IEICE TRANS. COMMUN., VOL.E85–B, NO.2 FEBRUARY 2002

PAPER

Reliability-Based Mirroring of Servers in Distributed Networks

Akiko NAKANIWA†, **Jun TAKAHASHI**†, ***Student Members***, **Hiroyuki EBARA**†, *and* **Hiromi OKADA**†, ***Regular Members***

SUMMARY In this paper, we consider optimal mirror allocation problems for the purpose of load balancing in network servers. We focus on constructing high-reliability networks and propose the optimal mirror allocation model such that the system reliability is maximized subject to costs and delays, in view of the trade-off between the reliability and cost. This optimization model is capable of dealing with various kinds of network topologies, although for simplicity, we assume the read-only situation. We formulate this optimization problem into a 0–1 integer programming model, and we use an approximate method for numerical analysis in order to analyze more large-scale systems. Our objective is to find the optimal mirror allocation by solving this model, and to show quantitatively the general characteristics of the load balancing and the improvement of the system reliability by the distributed mirror allocation.

key words: *load balancing, reliability, mirroring, distributed database system, Internet*

1. Introduction

There has been an explosive increase in the number of users in multi-media networks, particularly over the Internet. Furthermore, the increasing traffic over these networks to enable users to connect to the Internet from mobile computers including cellular phones is spreading rapidly. This causes network servers to become overloaded, and we are faced with several essential issues, such as the decline of reliability, the increase of the access delay and so on [3], [5]. In finding a solution for these issues, the mirroring of network servers has been considered and various studies concerning this issue have been done widely. Several mirror servers, which store copies of the same files and have the same function as network servers, are set in networks for the purpose of load balancing.

Distributed allocation of mirror servers results in prompt access of users and improvement of system reliability. However, this causes the cost of setting up mirror servers and managing the whole system to increase. It is obvious that there is a trade-off between the reliability and cost. In constructing effective networks and providing emerging multi-media services, one of the most important problems today lied in the allocation of mirror servers and copies of files on the networks. Though there have been many works on distributed file allocation for this purpose, most of them consider only costs and delays, and do not consider reliability [7]–[14].

We focus on constructing high reliability networks. In this paper, We propose the *Optimal Mirror Allocation Model* such that the system reliability is maximized subject to the cost and the delay, in view of the tradeoffs described above. In this model, we introduce a 0–1 integer programming formulation by setting the 0–1 variables on the allocation of mirror servers and files that exist on each server. We can find where each mirror server should be, and the concrete assignment of files to mirror servers [15].

Also, we give a network topology by an adjacency matrix, whose elements have weights relating to the distance between a pair of nodes. Most of the studies concerning the mirroring in networks consider only the tree structure as a network topology [16], [17]. However, the topology of current multi-media networks such as the Internet is not limited to the tree structure. We need to be able to consider other network structures. Our model is capable of dealing with various kinds of network structures, including tree structures, mesh structures, etc. In addition, we may take into account any problems with regards to communication costs and delays by using the weight of the distance. However, in this paper, we assume the read-only situation for simplicity and do not consider update costs and delays. we consider this matter as our future work.

Our objective is to find the optimal mirror allocation by solving this model, and demonstrate quantitatively the general characteristics on the improvement of the system reliability using the distributed allocation of mirror servers.

In this paper, we introduce the optimal mirror allocation model in Sect. 2. In Sect. 3, we present the approximate method we use in order to deal with practical large-scale systems, and in Sect. 4, we show the numerical results by this approximate method. Finally, we describe our conclusions in Sect. 5.

2. Optimal Mirror Allocation Model

2.1 System Model

We introduce a system model of a distributed system

Manuscript received October 2, 2000.
Manuscript revised May 30, 2001.

†The authors are with the Faculty of Engineering, Kansai University, Suita-shi, 564-8680 Japan.

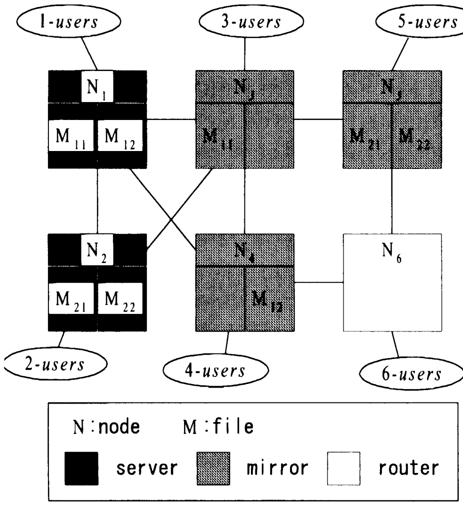


Fig. 1 An example of distributed network model.

for the optimal mirror allocation (see an example in Fig. 1). We give a topology of a network by an adjacency matrix $\mathbf{A}$, whose elements have weights relating to the distance between a pair of nodes. We calculate the shortest path matrix $\mathbf{Q}$, and formulate the reliability, the cost, and the delay by the matrix $\mathbf{Q}$.

There exist n nodes in the system, and each node is denoted by N_i $(1 \leq i \leq n)$. It is assumed that all nodes originally function as routers. s nodes $(s \leq n)$ work as servers, which hold individual files, and the rest of the nodes, say $(n-s)$ nodes, may be mirror servers of the s servers by storing all or a part of their files. In the case that node N_i is a server or a mirror server, it has the storage capacity of B_i. It is assumed that each user in the system is connected to one of the nodes, called a local node. The users connected with the node N_i are called *i-users*.

The number of kinds of files at each server N_i is m_i $(1 \leq i \leq s)$, and each file is denoted by M_{ik} $(1 \leq k \leq m_i)$. It is assumed that the size of the file M_{ik} is denoted by F_{ik}. Each mirror server can store these files as long as the total size of files does not exceed the storage capacity.

i-users generate file access flow, where the average is α_i per unit time. Also, let P_{ijk} denote the access probability from *i-users* for the files M_{jk} $(\sum_j \sum_k P_{ijk} = 1)$. The parameter P_{ijk} determines the traffic matrix in the system. As mentioned above, it is assumed in this paper that all file accesses are just to read files, and we do not consider the updating of files.

Moreover, we define λ_i as the failure rate of a node N_i, μ_i as the service rate, and t_{ij} as the communication rate of a link between the nodes N_i and N_j. In this paper, it is assumed for analytical simplicity that the communication rate of each link has the identical value t. We also do not consider the case of the failure of each link in networks.

2.2 Formulation

Our objective is to find the optimal allocation of mirror servers such that the system reliability is maximized. The reliability is a very important criterion to evaluate distributed database systems. However, some systems that optimize only the reliability is not very useful because the system may suffer from high costs and long delays. We take into account the cost to set up mirror servers, the total cost of managing the whole system, and the communication delay as the restrictive condition. In addition, we would like to consider the MTTR (mean time to repair), the detour of the path, etc. in the future to analyze more accurately system behavior where faults occur.

To formulate the optimal mirror allocation model, we define two kinds of 0–1 variables. We formulate this optimization problem as 0–1 integer programming problem using the following 0–1 variables.

1. The variables on the function of nodes S_{i1},S_{i2}.

$$S_{i1} = \begin{cases} 1 & \text{(the node } N_i \text{ is a server)} \\ 0 & \text{(otherwise)} \end{cases} \tag{1}$$

$$S_{i2} = \begin{cases} 1 & \text{(the node } N_i \text{ is a mirror server)} \\ 0 & \text{(otherwise)} \end{cases} \tag{2}$$

The variables on the allocation of files X_{ijk} [4].

$$X_{ijk} = \begin{cases} 1 & \text{(the file } M_{jk} \text{ is stored in the node } N_i) \\ 0 & \text{(otherwise)} \end{cases} \tag{3}$$

2.2.1 Primary Conditions

Under the assumptions presented in the previous section, we add the following capacity condition as a primary condition. Each node N_i can store files so long as the total size of files does not exceed the capacity of the node, that is

$$\sum_j \sum_k X_{ijk} F_{jk} \leq B_i \tag{4}$$

2.2.2 Objective Function

We formulate the reliability per unit time in the whole system as the objective function. In this paper, we define the system reliability as the mean of the success rate of each access from each user to each file. We consider the required nodes to complete the access from each user to each file, and calculate the reliability of the combination of these nodes using the reliability of each node as the success rate of each access. In the case that there are several paths between nodes, for the sake of analytical simplicity, we take into account only the shortest path. We should consider the detour of the path in the case of path failure in the future.

In this paper, we assume that the user accesses the node that has the shortest distance from the local node, if a file requested from a user is not stored in the local node and some remote nodes store the requested file. Using common probability formulas [1], [2], [6], the node reliability of a node N_i, $R_i(\mathbf{S},\mathbf{X})$, is formulated as follows.

$$\begin{aligned} R_i(\mathbf{S},\mathbf{X}) &= S_{i1}\cdot\exp\Bigg(-\lambda_i\cdot\sum_j\sum_k\frac{\alpha_j P_{jik}F_{ik}}{\mu_i} \\ &\quad\cdot\left(\Delta_{ji}+(1-\Delta_{ji})(1-X_{jik})\Delta_{jh_1}\right)\Bigg) \\ &\quad+S_{i2}\cdot\exp\Bigg(-\lambda_i\cdot\frac{1}{\mu_i}\sum_j\sum_k\Bigg(X_{ijk} \\ &\quad\cdot\alpha_i P_{ijk}F_{jk} \\ &\quad+\sum_g X_{ijk}\left(1-X_{gjk}\right)\Delta_{ih_2}\cdot\alpha_g P_{gjk}F_{jk}\Bigg)\Bigg) \\ &\quad+(1-S_{i1})(1-S_{i2}) \end{aligned} \tag{5}$$

Here, Δ_{ji} is defined as follows, and index h_1 and h_2 are equal to the value that minimizes the following functions on the positive side individually.

$$\Delta_{ji}=\delta\,(j-i)=\begin{cases}1 & (j=i)\\ 0 & (j\neq i)\end{cases} \tag{6}$$

$$h1_{min}(h_1)=X_{h_1ik}\cdot q_{jh_1} \tag{7}$$

$$h2_{min}(h_2)=X_{h_2jk}\cdot q_{gh_2} \tag{8}$$

In Eq. (5), the first term refers to the case where node N_i is a server, the second term refers to the case where the node is a mirror server, and the third term refers to the case the node only functions of a router. In the case that each node works as only a router, it is assumed that the failure rate of the node is equal to 0. This means that the reliability of the node is equal to 1.

Using this node reliability, we calculate the system reliability. In the following, we describe the procedure of the calculation.

1. We calculate *R_path* as the reliability of all nodes that are required to be available for the user to succeed in accessing each file as follows. In the case that there exist several nodes which store the requested file from the user, we calculate *R_path* for all paths from the user to these nodes.

$$R\_path=\prod_i R_i \tag{9}$$

2. In the case that there is only one node which stores the requested file, we calculate $R\_temp(j,M_{ik})$ as the success rate of the access from *j-user* to the file M_{ik} as follows.

$$R\_temp(j,M_{ik})=R\_path \tag{10}$$

Otherwise, $R\_temp(j,M_{ik})$ is as follows.

$$R\_temp(j,M_{ik})=1-\prod(1-R\_path) \tag{11}$$

3. We calculate the mean of $R\_temp(j,M_{ik})$ as the system reliability R_s.

$$R_s=\frac{\sum_j\sum_i\sum_k R\_temp(j,M_{ik})}{n\cdot\sum_i m_i} \tag{12}$$

2.2.3 Restrictive Conditions

We formulate the mirror cost, the total cost, and the communication delay as the restrictive conditions for the optimal mirror allocation problem.

1. Mirror Cost Cf
The mirror cost Cf is the primary cost to set up mirror servers. This depends on the storage capacity of files on mirror servers. Let Cf_{max} be defined as the maximum mirror set cost. Here, Cb is defined as the capacity cost coefficient, and Cm is the administration cost coefficient. Then, the mirror cost is as follows.

$$Cf=\sum_i S_{i2}\cdot(B_i\cdot\mathsf{Cb}+\mathsf{Cm})\le Cf_{max} \tag{13}$$

2. Total Cost C
The total cost is the cost to manage the whole system per unit time. It is assumed that the total cost C consists of the storage cost Cs and the communication cost Ct. Let C_{max} be defined as the maximum total cost, and the total cost is as follows.

$$C=Cs+Ct\le C_{max} \tag{14}$$

Next, we give full details of the storage cost and the communication cost.

- Storage Cost Cs
We assume that the storage cost is required for each node to store files. It is charged on the basis of unit time, and it is not dependent on the frequency of accesses. In this paper, the storage cost depends on the size of all the files stored at a node. Let Cs be the storage cost coefficient, which has a fixed value that is common for all nodes.

$$Cs=\mathsf{Cs}\cdot\sum_i\sum_j\sum_k X_{ijk}F_{jk} \tag{15}$$

- Communication Cost Ct
  We assume that the communication cost is required to communicate between a user site and his/her requested file site. For example, when an *i-user* accesses a file M_{jk}, the communication cost will be the cost to communicate between an *i-user* and a node that stores the file M_{jk}. If a local node N_i stores the file M_{jk}, the communication cost is the cost to communicate only between an *i-user* and the local node. Also, in this paper, if a file requested from a user is not stored in the local node and some remote nodes store the file, the user accesses the nodes that have the shortest distance from the local node, as described before. We define Ct_{jk} as the communication cost coefficient of the file M_{jk}. It is assumed that the communication cost coefficient Ct_{jk} is dependent on the size of the file F_{jk}. Ctt is the communication cost coefficient which has a fixed value that is common for all files.

$$\mathsf{Ct}_{jk} = \mathsf{Ctt} \cdot F_{jk} \tag{16}$$

  By using this coefficient and 0–1 variables X_{ijk}, we formulate the communication cost as follows.

$$Ct = \sum_i \left(\sum_j \sum_k (X_{ijk}\mathsf{Ct}_{jk} + (1 - X_{ijk})X_{h_3jk}q_{ih_3}\mathsf{Ct}_{jk})\alpha_i P_{ijk} \right) \tag{17}$$

  Here, it is assumed that index h_3 equals the value that minimizes the following function on the positive side.

$$h3_{min}(h_3) = X_{h_3jk} \cdot q_{ih_3} \tag{18}$$

3. Communication Delay Dt
   The communication delay is that spent to communicate in accessing from a user to a file. If we let Dt_{max} be defined as the maximum communication delay, and the communication delay is as follows. We define Dt_{jk} as the communication delay coefficient of the file M_{jk}. It is assumed that the communication delay coefficient Dt_{jk} is dependent on the size of the file F_{jk} and the communication rate t.

$$Dt = \sum_i \left(\sum_j \sum_k \left(X_{ijk}\mathsf{Dt}_{jk} + (1 - X_{ijk})X_{h_3jk}q_{ih_3}\mathsf{Dt}_{jk})\alpha_i P_{ijk} \right) \right. \leq Dt_{max} \tag{19}$$

$$\mathsf{Dt}_{jk} = \frac{F_{jk}}{t} \tag{20}$$

   Here, it is also assumed that index h_3 equals the value that minimizes the function $h3_{min}(h_3)$ (Eq. (18)) on the positive side, too.

3. Approximate Method

In the previous section, we introduced a 0–1 integer programming model for the optimal mirror allocation model. We use an approximate algorithm for the optimization problem to deal with practical large-scale systems. In solving optimization problems, the algorithm used is very significant. Our optimal mirror allocation model belongs to the class of NP hard problems. Using an exhaustive search, we may solve only a small model, and we may have to spend a long time to solve even the small problem. In this section, we compare the approximate method with the exhaustive search.

3.1 Approximate Algorithm

The approximate algorithm we use is based on the Greedy Method [13], which finds feasible solutions by directly fixing each variable on the basis of local critical values for contribution to an objective function. We can easily imagine that more popular files are likely to be distributed in distributed database systems. In this paper, we take the access frequency to the files as the local critical values, that is $\alpha_j \times P_{jik}$, which is denoted by Af_{jik} hereafter. We fix the variable $X_{jik} = 1$ in the order of the value of the corresponding access frequency Af_{jik}, so long as all restrictive conditions are satisfied. In the following, we describe our algorithm based on the Greedy Method.

[Approximate Algorithm]

1. First, we calculate the access frequency Af_{jik} to each file.
2. We sort the variables X_{jik} in decreasing order of Af_{jik}.
3. We repeat the procedures from 3.1 to 3.5 until the last variable of the sorted sequence in the previous procedure.

   3.1 Fix a variable $X_{jik} = 1$ in a sorted order.
   3.2 Check whether the primary condition is satisfied or not. If the primary condition is not satisfied, we return the value to 0, and go back to the step 3.1.
   3.3 Check whether the rest of restrictive conditions are satisfied or not. If the restrictive conditions are not satisfied, go back to the step 3.1.
   3.4 Fix variables S_{i1} and S_{i2} on the basis of the current values of X_{jik}.

Table 1 Traffic model.

Traffic Model	Mean Access Times α_j	Access Probability P_{jik}
1	identical	identical
2	identical	random
3	variant	identical
4	variant	random

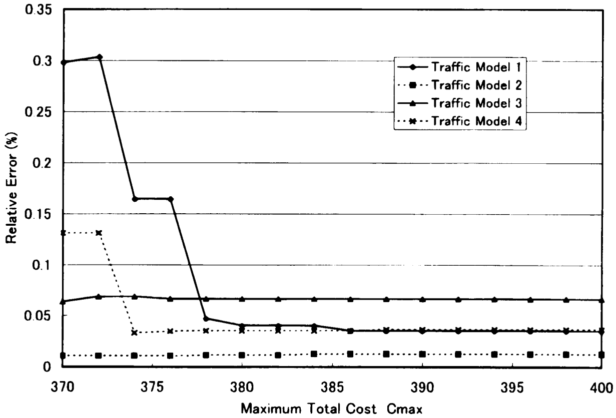


Fig. 2 A comparison of two algorithms.

3.5 Calculate the system reliability for the current allocation determined in the previous procedures, and commit the mirror allocation with the system reliability to memory.

4. We select the mirror allocation such that the system reliability is maximized for all allocation calculated in step 3.5, and consider it as the approximate solution.

3.2 Performance Evaluation

For the comparison between the approximate solution and the optimal solution by the exhaustive search, we assume four kinds of traffic models as shown in Table 1.

We define the relative error as the performance measure as follows.

$$\text{RelativeError}(\%) = \frac{\text{optimal reliability} - \text{approximate reliability}}{\text{optimal reliability}} \times 100 \quad (21)$$

In Fig. 2, We show the performance comparison between the approximate method and the exhaustive search for the above traffic models. Here, we consider the system parameters as follows. There exist 5 nodes ($n = 5$), and the capacity of each node is 20 [Gbyte] ($B_i = 20000, 1 \leq i \leq 5$). There are 2 servers ($s = 2$), and the rest of the nodes, say 3 nodes, may be mirror servers. Also, there exist 2 kinds of files on each server respectively ($m_i = 2, 1 \leq i \leq s$), and the size of each file is 30 [Mbyte] ($F_{ik} = 30$). This figure shows the relative error vs. the maximum total cost C_{max}. As illustrated, the value of the relative error of the approximate method is less than approximately 0.3% in all traffic models. In most cases, the relative error has a value under 0.1%. From these results, we can see that the approximate method is almost exact, and that it is practical enough.

Moreover, we compare the average time it takes to solve the optimization problem in Table 2. In this table, we show the average time [second] each algorithm takes in solving several kinds of scales of systems. Here, we used Sun UltraSPARC II$_i$ (334 MHz) for the numerical calculations. There is a remarkable difference between these two algorithms. In the case of the exhaustive search, the average time increases exponentially as systems scale. Therefore, the approximate method is useful in view of the calculation time.

In the case of the exhaustive search, the scale of the system model we can solve is basically limited to the scale $n = 5, s = 2, m_i = 3$. By using the approximate method, we show numerical results of the optimization problem for practical large-scale system models in the next section.

4. Numerical Results and Considerations

4.1 System Parameters for Numerical Results

We show quantitatively the general characteristics on the load balancing by the distributed allocation of mirror servers. In our model, we may consider various network environments such as the Internet, mobile network, etc., by setting system parameters. Here, for analytical simplicity, we consider the system parameters as follows. There exist 30 nodes ($n = 30$), and the capacity of each node is 20 [Gbyte] ($B_i = 20000, 1 \leq i \leq 30$) in case that the node is a mirror server. There are 10 servers ($s = 10$), and the rest of the nodes, say 20 nodes, may be mirror servers. Also, there exist 10 kinds of files on each server respectively ($m_i = 10, 1 \leq i \leq s$), and the size of each file is 10–100 [Mbyte] ($F_{ik} = 10$–100). However, we set the file size differently in the paragraph of characteristics on mirror restriction. Also, we assume the high rate of failure so that we can clearly see the difference in the system reliability due to the mirroring.

Considering the optimal mirror allocation problem, we define the number of mirrors in systems Nm and the distribution rate of files Rdf for performance measures as follows.

Table 2 Average time comparison between two algorithms in traffic model 2 [sec].

n	s	$\sum_i m_i$	Combinations of Variables	Exhaustive Search	Approximate Algorithms
4	2	4	2^8	0.13	0.00111
5	2	4	2^{12}	41.07	0.00144
4	2	6	2^{12}	38.5	0.00171
4	2	8	2^{16}	12411	0.00289
5	2	6	2^{18}	234792	0.00351

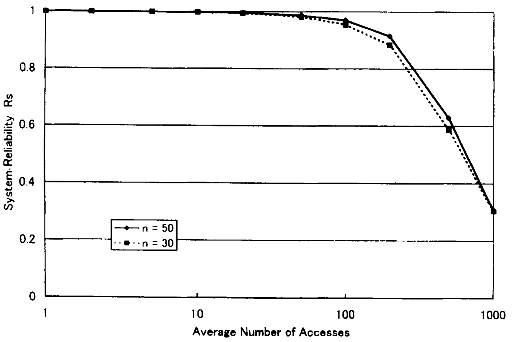


Fig. 3 Reliability characteristics on change of traffics.

$$Nm = \sum_i S_{i2} \tag{22}$$

$$Rdf = \frac{\sum_j \sum_i \sum_k X_{jik}}{\sum_i m_i} \tag{23}$$

Nm is the number of nodes which work as mirror servers in the system. Rdf is the ratio of the number of all files allocated in the system to the number of kinds of files. We obtain the assignment of mirrors to networks, and files to nodes as the optimal solution. On the basis of this assignment, we calculate the distribution rates, the system reliability, the total cost, the communication delay, etc. We describe our considerations regarding the optimization results according to these criterion.

4.2 General Characteristics

4.2.1 Reliability Characteristics on Change of Traffics

Firstly, we show the characteristics of the system reliability on the change of the average number of accesses from users.

In Fig. 3, we show the system reliability vs. the average number of accesses for the different scales of systems, i.e. the number of nodes n equal to 50, 30. We set the number of servers s as 10 here, as mentioned before. In this figure, we assume that the restrictive conditions do not affect the optimization at all, in order to investigate the influence of the change of traffic. As we may see from this figure, the system reliability decreases, as the number of accesses increase. This figure shows quantitatively the decline of the system reliability as the traffic increases.

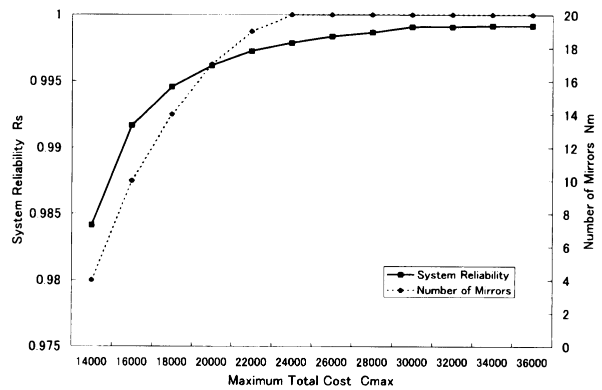


Fig. 4 An improvement of the system reliability by the mirror distribution.

We may guess that the allocation of more mirror servers can constrain such a decline of the system reliability. Therefore, we show quantitatively the improvement of the system reliability by the mirror distribution in the following.

4.2.2 Characteristics Using Cost Restriction

We show the characteristics of the mirror distribution using the restrictive condition of the total cost.

In Fig. 4, we show the system reliability and the number of mirrors vs. the maximum total cost C_{max} in the case that the average number of accesses is equal to 3. As shown in this figure, when C_{max} is small, files are not distributed in so many mirror servers and the system reliability is low. When C_{max} is large, that is, the total cost restriction is not so tight, all nodes, which can be mirror servers, work as mirror servers, and we see improvement in the system reliability. We can obtain the assignment of mirrors to networks, and files to nodes as the optimal solution. From this figure, we see clearly that the system reliability improves using the mirror distribution. This result also illustrates quantitatively the trade-off between the system reliability and the total cost.

In Fig. 5, we show the trade-off between the system reliability and the total cost in more detail. This figure shows the system reliability vs. the normalized value of the maximum total cost for the different numbers of accesses, that is, 1, 5, 10, 50, 100. As shown in this figure, the larger the maximum total cost, the higher the system reliability. Also, in the case that the average number of accesses has a larger value, the sys-

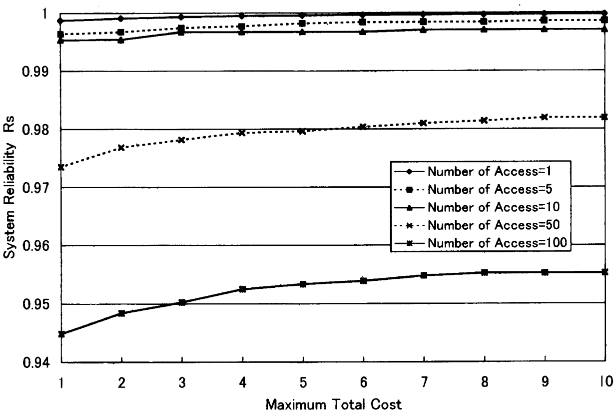


Fig. 5 A trade-off between the reliability and the cost.

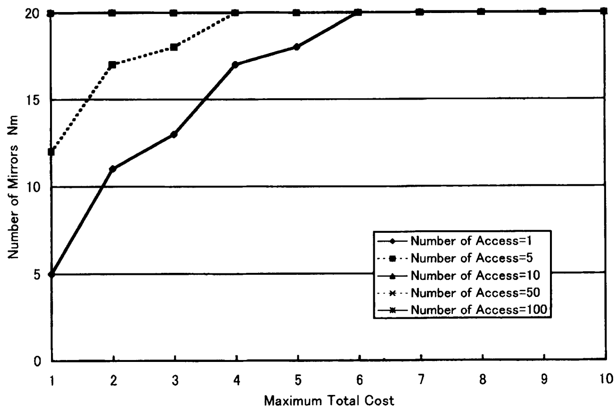


Fig. 6 Distribution characteristics of mirrors based on the cost restriction.

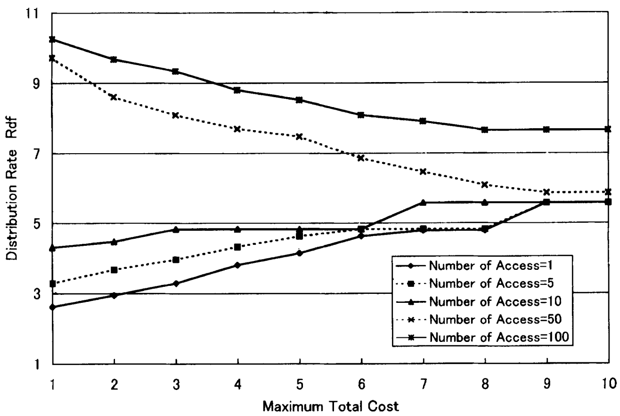


Fig. 7 Distribution characteristics of files based on the cost restriction.

tem reliability is lower as mentioned in Fig. 3. In the following figures, we may see how mirrors and files are distributed in this case.

Next, in Figs. 6 and 7, we show the distribution characteristics of mirrors and files using the total cost restriction for the different values of the average number of accesses. These figures show the number of mirrors

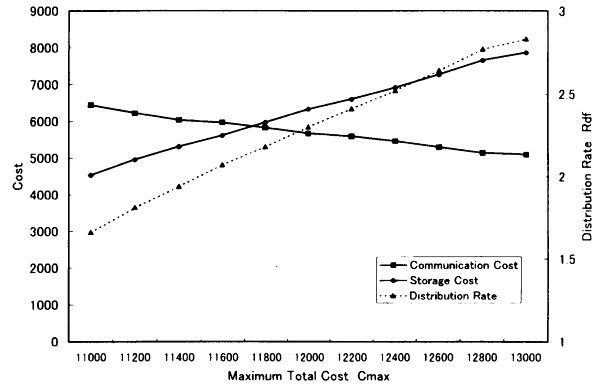


Fig. 8 A trade-off between the storage cost and the communication cost.

Nm and the distribution rate Rdf vs. the normalized value of the maximum total cost. In both figures, as the average number of accesses becomes larger, more files tend to be distributed on more mirror servers, so that it will be disinclined to forward requested files from remote servers, thus reducing the communication cost. Also, as shown in Fig. 6, when the maximum total cost is larger than 6, in all cases, the number of mirrors is equal to 20, the maximum number of mirror servers. This result means that mirror servers should be distributed as much as possible, in order to maximize the system reliability. We may imagine that the number of mirrors will increase if there are more nodes which can work as mirror servers. On the other hand, as in Fig. 7, as the total cost restriction becomes loose, the distribution rate Rdf tends to settle down to the specific value for each number of accesses individually. For example, in the case that the average number of accesses is equal to 1, 5 or 10, Rdf settles down to 5.58. This result shows that each file should be assigned to five or six nodes so that the system reliability is maximized, under the above system parameters.

In Fig. 8, we show the trade-off between the storage cost and the communication cost. This figure shows each cost and the distribution rate of files vs. the maximum total cost. Here, we set the average number of accesses to 3. As the maximum total cost becomes larger, the distribution rate Rdf and the storage cost Cs become larger, while the communication cost Ct becomes smaller. This tendency is due to the fact that files are stored on more mirror servers to improve system reliability.

4.2.3 Characteristics Using Delay Restriction

We show the characteristics of the mirror distribution using the communication delay restriction. In Fig. 9, we show the system reliability and the number of mirrors vs. the maximum communication delay Dt_{max}. Here, we set the average number of accesses to 10. As shown in this figure, when the maximum communication delay

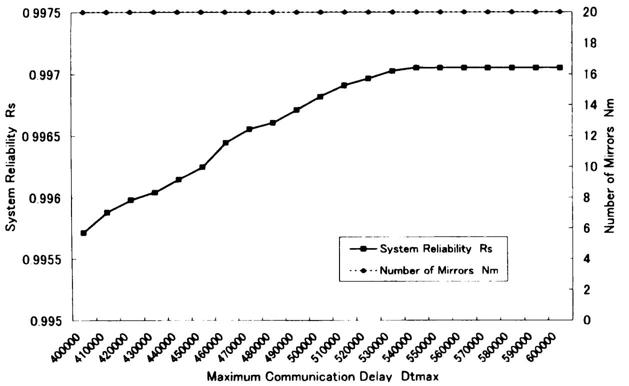


Fig. 9 Distribution characteristics of mirrors based on the delay restriction.

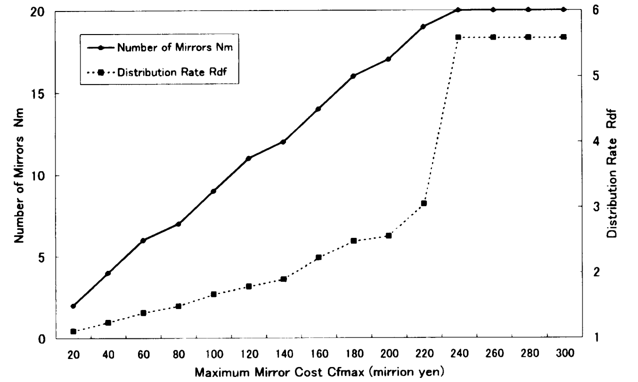


Fig. 11 Distribution characteristics of mirrors and files based on the mirror cost restriction.

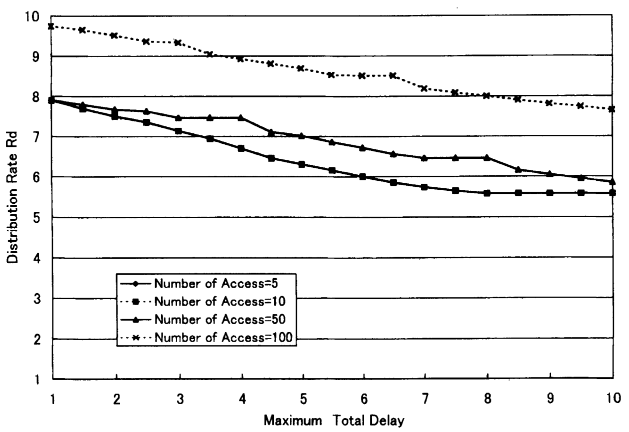


Fig. 10 Distribution characteristics of files based on the delay restriction.

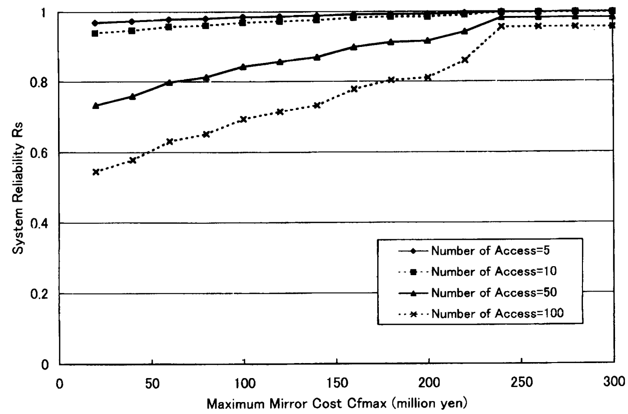


Fig. 12 A trade-off between the system reliability and the mirror cost.

is larger, the system reliability improves. On the other hand, the number of mirrors Nm is not affected by the delay restriction and always has an identical value 20, which is the maximum number of mirror servers. This result shows that we should assign as many mirrors as possible to maximize the system reliability.

Next, in Fig. 10, we show the distribution characteristics of files using the communication delay restriction for different numbers of accesses. As the communication delay restriction becomes looser, the distribution rate *Rdf* becomes smaller, while the system reliability is improved as you can see in Fig. 9. As we also see in Fig. 7., the distribution rate settles down to the specific value for each number of accesses. More files should be distributed to more nodes to reduce the communication delay when the restriction is more tight. As the restriction becomes looser, fewer files are distributed and the reliability of each node improves, since the number of files on each node decreases.

From both figures in this section, we may see quantitatively the trade-off between the reliability and the delay.

4.2.4 Characteristics Using Mirror Restriction

We show the characteristics of the mirror distribution using the restrictive conditions of the mirror cost and the capacity of the mirror servers.

In Fig. 11, we show the distribution characteristics of mirrors and files vs. the maximum mirror cost Cf_{max}. Here, we set the average number of accesses to 10. Also, in Fig. 12, we show the system reliability vs. the maximum mirror cost Cf_{max} for the different values of the average number of accesses respectively. As the maximum mirror cost increases, the number of mirrors and the distribution rate increase and the system reliability improves. In the case that the mirror cost restriction is tighter, all nodes are not able to work as mirror servers and files cannot be distributed to more nodes. However, as the mirror cost restriction becomes looser, all nodes start working as mirror servers, which improves the system reliability. This result clearly shows the trade-off between the system reliability and the mirror cost.

Next, we show the effect of the mirror capacity restriction. In Figs. 13 and 14, we show the distribution

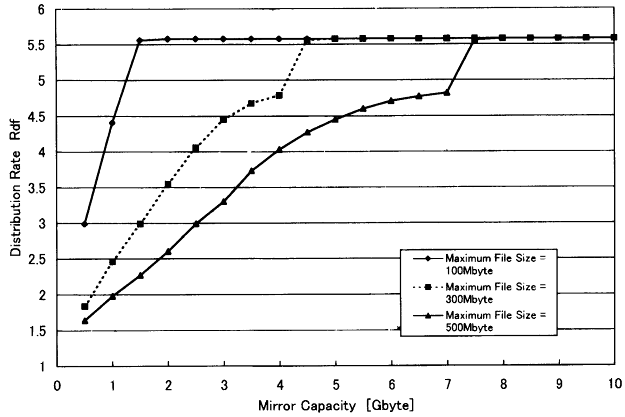


Fig. 13 Distribution characteristics of files based on the capacity restriction.

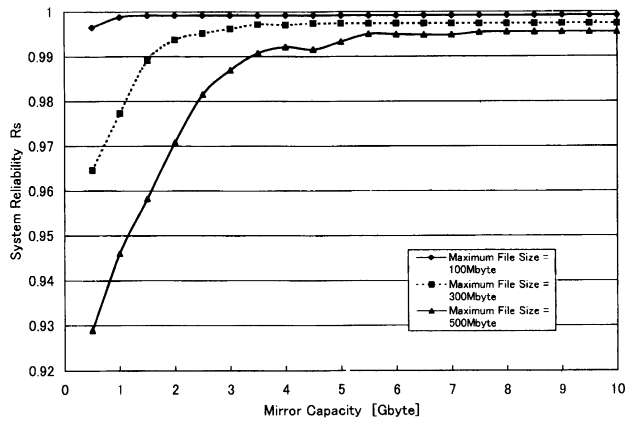


Fig. 14 System reliability characteristics based on the capacity restriction.

rate of files and the system reliability vs. the mirror capacity B_i for different values of the file size F_{ik}. Here, we set the average number of accesses to 3. As the mirror capacity becomes larger, more files tend to be distributed, and the system reliability improves. However, after the mirror capacity reaches a specific value, we cannot see the variation in the distribution rate and the system reliability. For example, in the case that the maximum size of files is 300 [Mbyte], 4.5 Gbyte is enough capacity for the optimal storage of files such that the system reliability is maximized. When the mirror capacity is larger than 4.5, the distribution rate of files has an identical value 5.58, as described in the paragraph on the characteristics using the cost restriction. For analytical simplicity, in this paper, the number of kinds of files is set to a value still smaller than that of real systems. In real systems, we must need more capacity on the nodes for the optimal storage of files.

From these numerical results, we see that there are benefits in assigning as many mirror servers as possible, while the distribution rate of files is dependent on various conditions. However, this result with regards to the number of mirror servers might be derived from our read-only assumption. The more general situation of updating files should be considered in the future.

5. Conclusions

In this paper, we have proposed the optimal mirror allocation model for the purpose of load balancing in network servers. In this model, we have formulated the optimal allocation problem of mirror servers such that the system reliability is maximized subject to the cost and the delay, as a 0–1 integer programming model. For numerical analysis, we used an approximate method, based on the Greedy Method in order to deal with more large-scale systems. From comparison results between this approximate method and exhaustive search, we have shown that this approximate method is very useful in terms of accuracy and calculation time.

As for the numerical results, we have analyzed the characteristics on load balancing using distributed mirror allocation. We have quantitatively shown the general characteristics, such as the improvement of the system reliability using the mirror distribution, the trade-off between the system reliability and the cost, the effects of the cost and the delay restriction, etc. In particular, we have obtained the followings results:

1. There are benefits in assigning as many mirror servers as possible.
2. The extent of the file distribution is considerably dependent on several conditions, such as the cost restriction, the delay restriction, etc.
3. When systems are released from the restrictive conditions, the distribution rate of files settles down to a specific value. That is, we may be able to find how many copies of each file are required to be stored in the whole systems so that the system reliability is optimized.

As a result, we can conclude that our optimal mirror allocation model is very useful in evaluating various issues in the construction of multi-media networks.

Our future work is to extend our model to a stochastic model so that we can take into account time dependent problems, such as MTTR (mean time to repair), MTBF (mean time between failure), the detour of the path, dynamic communication costs, etc. These problems should be considered to analyze the system behavior more accurately.

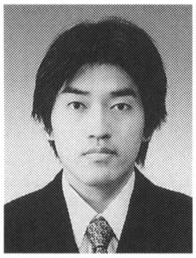
References

[1] A.K. Verma and M.T. Tamhankar, "Reliability-based optimal task-allocation in distributed-database management systems," IEEE Trans. Reliability, vol.46, no.4, Dec. 1997.

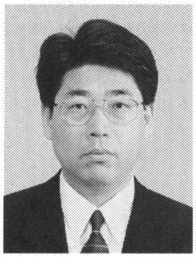
[2] S.M. Shatz and J. Wang, "Models & algorithms for reliability-oriented task-allocation in redundant distributed-computer systems," IEEE Trans. Reliability, vol.38, no.1, April 1989.

[3] S.T. Cheng, "Topological optimization of a reliable communication network," IEEE Trans. Reliability, vol.47, no.3, Sept. 1998.
[4] W.W. Chu, "Optimal file allocation in a multiple computer system," IEEE Trans. Comput., vol.C-18, no.10, pp.885–890, Oct. 1969.
[5] I. Shyu and S. Shieh, "Balancing workload and recovery load on distributed fault-tolerant VOD systems," IEEE Commun. Lett., vol.2, no.10, Oct. 1998.
[6] J.F. Lawless, Statistical Models and Methods for Lifetime Data, John Wiley & Sons, 1982.
[7] C.C. Bisdikian and B.V. Patel, "Issues on movie allocation in distributed video-on-demand systems," IEEE Proc. ICC'95, pp.250–255, 1995.
[8] C.C. Bisdikian and B.V. Patel, "Cost-based program allocation for distributed multimedia-on-demand systems," IEEE Multimedia, pp.62–72, Fall 1996.
[9] D. Deloddere, W. Verbiest, and H. Verhille, "Interactive video on demand," IEEE Commun. Mag., vol.32, no.5, pp.82–88, May 1994.
[10] V.O. K. Li, W. Liao, X. Qiu, and E.W. M. Wong, "Performance model of interactive video-on-demand systems," IEEE J. Sel. Areas Commun., vol.14, no.6, Aug. 1996.
[11] S.A. Barnett and G.J. Anido, "A cost comparison of distributed and centralized approaches to video-on-demand," IEEE J. Sel. Areas Commun., vol.14, no.6, Aug. 1996.
[12] A. Nakaniwa, H. Ebara, and H. Okada, "File allocation designs for distributed multimedia information networks," IEICE Trans. Commun., vol.E81-B, no.8, pp.1647–1655, Aug. 1998.
[13] A. Nakaniwa, M. Onishi, H. Ebara, and H. Okada, "File allocation in distributed multimedia information networks," Proc. IEEE GLOBECOM'98, Nov. 1998.
[14] A. Nakaniwa, M. Onishi, H. Ebara, and H. Okada, "Sensitivity analysis of file allocation for distributed information networks," Proc. IEEE ICC'99, June 1999.
[15] A. Nakaniwa, J. Takahashi, H. Ebara, and H. Okada, "Reliability-based optimal allocation of distributed mirror servers for internet," Proc. GLOBECOM2000, Nov. 2000.
[16] T. Yamamoto, D. Nakajima, H. Ohsawa, and Y. Yasuda, "A study on cache layout in hierarchical network," IEICE General Conference, 1999.
[17] M. Shiina, Y. Tajima, H. Morikawa, and T. Aoyama, "An adaptive network web caching system," IEICE General Conference, 1999.

Akiko Nakaniwa was born in 1975. She received B.S. and M.S. degrees from Kansai University, Japan, in 1997 and 1999, respectively. Currently she is studying in the graduate school of Kansai University. Her research interest includes the optimization design of distributed networks. She is a student member of IEEE.



Jun Takahashi was born in 1977. He received B.S. degree from Kansai University, Japan, in 2000. Currently he is studying in the graduate school of Kansai University. His research interest is the design of reliable networks. He is a student member of IEEE.



Hiroyuki Ebara was born in 1958. He received the B.S., M.S., and Ph.D. degrees in communication engineering from Osaka University, Osaka, Japan, in 1982, 1984, and 1987, respectively. In 1987 he became Assistant Professor of Osaka University. Since 1994 he has been with Kansai University, where he is currently Associate Professor. His main research interests are in computational geometry, combinatorial optimization, and parallel computing. Dr. Ebara is a member of IEEE, ACM, and SIAM.

Hiromi Okada received the B.S., M.S. and Ph.D. degrees on Communications Engineering, all from Osaka University, Japan, in 1970, 1972, and 1975, respectively. From 1975 to 1983, he was an assistant professor of Osaka university. From 1983 to 1987, he was an associate professor of Kobe university. From 1987 to 1996, He was with Osaka University as an associate professor. Since April 1996, he has been with the department of Electronics Engineering, Faculty of Engineering, Kansai University as a professor. His current research interest includes ATM multicasting networks, wireless ATM networks, ad-hoc wireless networks, performance evaluation and optimization of distributed information networks. He is the author of several books entitled "Information Networks" (in Japanese, Baifukan Pub.), "Introduction of Computer Systems" (in Japanese Shokodo Pbu.) and so on. He is a member of IEEE, IPS (Information Processing Society) Japan.