On Gracefully Degrading Multiprocessors With Multistage Interconnection Networks

Israel Koren, Senior member IEEE University of Massachusetts, Amherst Zahava Koren

University of Massachusetts, Amherst

Key Words — Graceful degradation, multiprocessor, multistage interconnection network, connectivity, bandwidth, processing power, maintenance.

Reader Aids — Purpose: Widen state of the art. Special math needed for explanations: Probability. Special math needed to use results: Same. Results useful to: Multiprocessor analysts.

Abstract - Multiprocessing systems consisting of many processors, memory modules and interconnection links are being designed and implemented. Improvements in technology have reduced the failure rates of all system components. However, the large increase in the number of components per system has more than offset the increase in reliability of a single component. Therefore, some of the hundreds (or even thousands) of system components are anticipated to fail while the system is operating. Important questions regarding these systems are: Should each component be repaired upon failure, or should maintenance be scheduled at regular intervals allowing a graceful degradation of the system between repairs? If the latter is chosen, how fast will the dependability and performance of the multiprocessing system degrade before a repair takes place? This paper studies the behavior of a multiprocessing system with a multistage interconnection network in the presence of faulty components. Measures for the connectivity and performance of these systems are proposed. including: The average number of operational paths, the average number of accessible processors and memories, the average number of fault-free processors (memories) that are connected to an accessible memory (processor), the bandwidth, and the processing power of the system. Based on these measures, a tight upper bound for the maximal fully connected system is suggested. The gracefully degrading system is then compared, through some numerical examples, to a system whose faulty components are repaired upon failure. Based on these comparisons, the anticipated reduction in system performance can be estimated and consequently, appropriate maintenance policies can be determined.

1. INTRODUCTION

Advances in VLSI technology enable the design and implementation of multiprocessing systems consisting of very many components. One important class of these multiprocessing systems includes the shared-memory multiprocessors where all processors can access a set of memory modules through a multistage interconnection network. When implementing a complex multiprocessor, some of its components (like processors, memory modules or interconnection links) will fail. One then has the choice between replacing the faulty components upon failure, or waiting until a scheduled maintenance is performed and using the system at a degraded rate of performance in the meantime.

If graceful degradation of the multiprocessor is allowed, the computational capabilities of the degrading system must be estimated. To measure these capabilities it is insufficient to count the number of fault-free processors, memories, and interconnection links, since a fault-free processor for example, is useless if it is disconnected from the rest of the system. The measures should indicate:

- The number of operational communication paths
- The size of the maximal fault-free fully connected system existing (where a set of processors and a set of memories form a fully connected system if each processor in the set can communicate with every memory in the corresponding set)
- Similar connectivity issues.

The problem of calculating the average maximal fully connected system (or, equivalently, of finding the probability of a fully connected system of given dimensions) is, in general, of exponential complexity. One approach to the problem is constructing a detailed Markov chain where at each state the exact location of faulty components is known. This approach was adopted [1] for the analysis of a 16 \times 16 network. The prohibitively large number of states (on the order of $2^{16+16+32} = 2^{64}$) was reduced there to 25 after making several simplifying assumptions. Some of these assumptions lead to a pessimistic view of the system. and the generalization of the state reduction process to other system sizes is not straightforward. In a second approach to the problem [2], a spatial Poisson distribution for the faults is assumed. This assumption is accurate only for very large systems and very small fault probabilities.

The approach adopted here is different. Measures for the *connectivity* of a multiprocessor system with a multistage interconnection network in the presence of faulty components are defined. These connectivity measures indicate how many fault-free processors are still connected to fault-free memories and vice versa. The connectivity measures are simple to calculate, do not require pessimistic assumptions, are not restricted to a given size of the system, and can be extended to other architectures of the interconnection network. Exact expressions for these measures are derived, and then a tight upper bound for the average fully-connected system is obtained.

0018-9529/89/0400-0082\$01.00©1989 IEEE

These novel measures for system connectivity are introduced in section 2 along with several basic assumptions and notation. Section 3 derives expressions for the connectivity measures. Based on these measures the processing power of the degrading system is calculated in section 4. Section 5 compares, through some numerical examples and using the above measures, a gracefully degrading system to a system where each component is repaired upon failure.

2. CONNECTIVITY MEASURES

Consider a multistage interconnection network which connects N processors ($N = 2^k$) to N memories. The $N \times$ N interconnection network is constructed of 2 × 2 switches arranged in $k = \log N$ stages, each containing N/2switches as illustrated in figure 1. Our analysis can be generalized to the case where the number of processors is not necessarily a power of 2, the number of memories is different from the number of processors, and the network is built of $a \times b$ switches. For the sake of clarity however, the discussion here is restricted to the simpler case.

Notation

t some given time instant

- $q_r(t), p_r(t)$ probability that a processor is faulty, faultfree at time $t p_r(t) + q_r(t) \equiv 1$.
- $q_m(t), p_m(t)$ probability of a faulty, fault-free memory at time t.
- $q_i(t), p_i(t)$ probability of a faulty, fault-free link at time t.
- Q(t) mean number, at time t, of operational processorto-memory paths.
- r* number of processors for the maximal, fully connected system.
- m* number of memories for the maximal, fully connected system.

Processors and memories are considered to be nondecomposable components and therefore, any fault in a processor for example, results in the removal of this processor from the system. However, in contrast to the pessimistic assumption in [1], interconnection switches are decomposed into links and if one link is faulty, the other links can still operate properly.

The functional form of $p_r(t)$, $p_m(t)$, $p_m(t)$, $p_l(t)$ depends upon the statistical model for the faults in the system. Our analysis applies to any statistical fault process, as long as $p_r(t)$, $p_m(t)$, $p_l(t)$ can be calculated. The model need not include repair of faulty components, and each component (namely, processor, link, and memory) may have a different distribution law. Thus our approach is less restrictive than the Markov chain approach [1].

In the numerical examples we employ the widely used statistical model according to which the failure rates for



Fig. 1. An 8×8 multi-stage interconnection network

each component are constants, viz λ_r , λ_m , λ_l for processors, memories and interconnection links, respectively. For the degrading system, the probability of a fault-free processor at time t is,

$$p_r(t) = \exp(-\lambda_r t) \tag{2.1}$$

Similar expressions are obtained for $p_m(t)$ and $p_i(t)$. For the numerical analysis of the system with repair-uponfailure we use the similarly well-known model, according to which the repair (hazard) rates are constant with parameters μ_r , μ_m , μ_i for the three types of components. For this model, the probability of a fault-free processor at time *t* is,

$$p_r(t) = \frac{\mu_r}{\lambda_r + \mu_r} + \frac{\lambda_r}{\lambda_r + \mu_r} \exp(-(\lambda_r + \mu_r)t)$$
(2.2)

and similarly for the memories and interconnection links.

To analyze the connectivity of multistage networks in this environment where processors, memories and interconnection links can fail, the connectivity measures should capture the capabilities of the gracefully degrading multiprocessor system, eg, the number of connected faultfree processors and memories, the number of operational paths within the interconnection network, and the number of memory requests (from the processors) that can be transmitted through the interconnection network simultaneously.

Let the average maximal fault-free fully connected system, at time t, be the tuple $(F_r(t), F_m(t))$. This tuple is the mean value of the tuple (r^*, m^*) . Since a 2-dimensional maximum is not well-defined, we choose among all local maxima the one for which r^* and m^* are the closest, ie, the one for which min $\{r^*, m^*\}$ is maximal. The problem of calculating $(F_r(t), F_m(t))$ is exponential. We suggest therefore, the following approximate measures.

A reasonable measure for connectivity is the mean number, at time t, of operational processor-to-memory *paths*. Define a *path* as a processor, a memory, and the links connecting them: for a path to be operational, all its components (processor, memory, links) must be fault-free. The use of Q(t) as a connectivity measure was first suggested in [2]. A shortcoming of this measure is that the number of paths by itself provides no indication as to how many distinct processors and memories are still accessible.

Define a processor as *accessible* (at time instant t) iff it is fault-free and is connected to at least one fault-free memory. Similarly, define a memory as *accessible* iff it is fault-free and is connected to at least one fault-free processor. We propose an additional measure for connectivity, namely, the tuple $(A_r(t), A_m(t))$ where $A_r(t)$ and $A_m(t)$ denote the mean number of accessible processors, and accessible memories (at time t), respectively. This tuple does not necessarily imply that a fully connected fault-free $A_r(t)$ $\times A_m(t)$ interconnection network exists, nor does it indicate how many fault-free memories are connected, on the average, to an *accessible* processor. However, by combining Q(t) and $(A_r(t), A_m(t))$, a more complete characterization of system connectivity is obtained, as shown in theorem 1.

Theorem 1: Given a multiprocessor system with a multistage interconnection network, let $N_m(t)$ $(N_r(t))$ denote the mean number of fault-free memories (processors) to which an *accessible* processor (memory) is connected, then:

a.
$$N_m(t) = Q(t)/A_r(t)$$

b.
$$N_r(t) = Q(t)/A_m(t)$$

Proof: (a) Let $\mu(t)$ be the random variable denoting the number of memories to which an arbitrary processor (not necessarily an accessible one) is connected at time t. E $\{\mu(t)\}$ is the mean number of operational processor-tomemory paths emanating from one processor. Then, if all processors are equivalent—

i.
$$Q(t) = N \cdot E\{\mu(t)\}.$$

 $\Pr{\{\mu(t) \ge 1\}}$, is the probability that a processor is accessible. Hence, the mean number of accessible processors, $A_r(t)$, can be written as—

$$ii. A_r(t) = N \cdot \Pr{\{\mu(t) \ge 1\}}.$$

Finally, $N_m(t)$ is according to its definition—

iii. $N_m(t) = \mathbb{E}\{\mu(t) \mid \mu(t) \ge 1\}.$

Evaluating the conditional *s*-expectation in *iii* using *i* and *ii* yields,

$$N_{m}(t) = E\{\mu(t) \mid \mu(t) \ge 1\} = \frac{\sum_{i=1}^{N} i \cdot \Pr\{\nu(t) = i\}}{\Pr\{\nu(t) \ge 1\}}$$
$$= \frac{\sum_{i=0}^{N} i \cdot \Pr\{\nu(t) = i\}}{\Pr\{\nu(t) \ge 1\}} = \frac{E\{\nu(t)\}}{\Pr\{\nu(t) \ge 1\}}$$
$$= \frac{N \cdot E\{\mu(t)\}}{N \cdot \Pr\{\nu(t) \ge 1\}} = \frac{Q(t)}{A_{r}(t)}$$

Equation b for $N_r(t)$ can be proved in the same way.

Example: This simple example clarifies theorem 1. Let $p_r(t) = 4/5$, $p_m(t) = 9/10$ and $p_n(t) = 1$ (indicating fault-free interconnections). It is easy to see that for this simple case,

$$Q(t) = \left(\frac{4}{5} N\right) \left(\frac{9}{10} N\right),$$
$$A_{r}(t) = \left(\frac{4}{5} N\right) \left(1 - \left(\frac{1}{10}\right)^{N}\right).$$

 $N_m(t)$, on the other hand, is defined as the mean number of fault-free memories to which an accessible processor in connected. Thus, it is defined as a conditional *s*-expectation (conditional on the event that the processor is accessible) and should be calculated as such. In this specific case for which $p_i(t) = 1$, the event "The processor is accessible" is equivalent to the event "The processor is fault-free and at least one memory is fault-free". Hence, $N_m(t)$ is the mean number of fault-free. This conditional *s*-expected value is:

$$N_m(t) = \frac{9}{10} N / [1 - (\frac{1}{10})^N]$$

and consequently,

~

$$N_m(t) = Q(t) / A_r(t)$$

Similarly,

$$A_{m}(t) = \left(\frac{9}{10}N\right)\left(1 - \left(\frac{1}{5}\right)^{N}\right),$$

$$N_{r}(t) = \frac{4}{5}N / \left[1 - \left(\frac{1}{5}\right)^{N}\right]$$

$$N_{r}(t) = Q(t) / A_{m}(t).$$

The measures $N_r(t)$ and $N_m(t)$ satisfy $N_r(t) \le A_r(t)$ and $N_m(t) \le A_m(t)$ but still $(N_r(t), N_m(t))$ is only an upper bound for the average maximal fully connected system $(F_r(t), F_m(t))$. However, due to the high correlation among the paths in the interconnecting network (resulting from the fact that each fault affects N paths), a reasonable assumption is that these two tuples are very close to each other (especially for large values of N and small fault probabilities). Simulation runs of which the results are reported in section 5, verified this assumption, suggesting that the proposed measures can adequately characterize the connectivity of a multistage multiprocessor in the presence of faulty components.

System connectivity can be the basis for measuring the performance of the system. One possible measure for performance is the processing power. The processing power of a multiprocessor system (at time t) is denoted by C(t) and is the mean number of accessible processors that are computing, ie, not communicating with the memory. Only accessible processors are considered since non-faulty processors which are disconnected from all memory modules can not perform any useful computation. The processing power is calculated based on the connectivity and the bandwidth BW of the multiprocessor system. The latter is usually defined as the mean number of requests for the shared memory which are accepted per cycle given that each processor generates, with probability p_a , a request during every cycle [4]. Since all the measures introduced before are time-dependent, a slightly different definition for the bandwidth is used here. BW () is the mean number of processors, at time t, which are communicating with some memory. Theorem 2 provides the relationship between the connectivity, the bandwidth and the processing power.

Theorem 2:

Given a multiprocessor system with a multistage interconnection network, let $A_t(t)$ and C(t) be defined as before, and let BW (t) denote the bandwidth at time t, then:

 $C(t) = A_t(t) - BW(t).$

Proof:

Let $\alpha(t)$, $\beta(t)$, $\gamma(t)$ be the *r*.v.'s denoting the number (at time *t*) of accessible processors, processors communicating with some memory, and computing processors, respectively. Clearly, for each time instant *t*, $\alpha(t) = \beta(t) + \gamma(t)$. Since $A_r(t) = E\{\alpha(t)\}$, BW (*t*) = $E\{\beta(t)\}$, and $C(t) = E\{\gamma(t)\}$, the theorem follows.

Expressions for the bandwidth of a multistage network have been derived before [4]. Here, the bandwidth is defined as a time-dependent measure and an expression for calculating it is presented in section 4.

3. THE CONNECTIVITY OF A MULTISTAGE SYSTEM

This and the next sections analyze the degradation over time of a multistage system in the presence of faulty components, using the different connectivity measures and the processing power as defined in the previous section. We use the very realistic assumption that the mean time between component failures is much larger than the average length of the communication period between a processor and a memory. This implies that the status of the system components (faulty or fault-free) is constant for a large enough period of time—thus allowing us to study the system behavior under a statistical steady-state.

In what follows the system is observed at some arbitrary time instant t which is fixed throughout the analysis and is omitted from the notation for the sake of simplicity. However, all resulting measures are still functions of t due to the dependence on t of $p_t(t)$, $p_m(t)$ and $p_k(t)$. These probabilities are now denoted by p_t , p_m , and p_t .

We start with the analysis of the system connectivity, as measured by Q, by A_r and A_m , and by the tuple (N_r, N_m) . In a non-redundant interconnection network there is exactly one path between a processor and a memory and consequently, the calculation of Q is straightforward. Q is obtained by multiplying the number of processor-memory pairs by the probability of a fault-free path. The latter probability is,

$$p_r p_l^{k+1} p_m \tag{3.1}$$

where (k + 1) is the number of links along the path. Therefore,

$$Q = N^2 \cdot p_r p_l^{k+1} p_m \tag{3.2}$$

Let Φ_r be the probability that a given processor (say processor 0) is accessible; it is $Pr\{v(t) \ge 1\}$ in the proof of theorem 1. A_r is obtained by multiplying N by Φ_r . Due to the overlapping of the paths leading to the same processor, the inclusion/exclusion principle has to be used to calculate Φ_r . Define E_j as the event in which memory j is connected to processor 0. Φ_r can now be expressed in terms of the E_j as the probability that at least one of the events E_j occurs,

$$\Phi_r = \Pr\{E_1 \cup E_2 \cup \dots \cup E_N\}$$
(3.3)

$$= \Pr\{ \cup E_j\} = \sum_{i=1}^{N} (-1)^{(i-1)} W(i)$$
(3.4)

where W(i) is the sum over all $\binom{N}{i}$ subsets $\{j_1, j_2, ..., j_i\}$ of size *i*, of the probability that all *i* paths in the subset are operational, namely,

$$W(i) \cong \sum_{(j_1,\dots,j_i)} \Pr\{E_{j_1} \cap E_{j_2} \cap \dots \cap E_{j_i}\}$$
(3.5)

For a subset of *i* paths to be operational, all links in the subset must be fault-free. Hence, $\Pr\{E_{j_1} \cap E_{j_2} \cap ... \cap E_{j_i}\}$ depends not only on *i* but also on the number of links that the paths in the given subset have in common, since each link must be taken into account exactly once. Let *d* denote the number of distinct links in the subset, then—

$$\Pr\{E_{j_1} \cap E_{j_2} \cap ... \cap E_{j_i}\} = p_m^i p_r p_l^d$$
(3.6)

$$W(i) = p_m^i p_r \sum_{d=k+1}^{2^{k+1}-1} S_{i,d} p_l^d$$
(3.7)

where $S_{i,d}$ is the number of subsets of size *i* which contain exactly *d* distinct links.

To find $S_{i,d}$, consider each subset of paths emanating

from one processor as a binary tree of height k, with the interconnection links as vertices of the tree. The link connecting processor 0 is the root of the tree and the links connected to the *i* memories are the leaves. The combinatorial problem that has to be solved is: How many binary trees of height k with *i* leaves have exactly d vertices? To solve this problem, the levels of the tree are numbered, beginning from 0 for the leaves up to k for the root. d can then be expressed as the sum $d_0 + d_1 + \ldots + d_k$ where d_n is the number of vertices at level n, with $d_k = 1$ and $d_0 = i$. We solve the above combinatorial problem recursively. Given that level n has d_n vertices, the number of vertices at level (n - 1) satisfies $d_n \leq d_{n-1} \leq 2d_n$ and the number of ways in which level (n - 1) can have d_{n-1} vertices is:

$$\binom{d_n}{d_{n-1}-d_n} 2^{2d_n-d_{n-1}}$$
 (n = k, ..., 1). (3.8)

Multiplying (3.8) from k to 1 yields,

$$S_{i,d} = 2^{d-2i+1} \sum_{d_0+d_1+\ldots+d_k=d} \prod_{n=1}^k \begin{pmatrix} d_n \\ d_{n-1} - d_n \end{pmatrix}$$
(3.9)

As an example, for the 8×8 interconnection network in figure 1 we obtain,

$$W(1) = (p_m p_l) p_r p_l 8p_l^{2}$$

$$W(2) = (p_m p_l)^{2} p_r p_l (4p_l^{2} + 8p_l^{3} + 16p_l^{4})$$

$$W(3) = (p_m p_l)^{3} p_r p_l (8p_l^{3} + 16p_l^{4} + 32p_l^{5})$$

$$W(4) = (p_m p_l)^{4} p_r p_l (2p_l^{3} + 4p_l^{4} + 48p_l^{5} + 16p_l^{6})$$

$$W(5) = (p_m p_l)^{5} p_r p_l (24p_l^{5} + 32p_l^{6})$$

$$W(6) = (p_m p_l)^{6} p_r p_l (4p_l^{5} + 24p_l^{6})$$

$$W(7) = (p_m p_l)^{7} p_r p_l 8p_l^{6}$$

$$W(8) = (p_m p_l)^{8} p_r p_l p_l^{6}$$

Substituting W(1), ..., W(N) into (3.4) and then multiplying by N yields A_r . Due to the symmetry between processors and memories, A_m is obtained similarly by interchanging p_r and p_m . Finally, N_r and N_m are calculated by dividing Q by A_m and A_r , respectively.

4. THE PROCESSING POWER OF A MULTISTAGE SYSTEM

This section derives an expression for the processing power of a multistage system in the presence of faults which requires an expression for the bandwidth of such systems. An analysis of the bandwidth for a fault-free interconnection network appears in [4]. The analysis has been generalized [3] to an environment in which faults can occur and is briefly summarized here for completeness. We adopt the simplifying assumption that the destinations of the memory requests are s independent and uniformly distributed among the N memories. Therefore, the network bandwidth can be obtained by multiplying the number of memories N by the probability that a given memory module is non-faulty and has a request at its input. This last probability is calculated iteratively, following a path leading to this memory, ie, the probability of a request on an output link of a switch is calculated from the probability that such a request has been accepted at the input links to the same switch.

To simplify the discussion, a link is in state 1 (0) if it has (has not) a request for the memory. A faulty link is in state 0. The probability of a request on a link is thus the probability that this link is in state 1. We assign numbers to the k = logN stages in a descending order so that stage 0 is the last stage and its output links are connected to the memories, stage (k - 1) is the first one and its inputs are connected to the processors (see Figure 1). Consider a switch in stage *i* and denote its outputs $X^{(i)}$, $Y^{(i)}$. Its input links are the outputs of (two different) switches in stage (i + 1) and are denoted by $X^{(i+1)}$ and $Y^{(i+1)}$. Based on our assumption that memory requests are uniformly distributed among the memories, the probability that an incoming request is routed to any output link is the same. Hence, it is sufficient to consider only a single output link and derive an expression for the probability that it is at state 1, ie, $\Pr\{X^{(i)} = 1\}$.

Since a request for a memory module can reach the output link of a switch through any of the two input links, the state probability $Pr\{X^{(i)} = 1\}$ of the given output link has to be calculated from the joint probabilities of these input links:

$$\Pr\{(X^{(i+1)}, Y^{(i+1)}) = (u,v)\}; u,v = 0,1.$$

This calculation is performed using transition probabilities which take into account the status (faulty or fault-free) of the (physical) input links and the destinations of the incoming requests. Since memory modules are assumed to be equivalent, the incoming requests are routed to any of the two output links with probability 0.5. Consequently, the transition probabilities between the two inputs and the output of a switch are,

$$Pr\{X^{(i)} = 1 / (X^{(i+1)}), Y^{(i+1)}) = (0,0)\} = 0$$

$$Pr\{X^{(i)} = 1 / (X^{(i+1)}, Y^{(i+1)}) = (0,1)\} = \frac{1}{2} p_i$$

$$Pr\{X^{(i)} = 1 / (X^{(i+1)}, Y^{(i+1)}) = (1,0)\} = \frac{1}{2} p_i$$

$$Pr\{X^{(i)} = 1 / (X^{(i+1)}, Y^{(i+1)}) = (1,1)\} = p_i - \frac{1}{4} p_i^2$$
(4.1)

Only input link faults are taken into account. Faults at the output links are considered as input link faults at the next stage.

The state probability $Pr{X^{(i)} = 1}$ of the output link is:

$$Pr\{X^{(i)} = 1\} = \frac{1}{2}p_{i}[Pr\{(X^{(i+1)}, Y^{(i+1)}) = (0,1)\} + Pr\{(X^{(i+1)}, Y^{(i+1)}) = (1,0)\}] + p_{i}(1 - \frac{1}{4}p_{i}) \times Pr\{(X^{(i+1)}, Y^{(i+1)}) = (1,1)\}$$
(4.2)

For a non-redundant network the inputs into each switch are s-independent. Hence,

$$Pr\{(X^{(i+1)}, Y^{(i+1)}) = (u, v)\} = Pr\{X^{(i+1)} = u\} \times Pr\{Y^{(i+1)} = v\}; u, v = 0, 1$$
(4.3)

Using (4.3) and the equation:

$$\Pr\{Y^{(i+1)} = 0\} = \Pr\{X^{(i+1)} = 0\} = 1 - \Pr\{X^{(i+1)} = 1\}$$

we obtain from (4.2) after some algebraic manipulations,

$$\Pr\{X^{(i)} = 1\} = p_i \times \Pr\{X^{(i+1)} = 1\}$$

- ¹/₄ $p_i^2 \times (\Pr\{X^{(i+1)} = 1\})^2$ (4.4)

This expression is identical to the one in [4] if fault-free ($p_i = 1$) 2 \times 2 switches are assumed. This simple recursion formula enables the calculation of the successive state probabilities, starting from the processors outputs up to the memory inputs. For the processors output links we have:

$$\Pr\{X^{(k)} = 1\} = p_a p_r \tag{4.5}$$

where p_a is the probability of a processor requesting a memory connection.

Recursively, $Pr{X^{(0)} = 1}$ is calculated. To compute the bandwidth note that the memory and its input link can be faulty as well, hence,

$$BW = N \cdot \Pr\{X^{(0)} = 1\} \cdot p_m p_l$$
(4.6)

On the basis of the bandwidth BW and the number of accessible processors A_r the processing power is calculated using:

$$C = A_r - BW \tag{4.7}$$

5. NUMERICAL RESULTS

This section numerically compares a gracefully degrading system and a system whose faulty components are repaired upon failure. The connectivity measures $A_r(t)$, $A_m(t)$, $N_r(t)$ and $N_m(t)$ of the two systems, both of size 16 × 16, have been calculated as a function of time and are depicted in figures 2 and 3; time is measured in $1/\lambda_r$ units. The values for the other two failure rates are $\lambda_m/\lambda_r = 0.7$ and $\lambda_t/\lambda_r = 0.2$ while the same repair rate was assumed for all three types of system components: $\mu_r = \mu_m = \mu_l = 20\lambda_r$.

An important conclusion from this comparison is that $N_r(t)$ and $N_m(t)$ degrade faster than $A_r(t)$ and $A_m(t)$, respectively. To find the average size of the maximal fully connected system we have simulated the gracefully degrading system for different time instances. The results of these simulation runs are shown in figures 2 and 3. The



Fig. 2. (a) The average number of accessible processors, A_r , for a 16 × 16 gracefully degrading system and a 16 × 16 repaired system ($\mu_r = \mu_m = \mu_l = 20\lambda_r$). (b) The average number of processors connected to an accesible memory, N_r , for the same two systems. (c) The average number of fully connected processors for the gracefully degrading system—results of simulation. The failure rates are $\lambda_m/\lambda_r = 0.7$ and $\lambda_l/\lambda_r = 0.2$.

average size of the fully connected system is very close to $N_r \times N_m$ thus demonstrating the usefulness of the proposed connectivity measures.

Figure 4 depicts the processing power of the degrading system and the system which is repaired upon failure as a function of time. Since the processing power depends on the probability that a processor requests a memory access we have calculated the processing power of the two systems (with the same parameters as in figures 2 and 3) for two values of p_a ; $p_a = 0.25$ and $p_a = 0.75$. The processing power for $p_a = 0.25$ is higher since the processors are less busy communicating with the memories. Figure 4 can be used to determine the interval of scheduled maintenance so that the processing power of the system will not fall below some predetermined value.

Figure 5 can be helpful (combined with an adequate cost function) in determining a cost-effective repair rate for unscheduled maintenance. This figure depicts the increase in system connectivity (as measured by N_r and N_m) and processing power when the repair rate is increased.

Similar results were obtained for other sets of failure rates, e.g., $\lambda_m/\lambda_r = 2$ and $\lambda_l/\lambda_r = 0.05$, indicating that the same qualitative conclusions can be reached for a wide range of system parameters.



Fig. 3 (a) The average number of accessible memories, A_m , for the same two systems. (b) The average number of memories connected to an accessible processor, N_m , for the same two systems. (c) The average number of fully connected memories for the gracefully degrading system—results of simulation.



Fig. 4. The processing power of the same two 16×16 systems as a function of t for (a) $p_a = 0.25$ and (b) $p_a = 0.75$.



Fig. 5 The processing power of a repaired 16×16 system as a function of μ ($\mu_r = \mu_m = \mu_l = \mu$) for (a) $p_a = 0.25$ and (b) $p_a = 0.75$. (c) The average number of processors connected to an accessible memory, N_r , as a function of $\mu 1$.

ACKNOWLEDGEMENTS

This research was supported in part by the US National Science Foundation under contract DCR-85-09423 and MIP-8805586. The valuable comments made by the referees are gratefully acknowledged.

REFERENCES

- J. Arlat, J. C. Laprie, "Performance-related dependability evaluation of supercomputer systems," Proc. 13-th Ann. Symp. Fault-Tolerant Computing, 1983 June, pp 276-283.
- [2] V. Cherkassy, M. Malek, "Graceful degradation of multiprocessor systems," Proc. 1987 Intern. Conf. Parallel Processing, 1987 Aug, pp 885-888.
- [3] I. Koren, Z. Koren, "Analyzing the bandwidth of multiprocessors with multistage interconnection networks," *Concurrent Computations: Algorithm, Architecture and Technology,* Tewsbury, Dickinson, Schwartz (eds.), Plenum, 1988, pp 525-540.
- [4] J. H. Patel, "Performance of processor-memory interconnection for multiprocessors," *IEEE Trans. Computers*, vol C-30, 1981 Oct, pp 771-780.
- [5] N. Tzeng, P. Yew, C. Zhu, "The performance of a fault-tolerant multistage interconnection network," Proc. 1985 Intern. Conf. Parallel Processing, 1985 Aug, pp 458-465.

AUTHORS

Dr. Israel Koren; Dept. of Electrical & Computer Engineering; University of Massachusetts; Amherst, Massachusetts 01003 USA.

Israel Koren (S'72, M'76, SM'87) is a professor at the University of Massachusetts. He received the BSc (Cum Laude), MSc, and DSc degrees from the Technion — Israel Institute of Technology, Haifa, in 1967, 1970, and 1975, respectively, all in Electrical Engineering. Previously he was with the Departments of Electrical Engineering and Computer Science at the Technion — Israel Institute of Technology, where he became the Head of the VLSI Systems Research Center in 1985. Prior to that he has held positions with the University of California, Berkeley; University of California, Santa Barbara; and the University of Southern California, Los Angeles. His research interests are fault-tolerant VLSI and WSI architectures, models for yield and performance, floor-planning of VLSI chips, and computer arithmetic. He was the chair'n of the IEEE Intern Workshop on Defect and Fault Tolerance in VLSI Systems, 1988 October. He is a Guest Editor for *IEEE Trans. Computers*, in 1989 April, special issue on High Yield VLSI Systems.

Dr. Zahava Koren; Dept. of Electrical & Computer Engineering; University of Massachusetts; Amherst, Massachusetts 01003 USA.

Zahava Koren is a visiting researcher at the University of Massachusetts. She received the BA and MA degrees in Mathematics and Statistics from the Hebrew University in Jerusalem in 1967 and 1969, respectively, and the DSc degree in Operations Research from the Technion — Israel Institute of Technology in 1976. She has held positions with the Department of Statistics; University of Haifa; Departments of Industrial Engineering and Computer Science at the Technion; and the Department of Business and Economics, California State University in Los Angeles. Her main interests are optimization in queuing processes, stochastic analysis of computer networks, and reliability of computer systems.

Manuscript TR88-351 received 1988 March 11; revised 1988 July 15; revised 1988 August 29.

IEEE Log Number 25466

◄ TR ►

Our Past Editors

The early Editors of this Transactions created something out of nothing, and then persevered in keeping that fragile something alive and growing. They deserve a great deal of our recognition and thanks. We, the present Editorial Board and readers of this Transactions, are pleased to recognize and thank our early Editors, as well as the more recent ones, by publishing their names here.

Some of the early history is clouded because the Editor's name was not published in this Transactions until 1960. Anyone who has information on the Editors in those early years, please write or call us so that we can complete this list.

Past Editors

?	1950-1954
P. K. McElroy	1954-1958
Ernie J. Breidung	1958-1960
W. X. Lamb, Jr.	1960-1964
John A. Connor	1965-1968
Ralph A. Evans	1969-1985
Richard A. Kowalski	1986-1987
Myron A. Wilson	1988-1988

This Transactions was published "aperiodically" from the time it began in 1952 through 1967. In 1968 it was promoted to a quarterly. It became a bimonthly (except for February when the *Proceedings of the Annual Reliability & Maintainability Symposium* are distributed) in 1973 and has continued with five issues each year since then. \triangleleft TR \blacktriangleright