File Distribution in Networks with Multimedia Storage Servers

Jeong-dong Ryoo and Shivendra S. Panwar

Center for Advanced Technology in Telecommunications
Polytechnic University
Five Metrotech Center, Brooklyn, NY 11201, USA

Abstract

In this paper, we consider an optimization problem in networks with storage servers for providing multimedia service. The design involves assigning communication link capacity, sizing the multimedia servers and distributing different types of content at each server, while guaranteeing an upper limit on the individual end-to-end blocking probability. We consider alternate methods for obtaining the end-to-end blocking probability with low computation time and present optimization procedures to obtain an optimal solution. Under a linear cost structure, our numerical investigations consider different scenarios that might be helpful in understanding how to distribute multimedia content for a cost-optimized solution.

1 Introduction

In the recent past, many researchers have investigated the issues related to the delivery of interactive video service in networks with video servers. Among them, there has been much interest in dealing with network design issues related to video-on-demand service. Several papers addressed the network and server requirements for designing cost-effective solutions for interactive video services [1, 2]. Another important focus in this area has been on the evaluation of the cost tradeoffs between centralized and distributed architectures [3, 4]. We previously addressed an optimization problem of designing networks for providing video-on-demand service [5].

The types of delivered service need not be confined to video-on-demand, but can be extended to a variety of multimedia services, e.g., digitized voice, high quality audio, and interactive video. The characteristics of multimedia service, such as required bandwidth for delivering a continuous stream of data, user access probability requirements, and holding times, influence the design of a cost-optimized architecture. This also further complicates the problem to be addressed in determining file distribution in the network.

In this paper, we consider an optimization problem in networks with storage servers for providing multimedia service. We assume that the connection blocking probability for an end user's request to obtain a desired service is the performance criterion of interest. Unlike other work that focused on the proportion to be placed at local or central servers, our design involves assigning communication link capacity, sizing the multimedia servers and distributing different types of service content at each server, while guaranteeing an upper limit on the individual end-to-end blocking probability. The solution reveals to the network designer the tradeoffs between storage cost, server cost and communication link cost in meeting a given grade of service.

By focusing on end-to-end blocking rather than constraining link blocking probabilities, as is often done, the network design can explicitly trade-off the cost of network elements and blocking. This would allow for lower cost feasible solutions where the elements with the highest cost contribute the most to the total end-to-end blocking, exceeding typical link blocking constraints. However, a constraint on the individual end-to-end blocking probability as compared to link blocking imposes a price in the form of high computation time in the optimization problem. Therefore, we suggest efficient methods for an optimization problem in networks with multimedia servers with end-to-end blocking constraints. The procedures find the best file distribution scheme as well as optimizes the size of the elements in a network with multimedia servers.

First, we start with a model formulation for providing multimedia service. After we formally define our optimization problem, we discuss how we can use approximations to compute the end-to-end blocking probability to reduce optimization time. In Section 3, we present two procedures to obtain an optimal solution, based upon the properties of the tradeoff between communication link cost and storage server cost. Under a linear cost structure, our numerical investigations consider different scenarios that illustrate how to distribute multimedia content for a cost-optimized solution.

2 Model Formulation

2.1 A structure for multimedia service networks

Consider a network with an arbitrary tree topology as shown in Fig. 1. Every residential customer is connected to a point, called a distribution cabinet in Fig. 1. This point corresponds to an *Optical Network Unit*(ONU) in an optical distribution network, a switch in the *Central Office*(CO) of a telephone network, or a headend

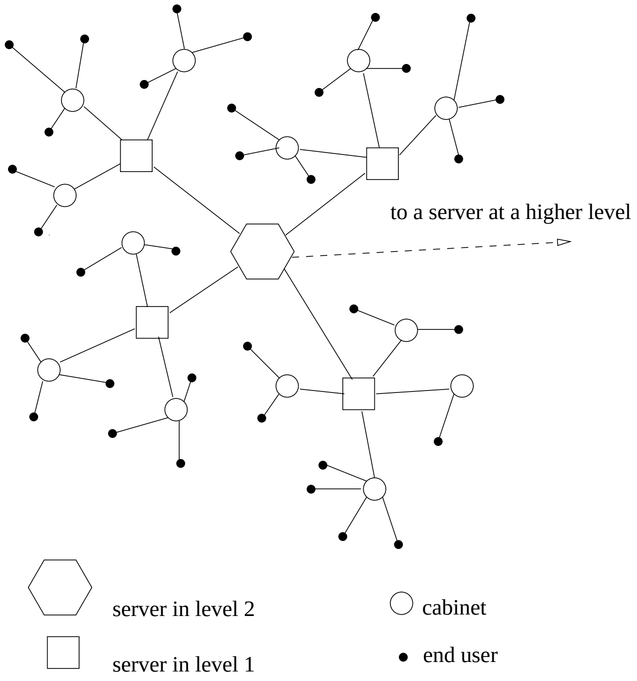


Figure 1: Network reference model

in a cable network, depending on the access network technology. From Asymmetric Digital Subscriber Line(ADSL) to Very-high-speed Digital Subscriber Line(VDSL), there exist a variety of *XDSL* technologies for delivering multimedia content over existing telephone lines. Hybrid Fiber/Coax(HFC) transport architectures are also regarded as a promising option for the delivery of interactive broadband services [6]. When this concentration point is absent and a customer request goes to a multimedia server directly, we can easily reformulate a new cost function and performance evaluation formula. Each node above this point is a potential site for a multimedia server. Multimedia servers can be placed in any node at any level of a tree network.

Once we focus on end-to-end or call blocking probability as a measure of grade of service, the two most important parameters of a multimedia server are the bandwidth of the I/O controller that is attached to the network and the maximum number of simultaneous streams that a server can support for a specific title. Thus a multimedia server is assumed to perform admission control for a new request based upon the simultaneous availability of the output bandwidth in the server and the requested file. Once the server cannot access a file or does not have enough bandwidth to serve a request for a file, the request is blocked.

In this paper, the number of streams that can access a specific title is assumed to be limited by the "number of copies" of the particular file. What we mean by the "number of copies" is the number of independent streams that can be supported simultaneously for the requested title, which in general may exceed the actual number of physical copies.

A request for a multimedia service is assumed to be assigned to a predetermined server that contains the requested file. The content files are distributed among the servers and each server contains a subset of the available files. We assume that files are placed, starting with the highest level server, in inverse order of their popularity.

Thus by placing the most popular titles closest to the users, the communication cost can be reduced, and the incremental storage cost is less due to the multiplexing gain. Through numerical investigations, we observed that this assumption of placing the most popular titles to low level servers appears to be valid for our network design problem under the end-to-end blocking constraint(Appendix A).

We classify several types of services according to their bandwidth requirements and mean holding time characteristics. The bandwidth requirement may be either a constant bit route(CBR) or the *effective bandwidth* if variable bit rate(VBR), when the core network operates using ATM technology [7]. It is also applicable to IP networks using streaming technology, which can define any "target" rate for the bandwidth requirement. Depending on the type of application, mean holding times can also vary.

In Fig. 2, we show the file distribution for a predetermined server selection in the case of an L level tree network. w_l denotes the number of nodes at level l of the tree, $l = 0, 1, \ldots, L$, where we define $w_L = 1$. Four types of services are available in this example. N_i denotes the number of available titles for service type $i \in \mathcal{I}$, where $\mathcal{I}$ is the set of service types, in increasing bandwidth order. The bandwidth of service type i is denoted by d_i. We will index the available titles from 1 to N_i, starting with the most popular file of type i. Let M_i^l, the *cutoff index*, be the index of the least popular type i file stored in a level l server. Therefore, the files ranging from the $(M_i^{l-1} + 1)$th most popular title to the M_i^lth one will be placed in a multimedia server in level l, where $M_i^0 = 0$. The required storage capacity m_l in a level l server is $\sum_{i \in \mathcal{I}} \sum_{j=M_i^{l-1}+1}^{M_i^l} t_{ij}^l b_{ij}$, where t_{ij}^l is the number of copies of the jth popular file of type i in multimedia server at level l and b_{ij} is the required space for storing one copy of the jth popular file of type i.

The M_i^l's are among the decision variables that we want to determine through

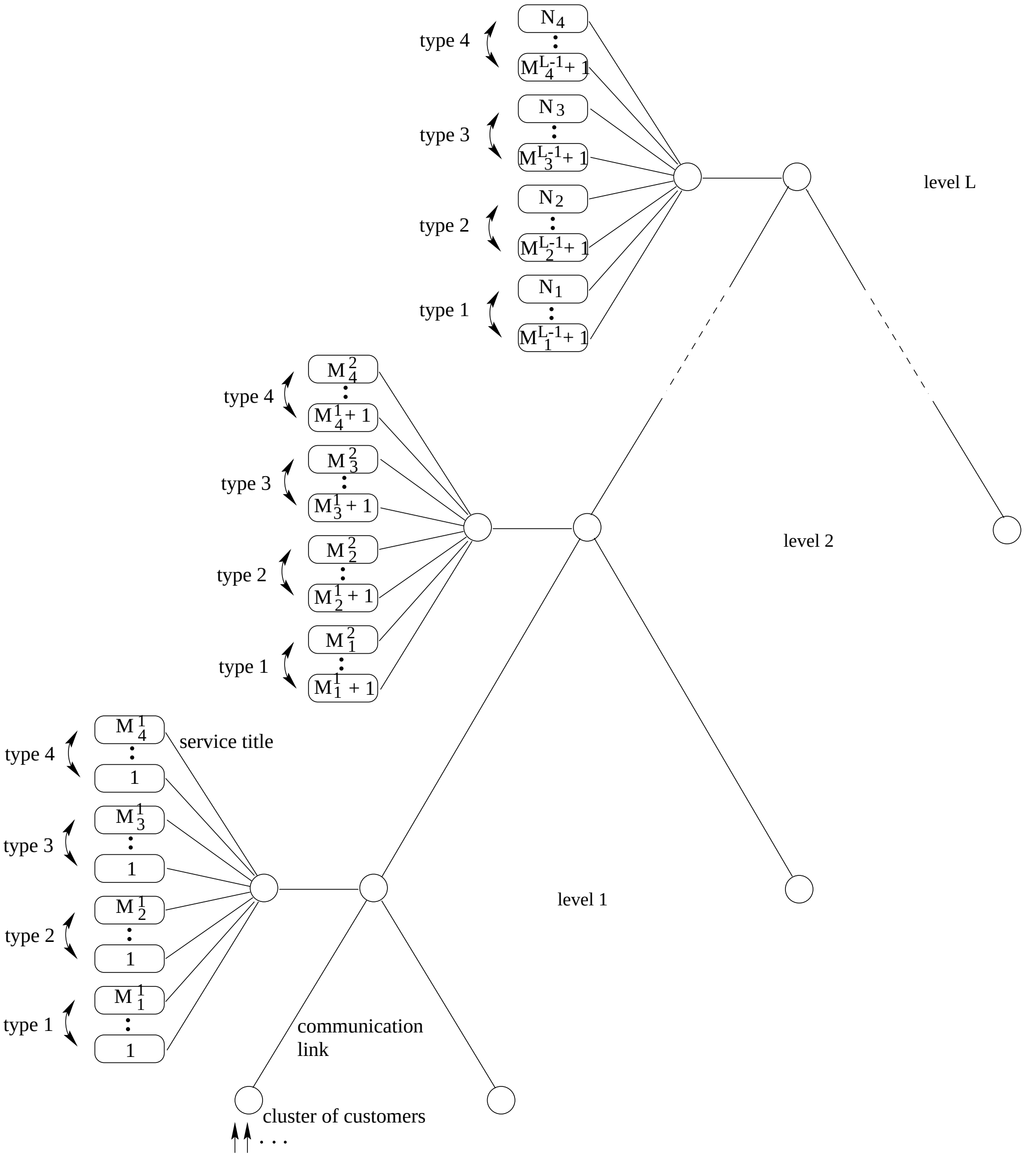


Figure 2: File distribution for predetermined server selection

optimization. The remaining problem is to find optimal values of link capacities, each server's output bandwidth, and the number of copies each file. When the characteristics of the customer demand changes, one can download the files that become less popular into a multimedia server at a higher level or vice-versa. We assume that the cost of transferring can be ignored since this operation can be done off-line.

2.2 Optimization model

The objective of the design is to minimize the overall cost, which is the sum of the costs of multimedia servers and communication link costs. For reasons that we will discuss in Section 3, the minimization of the total cost expression below will be divided into two steps, as denoted by the inner and outer minimizations:

$$\min_{\{M_i^l\}}\{\min_{\substack{\{k_l\}\\ \{s_l\}\\ \{t_{ij}^l\}}}\{\sum_{l=0}^{L-1} w_l\psi_l(k_l) + \sum_{l=1}^{L} w_l\phi_l(s_l, \sum_{i\in\mathcal{I}}\sum_{\substack{j=\\ M_i^{l-1}+1}}^{M_i^l} t_{ij}^l b_{ij})\}, \qquad (1)$$

with constraints

$$\begin{array}{ccccccc} & & & B_{ij}(\underline{k}, \underline{s}, \boldsymbol{t}) & \leq & \hat{B}_{ij} & \forall i\,, j \\ & & & k_l & \geq & 0 & \forall l \\ & & & s_l & \geq & 0 & \forall l \\ & & & t_{ij}^l & \geq & 0 & \forall i\,, j\,, l \\ M_i^1 & \leq & M_i^2 & \leq \quad \ldots & \leq & M_i^L & \forall i\,. \end{array}$$

The communication link cost, $\psi_l(k_l)$, is the cost of the link between level l and $l+1$ with capacity k_l. $\phi_l(s_l, m_l)$ represents the cost of a multimedia server at level l with output bandwidth s_l and storage capacity m_l. $B_{ij}(\underline{k}, \underline{s}, \boldsymbol{t})$ is the end-to-end blocking probability for the jth popular file of type i, which is constrained to be less than $\hat{B}_{ij}$. Depending on the values of the M_i^l's, some links and servers may be eliminated from the network(for example, when $M_i^{l-1} = M_i^l$ for all i, level l servers don't exist in final solution).

The total cost expression in (1) assumes symmetric traffic requirements, i.e., each cluster of users has the same demand characteristics for each title. Consequently, the results are obtained for a balanced tree, or in degenerate cases, a forest of balanced trees, all the links at any particular level having the same capacity. With minor modifications, this symmetric assumption can be loosened at the expense of increasing the decision variables.

Requests for a service title arrive according to a Poisson process. The route for the service title consists of communication links from the customer cluster to the server which provides the service title, the I/O of the server and a copy of the required title. Each element in the route has finite units of bandwidth, which is shared among different classes of traffics. A request takes predetermined units of bandwidth at each element on the route simultaneously, provided it is not blocked by any of the elements in the route. The length of time that the title is accessed, i.e., the service duration, has a general distribution with finite mean.

Since it is hard to compute the end-to-end blocking probability even for a moderate sized network, we consider two approximations: the reduced load approximation and the summation approximation. Details of these approximations can be found in [8], under the assumption that all service requests require the same bandwidth. Among several methods for calculating single element loss probability [9, 10, 11, 12, 13], we decided to use the Uniform Asymptotic Approximation(UAA) [13] for calculating the single element blocking probability for each type of traffic. In addition to its capability of handling real values of capacity, which is useful for our optimization, the UAA is quite accurate for calculations of single link blocking probabilities for services with heterogeneous bandwidths. Another attractive feature is that the complexity of computation does not increase with the link capacity.

From our numerical experience, even though the difference between the reduced load and summation approximations increases with higher loads and bandwidth requirements, the summation approximation provides a tight upper bound on the reduced load approximation at the loading region of 1% end-to-end blocking probability [14]. Generally, the repeated substitution method is used to get a set of converged solutions for the reduced load approximation [15]. The number of iterations in the repeated substitution depends on not only the stopping criterion of the iteration but also the values of traffic loads and system capacities. However, the summation approximation gives the solution in one iteration. Consequently, the summation approximation has an advantage in computation time, at the cost of somewhat reduced accuracy.

3 Optimization Method

In this section, we consider an optimization technique for solving the problem introduced in Equation (1). To avoid dealing with a nonlinear integer programming problem for the inner optimization, we assume a continuous cost function and allow all the decision variables except the M_i^l's to be real. Since it is difficult to decide the value of the cost function at non-integer values of M_i^l's, we keep the decision variables of the outer minimization as integers.

From our computational experience using the Matlab optimization package, which has an implementation of sequential quadratic programming, the CPU running time for one inner minimization routine(Inner_Min) under the reduced-load approximation of the end-to-end blocking constraint is about five times as large as the time under the summation approximation. However, using the solution from the summation approximation as a starting point, the iterations under the reduced load approximation terminate in about half the time that it takes from an arbitrarily chosen original initial point. Moreover, from extensive numerical investigation, we believe that the choice of the approximation schemes does not affect the optimal values of M_i^l's. Thus, the benefit in optimization time using the summation approximation becomes more significant for the outer minimization stage.

As we saw in [14], once the cost decreases by moving the most popular title from a higher level server down to lower level servers, the cost keeps decreasing until reaching the optimal cutoff index. After passing the optimal cutoff index, the cost increases as we further increase the value of M_i^l. This indicates that the optimization problem has a well-behaved cost structure with respect to the M_i^l's. Thus we suggest the following two alternative procedures for our optimization problem. In both procedures, the summation approximation is used for all but the last step.

Procedure 1(optimize each service independently)

Using a search algorithm, as described in the following pseudocode, find the optimal M_i^l for a specific i with the largest bandwidth service, while fixing the values of other M_j^l's, $j \neq i$, to 0. After finding the set of M_i^l values, perform the same optimization routine for the next largest bandwidth service, as in the previous step, now using the optimal M_i^l's obtained from the previous step and the remaining zero M_i^l's. We repeat the step until all the values of optimal M_i^l's are obtained.

```
$J_{old} = \infty$            ($J$ is the total cost)
$M_i^L = N_i$ , $M_i^l = 0$ for all $i$ and $1 \leq l < L$
for $i = |\mathcal{I}|$ down to 1      (start with largest bandwidth service)
    compute $J_{new}$, cost from inner minimization subroutine Inner_Min
    do the following routine while $J_{new} < J_{old}$ (search for optimal cutoff indices
        $J_{old} = J_{new}$                              for service i)
        $\hat{M}_i^l = M_i^l$ for all $l$
        for $l = 1$ to $L-1$     (calculate cost for new cutoff indices)
            for $j = l$ to $L-1$
                $M_i^j = \max\{M_i^l + Z_i^l, M_i^j\}$       ($Z_i^l$ is a step size.)
            compute $J^l$, the cost from Inner_Min
            $M_i^l = \hat{M}_i^l$ for all $l$
        end
        for $l = 1$ to $L-1$     (pick new point in "optimum" direction)
            if $J^l < J_{old}$
              $M_i^l = M_i^l + Z_i^l$
            else
              $M_i^l = \max\{0, M_i^l - Z_i^l\}$
              $Z_i^l = \max\{1, \lfloor \frac{Z_i^l}{2} \rfloor\}$
        end
        for $l = 1$ to $L-1$     (make cutoff indices consistent)
            $M_i^l = \max\{M_i^l, M_i^{l-1}\}$
        compute $J_{new}$ for new $M_i^l$'s using Inner_Min
    end(do)
end(for)
Perform final inner minimization using reduced-load approximation
```

Procedure 2(descent direction-based search)

Setting the initial starting points, $M_i^l = 0$ for all i and $1 \leq l < L-1$, we evaluate the cost from the inner minimization for each direction with respect to every i and l. Pick a new point that reduces the cost as the next "optimum" point, while adjusting *step sizes. Perform the same routine until no cost reduction is found for any i and l.*

```
$J_{old} = \infty$
$M_i^L = N_i$ , $M_i^l = 0$ for all $i$ and $1 \leq l < L$
compute $J_{new}$, the cost from Inner_Min
do the following routine while $J_{new} < J_{old}$
   $J_{old} = J_{new}$
   $\hat{M}_i^l = M_i^l$ for all $i$ and $l$
   for $i = 1$ to $|\mathcal{I}|$     (compute costs in all directions)
      for $l = 1$ to $L - 1$
         for $j = l$ to $L - 1$
            $M_i^j = \max\{M_i^l + Z_i^l, M_i^j\}$
         compute $J_i^l$, the cost from Inner_Min
         $M_i^l = \hat{M}_i^l$ for all $l$
      end
   end
   for $i = 1$ to $|\mathcal{I}|$     (pick new point in descent direction)
      for $l = 1$ to $L - 1$
         if $J^l < J_{old}$
           $M_i^l = M_i^l + Z_i^l$
         else
           $M_i^l = \max\{0, M_i^l - Z_i^l\}$
           $Z_i^l = \max\{1, \lfloor \frac{Z_i^l}{2} \rfloor\}$
      end
      for $l = 1$ to $L - 1$     (make cutoff indices consistent)
         $M_i^l = \max\{M_i^l, M_i^{l-1}\}$
```

```
    end
    compute $J_{new}$ for new $M_i^l$'s using Inner_Min
end(do)
Perform final inner minimization using reduced-load approximation
```

Z_i^l is initially an arbitrarily chosen positive integer value, which is adjusted during iterations as indicated above, and corresponds to a step size. An exhaustive search based outer minimization that finds a set of the optimal cutoff indices has complexity of $\mathcal{O}(\prod_{i=1}^{|\mathcal{I}|}(N_i)^{L-1})$. Using Procedure 1, the complexity is at most $\mathcal{O}(\sum_{i=1}^{|\mathcal{I}|}(N_i)(L-1))$. For Procedure 2, the inner "for" loop takes $|\mathcal{I}|(L-1)$ iterations, while the outer "do" loop iterates $\mathcal{O}(\max N_i)$ times in most cases from our computational experience. In practical situations, L tends to be a small value, such as two or three. The number of titles of each type of service N_i could be a much larger value, a few hundred or thousands. Also as the types of services increase, $|\mathcal{I}|$ gets larger. Thus the gain with the proposed procedure is considerable, and allows practical sized problems to be solved.

4 Numerical Investigations

In this paper, we consider a two-level system with one root server and several local servers. The constraint on the individual end-to-end blocking probability is assumed to be less than or equal to 0.01 for all i and j. For each type of service i, the user access probability for title j, p_j, is assumed to be the Zipf distribution with parameter $\theta = 0.271$ [16]. In both cases, we set $N = [10 \quad 10 \quad 10 \quad 10]\,, w_1 = 8\,, w_0 = 32$. Four different types of services are assumed to be available, and their bandwidth requirements, $\boldsymbol{d} = [1 \quad 2 \quad 12 \quad 24]$. This could correspond, for example, to services ranging from voice at 64kbps to compressed video at 1.5Mbps.

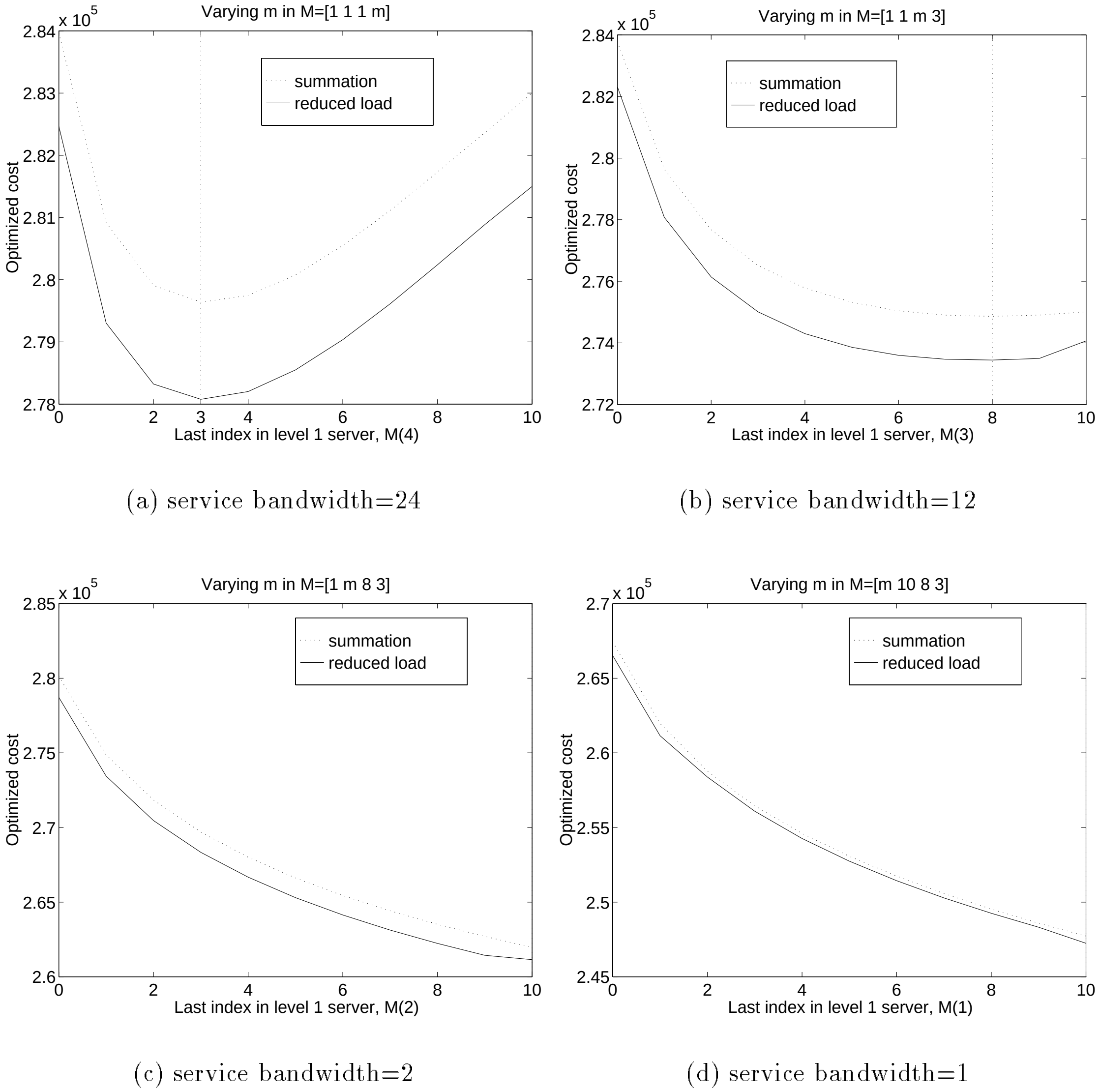
4.1 Case 1

This case deals with the situation that the total offered load for type i is the same for all types of services, i.e., $\sum_{j=1}^{N_i} \rho_{ij} d_i = 120$ for all i, where ρ_{ij} is the traffic intensity(the product of arrival rate and service duration) for the jth popular file of type i from one cluster of customers. Additionally, the mean service duration for all types of service is assumed to be the same. The cost function is assumed to be linear with coefficients shown in Table 1. In order to study the tradeoff between communication cost and storage cost, we chose the coefficient of the link cost between a level 1 server and a level 2 server, c, to be a variable parameter.

	Case1	Case2
one copy of type 1 file	4	1
one copy of type 2 file	8	6
one copy of type 3 file	48	84
one copy of type 4 file	96	600
one unit of server's output bandwidth	1	3
one unit of link bandwidth between level 0 and 1	5	10
one unit of link bandwidth between level 1 and 2	c	c

Table 1: Cost coefficients for Case 1 and Case 2

When $c = 5$, the effect of different file allocation schemes on total cost is shown in Fig. 3. The figure also shows the progress of Procedure 1. Every point in a curve is the optimized cost from the inner minimization of our optimization problem (1). We first vary the value of M_1, while the values of other three M_i's are fixed at 1. The best value from this step is $M_1 = 3$, as shown in Fig. 3(a). After finding the best value of M_1, we vary the next parameter, M_2. The same routine is performed until



(a) service bandwidth=24

(b) service bandwidth=12

(c) service bandwidth=2

(d) service bandwidth=1

Figure 3: Optimized costs with $c = 5$

the best values of all M_i's are obtained. Thus the solution for the outer minimization of (1) is $\boldsymbol{M} = [10 \quad 10 \quad 8 \quad 3]$. Up to this point all the optimization routines are performed using the summation approximation. Finally, to get the optimal values of decision variables in inner minimization, we perform the final step with the reduced load approximation. In this figure, in order to provide better understanding, we draw the curves for both approximation methods and all the possible value of M_i along intermediate steps of the procedure. Note that the reduced load and summation approximations find identical optimal values for the cutoff indices.

	M_i (Procedure 1/Procedure 2)				
Coefficient, c	Type 1	Type 2	Type 3	Type 4	Optimized Cost
0.01	4/4	0/1	0/0	0/0	204146/204012
0.1	5/5	1/1	0/0	0/0	205515/205515
0.3	7/7	2/2	0/0	0/0	208553/208553
0.5	10/10	6/4	0/0	0/0	211423/211303
1	10/10	8/8	0/0	0/0	219682/219683
2.5	10/10	10/10	0/2	0/0	234172/232776
3	10/10	10/10	3/3	1/1	236601/236601
4	10/10	10/10	5/5	2/2	243074/243074
5	10/10	10/10	8/8	3/4	247726/247619
6	10/10	10/10	10/10	5/9	250382/249193
7	10/10	10/10	10/10	10/10	250186/250186

Table 2: Optimal values of the cutoff indices for Case 1

When we vary the value of c, the impact on the optimal values of M_i's is shown in Table 2. Procedure 1 is used for the value before a slash mark. The results from Procedure 2 are shown after the slash mark. From the table, we can see that the files with smaller bandwidth requirement are distributed among multiple local servers, while one central server tends to keep the files with larger bandwidth requirements. The two procedures find the same cutoff indices in most cases. However, Procedure 2 gives slightly better results(always under 1% better) at the expense of higher optimization time.

4.2 Case 2

In this case, we assume that the sum of traffic intensities to the files with the same type of service, $\sum_{j=1}^{N_i} \rho_{ij} = 20$, is the same for all the types of services i's. The service time for the four types of service are assumed to be 3 min, 9 min, 21 min, and 75 minutes, respectively. The cost coefficients for unit storage for different service types are proportional to the product of their holding times and bandwidth.

The cost function is assumed to be linear with the coefficients listed in Table 1. The result from Procedure 1 is shown in Table 3. Again, we see that the central server keeps the files with the larger bandwidth requirements.

	M_i			
Coefficient, c	Type 1	Type 2	Type 3	Type 4
0.1	10	7	0	0
1	10	10	0	0
2	10	10	3	0
4	10	10	7	0
8	10	10	10	0
12	10	10	10	3
15	10	10	10	5
18	10	10	10	10

Table 3: Optimal values of the cutoff indices for Case 2

4.3 Exhaustive search

In order to examine the accuracy of the proposed procedure, we performed exhaustive searches for $N = [5 \quad 5 \quad 5 \quad 5]$. The total offered load for each type and the cost coefficients are identical to those of Case 1. The result is shown in Table 4.

	M_i			
Coefficient, c	Type 1	Type 2	Type 3	Type 4
1	5	5	1	0
2	5	5	2	1
3	5	5	4	2
4	5	5	4	4
5	5	5	5	5

Table 4: Optimal values of the cutoff indices from exhaustive search

Both procedures found the same solution as the exhaustive search, but with significantly less computation time. As an example, Procedure 2 consumed 12,731 seconds of CPU time in a Sun UltraSPARC2 machine to get the solution for $c = 1$, while the CPU time for exhaustive search was about 36 times as large.

5 Conclusion

In this paper, we consider an optimization problem in networks with storage servers for providing multimedia service under a constraint on the individual end-to-end blocking probabilities. We presented two procedures to find the best file distribution schemes as well as optimize the size of the elements in a network with multimedia servers. A well-managed multimedia file distribution scheme reduces storage cost and communication cost, making the service more affordable. This optimization can be used in content management as the demand characteristics of files vary with time, as well as in the initial design phase.

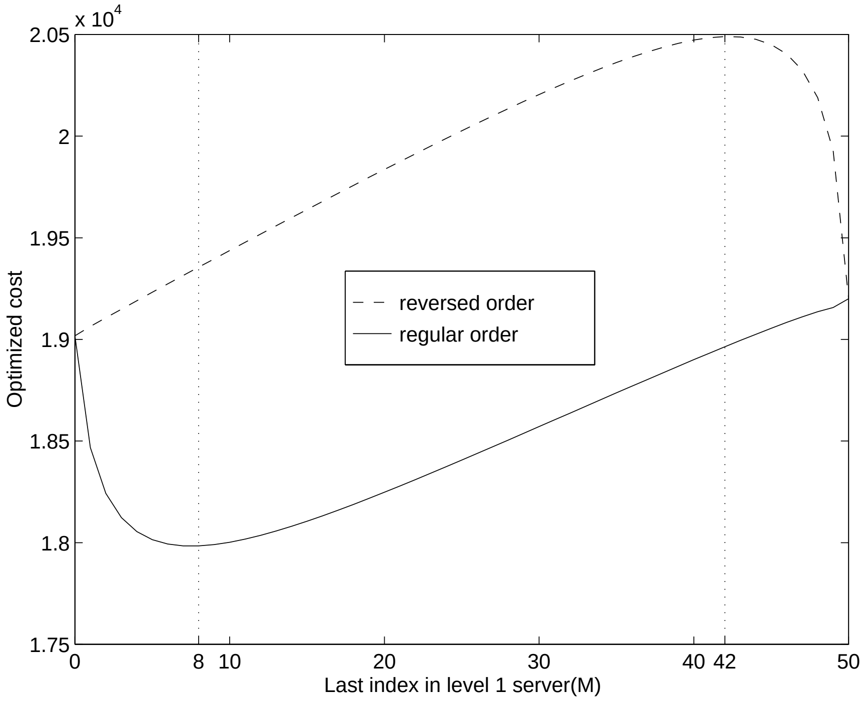


Figure 4: Optimized cost for reversed order

Appendix A Effect of File Allocation Order

In this section, we illustrate the effect of different ordering to the optimization cost with the end-to-end blocking constraint. We were unable to show that our assumption of placing the most popular titles to low level servers is valid by analytical means. Indeed, this is an interesting open problem. Through a numerical investigation, we observe that our assumption of placing the most popular titles to low level servers is valid for the cases tested. Also, we observed that the minimum cost exists for only one value of the cutoff index for the single rate case.

We consider two extreme cases of the file allocation according to the popularity. One of the cases is that the titles are indexed in order of their popularity and placed

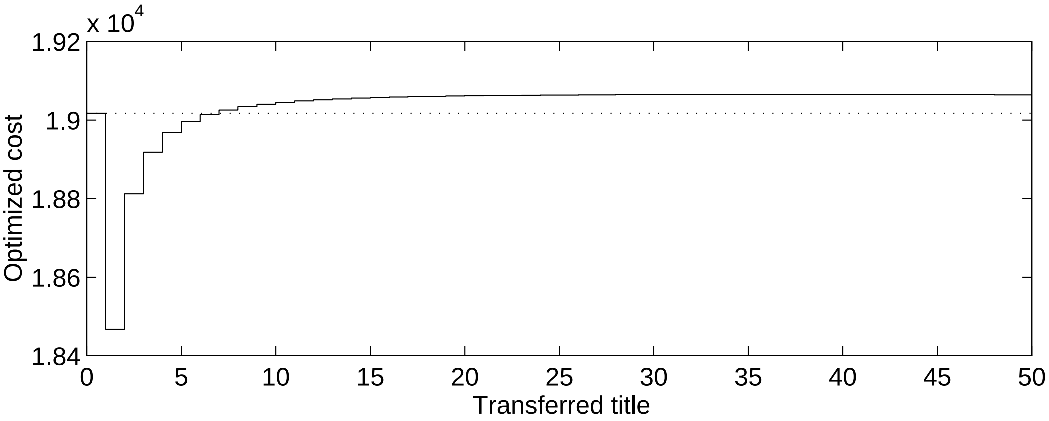


Figure 5: Optimized cost for transferring only one title to local servers

the most popular titles closest to the users(regular order). The other one is that the titles are indexed in inverse order of their popularity and placed the least popular titles closest to the users(reversed order), which we examined as the other extreme from our approach. Figure 4 shows the optimized costs for two different orderings. Two curves are obtained for $w_1 = 6, w_0 = 24, N = 50$ and $\sum_{j=1}^{N} \rho_j = 50$ erlangs, with a single service type. The cost function is assumed to be linear with coefficients 1 for one unit of server's output bandwidth, 4 for one copy of a service title, 2 for the link cost between a level 1 server and a level 2 server and 5 for a channel on a link between the level 0 node and the level 1 server. The solid curve has the minimum at $M = 8$, i.e., when the files ranging from the most popular title to the eighth popular title are placed in local servers. In the case of reversed ordering, as we move more titles from one root server down to multiple local servers, starting from the least popular title, the optimized cost increases until the eighth popular title(indexed 42 for the reversed ordering) is transferred to local servers.

The next case shows the cost tradeoff does not depend much on the total volume of all the service titles in each level server, but highly depends on the volume of each service title. In Figure 5, we keep only one title in local servers, while all other titles are placed in one root server. When the local servers have the most popular title only, the biggest cost reduction is achieved as compared to the case that root server contains all the titles. The cost reduction occurs up to the sixth-most popular title; any title that is less popular than the sixth title causes the cost to increase. This absolute cost reduction leads us to keep all the titles up to the sixth popular one in local servers. By moving the seventh and eighth titles, further cost reduction is possible since the reduction in communication link cost associated with moving those two titles exceeds the increased server cost and multiplexing gain of keeping them at the root server, as compared to the case of moving them one at a time. Hence the optimal cost is obtained when $M = 8$.

References

[1] Jean-Paul Nussbaumer, Baiju V. Patel, Frank Schaffa, and James P.G. Sterbenz, “Networking requirements for interactive video on demand,” *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 5, pp. 779–787, June 1995.

[2] Chatschik C. Bisdikian and Baiju V. Patel, “Issues on movie allocation in distributed video-on-demand systems,” in *Proceedings of the 1995 IEEE International Conference on Communications*, Seattle, Washington, USA, June 1995, IEEE, pp. 250–255.

[3] Scott A. Barnett and Gary J. Anido, “A cost comparison of distributed and centralized approaches to video-on-demand,” *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 6, pp. 1173–1183, Aug. 1996.

[4] Tsong-Ho Wu, Ibrahim Korpeoglu, and Bo-Chao Cheng, "Distributed interactive video system design and analysis," *IEEE Communications Magazine*, pp. 100–108, Mar. 1997.

[5] Jeong-dong Ryoo and Shivendra S. Panwar, "Optimization of video-on-demand networks to meet end-to-end blocking objectives," *Proceedings of the Sixth International Conference on Telecommunication Systems - Modeling and Analysis*, pp. 135–145, Mar. 1998.

[6] Andrew Paff, "Hybrid fiber/coax in the public telecommunications infrastructure," *IEEE Communications Magazine*, pp. 40–45, Apr. 1995.

[7] Keith W. Ross, *Multiservice Loss Models for Broadband Telecommunication Networks*, Springer, 1995.

[8] W. Whitt, "Blocking when service is required from several facilities simultaneously," *AT&T Technical Journal*, vol. 64, pp. 1807–1856, 1985.

[9] Joseph S. Kaufman, "Blocking in a shared resource environment," *IEEE Transactions on Communications*, vol. COM-29, no. 10, pp. 1474–1481, 1981.

[10] J. W. Roberts, "A service system with heterogeneous user requirements - application to multi-services telecommunications systems," in *Performance of Data Communication Systems and their Applications*, pp. 423–431. North Holland, 1981.

[11] Shun-Ping Chung and Keith W. Ross, "Reduced load approximations for multirate loss networks," *IEEE Transactions on Communications*, vol. 41, no. 8, pp. 1222–1231, Aug. 1993.

[12] Jean-François P. Labourdette and George W. Hart, "Blocking probabilities in multitraffic loss systems: Insensitivity, asymptotic behavior, and approximations," *IEEE Transactions on Communications*, vol. 40, pp. 1355–1366, Aug. 1992.

[13] Debasis Mitra and John A. Morrison, "Erlang capacity and uniform approximations for shared unbuffered resources," *IEEE Transactions on Networking*, vol. 2, pp. 558–570, 1994.

[14] Jeong-dong Ryoo and Shivendra S. Panwar, "File distribution schemes in networks with multimedia server: Multirate case," *CATT - Technical Report, Polytechnic University*, 1998.

[15] A. Girard and Y. Ouimet, "End-to-End Blocking for Circuit-Switched Networks: Polynomial Algorithms for Some Special Cases," *IEEE Transactions on Communications*, vol. COM-31, pp. 1269–1273, Dec. 1983.

[16] A.Dan, D.Sitaram, and P.Shahabuddin, "Scheduling policies for an on-demand video server with batching," in *Proceedings of ACM Multimedia 94 Conference*, Oct. 1994, pp. 15–23.