote "with a contingency pool" and "degree of interactiveness", respectively. The number of users admitted into the system is always greater for servers without a pool of contingency channels than it is for servers with such a pool. For a degree of interactiveness of 10%, there is no difference between the number of users admitted into the two systems, although as the degree of interactiveness increases, the difference between the number of admitted users increases. For 80% of interactiveness and a server capacity of 1000 channels, the number of users admitted into a system without a contingency pool is more than double the number in a system with a contingency pool. This difference decreases as the server capacity increases, since the number of users admitted into the system converges to the maximum value for a fixed load. The limitations of the presence of a contingency pool arise because such a system requires that channels be maintained idle to allow for eventual VCR operations, since without this need, these same channels could be used to admit a larger number of users.

Note that the number of users admitted into the system does not significantly increase despite an increase in the server capacity. This topping off effect can be attributed to the regulatory behaviour of batching policies, which admit users in a discrete, rather than continuous fashion. Users are admitted from time to time, and the inter admission interval is determined by the adopted batching policy. Therefore, it is not advantageous to increase the server capacity beyond a certain value, since the acquisition of new channels is regulated by the batching interadmission interval.

The probability of reneging in a system without a contingency pool is always less than it would be in the presence of such a pool (Fig. 4). The difference between number of channels needed to limit reneging to 1% in a system with a contingency pool and that in a system without one increases as the degree of interactiveness increases. For a low degree of interactiveness (10%), this difference is 40%, whereas for 80% interactiveness, the number of channels required in a system with a contingency pool is double that demanded by a system without such a pool.

Although for a system with a contingency pool with a server capacity above 300 channels, the number of denied VCR operations is less than 1%, in a system without such a pool this low percentage is only achieved with a server capacity of 700 channels for 10% interactiveness. When the interactiveness rises to 80%, a server capacity of 1,300 channels is required. Nevertheless, the apparent advantage of the presence of a contingency pool does not necessarily make such systems preferable, since the probability of reneging is greater in such systems. For instance, 700 channels are necessary to reduce the percentage of VCR operation denials to 1% if no contingency pool is provided, whereas in a system with such a pool, 1000 channels are needed to reduce the probability of reneging to 1% (10% interactiveness). In other words, when no contingency pool is provided, the target set of performance values is obtained with a lower server capacity. In such a system, when a high degree of interactiveness (80%) is involved, a server capacity of 1,400 channels is enough to maintain the percentage of denied VCR requests, as well as the reneging probability, below target values, while the provision of such a pool means that the server capacity must reach 3,500 channels. Therefore, for high degrees of interactiveness, systems without a contingency pool outperform those with such a pool.

There is thus a trade-off between denial of VCR operation requests and probability of reneging. The limitation of requests denied requires a large number of channels when no contingency pool is available, whereas the reduction of the probability of reneging in the presence of a contingency pool requires an even greater number of channels than when no such pool is available.

For high loads (60 requests per minute), the difference between the number of users admitted into a system with a contingency pool and the number of users admitted into a system without such a pool is larger than when low loads are involved, as illustrated in Figure 6. This difference increases, however, as the degree of interactiveness increases. For instance, for a system with 1000 channels and 10% interactiveness, the difference is 300 users; when Interactiveness rises to 80%, this difference rises to 1,100 users. For a given degree of interactiveness, to obtain the target reneging probability of systems with contingency channels requires a greater number of channels (Figure 7), with the difference between such probabilities increasing with the degree of interactiveness. Under high load conditions, without a contingency pool, the target percentage of denied VCR operations can only be achieved with a server capacity which is much higher than that needed for low loads (Figure 8). For high loads, for example, 80% interactiveness requires 5,800 channels, although for a low load, only 1,300 would be needed.

The server capacity under high load conditions as a function of the number of stored videos (server size) needed to reduce both probability of reneging and percentage of denied VCR operations to below 1% is shown in Figures 9 and 10, respectively. The increase in server capacity, coupled with the increase in server size, is more pronounced in systems with a contingency pool than in those without one, especially for medium to high degrees of interactiveness. Thus, for 80% interactiveness, an increase in server size from 100 to 300 stored movies leads to the need for an increase of 2000 channels in a system with a contingency pool, whereas this increase would be only 100 channels in systems without one. Figure 10 shows the relative irrelevance of server size increase in relation to channel demands for systems with a contingency pool.

A comparison of Figures 9 and 10 makes it clear that a system without a contingency pool is preferable. The server capacity increases to maintain the percentage of denied VCR operations below 1% required in such systems is much lower than that required to maintain the reneging probability below 1% in systems with a contingency pool. Moreover, this increase is lower for conditions of low loads than for high ones. Under the former conditions, an increase in capacity is significant only for systems with contingency pools and medium to high interactiveness. For example, an increase in server size from 100 to 300 stored movies leads to the need for an increase of 1,500 channels and 1000 channels for 80% and 40% interactiveness, respectively.

VoD Systems with Both Batching and Piggybacking

For a small server under conditions of low load, Figures 11, 12 and 13 show the number of users admitted into the system, the probability of reneging, and the percentage of denied VCR operations, respectively.

Under low load conditions, the number of users admitted into a system combining batching with piggybacking follows the same trend found in systems with batching only, with the difference increasing with the degree of interactiveness; the difference is greater when no contingency pool is provided, although for low degrees of interactiveness, this difference is not significant. The difference is greater in systems with batching only than in those using both batching and piggybacking, especially for high degrees of interactiveness.

For a given degree of interactiveness, systems without a contingency pool involve a lower probability of reneging than do those with a contingency pool (Figure 12). Moreover, in a system with both batching and piggybacking, specific probability values are obtained with a smaller number of channels (Figure 4). In a system with a contingency pool and medium to high interactiveness, a system with batching only doubles the channel demand of a system with both batching and piggybacking.

The pattern of VCR operation denial observed in a system with batching only (Figure 5) is similar to that observed in a system with both batching and piggybacking (Figure 13). A server capacity of 300 channels is required for a system with a contingency pool with batching only, whereas only 50 channels will be required with both batching and piggybacking. Results similar to those found for systems with batching only are found in systems with both batching and piggybacking. The reservation of a pool of contingency channels is thus not an attractive option, since their reservation means that a greater channel capacity will be required to maintain probability of reneging below a given target value (Figure 12).

The advantages of adopting both batching and piggybacking are especially striking when high loads are involved. With batching only, the maximum number of admitted users is 4,500 (Figure 6), whereas the adoption of both raises this number to 8,000 (Figure 14) since piggybacking makes it possible for channels to be released when video streams merge, thus allowing the acceptance of a larger number of users.

A higher load does still lead to the need for a higher server capacity to reduce the probability of reneging to the target value, as can be seen in the increase from 1,500 channels for high interactiveness under low load conditions (Figure 12) to 5,800 channels under high load conditions (Figure 15). The advantage of adopting both batching and piggybacking can also be appreciated by comparing the channel demands of a system with both (Figure 15) with those of one with batching only (Figure 7). For example, for a high degree of interactiviness and for a system with a contingency pool of buffers in a system with both batching and piggybacking the channel demand is 5,900 while it is 7,300 channels in a system with batching only.

Under high loads, conditions of reduced interactiveness reduce the negative impact of a pool of contingency channels, although under conditions of medium to high interactiveness server capacity demand is less when no contingency channels are provided.

Figures 17 and 18 show the server capacity as a function of server size in the maintenance of probability of reneging and percentage of denied VCR operations below 1% under high load conditions. Server capacity demands are lower when both batching and piggybacking are involved (Figure 17) than with batching only (Figure 9). For high interaction situations, for example, an increase in server size from 100 to 1000 videos leads to a server capacity demand of 1000 channels in a system with both batching and piggybacking (Figure 17), although this increases to 4000 channels in a system with batching only (Figure 9). The impact of using both batching and piggybacking is insignificant in relation to VCR operation denial.

It is worth noting that the assumption that a video stream which holds a contingency channel can only merge with the original multicast stream leads to an overestimation of the number of channel demand. The results presented in the present paper, however, correspond to a set of experiments with $N_{vcr} = 2$, which are close to the mean number of demanded channels, and, do not overestimate the number of demanded channels. Therefore, conclusions are not influenced by the adopted assumptions.

The execution time of the proposed method was assessed in a SPARC machine with clock rate 248 MHz. The execution time was of the order of μ-seconds. For instance, for a system with 2,000 users who perform VCR operations and request on average 4 VCR operations, the execution time was 390 μsec. Thus, the method introduced here is amenable for real time implementation.

VIII) Conclusions

This paper has introduced an approximation model to determine the number of channels required to support VCR operations in VoD systems. The model is an Erlang B queue with an arrival rate approximating the arrival rate of requests for VCR operation. The accuracy of the prediction of number of required channels has been checked against simulation results. The predicted value is an upper bound of the mean number of demanded channels, and a lower bound of the maximum number.

The performance of various VoD systems was analysed. The effectiveness of the reservation of a pool of contingency channels to handle VCR operations was investigated, and the results suggest that target performance values are obtained with a smaller server capacity when no contingency channels are available. Moreover, an interactive system with both batching and piggybacking admits a larger number of users, and provides lower reneging probabilities than does a system with batching only.

Follow up work comparing systems with delayed VCR requests with those in which such requests are rejected is suggested.

Acknowledgements

This work was partially sponsored by CNPq and CAPES. We would like to thank Edmundo Souza e Silva and Daniel Menascé for the discussion about queueing network algorithms.

References

[1] L. Golubchik, J. C. S. Lui and Muntz, “Adaptive Piggybacking: A Novel Technique for Data Sharing in Video-on-Demand Storage Servers”, *Multimedia Systems*, 4(3):140--155, 1996.

[2] C. C. Aggarwal, J. L. Wolf and P. S. Yu. “The Maximum Factor Queue Lengh Batching Scheme for Video-on-Demand Systems”, Techinical Report RC 20261 (91305), IBM Research Division, T. J. Watson Research Center, Yorktown Heights, November 1996.

[3] N.L.S. Fonseca and R. A. Façanha, “The Look-Ahead-Maximize-Batch Batching Policy”, to appear in *IEEE Transactions on Multimedia*.

[4] N. L. S. da Fonseca and R. A. Façanha, “Integrating Batching and Piggybacking in Video Server”, in Proc. of *IEEE Global Telecommunications Conference*, pg 1334-1338, 2000.

[5] J. K. Dey-Sircar, J. D. Salehi, J. F. Kurose and D. Towsley, “Providing VCR Capabilities in Large-Scale Video Servers”, in Proceedings of *2nd ACM International Conference on Multimedia*, pg 25-36, San Francisco, CA, 1994.

[6] A. Dan, P. Shahabuddin, D. Sitaram and D. Towsley, “Channel Allocation under Batching and VCR Control in Video-on-Demand Sysrems”, *Journal of Parallel and Distributed Computing,* vol 30, pg 168-179, 1995.

[7] Y. S. Chen, “Mathematical modeling of empirical laws in computer application: a case study”, *Comp. Math. Applicat.*, pp. 77-87, Oct. 1992.

[8] P. Branch, C. Edgar and B. Sonkin, “Modeling Interactive Behavior of a Video Based Multimedia System”, in Proc. of *IEEE International Conference on Communications*, 978-982, 1999.

[9] D. A. Menascé, V. A. F. Almeida and L. W. Dowdy, “Capacity Planning and Performance Modeling”, PTR Prentice Hall, 1994.

[10] E. de Souza e Silva and R.R. Muntz, “Queueing Networks: Solutions and Applications”, in *Stochastic Analysis of Computer and Communication Systems*, H. Takagi editor, North Holland, 1990

[11] A. E. Conway and N. D. Georganas, “Queueing Networks - Exact Computational Algorithms: A Unified Theory Based on Decomposition and Aggregation.”, The MIT Press, 1989.

[12] M. Reiser, “Mean-Value-Analysis and Convolution Method for Queue-Dependent Servers in Closed Queueing Networks”, *Performance Evaluation*, vol 1, pg 7-18, 1981.

[13] C. C. Aggarwal, J. Wolf and Philip S. Yu, “On Optimal Piggybacking Merging Policies for Video-on-Demand Systems”, in Proc. of the *ACM Sigmetrics*, vol 24, pp. 200--209, 1996

[14] R. A. Façanha and N.L.S. da Fonseca, “A Piggybacking Policy for Reducing the Bandwidth Demand of Video Servers”, *Managing QoS in Multimedia Networks and Services*, J.N. de Souza e R. Boutaba (editors), pg 225-236,Kluwer Academic Publishers, 2000.

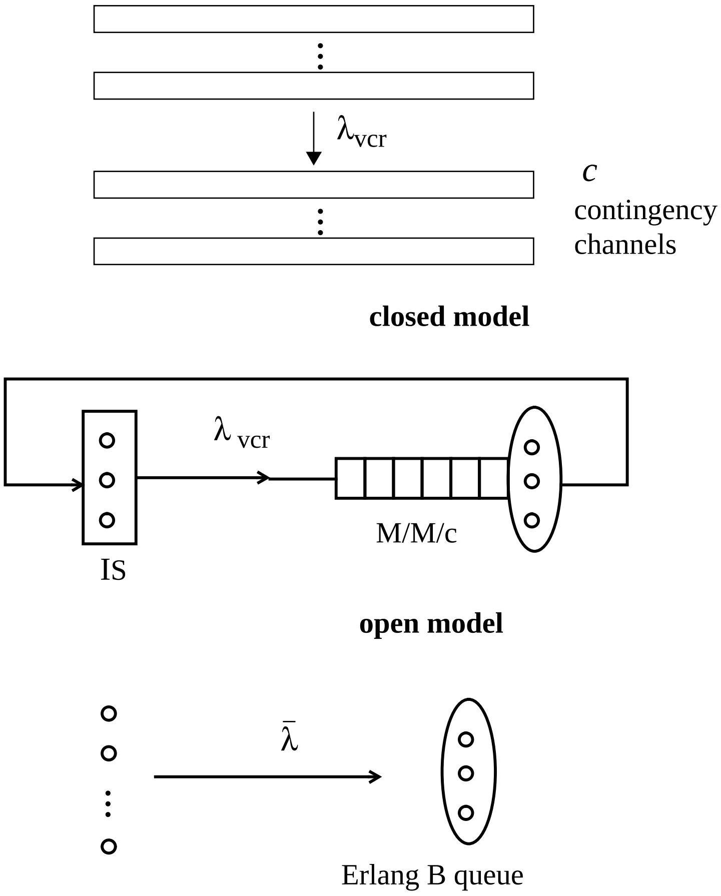

Figure 1: Approximate open model.

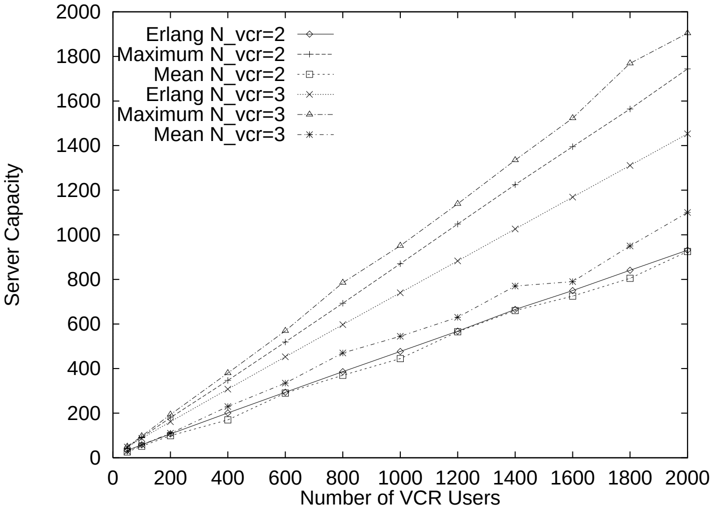

Figure 2: Estimated number of demanded channels x Simulation results.

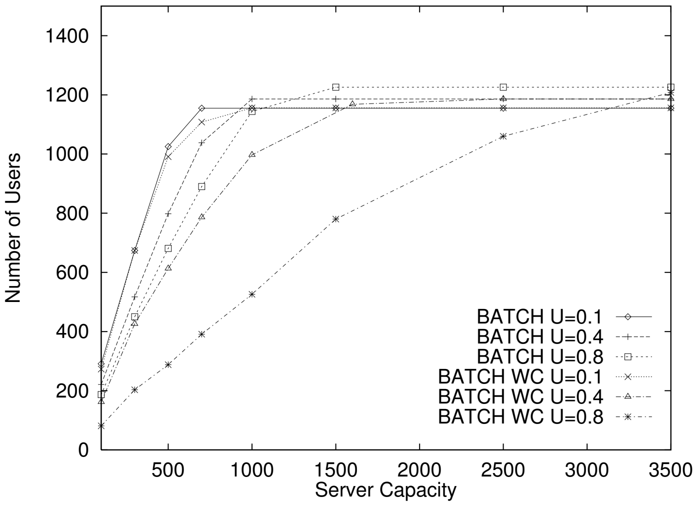

Figure 3: Number of Admitted Users X Server Capacity for an arrival rate of 10 req/min.

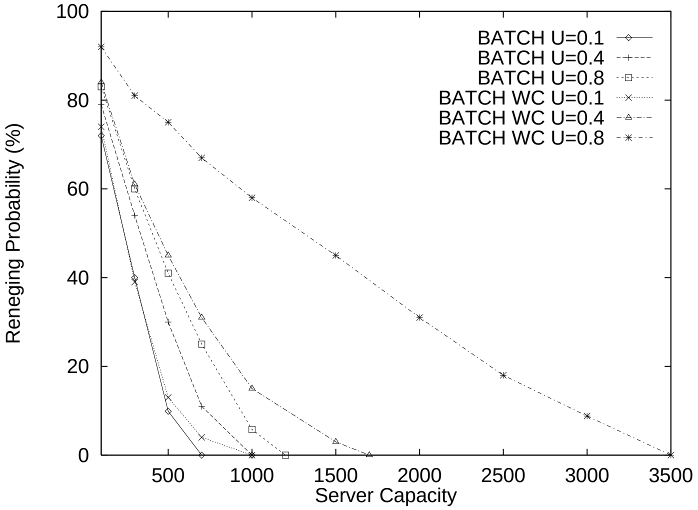

Figure 4: Reneging Probability X Server Capacity for an arrival rate of 10 req/min.

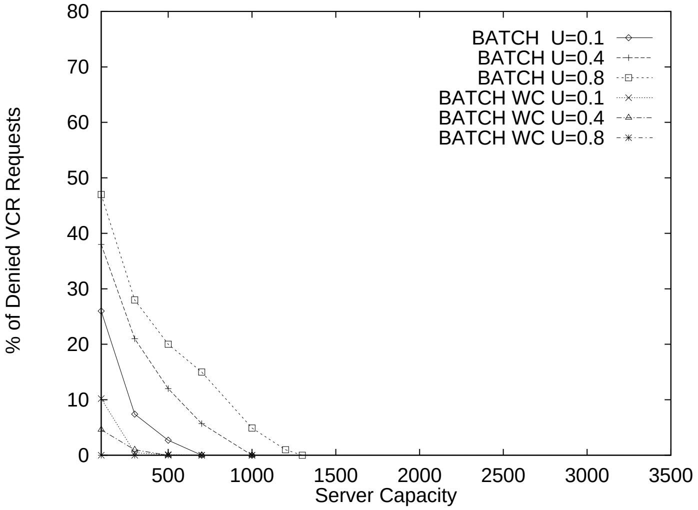

Figure 5: Percentage of Denied VCR Requests X Server Capacity for an arrival rate of 10 req/min.

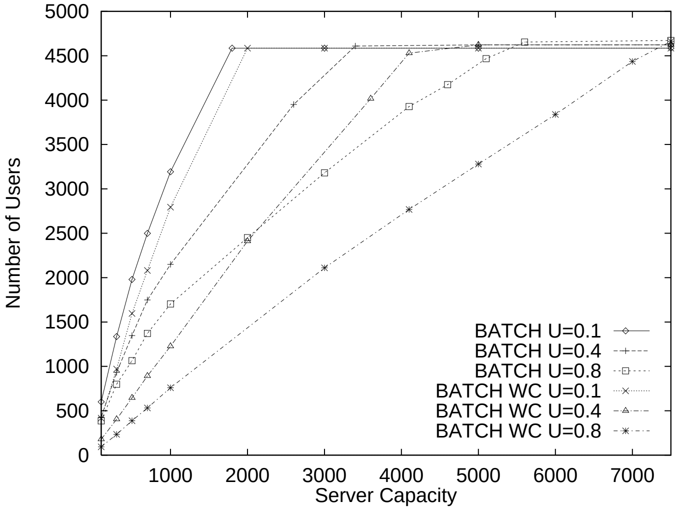

Figure 6: Number of Admitted Users X Server Capacity for an arrival rate of 60 req/min.

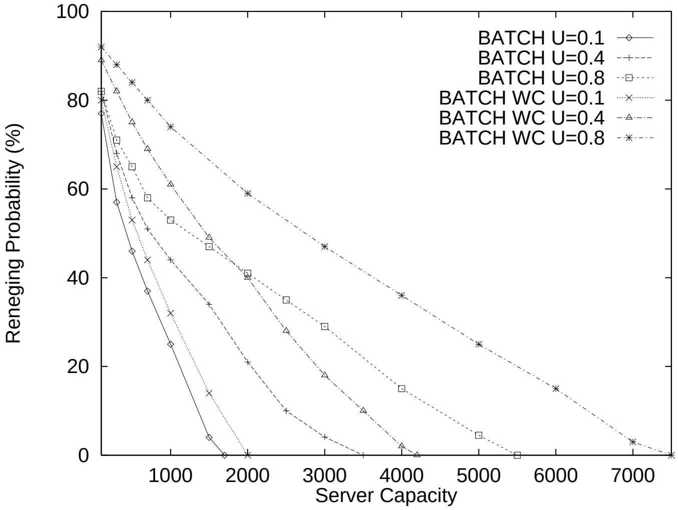

Figure 7: Reneging Probability X Server Capacity for an arrival rate of 60 req/min

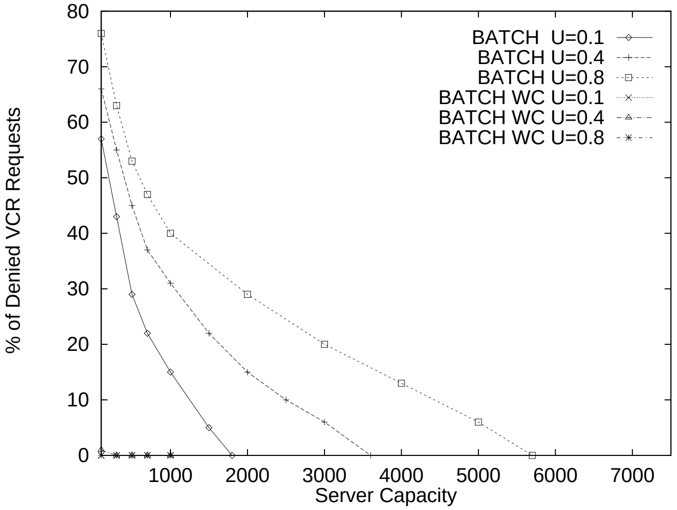

Figure 8: Percentage of Denied VCR Requests X Server Capacity for an arrival rate of 60 req/min.

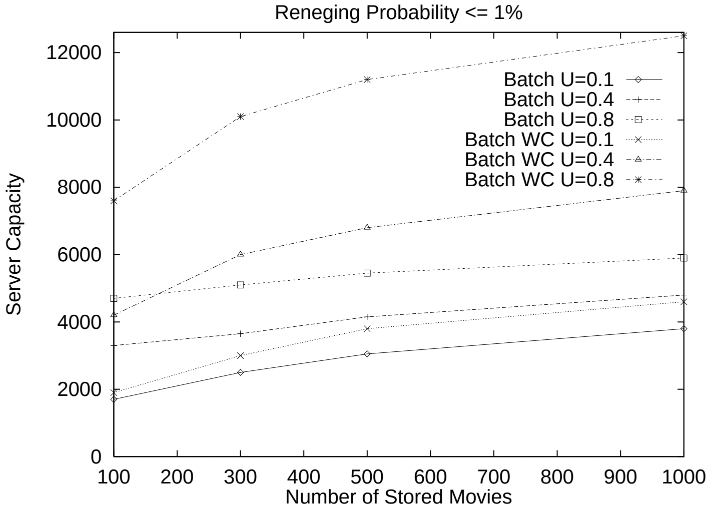

Figure 9: Server Capacity X Server Size for an arrival rate of 60 req/min, Reneging Probability ≤ 1%.

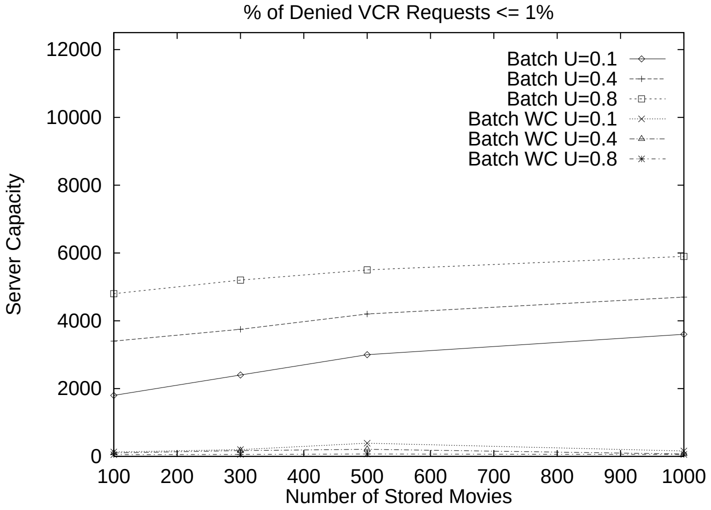

Figure 10: Server Capacity X Server Size or an arrival rate of 60 req/min., Percentage of Denied VCR Requests ≤ 1%.

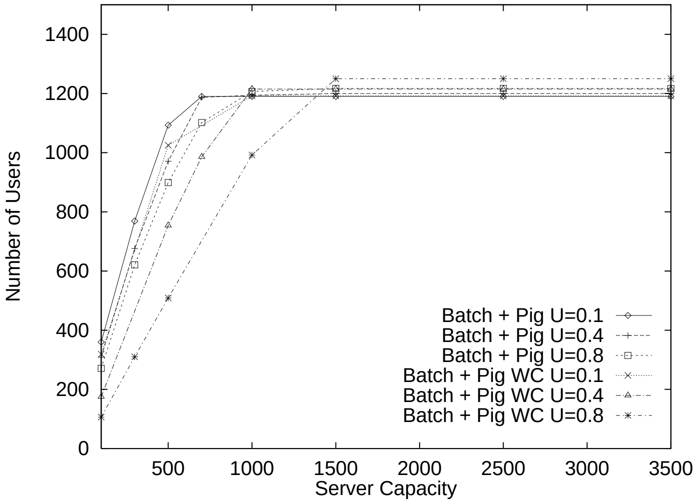

Figure 11: Number of Admitted Users X Server Capacity for an arrival rate of 10 req/min

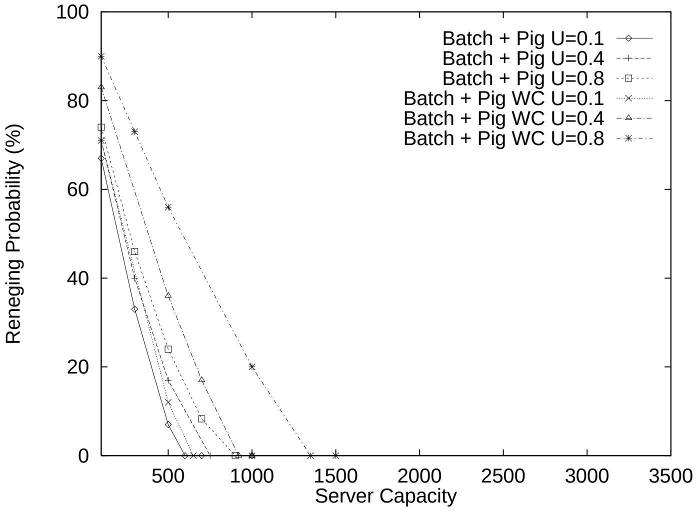

Figure 12: Reneging Probability X Server Capacity for an arrival rate of 10 req/min

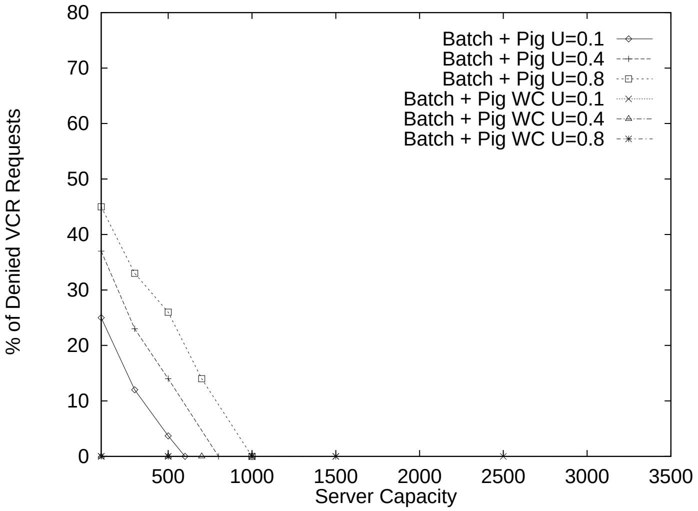

Figure 13: Percentage of Denied VCR Requests X Server Capacity for an arrival rate of 10 req/min.

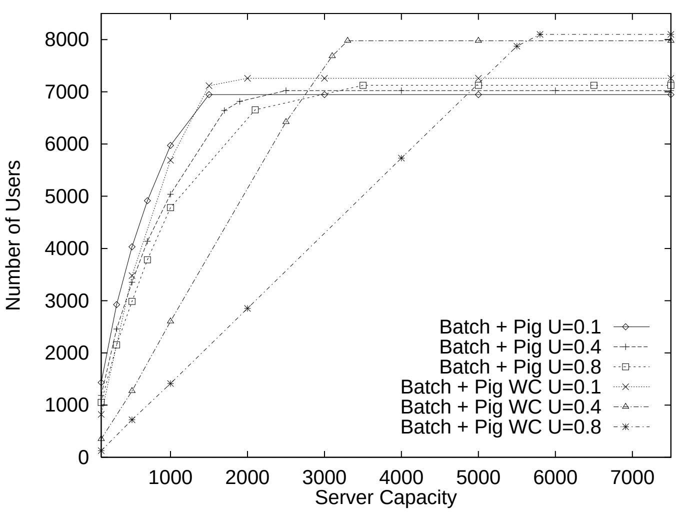

Figure 14: Number of Admitted Users X Server Capacity for an arrival rate of 60 req/min

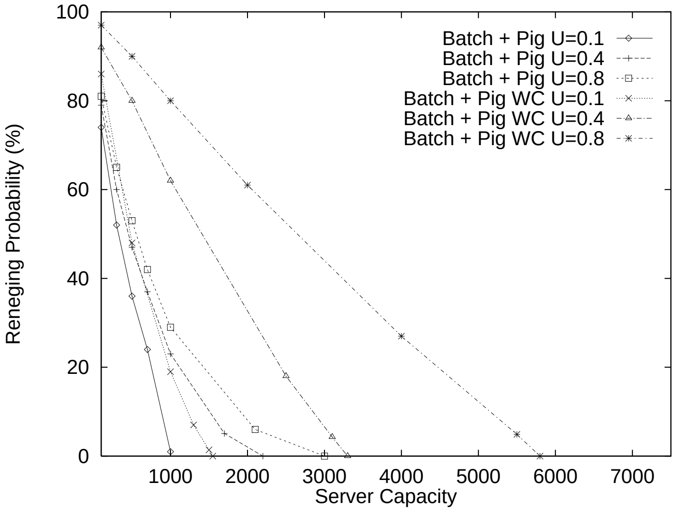

Figure 15: Reneging Probability X Server Capacity for an arrival rate of 60 req/mi

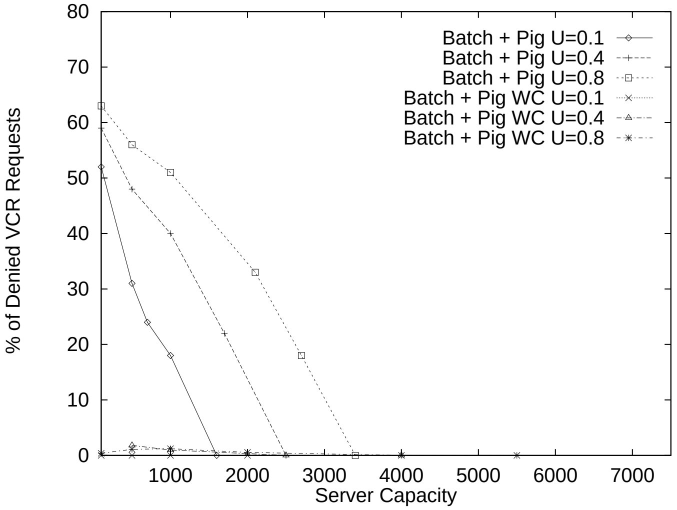

Figure 16: Percentage of Denied VCR Requests X Server Capacity for an arrival rate of 60 req/min

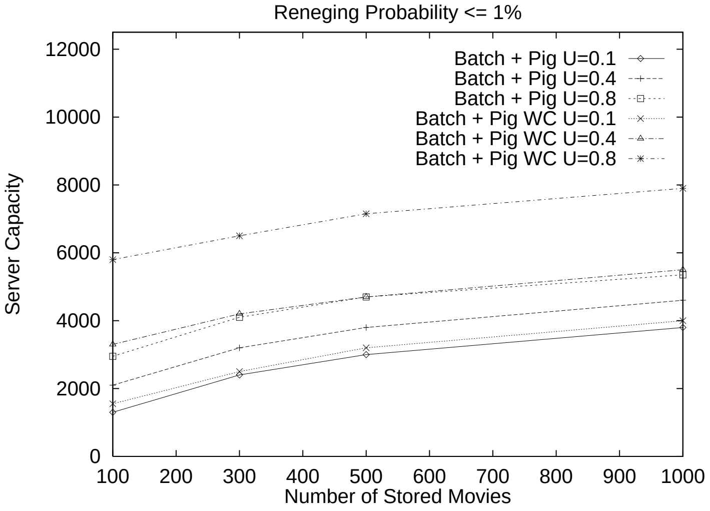

Figure 17: Server Capacity X Server Size for an arrival rate of 60 req/min,
Reneging Probability ≤ 1%.

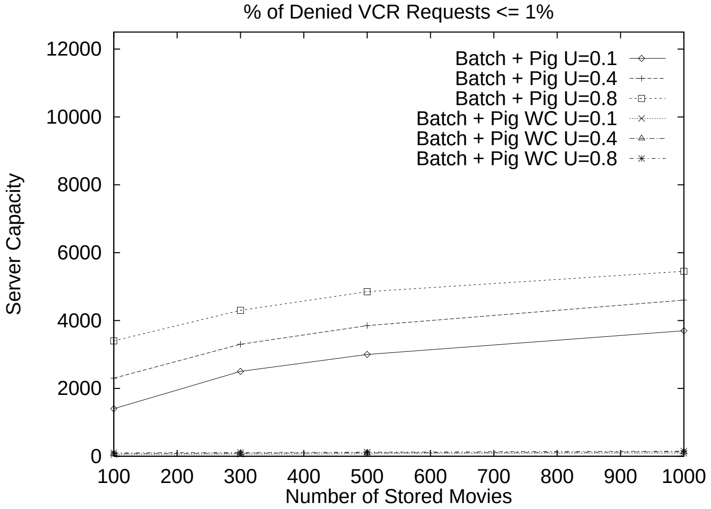

Figure 18: Server Capacity X Server Size or an arrival rate of 60 req/min., Percentage of Denied VCR Requests ≤ 1%.