Fault tolerant packet-switched network design and Its sensitivity

Tokumi Yokohira
Okayama University

Masashi Sugano
Mita Industrial Corporation

Takeshi Nishida
NEC Corporation

Hideo Miyahara
Osaka University

Fault Tolerant Packet-Switched Network Design and Its Sensitivity

Tokumi Yokohira, Member IEEE
Okayama University, Okayama
Masashi Sugano
Mita Industrial Corporation, Osaka
Takeshi Nishida, Member IEEE
NEC Corporation, Kawasaki
Hideo Miyahara, Member IEEE
Osaka University, Osaka

Key Words — **Packet-switched network, Sensitivity, Performance, Capacity assignment algorithm.**

Reader Aids —
Purpose: Examine sensitivities of networks designed with our previous methods
Special math needed for explanations: Probability
Special math needed to use results: Same
Results useful to: Network designers and reliability engineers

Summary & Conclusions — **Reliability and performance for telecommunication networks have been traditionally investigated separately in spite of their close relation. A design method integrating them for a reliable packet switched network, called a proofing method, was discussed in previous papers. This paper first presents the proofing method in detail. Then two heuristic design approaches (max-average, max-delay-link) for optimizing network cost in the proofing method are described. In order to verify their effectiveness and applicability, they are compared numerically for three example network topologies. Finally, the sensitivity of these two methods is examined with respect to changes in traffic demand and in link reliability.**

Numerical results show that the max-delay-link method provides a lower minimum network-cost than does the max-average method, for both a small and a large example network. The answers obtained by these two methods are not highly sensitive to changes either in traffic demand or in link reliability. Thus a network designed by these two methods is robust to a system change which is not considered at a design stage. The max-average method is superior to the max-delay-link method in terms of the sensitivity.

Many other sources of failures must be considered in our failure model, eg, node and software faults. Only statistically independent failures are considered. Statistical dependence effects must be included to make the model more realistic. The max-average and max-delay-link methods cannot prevent a state where there are no routes between a particular source and destination node pair. To cope with this, a topological design method must be added to these methods. The computational complexity of these methods needs to be clarified to identify how large a practical problem can be solved using them.

1. INTRODUCTION

All enterprises have become dependent upon networks or networked computing applications; thus the loss of network services is a serious outage, often resulting in unacceptable delays, loss of revenue, or temporary disruption. To obviate loss of network services, communication networks should be designed so that they remain operational and maintain as high performance level as feasible, even in presence of network component failure. Clearly, designers need to consider network performance and operability parameters in their designs. Researchers, however, have traditionally taken two distinct approaches based on different measures in designing reliable networks:

1. Reliability measures [1-3], eg, connectivity, probability of successful communication between any pair of nodes
2. Performance measures [4-6], eg, delay, throughput.

A design approach based only on reliability considerations does not necessarily avoid network performance degradation, even if the network is guaranteed to be connected during network element failure. In approach 2, networks are designed to optimize the performance measure without considering possible network failures. Consequently, the network performance can degrade drastically at an element failure. Neither approach is enough for a reliable network design. This incompleteness comes from the lack of a unified metric for specifying both reliability and performance in the traditional network design methods.

In recent years, efforts have been made to integrate performance and reliability in order to overcome the drawback in the traditional evaluation methods for networks [7-10]. The network systems are modeled by focusing on a change in performance levels in response to a system state change caused by failures. For this purpose, a new performance metric, performance-related reliability, is defined. It is a weighted sum of the performance in each network state; the weight is the probability of a state occurrence. Although all research efforts are devoted to modeling techniques, little work is done on an application of a performance-related reliability, eg, network design, bottleneck analysis, or sensitivity analysis.

For computer systems, especially multiprocessor systems, a modeling approach to integrate performance and reliability has been discussed in [13-15]. This approach is based on combining Markov reliability models with existing system performance models. Using this approach, the effect of reliability on performance has been examined [16], and sensitivity and bottleneck analysis have been performed [17,18] for multiprocessor systems.

In [11,12], a design method for a reliable packet switched network, referred to as a proofing method, uses the performance related reliability modeling approach cited above. The proofing method assigns to each link in advance a redundant capacity with a constraint of suboptimal network cost, so that it can accommodate any traffic detoured from failed links, even in the presence of any network link failure. Moreover, this method designs standby routes to which no capacity is assigned by a traditional performance-oriented method [6]. The method assures that —

0018-9529/91/1000-0452$01.00©1991 IEEE

- an average end-to-end packet delay remains under a designed value even in case of failures,
- the probability of successful communication between any pair of nodes is enhanced.

This paper first presents the proofing method. Then, two heuristic methods (max-average and max-delay-link) for optimizing network cost are described. In order to verify their effectiveness and applicability, some numerical comparisons are made for three network topologies (in [12] they are compared in a simple network topology). Finally, the sensitivity of these two approaches is examined from the following point of view.

Data required as input to a network design, eg, forecast of future demand, are often inaccurate. The design technique should therefore account for the accuracy of available data. Furthermore, network traffic volume and size evolve in response to future customer needs and to changes in networking technology. It is therefore desirable that network performance at the operational stage be relatively insensitive to input data such as future demand forecasts and network evolution. That is, it is desirable that solutions generated by design techniques be relatively insensitive to input data. Following this rationale, the design sensitivity to variation of input data is examined by changing the predicted probability of link failure, and by increasing the network traffic over the predicted value. The resulting analysis shows relative insensitivity of solutions generated by the two design methods to input data.

Section 2 defines a network model and states the assumptions used in this paper. Section 3 details the proofing method. Section 4 shows, by a small network, how network performance degrades as network failures occur — if networks are designed without considering failures. The remaining part of section 4 compares the max-average and max-delay-link methods with respect to network cost for both a small and a large network model. Section 5 discusses a sensitivity analysis to the proposed methods.

2. NETWORK MODEL

Notation

m,n	number of links, nodes in a network
C_i	capacity of link i
d_i	cost per unit capacity of link i
D_i	cost of link i: $D_i = d_i C_i$
G	total network cost: G is the sum of all D_i, $i=1,\ldots,m$
γ_{uv}	mean of the traffic pattern from source node u to destination node v; $\gamma_{uu}=0$
γ	mean total traffic of the network
μ	reciprocal of mean packet length
T_i	packet delay in link i
λ_i	average flow on link i
X_i	state of link i; 0 = operable, 1 = failed
p_i	failure probability of link i; see assumption 5
T	network delay
$T_{\max}$	maximum allowable T
S'_y	network state: $(X_m, X_{m-1},\ldots,X_1)$ such that $y = \sum_{i=1}^{m} 2^{i-1} X_i$
S_j	network state with special ordering, j; see (2) and its following paragraph for the ordering definition
M	a number large enough such that $\{S_j: j = 0,1,\ldots,M-1\}$ covers a high portion of the state space; see section 3.2
$T(j)$	network delay for state S_j
P_j	$\Pr\{S_j\}$
$\mathrm{CAP}(i,j)$	capacity of link i in network state, S_j
α_{ij}	bias value (weight factor, importance level) for link i in state S_j
α	bias value, independent of i,j; see sections 3.3 & 3.4
α^*	the value of α that minimizes network cost, in a particular situation

Other, standard notation is given in ''Information for Readers & Authors'' at the rear of each issue.

Nomenclature, Acronyms, Definitions

Network delay: Mean total time a packet spends in the network
Network state: The m-tuple of link states, from link m to link 1
Unreachable traffic: For a particular network state, there is no route from a source node to a destination node, due to link failures; see procedure P-2 in section 3.2
End-to-end traffic requirement: Mean of the traffic pattern from a particular source node to a particular destination node; see γ_{uv}
CFA algorithm: Algorithm for calculating capacity and flow assignment [4].

Assumptions

1. A network is constructed with m links and n nodes, and the network topology is given. Only links can fail; the nodes can not fail. If a link fails, then its capacity becomes zero. Link failures are mutually statistically independent. The statistical independence assumption of failure sometimes leads to incorrect results [19]. Thus the effect of statistical dependence will be treated in our future work.

2a. The cost of a link is directly proportional to its capacity: $D_i = d_i C_i$, $i=1,\ldots,m$.

2b. Since the capacity and flow assignment are considered only for each link, the cost of each node does not influence the optimization problem. Without loss of generality, the cost of all nodes is therefore set to zero.

2c. The network construction cost is the sum of the costs of each link.

3a. The traffic pattern from source node u to destination node v obeys a Poisson process with a mean γ_{uv}. The mean total traffic of the network is, therefore, the sum of γ_{uv} over all u,v $(u,v=1,\ldots,n)$.

3b. The packet-length distribution is exponential with mean $1/\mu$.

4. There is a fixed routing scheme, ie, all packets from source node u to destination node v are routed on a fixed path, and another fixed route is selected when a link failure occurs.

5. The probability of link failure is less than 1/2; this is based on recent high-reliability media technology.

6. In procedure P-2, the unreachable traffic is zero, ie, the unreachable traffic does not enter the network. ☐

From assumption 3 and the Kleinrock independence assumption [4] —

$$T_i = 1/(\mu C_i - \lambda_i).$$

Thus the network delay is [4]:

$$T = \frac{1}{\gamma} \sum_{i=1}^{m} T_i \lambda_i. \tag{1}$$

Examples of network states are:

S_0' represents (0,0,...,0) — no links are failed
S_3' represents (0,..., 0,1,1) — links 2 & 1 are failed

From assumption 5, the most probable network state is S_0'. Because —

$$P_y' = \prod_{i=1}^{m} \{p_i X_i + (1-p_i)(1-X_i)\}. \tag{2}$$

Without loss of generality, the S_y' are reordered in decreasing order with respect to P_y', and new state variable is S_j ($j = 0,\ldots,2^m-1$). S_0 corresponds to S_0' because $p_i < 1/2$ (from assumption 5).

3. PROOFING METHOD

3.1 *Network Design Goal*

The classical capacity and flow assignment problem is formulated as:

Given:

Network topology
End-to-end traffic requirement, γ_{uv}
The maximum network delay, $T_{\max}$

Minimize:

Network cost, *D*

With respect to:

Link capacities, C_i, $i = m, m-1, \ldots, 1$
Link flow, λ_i, $i = m, m-1, \ldots, 1$

Constraint:

$T \le T_{\max}$. ☐

This problem is solved by the CFA algorithm [4]; it provides a solution for S_0 (no failed links). If any links fail, however, the *constraint* is not always satisfied, because the traffic routed on the failed links goes to the other links, and this can cause congestion. A highly reliable network must operate with low network delay, even if this network is in any failure state. Our design goal is to prevent drastic performance degradation even in a failure state. Therefore the following additional constraint is needed.

New constraint:

$$T(j) \le T_{\max}, \text{ for all } j \ (0 \le j \le M-1). \tag{3}$$

3.2 *Proofing Method*

The link capacity is determined with procedures, P-1 to P-3.

P-1. Generate S_j, $j=0,\ldots,M-1$ by using algorithm ORDER [9].

P-2. Obtain CAP(i,j) by applying the CFA algorithm to the network in state S_j, so that the network cost is minimized under the *new constraint*, T(j) $\le T_{\max}$; see assumption 6

P-3. Calculate the capacity of link *i*:

$$C_i = \sum_{j=0}^{M-1} \alpha_{ij} P_j \, \text{CAP}(i,j) \tag{4}$$ ☐

A crucial point in this proofing method is how to determine *M* and the α_{ij} (bias values, or link *i* importance level in S_j). The value of *M* depends on the designer's decision: To what degree must the network be robust? A large *M* results in a very robust network; a small *M* results in a very weak network. Deriving an algorithm to determine the value of *M* is beyond the scope of this paper, but a value of *M* such that:

$$\sum_{j=0}^{M-1} P_j \ge 0.99$$

ie, a high proportion of the state space is covered, may be reasonable. It is difficult to determine the bias values; they are a key design parameter in the sense that they affect two tradeoff measures: Network cost and performance. Refs [11,12] propose two heuristic methods (max-average and max-delay-link) to determine the bias values. They are explained in sections 3.3 & 3.4.

Nomenclature, Acronyms, Definitions

DT product: product of the packet delay and the flow for each link
NA method: A design method using the CFA algorithm, viz, no-augmentation
MA method: max-average method
MDL method: max-delay-link method.

3.3 *Max-Average Method*

This method treats link *i* such that CAP$(i,0) = 0$; ie, a capacity for link *i* is not assigned in state S_0. Such a link has zero-capacity. A unique bias value is chosen for the links other than zero-capacity links. Procedure P-3 is replaced by procedures: MA-3.1, MA-3.2, MA-3.3.

MA-3.1. For all i such that $\mathrm{CAP}(i,0)=0$, determine —

$$\alpha_{ij} = \begin{cases} 1/P_j, & \mathrm{CAP}(i,j) = C_{\max}(i) \\ 0, & \text{otherwise.} \end{cases}$$

$$C_{\max}(i) \equiv \max\{\mathrm{CAP}(i,0),\mathrm{CAP}(i,1),\ldots,\mathrm{CAP}(i,M-1)\}.$$

To determine α_{ij} by this equation is equivalent to $C_i = C_{\max}(i)$; see (4).

MA-3.2. For all i such that $\mathrm{CAP}(i,0) \neq 0$ —

$$C_i = \sum_{j=0}^{M-1} \alpha\, P_j\, \mathrm{CAP}(i,j) \tag{5}$$

ie, $\alpha_{ij} = \alpha$

MA-3.3. For all states, assign the flow of each link using C_i (by the flow-assignment algorithm [4]), and calculate $T(j)$. If all $T(j)$ satisfy the *new constraint*, then decrease α to $\alpha - \Delta\alpha$ ($\Delta\alpha$ is sufficiently small) and go to MA-3.2; else Stop. □

The reason that maximum capacity is assigned to the zero-capacity links in (MA-3.1) is:

The P_0 is so large (see assumption 5: $p_i < 1/2$) that the other states have little effect in (4). Consequently, if a fixed bias value (independent of state and link number) is chosen for all links, then sufficient capacity for accommodating the detoured traffic is not assigned to the zero-capacity links.

In the MA method, the network cost decreases linearly as a smaller value of α is chosen. Therefore, α^* is the minimum α for which the *new constraint* is satisfied. In order to shorten computation time for the algorithm, the initial value of α should be chosen carefully; $\alpha_{\max}$ is the best initial value.

$$\alpha_{\max} = \max\{\alpha_{\max}(1),\alpha_{\max}(2),\ldots, \alpha_{\max}(m)\},$$

$$\alpha_{\max}(i) \equiv C_{\max}(i) \Big/ \sum_{j=0}^{M-1} P_j\, \mathrm{CAP}(i,j)$$

For $\alpha \geq \alpha_{\max}$, every link is assigned the capacity which is at least the maximum link capacity among the M possible states. Thus the *new constraint* is satisfied.

The MA method is simple and therefore reduces the computation time. This, however, sacrifices the network-cost optimization, largely because the algorithm does not consider the bias value of each link in a specific state (a state where there is a bottleneck link, see section 3.4). The bias value level can be represented in various ways. Section 3.4 presents an alternative algorithm for considering the bias value.

3.4 *Max-Delay-Link Method*

From (1), the DT product determines the network delay when the link DT product is much larger than the other DT products and m is small. In such a situation and when the *new constraint* is not satisfied, the link is a bottleneck (for the network delay). To satisfy the *new constraint*, the capacity of a bottleneck link should be increased. This MDL method is based on that idea [12]. Procedure P-3 is replaced by procedures: MDL-3.0, MDL-3.1, MDL-3.2, MDL-3.3.

MDL-3.0. Select $\alpha_{\max}$ as the initial value of α. Put all links into a marked-link list, viz, the list of no-bottleneck links.

MDL-3.1. Use the current value of α. Then —

$$C_i = \sum_{j=0}^{M-1} \alpha\, P_j\, \mathrm{CAP}(i,j),$$

for i in marked-link list

$$C_i = C_{\max}(i), \text{ otherwise.} \tag{6}$$

If there are no links in the marked-link list, then Stop.

MDL-3.2. For all states, assign the flow (of each link) using C_i, and calculate $T(j)$. If all $T(j)$ satisfy the *new constraint*, then decrease α to $\alpha - \Delta\alpha$ and go to MDL-3.1.

MDL-3.3. Find k such that S_k is the state where the network delay is the maximum among the M most probable states: $T(k) \geq T(j)$, for $j \neq k$. In S_k, calculate the DT product of all links in the marked-link list. Delete (from the marked-link list) the link for which DT product is the largest among all links. Go to MDL-3.1. □

Since, for all α in $[0, \alpha_{\max}]$, links are assigned the maximum capacity until the *new constraint* is satisfied, α^* is in $[0, \alpha_{\max}]$.

4. EFFECTIVENESS OF THE PROOFING METHOD

Section 4.1 shows, by means of a sample network —

- how performance degrades as links fail, if the network is designed by assuming no failures;
- that MA & MDL methods can eliminate such performance degradation.

Section 4.2 evaluates the MA & MDL methods and compares them with respect to network cost.

Example

1. In case of network failure, routes are changed to detour around the failed links and to minimize the network delay.
2. Such routes are determined using the flow-deviation algorithm [6].
3. Figure 1 shows the topology of the network.
4. Table 1 shows the end-to-end traffic requirement γ_{uv}, unit cost d_i, and failure probability p_i.
5. Table 2 shows the network state S_j ($j=0,\ldots,24$), the failed link(s) in S_j, and P_j.
6. The mean packet length $1/\mu$ is 10 kbit; $T_{\max} = 1$ sec.

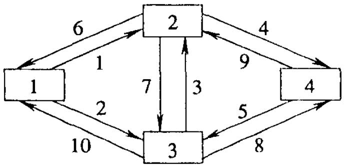


Figure 1. Model for Sample-Network

TABLE 1
End-to-end Traffic Requirements and Link Attributes

		Destination			
		1	2	3	4
Source	1	0.00	0.02	0.03	0.04
	2	0.02	0.00	0.02	0.03
	3	0.03	0.01	0.00	0.03
	4	0.03	0.04	0.01	0.00

(a) End-to-end traffic requirements (packets&sec.)

Link No.	Failure Prob. (p_i)	Unit Cost (d_i)
1	0.045	0.045
2	0.040	2.5
3	0.035	3.0
4	0.030	1.0
5	0.025	3.5
6	0.020	1.5
7	0.015	2.5
8	0.0075	2.0
9	0.0075	1.3
10	0.0050	2.3

(b) Link attributes

4.1 *Performance Comparison of the Proofing and Classical Methods*

The capacity assignment obtained by the NA method is shown in table 3. No capacity is assigned to links 2, 3, 5; ie, those links are redundant for the minimum cost design in the NA method. For $M = 20$, table 3 also shows the capacity assignment which minimizes the network cost in the MA method ($\alpha^* = 1.375$) and the MDL method ($\alpha^* = 0.867$).

The network delay in S_j ($j = 0, \ldots, 19$) by the NA, MA, MDL methods is shown in figure 2. From tables 2 & 3, the states S_0, S_1, S_4, S_6 – S_{10}, S_{14}, S_{19} can occur (the other states can not occur) in the NA method because no capacity is assigned to links 2, 3, 5. In figure 2, the network delay in the states which can occur is shown only for the NA method. In all three methods, the network delay in the states where unreachable traffic exists is plotted with a box mark, "□".

From figure 2, unreachable traffic exists in 5 states (S_1, S_7, S_9, S_{14}, S_{19}) in the NA method. The network delay in S_7 & S_9 is smaller than $T_{\max}$ because total traffic becomes light

TABLE 2
Network States in Sample Network

State	Probability	Failed Link
0	0.78964	*
1	0.03721	1
2	0.03290	2
3	0.02864	3
4	0.02442	4
5	0.02025	5
6	0.01612	6
7	0.01202	7
8	0.00798	8
9	0.00598	9
10	0.00397	10
11	0.00155	1,2
12	0.00135	1,3
13	0.00119	2,3
14	0.00115	1,4
15	0.00102	2,4
16	0.00095	1,5
17	0.00089	3,4
18	0.00084	2,5
19	0.00076	1,6
20	0.00073	3,5
21	0.00067	2,6
22	0.00063	4,5
23	0.00058	3,6
24	0.00057	1,7

TABLE 3
Capacity Assignment in No-Augumentation, Max-average and Max-delay-link Methods

Link No.	No-augumentation	Max-average ($\alpha = 1.375$)	Max-delay-link ($\alpha = 0.867$)
1	31.554	40.801	25.727
2	0.000	18.641	18.641
3	0.000	17.920	0.093
4	17.843	23.413	14.763
5	0.000	13.926	13.926
6	12.330	16.871	10.638
7	10.638	13.796	16.887
8	8.236	11.889	19.141
9	16.873	23.211	14.636
10	8.945	12.103	14.932

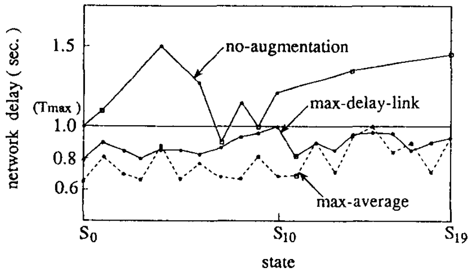


Figure 2. Network Delay in Each State

as a result of removing unreachable traffic. Even if there is no unreachable traffic, the network delay exceeds $T_{\max}$ in such states (except S_0), eg, in S_4.

Performance degrades as links fail, when the network is designed without allowing for link failure.

On the other hand, the network delay in all states S_0 – S_{19} does not exceed $T_{\max}$ in the MA & MDL methods, and there is no unreachable traffic in any states except S_{11}. This situation (unreachable traffic in S_{11}) cannot be prevented using any capacity assignment method, since S_{11} is a state where links 1 & 2 fail. A topological design, which is beyond the scope of this paper, is needed to avoid this situation.

4.2 *Cost Comparison of the MA & MDL Methods*

Figure 3 shows the ratio of [network cost for the MA or MDL method] to [network cost for the NA method]; α is a parameter. The network cost for the MA method decreases linearly as α decreases. When α is smaller than a threshold value (in this example, 1.375), the *new constraint* is not satisfied; thus the network cost is plotted with a dashed line. For the MA method, $\alpha^* = 1.375$; the minimum network cost is 2.448.

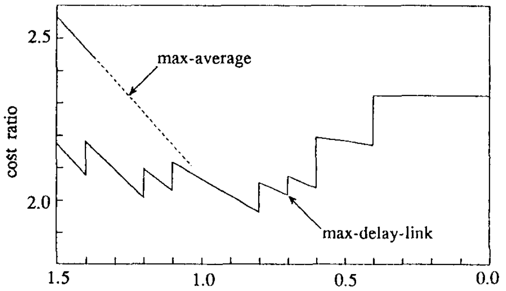


Figure 3. Cost Ratio in Max-Average & Max-Delay-Link Methods

Although the network cost for the MDL method also decreases linearly as α decreases, the network cost drastically increases at some value of α. This is because a bottleneck link appears as α decreases, and the maximum capacity is assigned to the bottleneck link. The rate of network-cost decrease slows as α decreases, because the number of links in the marked-link list decreases as α decreases; see MDL-3.1.

For the small (and large) value regions of α, the network-cost increase by assigning maximum capacity is, therefore, larger (or smaller) than the cost decrease by decreasing the value of α, respectively.

For the MDL method, $\alpha^* = 0.867$ which is in $[0, \alpha_{\max}]$; the minimum network cost is 1.999.

Figure 4 shows the minimum network cost obtained by the MA & MDL methods for various values of M. There is a close relation between the value of M and the minimum network cost. The cumulative probability (sum of the probabilities for each state) is also shown in figure 4. For $M = 20$, the cumulative probability = 0.99. The cost drastically increases at $M = 2$ & $M = 10$, because when there is a new failure scenario, some links whose capacities were not assigned do become necessary (as alternate paths) and thus capacities are assigned to them. For example, when $M = 2$ (S_0 & S_1 are considered), then link 2 becomes necessary for node 1, because link 1 fails in S_1.

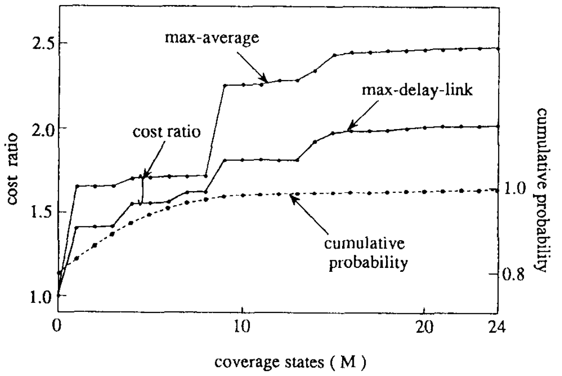


Figure 4. Cumulative Probability and Cost with Changine the Number of Most Probable States

Two large networks (I & II) are defined in figure 5. Figure 6 illustrates their minimum network cost and their cumulative probability for various values of M. Table 4 shows their end-to-end traffic requirements γ_{uv}, $(u,v = 1,\ldots,8)$, unit cost d_i, and failure probability p_i $(i = 1,\ldots,28)$. Figures 3, 4, 6 show that the MDL method provides lower network-cost design than the MA method.

The computational complexity of the MA & MDL methods is an open issue for us, because it is hard to estimate the number of flow assignments, which depends on network topology and initial-flow for the flow-assignment algorithm in MA-3.3 & MDL-3.2. The computation time for the cost ratio for each value of M in figures 4 & 6 is 2 to 5 minutes (including I/O time) on a 7.5 MIPS workstation.

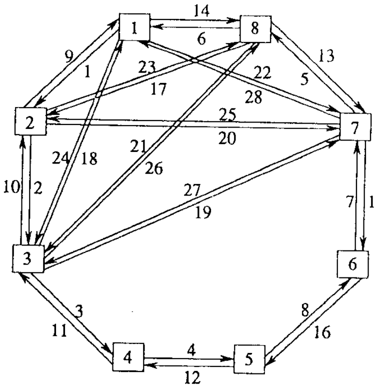
5. SENSITIVITY ANALYSIS

Two sensitivities are investigated:

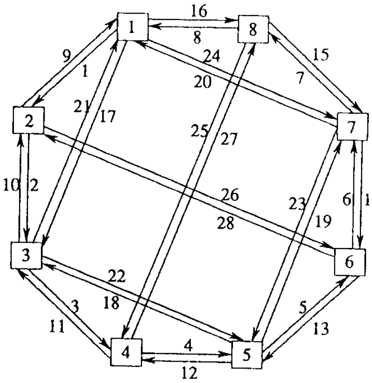
S-1. How does a network performance change when a network state goes into S_j for $j \geq M$? This situation can occur when the predicted value of p_i is inaccurate.

S-2. How does network performance change when the actual end-to-end traffic requirement at the network operational stage is larger than the predicted requirement? This situation can occur when the predicted end-to-end traffic requirement at the network design stage is underestimated or a user requirement increases as time being. □

Figure 7a shows the network delay change of the example small network for states up to S_{50}; that network is designed for



(a) Large network I



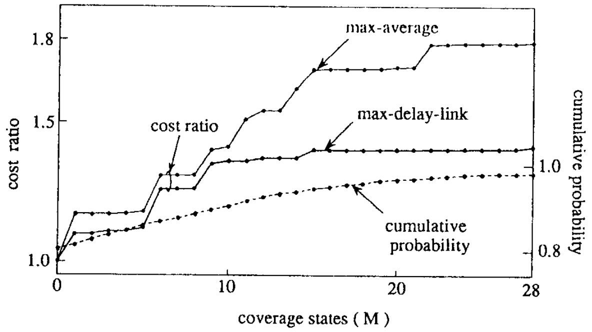
(b) Large network II

Figure 5. Models for Large Networks

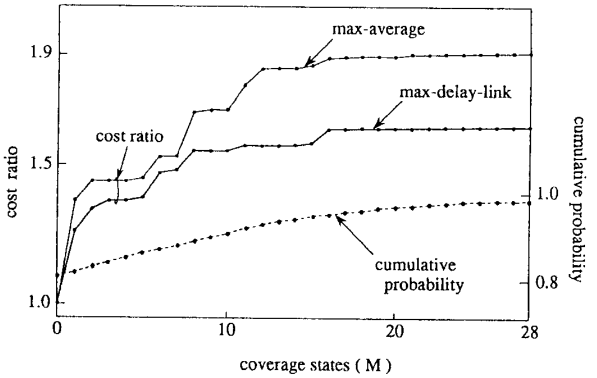
$M=20$. Similarly, figures 7b & 7c show the network delay change of large networks I & II for states up to S_{59}; those networks are designed for $M=29$. In figure 7, although the network delay exceeds $T_{\max}$ in some states in the right-hand region beyond the vertical dashed line (the state boundary in the design stage), the difference between $T_{\max}$ and the network delay in those states is 0.2 seconds at most, viz, a 20% performance degradation. Therefore the networks designed with the MA & MDL methods are robust for situation S-1.

Figure 8 shows the network delay for states up to S_{M-1} when all the γ_{uv} increase 10% and 20%, compared to the predicted volume. Though the network delay exceeds $T_{\max}$ in some states, the difference between $T_{\max}$ and the network delay in those states is 0.1 seconds at most, viz, a 10% performance degradation. Therefore the networks designed with the MA & MDL methods are robust for situation S-2.

Figures 7 & 8 show that the number of states where the network delay exceeds $T_{\max}$ in the MA method is smaller than the number in the MDL method. This is because more redundant capacity is assigned to each link in the MA method. The MA method can be superior to the MDL method in terms of sensitivities to situations S-1 & S-2.



(a) Cumulative probability and cost with changing the number of most probable states in large network I



(b) Cumulative probability and cost with changing the number of most probable states in large network II

Figure 6. Cumulative Probability and Cost with Changing the Number of Most Probable States in Large Networks

TABLE 4
End-to-end Traffic Requirements and Link Attributes in Large Networks

		Destination							
		1	2	3	4	5	6	7	8
Source	1	0.00	0.06	0.05	0.04	0.02	0.02	0.04	0.05
	2	0.05	0.00	0.06	0.02	0.04	0.03	0.03	0.04
	3	0.05	0.06.	0.00	0.03	0.02	0.02	0.04	0.07
	4	0.03	0.03	0.02	0.00	0.01	0.01	0.02	0.02
	5	0.03	0.02	0.01	0.01	0.00	0.01	0.03	0.02
	6	0.03	0.03	0.04	0.03	0.02	0.00	0.04	0.02
	7	0.04	0.05	0.04	0.03	0.02	0.04	0.00	0.06
	8	0.07	0.06	0.07	0.04	0.03	0.02	0.04	0.00

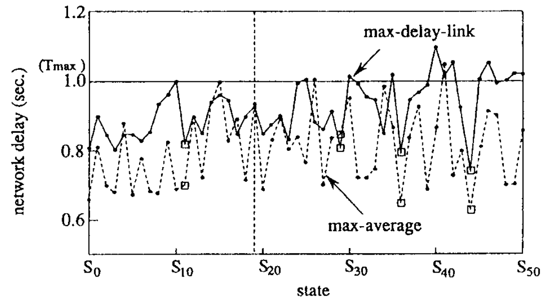
(a) End-to-end traffic requirements (packets&sec.)

(continued)

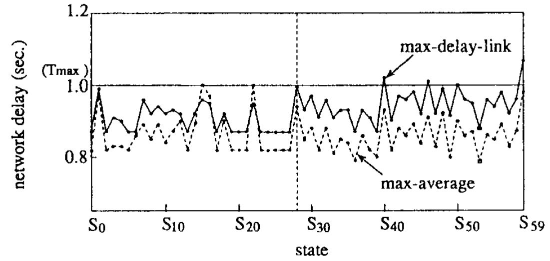
TABLE 4 (*continued*)

Link No.	Failure prob. (p_i)	Unit cost (d_i) A	B
1	0.0145	1.5	0.5
2	0.014	3.5	0.7
3	0.0135	1.2	0.6
4	0.013	0.6	1.0
5	0.0125	2.5	1.2
6	0.012	1.6	1.8
7	0.0115	0.9	1.1
8	0.011	0.5	0.6
9	0.0105	2.0	1.1
10	0.01	2.4	1.2
11	0.0095	1.0	1.0
12	0.009	0.7	0.9
13	0.0085	4.0	2.3
14	0.008	1.5	3.0
15	0.0075	1.0	0.9
16	0.007	0.8	1.0
17	0.0065	4.0	1.0
18	0.006	2.4	2.0
19	0.0055	4.0	1.3
20	0.005	3.8	1.7
21	0.0045	3.3	0.9
22	0.004	2.0	1.8
23	0.0035	2.8	1.0
24	0.003	3.5	1.5
25	0.0025	3.0	1.3
26	0.002	2.5	1.2
27	0.0015	3.0	1.8
28	0.001	2.4	1.9

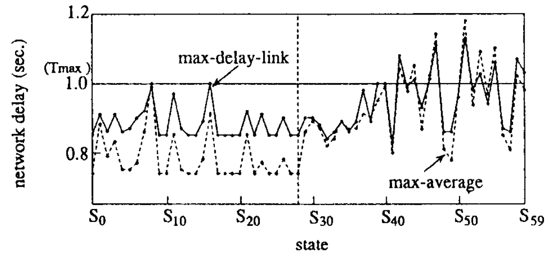
(b) Link Attribute



(a) Small network



(b) Large network I



(c) Large network II

Figure 7. Sensitivity to State

REFERENCES

[1] R. Wilkov, "Analysis and design of reliable computer networks", *IEEE Trans. Communications,* vol COM-20, 1972 Mar, pp 660–678.

[2] H. Frank, "Survivability analysis of command and control communication networks, Part 1", *IEEE Trans. Communications,* vol COM-22, 1974 May, pp 589–595.

[3] T. G. Robertazzi, P. E. Sarachik, "Self-organizing communication networks", *IEEE Communications Mag.*, vol 24, num 1, 1986, pp 28–33.

[4] L. Kleinrock, *Queueing Systems,* vol 2, 1976; John Wiley & Sons.

[5] D. G. Cantor, M. Gerla, "The optimal routing of messages in a computer network via mathematical programming", *Proc. IEEE Computer Conf.*, 1972, pp 167–170; San Francisco, California.

[6] L. Fratta, M. Gerla, L. Kleinrock, "The flow deviation method - An approach to store and forward communication network design", *Networks,* num 3, 1973, pp 97–133.

[7] B. L. Deuermeyer, "A new approach for network reliability analysis", *IEEE Trans. Reliability,* vol R-31, 1981 Oct, pp 350–358.

[8] D. F. Larzaroiu, E. Staicut, "Congestion-reliability-availability relationship in packet-switching computer networks", *IEEE Trans. Reliability,* vol R-32, 1983 Oct, pp 354–357.

[9] V. O. Li, J. A. Silvester, "Performance analysis of networks with unreliable components", *IEEE Trans. Communications,* vol COM-32, 1984 Oct, pp 1105–1110.

[10] K. K. Aggarwal, "Integration of reliability and capacity in performance measure of a telecommunication network", *IEEE Trans. Reliability,* vol R-34, 1985 Jun, pp 184–186.

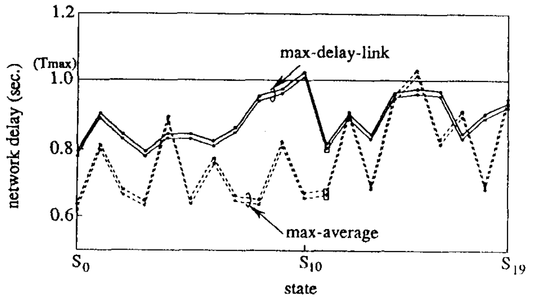
[11] T. Nishida, H. Miyahara, "Fault tolerant packet switched network design using capacity augmentation", *Proc. IEEE InfoCom88,* 1988, pp 9C.1.1–9C.1.7.

[12] T. Yokohira, M. Sugano, T. Nishida, H. Miyahara, "Capacity assignment algorithm for packet-switching network with unreliable links", *Trans. IEICE of Japan,* vol J73-B-I, num 10, 1990, pp 744–750. (in Japanese)

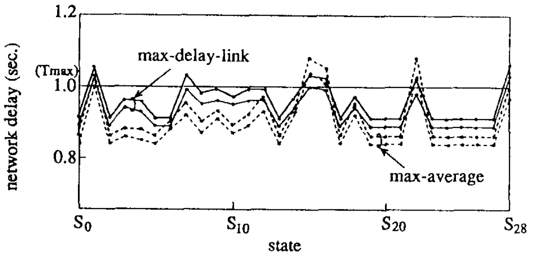
[13] M. D. Beaudry, "Performance-related reliability measures for computing systems", *IEEE Trans. Computers,* vol C-27, 1978 Jun, pp 540–547.

[14] J. F. Meyer, "On evaluating the performability of degradable computing systems", *IEEE Trans. Computers,* vol C-29, 1980 Aug, pp 720–731.

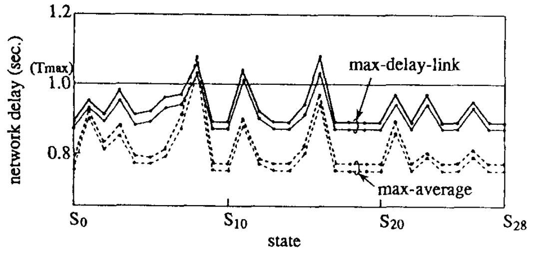
[15] V. G. Kulkarni, V. F. Nicola, K. S. Trivedi, R. M. Smith, "Numerical evaluation of performability measures and job completion time in repairable fault-tolerant systems", *Proc. FTCS-16,* 1986, pp 252–257; Vienna, Austria.



(a) Small network



(b) Large network I



(c) Large network II

Figure 8. Sensitivity to End-to-End Traffic

[16] V. P. Kumar, A. L. Reibman, "Failure dependent performance analysis of a fault-tolerant multistage interconnection network", *IEEE Trans. Computers,* vol 38, 1989 Dec, pp 1703-1713.

[17] J. T. Blake, A. L. Reibman, K. S. Trivedi, "Sensitivity analysis of reliability and performability measures for multiprocessor systems", *Proc. ACM SIGMETRICS, Conf. Measurement Modeling Computer Systems,* 1988, pp 177-186; Santa Fe, New Mexico.

[18] A. Goyal, W. C. Carter, E. de Sauza e Silva, S. S. Lavenberg, K. S. Trivedi, "The system availability estimator", *Proc. FTCS-16,* 1986, pp 84-89; Vienna, Austria.

[19] H. Heffes, A. Kumar, "Stochastic damage models and dependence effects in the survivability analysis of communication networks", *IEEE J. Selected Areas in Communications,* vol SAC-4, num 2, 1986, pp 234-242.

AUTHORS

Dr. Tokumi Yokohira; Department of Information Technology; Faculty of Engineering; Okayama University; 3-1-1, Tsushima-naka, Okayama-shi, 700 JAPAN.

Tokumi Yokohira (M'91) was born in Nagasaki, Japan, in 1960 December. He received the BE, ME, and DE in Information and Computer Sciences from Osaka University, Osaka in 1984, 1986, and 1989. He is an Assistant Professor in the Department of Information Technology, Faculty of Engineering, Okayama University, Okayama. His research interests include fault tolerance and performance evaluation of computer communication networks. He is a member of the IEEE, the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan, and the Information Processing Society (IPS) of Japan.

Masashi Sugano; Information Products Research Section; Mita Industrial Co., Ltd.; 1-2-28, Tamatsukuri, Chuo-ku, Osaka-shi, 540 JAPAN.

Masashi Sugano was born in Ishikawa, Japan in 1963. He received the BE and ME in Information and Computer Sciences from Osaka University in 1986 and 1988. From 1985 to 1988 he studied design and management of computer communication networks. In 1988 he joined Mita Industrial Co., Ltd., Osaka. He is a staff engineer of the information products research section. He received the IEICE paper award in 1990. He is a member of the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan, and the Information Processing Society (IPS) of Japan.

Dr. Takeshi Nishida; C&C System Research Laboratories; NEC Corporation; 4-1-1, Miyazaki, Miyamae-ku, Kawasaki-shi, 213 JAPAN.

Takeshi Nishida (M'84) was born in Ishikawa, Japan in 1955. He received the BA in Physics from Kyoto University, Kyoto in 1980, and the ME and DE in Information and Computer Sciences from Osaka University, in 1982 and 1986. From 1982 to 1987 he was an Assistant Professor in the Department of Information and Computer Sciences, Faculty of Engineering Science, Osaka University. Since 1987 he has been a researcher in the C&C System Research Laboratories, NEC Corp. During the summer of 1986 he worked at IBM Thomas J. Watson Research Center, Yorktown Heights. His research interests include high-speed network protocols, network design and management, and intelligent networks. He received the IEICE paper award in 1990. He is a member of the IEEE and the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan.

Dr. Hideo Miyahara; Department of Information and Computer Sciences; Faculty of Engineering Science; Osaka University; 1-1, Machikaneyama-chou, Toyonaka-shi, 560 JAPAN.

Hideo Miyahara (S'68,M'72) was born in Osaka, Japan, in 1943 June. He received the ME and DE in Communication Engineering from Osaka University, in 1969 and 1973. From 1973 to 1980 he was an Assistant Professor in the Department of Applied Mathematics and Physics, Faculty of Engineering, Kyoto University. From 1980 to 1986 he was with the Department of Information and Computer Sciences, Faculty of Engineering Science, Osaka University, as an Associate Professor. Since 1986 he has been a Professor with the Department of Information and Computer Sciences, Faculty of Engineering Science, Osaka University. From 1983 to 1984 he was a Visiting Scientist at the IBM Thomas J. Watson Research Center. His research interests include performance evaluation of computer communication networks, local area networks, and distributed systems. He received the IEICE paper award in 1990. He is a member of the IEEE, the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan, and the Information Processing Society (IPS) of Japan.

Manuscript TR91-319 received 1990 November 29; revised 1991 March 17; revised 1991 May 7.

IEEE Log Number 02227

◄TR►