Optimal File Placement in VOD System Using Genetic Algorithm

Kit-Sang Tang, *Member, IEEE*, King-Tim Ko, *Member, IEEE*, Sammy Chan, *Member, IEEE*, and Eric W. M. Wong, *Senior Member, IEEE*

***Abstract*—Advances in computing and networking are generating a significant demand for video-on-demand (VOD) applications. In this paper, the condition for minimum blocking probability of a VOD system is derived. The optimal load-sharing requirement in such a system is found so that the minimum blocking probability is achieved. A combination of genetic algorithm and modified bin-packing algorithm is then developed for the multimedia file placement exercise. It is demonstrated that a specified blocking probability can be achieved while the capacity usage is minimized simultaneously.**

***Index Terms*—Blocking probability, file placement, genetic algorithm, video on demand.**

I. Introduction

WITH THE advanced technologies now developed in the areas of high-speed networks and multimedia, video-on-demand service (VOD) is considered as the emerging trend in home entertainment, as well as in education, banking, home shopping, and interactive games [14]. Such multimedia experience is expected to blossom when the broadband access networks are available.

In order to provide a large variety of program contents, thousands of multimedia files, encoded in MPEG format, are stored in a video server which consists of multiple disks (or disk arrays) [21]. These multimedia files can be obtained on demand by a large number of geographically distributed users. Once there is a user's request, the particular file, in the form of a data stream, is directed and transmitted from the disk to the set-top box installed in the user's premises using a broadband connection.

The task of allocating multimedia files to the video server is currently handled by an experienced operator which implies a suboptimal and personnel-dependent solution. The complexity of such a problem increases exponentially with the number of files and disks in the video server. The level of difficulty thereby escalates, and the current solution will give rise to higher request blocking ratio and dissatisfaction to customers.

The design of an optimal file placement is considered as being a multiobjective, highly constrained, and noncontinuous problem. Although different heuristic approaches have been proposed [11], [20], they are focused on a system with multiple servers in a VOD network, assuming a single disk per server.

In [15], a popularity-based approach is developed for multiple disks based on the information of popularity. The algorithm is designed for a homogeneous system with identical disks, and assuming that all multimedia files have the same duration. A similar analysis of the storage cost is given in [1] for distributing movies across disks. However, different types of disks may coexist in the video server and the durations of the multimedia files usually vary from a few minutes for short video clips to more than two hours for movies.

In this paper, an optimal condition of load sharing for minimum blocking probability is constructed numerically in such a heterogeneous system while the solution is obtained effectively by an iterative approach. A genetic algorithm, together with a heuristic bin-packing algorithm, are designed such that the optimal number of copies of the multimedia files and their corresponding disk locations are determined.

The organization of this paper is as follows: In Section II, the condition for minimum blocking probability of the heterogeneous VOD system is derived. The optimal file placement strategy using the genetic algorithm (GA) and modified bin-packing algorithm is explained in Section III. The effectiveness of the proposed strategy is demonstrated in Section IV. Finally, conclusions are made in Section V.

II. Minimum Blocking Probability in VOD System

Fig. 1 depicts a centralized multimedia file storage system [1] with multiple storage disks for a VOD system. Due to the drop in storage cost and the expanding I/O bandwidth using the disk striping technique [16], such a system becomes economically feasible.

An entire multimedia file, in MPEG compressed format, is stored on a single disk. Interleaving is not recommended because of the adverse effect on its reliability with a single disk failure [8]. The number of concurrent accesses to a particular file is limited by the disk throughput. For an MPEG-1 video, 1–2 Mbps I/O bandwidth is required for its transmission (and 2–20 Mbps transmission for MPEG-2).

The demand rate for a multimedia file creates its popularity profile, defined as the probability of a file being requested by a user. Zipf's law is usually employed [19] to model the popularity profiles within a site. Another commonly used assumption is the uniform popularity, which means that all files are equally likely to be requested.

Manuscript received March 31, 2000; revised January 29, 2001. Abstract published on the Internet July 31, 2001. This work was supported by the City University of Hong Kong under Grant 7001125.

The authors are with the Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong (e-mail: kstang@ee.cityu.edu.hk; ewong@ee.cityu.edu.hk).

Publisher Item Identifier S 0278-0046(01)08774-3.

0278–0046/01$10.00 © 2001 IEEE

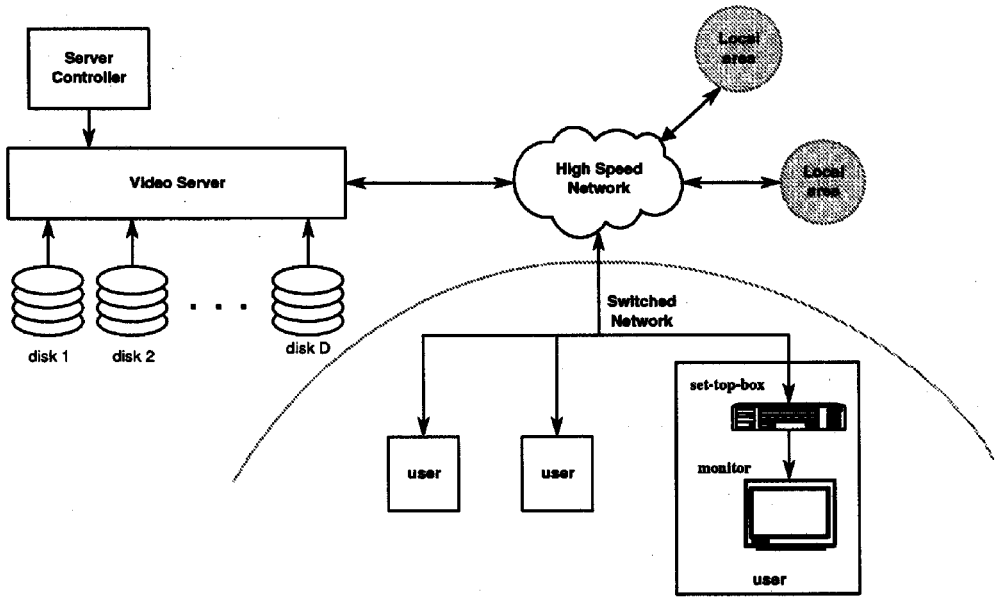

Fig. 1. Basic configuration of the VOD system.

Assuming that the requests can be considered as a Poisson arrival process with average rate λ and mean service time $1/\mu$, the traffic intensity of the VOD system is computed as

$$A = \frac{\lambda}{\mu}. \tag{1}$$

If the request of a multimedia file is randomly assigned to one of the disks storing that file and there is no retry mechanism, each disk can be modeled by an $M/G/n/n$ queueing system where M stands for Markovian or memoryless interarrival time, G refers to a general service time distribution, and n is the number of I/O streams available in the disk. Assuming that q_j portion of the total traffic is allocated to disk j, the blocking probability of disk j having L_j I/O streams can be computed using the following Erlang B formula [2]:

$$B_{q_j} = \frac{(Aq_j)^{L_j}/L_j!}{\sum_{i=0}^{L_j}(Aq_j)^i/i!} \tag{2}$$

where $q_j \geq 0$.

The blocking probability of the overall system is then derived [9] by

$$B = \sum_{j=1}^{D} q_j B_{q_j} = \sum_{j=1}^{D} q_j \frac{(Aq_j)^{L_j}/L_j!}{\sum_{i=0}^{L_j}(Aq_j)^i/i!} \tag{3}$$

where $\sum_{j=1}^{D} q_j = 1$ and D is the total number of disks.

Using the method of Lagrange multipliers, define

$$G = \sum_{j=1}^{D} q_j \frac{(Aq_j)^{L_j}/L_j!}{\sum_{i=0}^{L_j}(Aq_j)^i/i!} - K\left(\sum_{j=1}^{D} q_j - 1\right). \tag{4}$$

Differentiating G with respect to q_j, and setting the result to zero for minimization, we have

$$\frac{\partial G}{\partial q_j} = \frac{\partial}{\partial q_j} q_j \frac{(Aq_j)^{L_j}/L_j!}{\sum_{i=0}^{L_j}(Aq_j)^i/i!} - K = 0. \tag{5}$$

Hence, the condition for minimization is to have $q^{\text{opt}} = \{q_1^{\text{opt}}, q_2^{\text{opt}}, \ldots, q_D^{\text{opt}}\}$ such that

$$\frac{\frac{(Aq_j^{\text{opt}})^{L_j}}{L_j!}\sum_{i=0}^{L_j}\frac{\left(Aq_j^{\text{opt}}\right)^i (L_j+1-i)}{i!}}{\left(\sum_{i=0}^{L_j}\frac{\left(Aq_j^{\text{opt}}\right)^i}{i!}\right)^2} = K \quad \text{or}$$

$$B_{q_j^{\text{opt}}} \cdot \left[L_j + 1 - Aq_j^{\text{opt}} \cdot \left(1 - B_{q_j^{\text{opt}}}\right)\right] = K \tag{6}$$

where $B_{q_j^{\text{opt}}} = ((Aq_j^{\text{opt}})^{L_j}/L_j!)/(\sum_{i=0}^{L_j}(Aq_j^{\text{opt}})^i/i!)$; $j = 1, 2, \ldots, D$ and K is a constant.

The vector q^{opt} can be found by an iterative approach. Defining

$$\varphi(q_j, L_j) = B_{q_j} \cdot \left[L_j + 1 - Aq_j \cdot (1 - B_{q_j})\right]$$

the following procedures are designed.

1. Set the initial value $q_1 = 1/D$;
2. Let $K = \varphi(q_1, L_1)$ and find q_j such that $\varphi(q_j, L_j) = K$ for $j = 2, 3, \ldots D$;
3. Calculate $q_{\text{sum}} = \sum_{j=1}^{D} q_j$;
4. if $(|q_{\text{sum}} - 1| < \epsilon)$
   {
   $q_j = q_j - (q_{\text{sum}} - 1)/D$ where $j = 1, 2, \ldots, D$;
   Stop;
   }
   else
   {
   $q_1 = q_1 - (q_{\text{sum}} - 1)/D$;
   Goto step 2;
   }

In order to compute the value of q_j in step 2 such that $\varphi(q_j, L_j) = K$, a gradient search with the following procedures can be used.

1. Set the initial value $q_j = 1/D$;
2. while $(|(\varphi(q_j, L_j) - K)/K| \geq \delta)$
 {
 $q_j = q_j + (\alpha[K - \varphi(q_j, L_j)]/\varphi'(q_j, L_j))$ where α is a small positive constant and $\varphi'(q_j, L_j) = d\varphi(q, L_j)/dq|_{q=q_j}$
 }

The small positive constants ϵ and δ are used to govern the accuracy of the solution, while α controls the rate of convergence of the gradient search.[1]

Example: It is assumed that there are 20 storage disks in the video server of a VOD system. Disk #1–#15 are type-A disks which can support 50 I/O streams simultaneously while disk #16– #20 are type-B disks which can only support 30 I/O streams.

Let $\lambda = 10$ requests per minute and $(1/\mu) = 74.43$ minutes, the optimal values of q_j for type A and type B satisfying (6) are 0.056 053 1 and 0.031 840 6, respectively. The minimum system blocking probability is 0.029 890 5. It takes about 90 ms in a Pentium 133 machine with α, δ and ϵ set as 0.01, 10^{-3} and 10^{-7}, respectively, to calculate the above results using the proposed procedures.

III. Optimal MPEG Placement Scheme

In order to set up the video server in a VOD system, it is required to identify the number of copies of each multimedia file and their corresponding disk locations. There are two major objectives in order to optimize the system performance and provide a specified service quality:

1) to minimize the total capacity usage of the server;
2) to minimize the blocking probability of the VOD system.

The problem is also subjected to the constraint on the disk capacity. In other words, the capacity of each disk must not be exceeded.

In this paper, the GA [17] together with a modified bin-packing algorithm, known as Highest Load First (HLF), are proposed. The gene of the chromosome in the GA is to reflect the number of copies of each multimedia file, while the HLF algorithm will search for the optimal file placement scheme of that chromosome such that a minimum blocking probability is found.

A. HLF for Optimal Blocking Probability

Assuming that the request of a multimedia file is randomly allocated to one of the disks storing that particular file, the probability of accessing each copy of multimedia file i can be computed as p_i/n_i where n_i and p_i are the number of copies and the popularity of file i, respectively.

If no retry is allowed in the system, the probability in accessing a disk j can be computed as

$$P_{d_j} = \sum_{i \in d_j} \frac{p_i}{n_i} \tag{7}$$

where $i \in d_j$ implies that file i is stored in disk j.

[1]Since $q_m = q_n$ if $L_m = L_n$ and $\varphi(q_m, L_m) = \varphi(q_n, L_n)$, the computation time is reduced by only finding the value of q_j in step 2 when the number of I/O streams of disk j is different from the previous disks.

The mean service time $(1/\mu_j)$ and the arrival rate (λ_j) of disk j can then be derived by

$$\lambda_j = \lambda P_{d_j}$$
$$\frac{1}{\mu_j} = \frac{1}{P_{d_j}} \sum_{i \in d_j} \frac{p_i}{n_i \mu_i}$$

where $1/\mu_i$ is the average holding time of multimedia file i. $1/\mu_i$ is generally proportional to the duration of the video (t_i), or $(1/\mu_i) = f_i t_i$ with f_i determined by the customer behavior, like fast forward, pause, and rewind actions. For simplicity, it is assumed that $f_i = 1$ or $(1/\mu_i) = t_i, \forall i$.

Hence, given a set of the number of copies of the multimedia files, $\{n_1, n_2, \ldots, n_M\}$ where M is the number of multimedia files, the minimum blocking probability is achieved when

$$\lambda \sum_{i \in d_j} \frac{p_i}{n_i \mu_i} = \frac{\lambda_j}{\mu_j} = A_j = A q_j^{\text{opt}} \quad \text{or}$$
$$\sum_{i \in d_j} \frac{p_i}{n_i \mu_i} = \frac{q_j^{\text{opt}}}{\mu} \quad \text{for} \quad j = 1, 2, \ldots D \tag{8}$$

where $(1/\mu) = \sum_i (p_i/\mu_i) = \sum_i p_i t_i$ and q_j^{opt} is the optimal load sharing of disk j obtained in Section II.

Such a problem can be reduced into a bin-packing problem as putting the elements of size $p_i/(n_i \mu_i)$ into the bins with space q_j^{opt}/μ for $j = 1, 2, \ldots, D$. Unlike the conventional bin-packing problem [3], [13] in which the number of bins is to be minimized, our objective is to allocate the elements such that all the bins are "just full" meaning that (8) is satisfied.

Similarly to the conventional bin-packing problem, the sequence of the elements to be put into the bin may affect the effectiveness of the algorithm. In general, there are four different reordering approaches.

1) The descending approach arranges the elements from largest size to the smallest.
2) The ascending approach arranges the elements from smallest size to the largest.
3) The random approach arranges the elements randomly.
4) The regular approach maximizes the distances between similarly sized elements.

Due to the discretization effect of the element's size, it is expected that the descending approach will be more effective as the element with larger size is allocated first. Such advantage is also demonstrated in our simulation results given in Section IV.

The file placement algorithm is, hence, designed as follows.

1. Compute the $(1/\mu) = \sum_i p_i t_i$, assuming $f_i = 1$ where $i = 1, 2, \ldots, M$;
2. Find the value of q_j^{opt} based on (6), and the bin space of disk j is defined as q_j^{opt}/μ where $j = 1, 2, \ldots, D$;
3. For each multimedia file i, introduce n_i elements with size $p_i/n_i \mu_i$;
4. Sort the elements in descending order according to the size;
5 While (not finished)
 {
 - Select the bin j with maximum available space, providing that it does not contain the file i and its capacity would

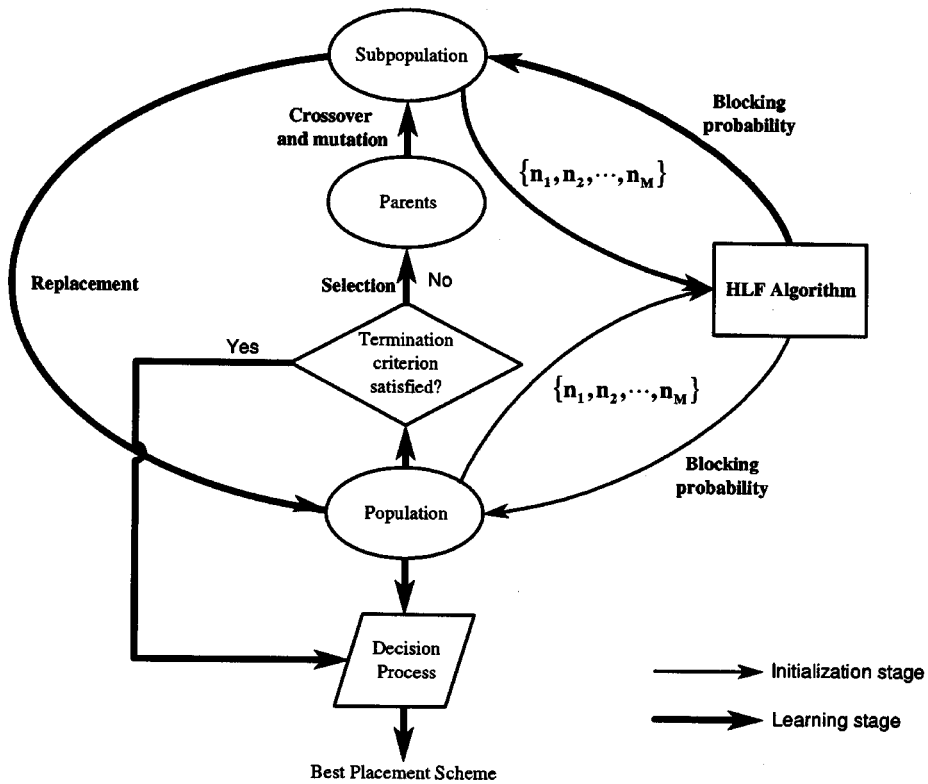

Fig. 2. Operational cycle of the overall system.

not be exceeded after the file allocation. If no suitable bin can be assigned, issue a constraint violation flag and exit.

- Allocate the file i into bin j and reduce the available space of the bin j by $p_i/(n_i\mu_i)$.

}

B. GA

The GA [12] has been demonstrated as a powerful optimization tool for the multiobjective problems [7], [17]. It is inspired by the mechanism of natural selection, where stronger individuals would likely be the winners in a competing environment.

In this placement problem, the chromosome is in the form of an integer string $I = \{n_1, n_2, \ldots n_M\}$. The upper bound of each gene (n_i^{ub}) is D, while the lower bound (n_i^{lb}) is determined as the smallest positive integer satisfying the relationship $p_i/(n_i^{lb}\mu_i) \leq \max_j q_j^{\text{opt}}/\mu$. The operational cycle of the overall system is depicted in Fig. 2. A detailed implementation of the GA may be found in [17] and [18].

1) Crossover and Mutation: A multipoint crossover [17], [18] is adopted in our system to exchange the information between chromosomes. For mutation, each gene of the chromosome n_i is randomly altered within the searching domain $[n_i^{lb}, n_i^{ub}]$ with a small probability (p_m) [18].

2) Objective Functions: The two objectives explained in Section III are formulated as follows.

1) The objective function O_1 is defined as the total capacity usage

$$O_1 = \sum_{i=1}^{M} s_i \cdot n_i \tag{9}$$

where s_i is the size of the multimedia file i.

2) The objective function O_2 is defined as the percentage deviation between the optimal blocking probability (B) obtained by the HLF algorithm in Section III-A and the minimum blocking probability $(B_{\min})$ computed as explained in Section II. Hence,

$$O_2 = \frac{B - B_{\min}}{B_{\min}} \times 100\%. \tag{10}$$

3) Constraint Handling: The solution can only be found by the HLF if the disk capacity constraint is fulfilled. If no solution is found, penalty values are assigned to the objectives so as to reflect the condition of the low performer.

4) Pareto-Based Fitness Assignment: Instead of aggregating the objectives with a weighting function, the multiobjective approach [5], [6] is applied. Assuming that chromosome I is dominated by other p chromosomes in the population, its rank is determined as

$$\text{rank}(I) = 1 + p. \tag{11}$$

Hence, a Pareto-based fitness can be assigned to each chromosome according to its rank in the population.

Pareto-based ranking can correctly assign all nondominated chromosomes with the same fitness. However, the Pareto set may not be uniformly sampled. Usually, the finite population will converge to only one or some of these chromosomes, due to stochastic errors in the selection process. Therefore, the techniques of fitness sharing [10] and mating restriction [4] are adopted in our design to prevent the drift, and to promote the sampling of the whole Pareto set by the population.

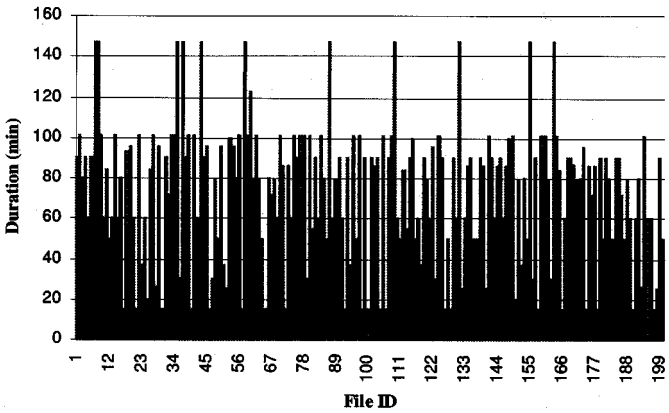

Fig. 3. Duration of the videos with their file IDs.

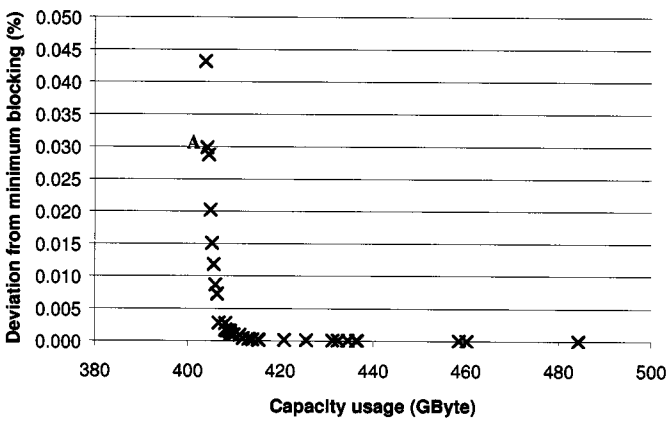

Fig. 4. Rank 1 solution in the final population.

IV. Experimental Results

The system configuration of the example given in Section II is used in the following tests. The popularities of the multimedia files are assumed to be governed by Zipf's law and computed as follows:

$$p_i = \frac{c}{i^{\zeta}} \quad i = 1, 2, \ldots, K \tag{12}$$

where $c = (\sum_{i=1}^{K}(1/i^{\zeta}))^{-1}$ and ζ is a constant describing the popularity distribution of the files.

A. Test 1

It is assumed that the contents of the VOD system consist of 200 multimedia files encoded in MPEG 2, randomly selected from several types of videos. The durations of the videos are depicted in Fig. 3, and the popularities are assigned according to the file IDs using (12) with $\zeta = 0.271$ [1].

The traffic intensity is assumed to be 744.3 with $\lambda = 10$ requests per minute and $(1/\mu) = 74.43$ minutes. Fig. 4 shows all the rank 1 solutions in the final population after 10 000 GA cycles.

The mark "A" indicates the best solution found if a maximum deviation of 0.03% from minimum blocking probability is allowed for customer requirement. The blocking probability and the capacity usage are 0.029 899 5 (0.029 874% deviated from the minimum blocking probability) and 404.274 GByte, respectively. Only one file requires two copies as reflected by the very small increment in the capacity usage comparing with 403.92 GB, the sum of the size of all multimedia files.

TABLE I
Blocking Probability With Different Reordering Methods

	Blocking probability
Minimum blocking	0.0298905
Descending	0.0298995
Ascending	0.0404577
Random	0.0350503
Regular	0.0316189

TABLE II
Solution Obtained in Test 3 Using GA

	blocking probability (% deviated from ideal case)	capacity usage in GByte
Case 1	0.0298994 (0.0298%)	405.688
Case 2	0.0298942 (0.0124%)	405.688
Case 3	0.0298995 (0.0299%)	404.274

TABLE III
Optimal Placement in Test 3 With Single Copy Only

	blocking probability (% deviated from ideal case)	capacity usage in GByte
Case 1	0.0299017 (0.0373%)	403.920
Case 2	0.0299135 (0.0767%)	403.920
Case 3	0.0299034 (0.0432%)	403.920

B. Test 2: Different Reordering Approaches

Using the solution of the file copies obtained in Test 1, the four different reordering approaches explained in Section III-A are applied. The results are tabulated in Table I for comparison.

It can be observed that the descending approach has the best performance and the result obtained is close to the ideal blocking probability, while the ascending approach has the worst result.

C. Test 3: Duration and Popularity

Since the relationship of the popularity and the duration may vary from case to case, three different situations are studied.

- *Case 1:* The video with shorter duration has the higher popularity.
- *Case 2:* The video with longer duration has the higher popularity,
- *Case 3:* The rank of the popularity is randomly assigned (same as Test 1).

All the conditions in Test 1 are kept, and the solution found by the GA on these three different cases are tabulated in Table II.

Table III tabulates the best solution using the HLF if only a single copy is allowed for each file. It can be easily observed

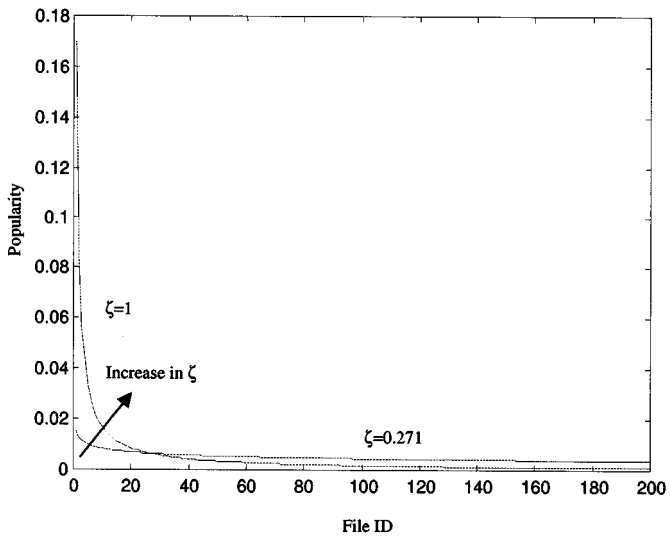

Fig. 5. Popularity distributions with different ζ.

that the design specification is violated in all three cases. By comparing Tables II and III, a small increment in capacity usage makes the customer specification achievable.

D. Test 4: Popularity Distribution

The distribution of the popularity implies different customer behavior in request. If ζ is small, the popularity is more uniform. While requests are concentrated onto a few multimedia files, ζ is increased, as illustrated in Fig. 5.

Assuming that the traffic intensity A is fixed and equal to 744.30, the following different values of ζ are tested:

- $\zeta = 0$ (uniform distribution);
- $\zeta = 0.271$ (same as Test 1);
- $\zeta = 1.0$.

Similarly, a maximum deviation of 0.03% from minimum blocking probability is specified for customer requirement. The solutions obtained by the GA are tabulated in Table IV. Similarly to Test 3, a small increment in the capacity usage causes a significant improvement as compared in Tables IV and V.

Table VI tabulates the best solution using the HLF if two copies are made for each file. It shows that the design specification can only be achieved when $\zeta = 0.271$. Together with Table V, it is concluded that proper distribution of copies should be assigned in order to minimize the blocking probability as well as the capacity usage.

TABLE IV
SOLUTIONS OBTAINED IN TEST 4 USING GA

	blocking probability (% deviated from minimal case)	capacity usage in GByte
$\zeta = 0$	0.0298988 (0.0277%)	406.749
$\zeta = 0.271$	0.0298995 (0.0299%)	404.274
$\zeta = 1.0$	0.0298924 (0.0062%)	416.514

TABLE V
OPTIMAL PLACEMENT IN TEST 4 WITH SINGLE COPY ONLY

	blocking probability (% deviated from minimal case)	capacity usage in GByte
$\zeta = 0$	0.0299625 (0.2407%)	403.920
$\zeta = 0.271$	0.0299034 (0.0432%)	403.920
$\zeta = 1.0$	0.1608380 (438.09%)	403.920

TABLE VI
OPTIMAL PLACEMENT IN TEST 4 WITH TWO COPIES

	blocking probability (% deviated from minimal case)	capacity usage in GByte
$\zeta = 0$	0.0299018 (0.0378%)	807.840
$\zeta = 0.271$	0.0298994 (0.0297%)	807.840
$\zeta = 1.0$	0.0670336 (124.26%)	807.840

V. CONCLUSION

In this paper, the minimum blocking probability of a heterogeneous VOD system with different types of storage disks and multimedia files was derived. It provides not only the lower bound of the blocking probability, but also the required load sharing of the system.

A heuristic approach based on the concept of bin packing is developed to achieve the optimal load sharing. Together with the GA, the optimal number of copies of each multimedia file and their corresponding disk locations were determined. The results demonstrated that a specified blocking probability can be achieved in such a heterogeneous VOD system while the capacity usage is minimized at the same time.

REFERENCES

[1] S. A. Barnett and G. J. Anido, "A cost comparison of distributed and centralized approaches to video-on-demand," *IEEE J. Select. Areas Commun.*, vol. 14, pp. 1173–1183, Aug. 1996.

[2] D. Bertsekas and R. Gallager, *Data Networks*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1992, p. 179.

[3] J. E. G. Coffman, M. R. Garey, and D. S. Johnson, "Approximation algorithms for bin-packing—An updated survey," in *Algorithm Design for Computer System Design*. New York: Springer-Verlag, 1984, pp. 49–99.

[4] K. Deb and D. E. Goldberg, "An investigation of niche and species formation in genetic function optimization," in *Proceedings of the 3rd International Conference on Genetic Algorithms*, J. D. Schaffer, Ed. San Mateo, CA: Morgan Kaufmann, 1989, pp. 42–50.

[5] C. M. Fonseca and P. J. Fleming, "Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization," in *Genetic Algorithms: Proceedings of the Fifth International Conference*, S. Forrest, Ed. San Mateo, CA: Morgan Kaufmann, 1993, pp. 416–423.

[6] ——, "Multiobjective optimization and multiple constraint handling with evolutionary algorithms—Part I: A unified formulation," *IEEE Trans. Syst., Man, Cybern. A*, vol. 28, pp. 26–37, Jan. 1998.

[7] ——, "Multiobjective optimization and multiple constraint handling with evolutionary algorithms—Part II: Application example," *IEEE Trans. Syst., Man, Cybern. A*, vol. 28, pp. 38–47, Jan. 1998.

[8] D. J. Gemmell, H. M. Vin, D. D. Kandlur, P. V. Rangan, and L. A. Rowe, "Multimedia storage servers: A tutorial," *IEEE Computer*, vol. 28, pp. 40–49, May 1995.

[9] A. Girard, *Routing and Dimensioning in Circuit-Switched Networks*. Reading, MA: Addison-Wesley, 1990.

[10] D. E. Goldberg and J. Richardson, "Genetic algorithms with sharing for multimodal function optimization," in *Proceedings of the 2nd International Conference on Genetic Algorithms*, J. J. Grefenstette, Ed. Hillsdale, NJ: Lawrence Erlbaum, 1987, pp. 41–49.
[11] R. H. Hwang and Y. C. Sun, "Optimal video placement for hierarchical video-on-demand systems," *IEEE Trans. Broadcasting*, vol. 44, pp. 392–401, Dec. 1998.
[12] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Cambridge, MA: MIT Press, 1975.
[13] N. Karmarkar and R. M. Karp, "An efficient approximation scheme for the one-dimensional bin packing problem," in *Proc. 23rd Annual FOCS*, 1982, pp. 312–320.
[14] T. D. C. Little and D. Venkatesh, "Prospects for interactive video-on-demand," *IEEE Multimedia*, vol. 1, pp. 14–24, Autumn 1994.
[15] ——, "Popularity-based assignment of movies to storage devices in a video-on-demand system," *Multimedia Syst.*, vol. 2, pp. 280–287, Jan. 1995.
[16] P. Lougher and D. Shepherd, "The design of a storage server for continuous media," *Computer J.*, vol. 36, no. 1, pp. 32–42, 1993.
[17] K. F. Man, K. S. Tang, and S. Kwong, *Genetic Algorithms: Concepts and Designs*. London, U.K.: Springer-Verlag, 1999.
[18] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Program*, 3rd ed. Berlin, Germany: Springer-Verlag, 1996.
[19] J. P. Nussbaumerm, B. V. Patel, F. Schaffa, and J. P. G. Sterbenz, "Networking requirements for interactive video on demand," *IEEE J. Select. Areas Commun.*, vol. 13, pp. 779–787, June 1995.
[20] I. Ouveysi, K. C. Wong, S. Chan, and K. T. Ko, "Video placement and dynamic routing algorithms for video-on-demand networks," in *Proc. Global Telecommunications Conf.*, vol. 2, 1998, pp. 658–663.
[21] W. D. Sincoskie, "System architecture for a large scale video on demand service," *Comput. Networks ISDN Syst.*, vol. 22, pp. 155–162, 1991.

Kit-Sang Tang (M'96) received the B.Sc. degree from the University of Hong Kong, Hong Kong, and the M.Sc. and Ph.D. degrees from the City University of Hong Kong, Hong Kong, in 1988, 1992, and 1996, respectively.

He is currently an Assistant Professor in the Department of Electronic Engineering, City University of Hong Kong. He is a member of the IFAC Technical Committee on Optimal Control (Evolutionary Optimization Algorithms). His research interests include evolutionary algorithms and chaotic theory.

Dr. Tang is a member of the Intelligent Systems Committee of the IEEE Industrial Electronics Society.

King-Tim Ko (S'78–M'81) was born in Hong Kong. He received the B.Eng. (Hons.) degree with First Class Honors in electrical engineering and the Ph.D. degree in communication engineering from The University of Adelaide, Adelaide, Australia, in 1977 and 1982, respectively.

From 1982 to 1986, he was a Senior Engineer with Telecom Australia Research Laboratories, Melbourne, Australia. In 1986, he joined the Department of Electronic Engineering, City University of Hong Kong, Hong Kong, where he is currently an Associate Professor. His main research interests include design and performance of communication networks, as well as multimedia applications. He is currently a member of the Working Group on E-Commerce for the Hong Kong SAR Government.

Sammy Chan (S'87–M'91) received the B.E. and M.Eng.Sc. degrees in electrical engineering from the University of Melbourne, Melbourne, Australia, and the Ph.D. degree in communication engineering from Royal Melbourne Institute of Technology, Melbourne, Australia, in 1988, 1990, and 1995, respectively.

From 1989 to 1994, he was with Telecom Australia Research Laboratories, first as a Research Engineer, and between 1992 and 1994, as a Senior Research Engineer and Project Leader. Since December 1994, he has been with the Department of Electronic Engineering, City University of Hong Kong, Hong Kong, where he is currently an Associate Professor. His current research interests are video on demand and wireless ATM networks.

Eric W. M. Wong (S'88–M'88–SM'00) received the B.Sc. and M.Phil. degrees in electronic engineering from the Chinese University of Hong Kong, Hong Kong, and the Ph.D. degree in electrical and computer engineering from the University of Massachusetts, Amherst, in 1988, 1990, and 1994, respectively.

In 1994, he joined the City University of Hong Kong, Hong Kong, where he is currently an Associate Professor in the Department of Electronic Engineering and a member of the Networking Group within that department. His research interests are high-speed networks, video on demand, satellite communications, and dynamic routing. The most notable of these involved the analytical modeling of the least-loaded routing scheme in circuit-switched networks. The model drastically reduces the computational complexity of designing and dimensioning telephone systems. The model has also served as a core for the analysis of many other advanced routing schemes, such as the Real-Time Network Routing currently implemented in the AT&T telephone network.