On Probabilistic Diagnosis of Multiprocessor Systems Using Multiple Syndromes

Sunggu Lee, *Member, IEEE*, and Kang G. Shin, *Fellow, IEEE*

***Abstract*—This paper addresses the distributed self-diagnosis of a multiprocessor/multicomputer system based on fault syndromes formed by comparison testing. We show that by using multiple fault syndromes, it is possible to achieve significantly better diagnosis than by using a single fault syndrome, even when the amount of time devoted to testing is the same. We derive a multiple syndrome diagnosis algorithm that in terms of the level of diagnostic accuracy achieved, is globally suboptimal, but optimal among all diagnosis algorithms of a certain type to be defined. Our diagnosis algorithm produces good results, even with sparse interconnection networks and interprocessor tests with low fault coverage. It is also proven that our diagnosis algorithm produces 100% correct diagnosis as N, the number of nodes in the system, approaches ∞, provided that the interconnection network has connectivity greater than or equal to 2 and that the number of syndromes produced grows faster than $\log N$. Our solution and another multiple syndrome diagnosis solution by Fussell and Rangarajan are comparatively evaluated, both analytically and with simulations.**

***Index Terms*— Distributed self-diagnosis, fault-tolerant computing, intermittent fault, multicomputer, multiprocessor, probabilistic diagnosis, self-test, system-level diagnosis**

I. Introduction

THIS PAPER addresses the problem of the distributed on-line self-diagnosis of processing nodes in a multiprocessor or multicomputer system. Each node diagnoses itself as faulty or nonfaulty based on comparisons of task outputs with its neighboring nodes. This type of method can be viewed as an on-line self-testing method in which a node tests itself by comparing itself to its neighboring nodes, instead of executing special routines to test its internal circuitry.

Informally, a *fault syndrome* is a set of task output comparisons viewed as a collection of binary pass-fail interprocessor test results. We consider diagnosis using multiple fault syndromes, corresponding to diagnosis based on multiple-task output comparisons. Diagnosis using this type of fault syndrome has its origins in the PMC model [9]. Unlike most of the system-level diagnosis methods based on the PMC model, however, we do not place an upper bound on the number of permitted faulty processors, nor do we assume interprocessor tests with perfect fault coverage. In addition, for ease of implementation, we assume that processors test one another by comparing the outputs of identical tasks (commonly referred to as *comparison-testing*). Since interprocessor tests with imperfect fault coverage, and since intermittently faulty processors can result in the same types of fault syndromes, we can handle intermittent as well as permanent faults.

Several authors [7], [8], [11] addressed the problem of diagnosing intermittent faults in t_i-diagnosable systems, in which, if no more than t_i nodes are intermittently faulty, a nonfaulty node will never be diagnosed as faulty [7]. Because a node is identified as faulty only if there is sufficient evidence to definitely identify it as faulty, given the upper bound t_i on the number of faulty nodes, however, these methods are overly conservative in identifying faulty nodes, and thus rarely achieve correct diagnosis (by which we mean that the diagnosed fault set is the same as the actual fault set).

Other authors [1], [3]–[5], proposed probabilistic diagnosis algorithms that achieve correct diagnosis with high probability, given intermittently faulty processors. Such probabilistic diagnosis algorithms offer the most general solutions with the highest level of *diagnostic accuracy*, defined as the percentage of diagnoses that are correct. The following are three arguments used to support probabilistic diagnosis algorithms:

1) using analysis to show that high diagnostic accuracy is achieved in certain situations [4],
2) guaranteeing that the set of nodes most likely to have caused the syndrome is found [3], and
3) showing that as the number of nodes in the system grows to infinity, diagnostic accuracy approaches 100% [1], [5].

Although argument 2)—guaranteeing the most probable diagnosis—is the most appealing, it has been shown that finding the most probable diagnosis, given the global syndrome information, is an NP-hard problem [2], [6]. From a practical perspective, argument 3) is insufficient, because it says nothing about the quality of diagnosis produced for finite systems. Because automated diagnosis is particularly important for large systems, however, asymptotically correct diagnosis is certainly a desirable property of any probabilistic diagnosis algorithm. In this paper, a diagnosis algorithm is presented that has the property of asymptotically correct diagnosis being *optimal* (in diagnostic accuracy) among all diagnosis algorithms of a certain type to be defined. Our diagnosis algorithm is further supported by simulations using square mesh and TMR structure.

Manuscript received January 13, 1992; revised March 8, 1993. This work was supported in part by the National Science Foundation under Grant MIP-9012549, and in part by the Korean Research Institute for Science and Technology (RIST) under Grant R92071.

S. Lee is with the Department of Electronic and Electrical Engineering, Pohang University of Science and Technology, Pohang 790-600, Republic of Korea.

K. G. Shin is with the Real-Time Computing Laboratory, Division of Computer Science and Engineering, Department of Electrical Engineering and Computer Science, The University of Michigan, Ann Arbor, MI 48109-2122, USA.

IEEE Log Number 9401198.

Blough *et al.* [1] showed that they could asymptotically achieve 100% correct diagnosis in an N processor system as $N \rightarrow \infty$, provided that $\alpha(N) \log N$ tests were performed on each processor, where $\alpha(N) \rightarrow \infty$ arbitrarily slowly as $N \rightarrow \infty$. In their method, the number of tests on processor u_i is equivalent to the number of processors testing u_i. Fussell and Rangarajan [5] used a different testing model to show that the same asymptotic result as [1] can be obtained for systems with lower connectivity (e.g., meshes or rings) if each pair of processors conducts multiple tests and if the number of *these* tests on each processor grows faster than $\log N$. Fussel and Rangarajan's algorithm can be viewed as a *multiple syndrome diagnosis* algorithm, in which testing is conducted in stages and a fault syndrome is collected after each testing stage.

In this paper, we improve upon Fussell and Rangarajan's (FR) algorithm [5] for probabilistic diagnosis based on multiple syndromes. Because multiple syndromes can be formed in many different ways, and because many different types of syndrome information can be used in the diagnosis, we define a specific category of multiple syndrome diagnosis (of which the FR algorithm is a member) and restrict our analysis to this category. Our diagnosis algorithm is *provably optimal* among all multiple syndrome diagnosis algorithms that use the same type of syndrome information as the FR algorithm. In addition, our optimal multiple syndrome diagnosis algorithm has the same desirable asymptotic properties as the FR algorithm.

The main contributions of this paper are 1) the use of probability analysis to show the advantage of multiple- versus single-syndrome diagnosis, and 2) the development of a multiple-syndrome diagnosis algorithm that is optimal among all diagnosis algorithms of a specific category. Simulations are presented in Section V to provide a comparison of the FR algorithm, our multiple syndrome diagnosis algorithm, and an single syndrome diagnosis algorithm [6].

II. Background

A. Preliminaries

A system S is composed of N processing nodes, denoted by the set $V = \{u_0, \cdots, u_{N-1}\}$. Interprocessor testing is assumed to be done by comparison testing, in which a test between two processors u_i and u_j is actually a comparison of their execution results or outputs for an identical task. The set of tests executed in this manner is represented by an undirected graph $G = (V, E)$, called the *testing graph*, where vertex $u_i \in V$ represents a processing node and undirected edge $e_{ij} \in E$ represents the fact that a comparison test is performed between u_i and u_j. The testing graph is assumed to be a subgraph of the graph representing the interconnection network of the system. A (fault) *syndrome SD* is a mapping from E to $\{0, 1\}$. $SD(e_{ij})$ is denoted by a_{ij} and is equal to 1 if and only if nodes u_i and u_j produce different execution results for the same task. A diagnosis is said to be *correct* if the set of nodes diagnosed to be faulty is the same as the actual fault set. For a given node u_i, let $\Gamma(u_i)$ denote the set of nodes that u_i tests (which are the same nodes that test u_i), and let $d(u_i) = |\{u_j \in \Gamma(u_i) : a_{ij} = 1\}|$, where $|A|$ is the cardinality of the set A. The fault status of u_i is denoted by δ_i for "u_i is faulty" and $\overline{\delta}_i$ for "u_i is nonfaulty."

The following probability parameters are used. Given a node $u_i \in V$, f_i is the prior fault probability of u_i. Let p_{ik} be the probability that $u_i \in V$ produces an incorrect result for a test task t_k, given that u_i is faulty. p_{ik} values will be referred to as *fault coverage*. In part of the probability analysis, we will require the use of *average* probability parameter values. The average values of parameters will be denoted by the corresponding letters without subscripts. For example, f and p refer to the average f_i and p_{ik} values, respectively.

The testing methods used in single- and multiple-syndrome diagnoses are referred to as *single-syndrome testing* and *multiple-syndrome testing*, respectively. As implied by their names, the main difference between single- and multiple-syndrome testing is that multiple-syndrome testing generates multiple (normally more than 1) syndromes by using multiple testing stages. As used in this paper, however, there is another major difference between the two methods regarding the manner in which the comparison tests are generated.

In single syndrome testing, it is assumed that the comparison test between a node u_i and another node $u_j \in \Gamma(u_i)$ is independent of any other comparison test. If two nodes u_i and u_j execute and compare more than one task, then $a_{ij} = a_{ji} = 1$ if any of the task outputs are different for the two nodes. Most of the previous work on diagnosis with comparison testing has assumed single-syndrome testing.

In multiple-syndrome testing, on the other hand, testing is done in stages, and in each testing stage, it is assumed that the same task is used in the comparison tests between a node u_i and the nodes in $\Gamma(u_i)$. In other words, all nodes in the same connected component of the testing graph must execute the same test task in a single testing stage. The results of the comparison tests in different testing stages are assumed to be statistically independent of each other. A "new" fault syndrome is formed after each testing stage using the same testing graph. The number of testing stages used in multiple syndrome testing is denoted as R.

There are several ways in which diagnosis can be done using the syndromes generated by the multiple syndrome testing method. Let us define the *most probable diagnosis*, given a testing graph and syndrome as the diagnosis (identification of faulty and nonfaulty nodes) that has the highest posterior probability value, given the testing graph and syndrome information. Then the optimal diagnosis method (in terms of diagnostic accuracy) is to produce the most probable diagnosis, given all of the information contained in the multiple syndromes. It has been shown that even with a single syndrome, however, finding the most probable diagnosis is an NP-hard problem [6]. In addition, an extremely high communication overhead is required to reliably distribute the syndrome information to all of the nodes. This problem is compounded when multiple syndromes are used.

Therefore, to obtain an efficient and practical diagnosis algorithm, we consider diagnosis algorithms in which each node u_i is aware of only the results of its tests with its immediate neighbors, referred to as *local* syndrome information. Note that in this case, $\Gamma(u_i)$ is the set of u_i's neighbors in the

TABLE I
CATEGORIZATION OF DIAGONOSIS USING LOCAL SYNDROMEINFORMATION

Category	Syndrome Information	Interpretation
1	Global	all a_{ji} values for all R stages
2	Local & iterative	a_{ji} values for all $u_j \in \Gamma(u_i)$ and F
2A	Summar. local & iterative	$\{d^k(u_i) : 1 \le k \le R\}$ and F
3	Local	a_{ji} values for all $u_j \in \Gamma(u_i)$
3A	Summarized local	$\{d^k(u_i) : 1 \le k \le R\}$
3AM	m-threshold local	$\lvert\{k : 1 \le k \le R \text{ and } d^k(u_i) > m\}\rvert$

interconnection network. For a node $u_i \in V$, *summarized local* syndrome information is defined as $\{d^k(u_i) : 1 \le k \le R\}$, where $d^k(u_i) = d(u_i)$ for testing stage k (the number of nodes that test u_i to be faulty in stage k). There are two dimensions to the syndrome information: One dimension is $d^k(u_i)$ for a fixed k, and the other dimension is the number of testing stages in which $d^k(u_i)$ is greater than a fixed threshold. Given an integer m, *m-threshold local* syndrome information is defined as $|\{k : 1 \le k \le R \text{ and } d^k(u_i) \ge m\}|$.

With these definitions, several categories of diagnosis can be defined. Category 1, the most general category, is defined as diagnosis using *all* of the global syndrome information. Exponential complexity algorithms such as [3] are category 1 diagnosis algorithms. In [1] and [4], the authors employ an iterative diagnosis method in which faulty nodes are identified one at a time, using summarized local syndrome information *and* the identity of the nodes identified as faulty in previous iterations; this is defined as category 2A diagnosis. (Category 2 uses local syndrome information and previously identified faulty node information.) Finally, category 3, 3A, and 3AM diagnosis are defined as diagnosis using local, summarized local, and m-threshold local (for any fixed m) syndrome information, respectively. This categorization is shown in Table I. For categories 2 and 2A, F refers to the set of nodes currently identified as faulty by the diagnosis algorithm. The diagnosis algorithm derived in this paper is the optimal category 3AM multiple-syndrome diagnosis algorithm.

B. Description of Fussell and Rangarajan's Algorithm

In this section, we describe Fussell and Rangarajan's (FR) algorithm [5] in detail, because it is characteristic of category 3AM diagnosis. In the FR algorithm, testing is conducted in stages, and two thresholds, kv_i and sv_i, are used. In testing stage k, it is assumed that all processors execute the same test task t_k. Let $T = \{t_1, \cdots, t_R\}$ be the set of R test tasks executed on all processors, where all tasks are treated identically, and let M be the total number of possible distinct results that a faulty processor can produce for a given test task. It is assumed that an incorrect result produced is uniformly distributed among the M possible distinct results. In Step 1 of the FR algorithm shown below, kv_j is chosen to be $|\Gamma(u_j)| - 1$, and a range of values is indicated as being acceptable for the choice of sv_j. These thresholds were simply chosen in order for the algorithm to satisfy desirable asymptotic properties. The authors of [5] proved that as $N \to \infty$, the diagnostic accuracy of the FR algorithm asymptotically approaches 100%. An earlier algorithm by the same authors [10] can be considered to be the same as the FR algorithm with $sv_j = 0$ for all $u_j \in V$.

Algorithm FR

0. Let $F \leftarrow \emptyset$ be the set of diagnosed faulty nodes;
1) **For** each $u_j \in V$ **do**
   $kv_j \leftarrow |\Gamma(u_j)| - 1$;
   $sv_j \leftarrow R - k_r R\{1 - p(1 - \frac{p}{M})^{|\Gamma(u_j)|}\}$,
   where $1 \le k_r \le 2$;
2) **For** each $t_i \in T$ **do**
   for each $u_j \in V$ **do**
   if $d(u_j) > kv_j$;
   then $L(i,j) = 1$;
   else $L(i,j) = 0$;
3) **For** each $u_j \in V$ **do**
   if $\sum_{t_i \in T} L(i,j) > sv_j$
   then $F \leftarrow F \cup \{u_j\}$;

III. ANALYSIS OF MULTIPLE-SYNDROME TESTING

The two main differences between multiple and single syndrome testing are the use of multiple versus single syndromes and the constrained manner in which the syndromes are formed in multiple syndrome testing. Regarding the first difference, examples are given in [10] to show that the additional information available when multiple syndromes are used permits correct diagnosis in some fault situations where an "accumulated" single syndrome results in incorrect diagnosis. In this section, we focus on the second difference and show that the way in which syndromes are formed in multiple and single syndrome testing also results in a significant difference in diagnostic capability.

Suppose we take all of the syndromes associated with the multiple testing stages in a multiple syndrome testing method and form an updated syndrome. The single syndrome generated by using this process has the property that for a given node $u_i \in V$, all of the tests e_{ij} for $u_j \in \Gamma(u_i)$ use the same set of tasks. However, the syndrome used in single syndrome testing does not have this property. In single syndrome testing, it is assumed that the test $e_{ij} \in E$ is independent of all of the other tests in E. As shown shortly, this "small" difference in testing method results in a significant difference in diagnostic capability.

First of all, let us consider the number of tasks executed per test for the two testing methods, given the same amount of testing time. Suppose that it takes τ units of time to execute each task, that each task has the same level of fault coverage p, and that test tasks are statistically independent in terms of the faults detected. For multiple syndrome testing, all nodes can be assigned to execute the same set of R tasks in time $R\tau$, resulting in R tasks being used in each test. The reason for this is that the *tests* are not required to be statistically independent of each other; a dependence is actually enforced by the requirement that the nodes in each connected component of the testing graph must execute the same set of tasks. For single syndrome testing, there is no specific requirement that the set of tasks executed by different nodes be identical. For a node u_i to test and be tested by another node $u_j \in \Gamma(u_i)$, however, u_i and u_j must have at least one task in common. This holds for u_i and all nodes $u_j \in \Gamma(u_i)$. In addition, if each test e_{ij} is to be statistically

independent for different nodes $u_j \in \Gamma(u_i)$, the set of tasks executed by different nodes in $\Gamma(u_i)$ must be disjoint. In time $R\tau$, node u_i can execute at most R tasks. Then, assuming that the set of R test tasks for node u_i is distributed uniformly for all tests e_{ik}, each test uses $R\tau/|\Gamma(u_i)|$ tasks. This is to be compared with R tasks for each test in multiple syndrome testing.

It may still be argued that this comparison is unfair, and that R tasks should be executed for each test in single syndrome testing also. In this case, however, the tests e_{ik} will not be statistically independent, and the single syndrome diagnosis algorithm, which assumes statistical independence of tests, will be operating on an incorrect assumption. It is noted, though, that if the tests e_{ik} are not limited to being comparison tests, this argument may not hold.

To continue with the analysis, let us compare the difference in diagnostic capability between single and multiple syndrome testing when only one syndrome is used in the multiple syndrome testing method. For simplicity of analysis, it is assumed that all nodes have identical parameter values and that the testing graph is regular with node-degree γ. The analysis can easily be individualized for each node (average parameter values are used for a given node's neighbors). Given a node u_i and any node $u_j \in \Gamma(u_i)$, let $A = A_{ji} = P(a_{ji} = 1 \mid \delta_i) = p_i((1-f_j)+f_j(1-\frac{p_j}{M}))+(1-p_i)f_ip_j = (1-f)p+fp(2-p-\frac{p}{m})$ and $B = B_{ji} = P(a_{ji} = 1 \mid \bar{\delta}_i) = f_jp_j \approx fp$. A (B) is the probability of node u_j testing u_i to be faulty, given that u_i is faulty (nonfaulty). It is assumed that $A > B$, which simply says that a nonfaulty node has a lower probability of being accused as faulty than a faulty node does. Note that the easier assumption that $f < 0.5$ (prior node failure probability less than 0.5) is *sufficient* for the assumption $A > B$ to hold.

For single syndrome testing, the probability of having $z = d(u_i)$ one-links incident on u_i out of a maximum of γ links (denoted as z *one-links* : γ), given that u_i is faulty and nonfaulty, is calculated as follows:

$$P(z \text{ one-links} : \gamma \mid \delta_i) = \binom{\gamma}{z} A^z(1-A)^{\gamma-z}, \quad \text{(1a)}$$

$$P(z \text{ one-links} : \gamma \mid \bar{\delta}_i) = \binom{\gamma}{z} B^z(1-B)^{\gamma-z}. \quad \text{(1b)}$$

For (1a), z of the γ neighbors of u_i are chosen such that those z neighbors test u_i to be faulty, and the $\gamma - z$ other neighbors test u_i to be nonfaulty. The result in (1b) is obtained similarly. It follows that:

$$\begin{aligned} &P(\delta_i \mid z \text{ one-links} : \gamma) \\ &= \frac{P(z \text{ one-links} : \gamma \mid \delta_i) f_i}{P(z\text{one-links} : \gamma \mid \delta_i) f_i + P(z \text{ one-links} : \gamma \mid \bar{\delta}_i)(1-f_i)} \\ &= \frac{1}{1 + \frac{1-f}{f}\left(\frac{B}{A}\right)^z \left(\frac{1-B}{1-A}\right)^{\gamma-z}}. \end{aligned} \quad \text{(1c)}$$

For multiple syndrome testing with a single syndrome, the probability that there are z one-links incident on a node $u_i \in V$, given that u_i is nonfaulty and faulty, is calculated as follows:

$$P(zone - links : \gamma \mid \bar{\delta}_i) = \sum_{j=z}^{\gamma} \binom{\gamma}{j} f^j(1-f)^{\gamma-j} \binom{j}{z} p^z(1-p)^{j-z} = h(z), \quad \text{(2a)}$$

$$\begin{aligned} P(zone - links : \gamma \mid \delta_i) = {} & p \sum_{j=0}^{z} \binom{\gamma}{j} (1-f)^j f^{\gamma-j} \binom{\gamma-j}{z-j} \\ & \times \left(1-\frac{p}{M}\right)^{z-j} \left(\frac{p}{M}\right)^{\gamma-z} \\ & + (1-p)h(z) \\ = {} & pg(z) + (1-p)h(z). \end{aligned} \quad \text{(2b)}$$

For (2a), u_i must have z or more faulty neighbors, of which exactly z execute the test task incorrectly. For the first part of (2b), u_i produces an incorrect test result and has $j \le z$ nonfaulty neighbors and $(z-j)$ faulty neighbors that either execute the test task correctly or produce a different incorrect result from u_i. It follows that the posterior fault probability of u_i, given z one-links incident on u_i, is calculated as follows:

$$\begin{aligned} &P(\delta_i \mid zone - links : \gamma) = \\ &\qquad \frac{f[pg(z) + (1-p)h(z)]}{f[pg(z) + (1-p)h(z)] + (1-f)h(z)} \\ &\qquad \approx \begin{cases} 1 & \text{if } z = \gamma, \\ f(1-p)/(1-fp) & \text{if } z \neq \gamma. \end{cases} \end{aligned} \quad \text{(2c)}$$

The approximation holds if M is large and $\gamma \le 4$. Then, if f (prior fault probability) values are small, (2c) is close to a δ function, with a spike at $z = \gamma$. If $\gamma > 4$, then (2c) becomes close to a step function. In general, (1c) is a much more smoothly increasing function of z than (2c) is.

In Fig. 1, we have shown the distributions of $z = d(u_i)$, given u_i faulty and u_i nonfaulty for both single syndrome testing ((1a) and (1b)) and multiple syndrome testing ((2a) and (2b)) using $M = 1000$, $\gamma = 4$, $p = 0.4$, and two different values of f. The fault coverage value p used for multiple syndrome testing is actually $1-(1-p)^\gamma = 0.87$, because this level of fault coverage can be obtained in the same amount of testing time required to achieve fault coverage of p for single syndrome testing. A high level of diagnostic accuracy can be achieved if the syndrome information perceived when u_i is faulty is drastically different from the syndrome information perceived when u_i is nonfaulty. From our analysis, it can be seen that the syndrome used in multiple syndrome testing fits this mold much more closely than the syndrome used in single syndrome testing.

IV. Optimal Multiple Syndrome Diagnosis

A. Probability Analysis

Consider category x diagnosis, where x can be any of the categories defined in Table I. Suppose the syndrome information used in diagnosing node $u_i \in V$ is denoted by

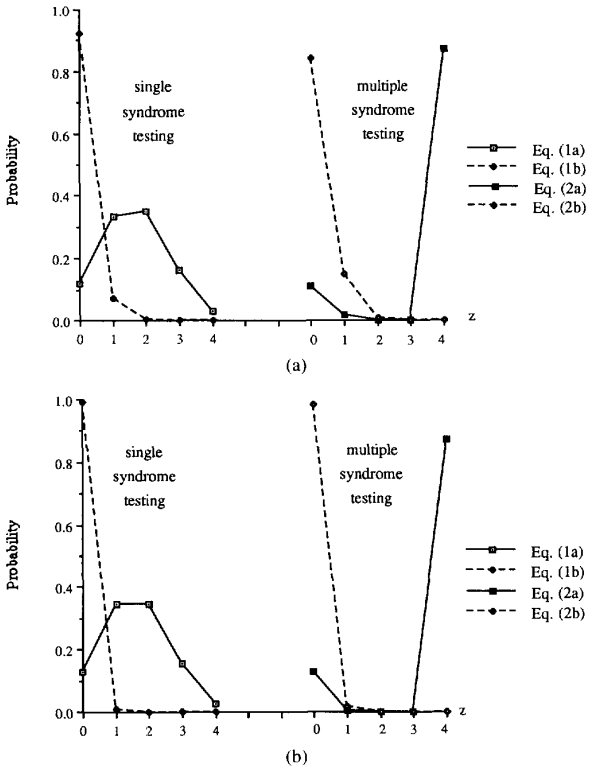


Fig. 1. Probability distribution. (a) $f = 0.0484$. (b) $f = 0.0050$.

SD_i. SD_i is a restricted form of the information present in the syndrome SD. For a given node u_i, its fault status set is defined as *Status_Set*$_i = \{\delta_i, \overline{\delta}_i\}$. Let us consider an arbitrary category x diagnosis algorithm A. In order for the diagnosis by A to be correct, A's diagnosis of each node $u_i \in V$ must be correct. Let A_i denote the part of A that diagnoses the fault status of node u_i, and define as basic events the pairs (SD_i, fs_i), where SD_i is a "partial syndrome" and $fs_i \in$ *Status_Set*$_i$. The set of all possible SD_i's will be denoted by SD_i^{all}. The diagnosis of u_i by A, when executed on a syndrome containing the partial syndrome SD_i, is denoted by $\text{Diag}_{A_i}(\text{SD}_i) \in$ *Status_Set*$_i$.

For category x diagnosis of node u_i, the sample space is $\Omega_i^x = \{(\text{SD}_i, fs_i) : \text{SD}_i \in \text{SD}_i^{\text{all}}, fs_i \in \textit{Status\_Set}_i\}$, the event space Θ_i^x is the set of all possible subsets of Ω_i^x, and the probability measure P_i^x is defined for category x diagnosis such that it is a legitimate probability measure. Given a testing graph G and a diagnosis algorithm A, let $\textit{Correct}_G(A_i) = \{(\text{SD}_i, fs_i)$: $\text{Diag}_{A_i}(\text{SD}_i) = fs_i\}$. For a testing graph G, the probability of correct diagnosis of u_i by A is as follows:

$$P_i^x(\textit{Correct}_G(A_i)) = \sum_{\text{SD}_i \in \text{SD}_i^{\text{all}}} P_i^x(\text{SD}_i, \text{Diag}_{A_i}(\text{SD}_i)).$$

For category x diagnosis, let OPTx denote the algorithm that makes the most probable diagnosis for node u_i, given SD_i. Then the following theorem shows that OPTx is the optimal category x diagnosis algorithm.

Theorem 1: For any category x diagnosis algorithm A, P($\{A$ produces correct diagnosis$\}$) $\leq$ P($\{OPTx$ produces correct diagnosis$\}$).

Proof: Consider a testing graph G and an arbitrary category x diagnosis algorithm A. A produces correct diagnosis if and only if the diagnosis of each node is correct. Let A_i denote the diagnosis of node u_i by A. For an arbitrary node $u_i \in V$, the following condition exists:

$$\begin{aligned} &P_i^x(\textit{Correct}_G(A_i)) \\ &\quad = \sum_{\text{SD}_i \in \text{SD}_i^{\text{all}}} P_i^x(\text{SD}_i, \text{Diag}_{A_i}(\text{SD}_i)) \\ &\quad = \sum_{\text{SD}_i \in \text{SD}_i^{\text{all}}} P_i^x(\text{Diag}_{A_i}(\text{SD}_i) \mid \text{SD}_i) P_i^x(\text{SD}_i) \\ &\quad \leq \sum_{\text{SD}_i \in \text{SD}_i^{\text{all}}} P_i^x(\text{Diag}_{\text{OPT}_{x_i}}(\text{SD}_i) \mid \text{SD}_i) P_i^x(\text{SD}_i) \\ &\quad = P_i^x(\textit{Correct}_G(\text{OPT}x_i)). \end{aligned}$$

Since $P_i^x(\textit{Correct}_G(A_i)) \leq P_i^x(\textit{Correct}_G(\text{OPT}x_i))$ for all $u_i \in V$, the theorem follows. **Q.E.D.**

Category 3AM multiple syndrome diagnosis is based on m-threshold local syndrome information for any fixed m. As shown in Table I, m-threshold local syndrome information gives the number of testing stages in which the number of one-links incident on a node $u_i \in V$ exceeds a threshold m, for any given m. Given this information for all possible values of m, what is the optimal diagnosis of node u_i?

From Theorem 1, we know that the most probable diagnosis for node u_i, given its syndrome information, is the optimal diagnosis. If it can be shown that the posterior fault probability of u_i is a strictly increasing function of $d^k(u_i)$, the number of one-links incident on u_i in testing stage k, then a threshold z_{th_i} can be chosen for $d^k(u_i)$ such that u_i is more probably faulty if $d^k(u_i) > z_{\text{th}_i}$ and more probably nonfaulty otherwise. Denoting this optimal threshold by $z^*_{\text{th}_i}$, the relevant category 3AM syndrome information for u_i can be reduced to $|\{k : 0 \leq k \leq R \text{ and } d^k(u_i) > z^*_{\text{th}_i}\}|$. Furthermore, if it can be shown that the posterior fault probability of u_i is a strictly increasing function of the number of testing stages in which $d^k(u_i) > z^*_{\text{th}_i}$, then the most probable diagnosis of u_i can be based on a threshold H_{th_i} on this quantity.

In summary, if the requirements stated in the above paragraph can be shown to hold, optimal category 3AM diagnosis can be based on the proper selection of the two thresholds z_{th_i} and H_{th_i}. In the FR algorithm, the threshold values $z_{\text{th}_i} = kv_i = \gamma - 1$ and $H_{\text{th}_i} = sv_i$ were used. The thresholds kv_i and sv_i in the FR algorithm were simply chosen to prove desirable asymptotic properties of the algorithm.

Optimal threshold values can be obtained by calculating posterior fault probabilities. The optimal choice for z_{th_i} is obtained by using (2c).

Theorem 2: The optimal z_{th_i} value, denoted as $z^*_{\text{th}_i}$, is equal to z such that $P(\delta_i | z \textit{ one-links} : \gamma) \leq 0.5$ and $P(\delta_i | z+1 \textit{ one-links} : \gamma) > 0.5$.

Proof: First of all, it is claimed that (2c) is a monotonically increasing function of z. From (2a) and (2b), it can be seen that $g(z)$ is an increasing function of z and that

$h(z)$ is a decreasing function of z. In (2c), the numerator and denominator can be rewritten to separate the terms multiplying $g(z)$ and $h(z)$. When this is done, the terms multiplying $g(z)$ are the same, and the term multiplying $h(z)$ in the denominator is greater than the corresponding term in the numerator. Thus, the stated claim follows. Given that z one-links are incident on a node $u_i \in V, u_i$ is more probably faulty (nonfaulty) if $P(\delta_i \mid z \textit{ one-links} : \gamma) \leq 0.5$ ($P(\delta_i \mid z \textit{ one-links} : \gamma) > 0.5$). Thus, based upon the claim and Theorem 1, Theorem 2 follows. **Q.E.D.**

Although calculation of (2c) for large values of γ is computationally expensive, because γ is at most the node-degree of the interconnection network, very large γ values will rarely be needed, because most practical systems are sparsely connected. Also, from the analysis done in Section III (approximation for (2c)), it is evident that when $\gamma \leq 4$ and $f < 0.5, z^*_{\text{th}_i} = \gamma - 1$. Because (2c) is a monotonically increasing function of z, we can obtain $z^*_{\text{th}_i}$ for $\gamma > 4$ by evaluating (2c) for several values of z near γ.

We now use probability analysis to derive $H^*_{\text{th}_i}$, the optimal value of H_{th_i}. Given the set of faulty nodes F', let $\Delta_i \equiv |\Gamma(u_i) \cap F'|$ (the number of faulty neighbors of u_i). Then the probabilities of having H syndromes in which $d(u_i) > z^*_{th_i}$ (denoted as H *passes*), given that u_i is faulty and nonfaulty, are as follows:

$$P(H \textit{ passes} \mid \delta_i) = \sum_{k=0}^{\gamma} P(\Delta_i = k)\binom{R}{H} A_k^H (1 - A_k)^{R-H}, \quad \text{(3a)}$$

and

$$P(H \textit{ passes} \mid \bar{\delta}_i) = \sum_{k=0}^{\gamma} P(\Delta_i = k)\binom{R}{H} B_k^H (1 - B_k)^{R-H}, \quad \text{(3b)}$$

where $A_k \equiv P(d(u_i) > z^*_{\text{th}_i} \mid \delta_i, \Delta_i = k)$ and $B_k \equiv P(d(u_i) > z^*_{\text{th}_i} \mid \bar{\delta}_i, \Delta_i = k)$. Using (3a) and (3b), the posterior fault probability of u_i, given that $d(u_i) > z^*_{th_i}$ for H syndromes, is as follows:

$$P(\delta_i \mid H\textit{passes}) = \frac{P(H \textit{ passes} \mid \delta_i)P(\delta_i)}{P(H \textit{ passes} \mid \delta_i)P(\delta_i) + P(H \textit{ passes} \mid \bar{\delta}_i)P(\bar{\delta}_i)}. \quad \text{(3c)}$$

For simplicity of analysis, let us assume that $\gamma \leq 4$ and $f < 0.5$, so that $z^*_{\text{th}_i} = \gamma - 1$. (The changes required in the analysis when $\gamma > 4$ are discussed at the end of this section.) Then, using (2a) and (2b), we get the following equations:

$$A_k = p\left(1 - \frac{p}{M}\right)^k + (1-p)B_k, \quad 0 \leq k \leq \gamma, \quad \text{(4a)}$$

$$B_k = \begin{cases} p^\gamma & , k = \gamma, \\ 0 & , 0 \leq k \leq \gamma - 1. \end{cases} \quad \text{(4b)}$$

Theorem 3: If $f < 0.5$, $M > \frac{p}{1-p}$, and $2 \leq \gamma \leq 4$, the optimal H_{th_i} value, $H^*_{th_i}$, is equal to H such that $P(\delta_i \mid H \textit{ passes}) \leq 0.5$ and $P(\delta_i \mid H + 1 \textit{ passes}) > 0.5$.

Proof: Following the proof method of Theorem 2, we first show that (3c) is a monotonically increasing function of H. Equation (3c) can be rewritten as follows:

$$P(\delta_i \mid H \textit{ passes}) = \frac{1}{1 + \frac{(1-f)P(H \textit{ passes}|\bar{\delta}_i)}{fP(H \textit{ passes}|\delta_i)}} = \frac{1}{1 + Z(H)}. \quad \text{(5)}$$

From (5), if $\frac{d}{dH}Z(H) < 0$, then $\frac{d}{dH}P(\delta_i \mid H \textit{ passes}) > 0$. Using (4a) and (4b) for A_k and B_k, and performing the necessary calculations for $\frac{d}{dH}Z(H)$, it can be shown that $\frac{d}{dH}Z(H) < 0$, provided that $A_k > B_\gamma (0 \leq k \leq \gamma)$. Using (4a) and (4b), it can easily be shown that this condition holds when $\gamma \geq 2$ and $M > \frac{p}{1-p}$. Then, from Theorem 1 and the result that (3c) is a monotonically increasing function of H, Theorem 3 follows. **Q.E.D.**

B. Description of OPTM Algorithm

Algorithm OPTM, the optimal category 3AM multiple syndrome diagnosis algorithm, is essentially the same as the FR algorithm, except that the thresholds are chosen differently. This, however, is a crucial difference, because the performance of the algorithm hinges upon the choice of thresholds. $z^*_{\text{th}_j}$ is obtained by using (2c), and $H^*_{\text{th}_j}$ is obtained by using Theorem 3. Our algorithm is described below, using the same assumptions as the FR algorithm with test tasks $T = \{t_1, \cdots, t_R\}$.

Algorithm OPTM

1. **For** each $u_j \in V$ **doparallel**
   calculate $z^*_{\text{th}_j}$ using (2c);
   calculate $H^*_{\text{th}_j}$ using Theorem 3;
2. **For** each $u_j \in V$ **doparallel**
   for each $t_i \in T$ **do**
   if $d(u_j) > z^*_{\text{th}_j}$
   then $L(i,j) = 1$;
   else $L(i,j) = 0$;
3. **For** each $u_j \in V$ **doparallel**
   if $\sum_{t_i \in T} L(i,j) > H^*_{\text{th}_j}$
   then u_i is faulty;
   else u_i is non-faulty;

Theorem 4: The OPTM algorithm is the optimal category 3AM diagnosis algorithm, provided that $2 \leq \gamma \leq 4, f < 0.5$, and $M > \frac{p}{1-p}$.

Proof: From Theorems 2 and 3, $z^*_{\text{th}_i}$ and $H^*_{\text{th}_i}$ are the optimal thresholds for the diagnosis of a node $u_i \in V$, given category 3AM syndrome information for u_i. Also, from the proofs of Theorems 2 and 3 (monotonically increasing property of the posterior probability function), it is clear that using the two thresholds $z^*_{\text{th}_i}$ and $H^*_{\text{th}_i}$ is sufficient for the optimal diagnosis of node u_i. Then, because the OPTM algorithm performs the optimal diagnosis of all nodes $u_i \in V$, given the syndrome information, the theorem follows. **Q.E.D.**

Although the OPTM algorithm is the optimal category 3AM diagnosis algorithm, it is not the optimal category 3, or even category 3A, diagnosis algorithm. Because the effort required

to obtain and use the syndrome information for category 3 or 3A is only slightly more than for category 3AM, it might seem worthwhile to derive the optimal category 3 and 3A multiple syndrome diagnosis algorithms. The diagnostic accuracy achieved by the OPTM algorithm is very close to the best possible, however, even when category 3 or 3A syndrome information is used.

To see the reason for this, let us refer to the analysis of Section III and Fig. 1. Given a nonfaulty node u_i, it is most likely not to have any one-links incident on it. Given a faulty node u_i, if it fails a test task t_k (in other words, t_k *covers* the fault), then it is most likely to have γ one-links incident on it; otherwise, if u_i passes t_k, then it is most likely not to have any one-links incident on it. Therefore, in all cases, u_i will most likely either have γ one-links or zero one-links incident on it. Given this observation, the syndrome information used in category 3AM contains almost all of the important category 3A syndrome information. In simulations using the experimental setup to be described in Section VI, out of 80 000 syndromes generated for a 100-node torus-wrapped square mesh, there was exactly one syndrome in which a node u_i had neither $d(u_i) = \gamma$ nor $d(u_i) = 0$. Thus, the OPTM algorithm produced the optimal category 3A diagnosis over 99.998% of the time. Also, as explained in [6], the optimal category 3A diagnosis algorithm approximates the behavior of the optimal category 3 diagnosis algorithm, because *average* probability parameter values are used in the analysis for category 3A. In summary, the OPTM algorithm is the optimal category 3AM multiple syndrome diagnosis algorithm and the "near-optimal" category 3A and category 3 multiple syndrome diagnosis algorithm.

C. Asymptotically Correct Diagnosis

One of the main desirable aspects of the FR algorithm was that it was shown to asymptotically achieve 100% correct diagnosis in an N processor system as $N \rightarrow \infty$, provided that $\gamma \geq 2$ and $R \geq \alpha(N) \log N$ testing stages are used, where $\alpha(N) \rightarrow \infty$ arbitrarily slowly as $N \rightarrow \infty$. Because the OPTM algorithm is the optimal category 3AM diagnosis algorithm and the FR algorithm is in category 3AM, it follows that OPTM also asymptotically achieves 100% correct diagnosis under the same conditions.

In order to prove that the FR algorithm [5] achieves asymptotically correct diagnosis, the authors used $z_{\text{th}_i} = kv_i = \gamma - 1$ and specified a range of acceptable values for $H_{\text{th}_i} = sv_i$. It is interesting to note that $H^*_{\text{th}_i}$ does not necessarily fall in the range of acceptable values for sv_i. In the FR algorithm, limits were set on sv_i so that asymptotic approximations could be used to prove the asymptotic correctness of the algorithm. However, the threshold H_{th_i} need not necessarily belong to the range of acceptable values for sv_i in order to achieve asymptotically correct diagnosis.

Regarding the z_{th_i} threshold, the OPTM algorithm also uses $z^*_{\text{th}_i} = \gamma - 1$. However, this value of z_{th_i} was chosen after careful posterior probability analysis. If $\gamma > 4$ or $f \geq 0.5$, then (2c) may yield a different value for $z^*_{\text{th}_i}$. The methodology described in this paper is still applicable in deriving $H^*_{\text{th}_i}$ and the resulting optimal category 3AM algorithm. This case is discussed in the next section.

D. Analysis for Higher Connectivity Networks

In deriving the thresholds for the OPTM algorithm, we have assumed that $2 \leq \gamma \leq 4, f < 0.5$ and $M > \frac{p}{1-p}$. The restrictions $f < 0.5$ and $M > \frac{p}{1-p}$ are reasonable, because f is the prior fault probability, and because M, the number of failure modes of a faulty node, can be made large by choosing test tasks with a large number of possible output results. The restriction on γ is undesirable, however, because higher connectivity networks are useful for many applications. If $\gamma > 4$, then it is possible that $z^*_{\text{th}_i} < \gamma - 1$. In that case, the A_k and B_k values used in (4a) and (4b) must be changed.

Using (2a) and (2b) and generalizing for arbitrary $z^*_{\text{th}_i}$ values, we get the following equation:

$$A_k = p \sum_{j=z^*_{\text{th}_i}+1}^{\gamma} \binom{k}{j-\gamma+k} \left(1 - \frac{p}{M}\right)^{j-\gamma+k} \left(\frac{p}{M}\right)^{\gamma-j} + (1-p)B_k, \tag{6a}$$

$$B_k = \sum_{j=z^*_{\text{th}_i}+1}^{k} \binom{k}{j} p^j (1-p)^{k-j}. \tag{6b}$$

After (6a) and (6b) have been used to calculate A_k and B_k for all $k (0 \leq k \leq \gamma)$, (3a)–(3c) can be used with the new A_k and B_k values to calculate the posterior fault probabilities required by the OPTM algorithm.

V. Simulations

To evaluate the performance of the diagnosis algorithms studied, numerous simulation experiments were conducted on a SUN 4/280 for a 100-node torus-wrapped square mesh and 300-node TMR structure. The TMR structure is a 2-regular graph in which nodes are clustered into completely connected components of size 3 each. In multiple syndrome diagnosis, diagnosis decisions are made for each node, independently of the other nodes. Because a diagnosis result is correct only if the diagnosis decisions at *all* nodes are correct, however, diagnostic accuracy decreases with increasing number of nodes for the two types of structures considered. For diagnostic accuracy to increase with increasing numbers of nodes, each node must be connected to at least $\log N$ other nodes, where N is the total number of nodes (refer to Section IV-C).

Assumptions widely used in the fault-tolerant computing field were followed to generate probability parameter values. For each node $u_i \in V$, failure arrivals were assumed to follow a Poisson process and a time value τ_i, corresponding to the length of time u_i has been in the system, was generated from a uniform distribution over the interval $[0, T]$ for some T. Then, for node $u_i, f_i = 1 - e^{-\lambda \tau_i}$, where $\lambda = \text{MTTF}^{-1}$ is the mean failure arrival rate. The simulated experiments were conducted with $T = 1$ K hr and MTTF values of 50 K and 100 K hr, resulting in λT values of 0.1 and 0.01, respectively. Clearly, the same results can be obtained by decreasing T and increasing MTTF by the same factor. Thus, this can model components that have been in the system for different lengths of times and components that have different MTTF values (their τ_i values can be adjusted). Given MTTF values of 50 K and 100 K, $E[f_i] = 0.0099$ and 0.0050, respectively.

TABLE II
THRESHOLD VALUES FOR TORUS-WRAPPED SQUARE MESH ($\gamma = 4$)

	H_{th}^*	sv_i	sv_i^{mid}
$p = 0.1$	0	0 — 1.60	0.80
$p = 0.2$	0	0 — 3.20	1.60
$p = 0.3$	0	0 — 4.80	2.40
$p = 0.4$	0	0 — 6.40	3.20
$p = 0.5$	0	0 — 8.00	4.00

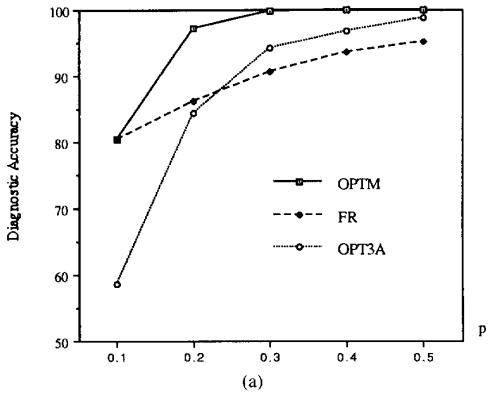
TABLE III
THRESHOLD VALUES FOR TMR STRUCTURE ($\gamma = 2$)

	H_{th}^*	sv_i	sv_i^{mid}
$p = 0.1$	0	0 — 1.60	0.80
$p = 0.2$	0	0 — 3.20	1.60
$p = 0.3$	0	0 — 4.80	2.40
$p = 0.4$	0	0 — 6.40	3.20
$p = 0.5$	1	0 — 8.00	4.00
$p = 0.7$	4 ($f = 0.0099$)	6.40 — 11.20	8.80

For all diagnosis algorithms evaluated, 1000 fault situations were produced and diagnosed, assuming p values of 0.1 to 0.5 (in 0.1 increments) and the two MTTF values given above. For each fault situation, $R = 16$ testing stages were used, thus resulting in 16 syndromes. The OPTM algorithm was compared with the FR algorithm and the OPT3A algorithm, a category 3A single syndrome diagnosis algorithm described in detail in [6]. The OPT3A algorithm is used for comparison purposes, because it has been shown to be the optimal category 3A single syndrome diagnosis algorithm [6]. Assuming that it takes τ units of time to execute a single test task, multiple syndrome testing requires $R\tau$ time units, and single syndrome testing requires $\gamma\tau$ time units. Thus, for single syndrome diagnosis, if the same amount of time is devoted to testing, it is possible to use test tasks that are R/γ times as long as those used in multiple syndrome diagnosis. Therefore, in the simulations for the OPT3A algorithm, $p' = 1 - (1 - p)^{R/\gamma}$ was used as the fault coverage value.

In the simulations for the FR algorithm, we must choose values for kv_i and sv_i. The FR algorithm specifies that $kv_i = \gamma - 1$, but indicates that a range of values is acceptable for sv_i (refer to Section II-B). If the equation for sv_i in Section II-B is used, it is possible to get a negative value for sv_i. Because a negative sv_i threshold value implies that all nodes in V will be diagnosed to be faulty, this possibility is discounted. Then the modified equation for sv_i is $\max\{0, R-2R(1-p(1-\frac{p}{M})^\gamma)\} \leq sv_i \leq R - R(1 - p(1 - \frac{p}{M})^\gamma)$. In our simulations, sv_i was chosen to be the value halfway between the lower and upper bounds for sv_i. M was chosen to be 1000. Tables II and III show the values of H_{th}^* and sv_i for the torus-wrapped square mesh and TMR structure, respectively. H_{th}^* values shown in Tables II and III are for both $f = 0.0099$ and $f = 0.0050$, unless otherwise specified. sv_i and sv_i^{mid} (used in the simulations) are independent of f. In Table III, threshold values for $p = 0.7$ and $f = 0.0099$ are shown to demonstrate that H_{th}^* and sv_i values do diverge. This shows that the optimal threshold value for H_{th_i} does not necessarily lie in the region specified for sv_i [5].

Figs. 2 and 3 show the results of the simulations for the torus-wrapped square mesh and TMR structure, respectively. In all cases, the OPTM algorithm performs significantly better than the OPT3A algorithm, with the difference being more acute when p is small. The FR algorithm performs the same as the OPTM algorithm for $p = 0.1$, but then quickly falls off in accuracy as different $\lceil sv_i^{\text{mid}} \rceil$ threshold values are used. Similar results were obtained for all simulations attempted.



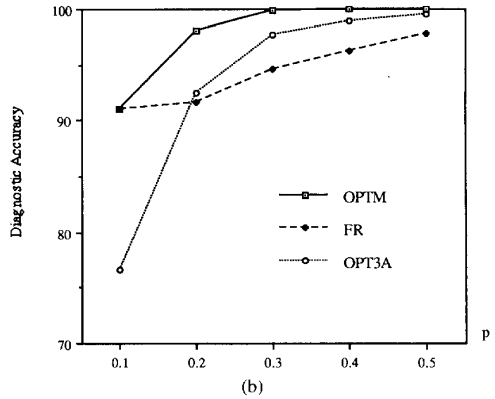


Fig. 2. Accuracy with square mesh and MTTF. (a) 50 K hr. (b) 100 K hr.

VI. CONCLUSION

In this paper, we have derived an optimal category 3AM (and near-optimal category 3A and category 3) multiple syndrome diagnosis algorithm. Using probability analysis, multiple syndrome testing is shown to be more effective than single syndrome testing. Our simulation results support the probability analysis. It is proven that Algorithm OPTM, the optimal category 3AM multiple syndrome diagnosis algorithm, achieves 100% correct diagnosis in an N processor system as $N \rightarrow \infty$, provided that $R \geq \alpha(N) \log N$ testing stages are used, where $\alpha(N) \rightarrow \infty$ arbitrarily slowly as $N \rightarrow \infty$. If γ is considered as a constant, the computational complexity of the OPTM algorithm is $O(R)$, which is the minimum possible for any multiple syndrome diagnosis algorithm, because R testing stages are required.

The OPTM algorithm requires each processor to execute identical tasks with its neighbors, to send and receive the results of the tasks from its neighbors, to compare the results received with its own results, and to execute a diagnosis procedure to determine whether it should diagnose itself to be faulty or nonfaulty. Note that a TMR or NMR system also has all of the above requirements, excluding the diagnosis

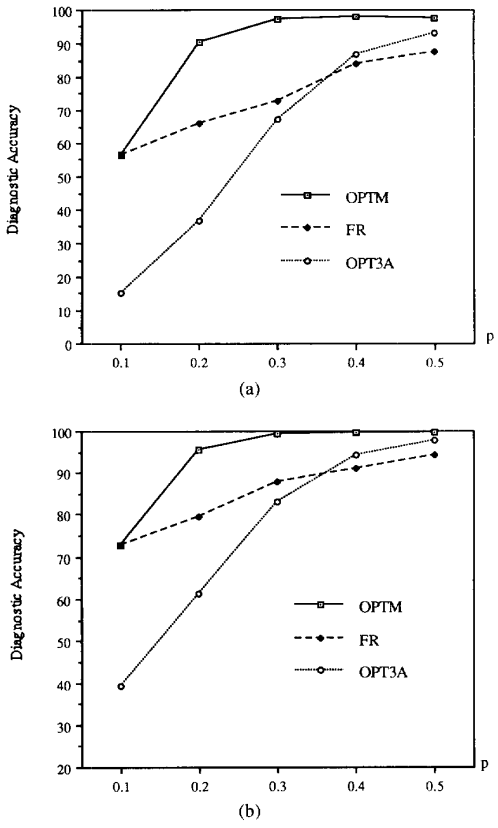


Fig. 3. Accuracy with TMR structure and MMTF. (a) 50 K hr. (b) 100 K hr.

requirement.[1] The diagnosis procedure must be executed by a diagnostic component that operates in a fail-safe mode or is part of the *hard-core* of the processor. There must also be reliable communication between neighboring nodes. This can be accomplished by sending identical messages along multiple disjoint paths [6]. The thresholds used by the OPTM algorithm can be precomputed. Thus, the OPTM algorithm simply requires the diagnostic component to accumulate integer quantities and compare them against precomputed threshold values. The diagnostic component can therefore be a very simple digital circuit.

REFERENCES

[1] D.M. Blough, G.F. Sullivan, and G.M. Masson, "Almost certain diagnosis for intermittently faulty systems," *Digest of Papers*, FTCS-18, pp. 260–265, 1988.
[2] D.M. Blough, "Fault detection and diagnosis in multiprocessor systems," Ph.D. dissertation, Johns Hopkins Univ., Baltimore, MD, 1988.
[3] M.L. Blount, "Probabilistic treatment of diagnosis in digital systems," *Digest of Papers*, FTCS-7, pp. 72–77, 1977.
[4] A.T. Dahbura, K.K. Sabnani, and L.L. King, "The comparison approach to multiprocessor fault diagnosis," *IEEE Trans. Comput.*, vol. C-36, no. 3, pp. 373–378, Mar. 1987.
[5] D. Fussell and S. Rangarajan, "Probabilistic diagnosis of multiprocessor systems with arbitrary connectivity," *Digest of Papers*, FTCS-19, pp. 560–565, 1989.
[6] S. Lee, "Probabilistic multiprocessor and multicomputer diagnosis," Ph.D. dissertation, Univ. of Michigan, Ann Arbor, 1990.

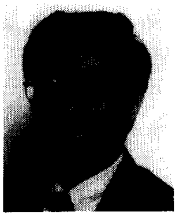
[1] Although a voting circuit is typically used to do the comparisons in TMR or NMR, the communications and comparison requirements are the same.

[7] S. Mallela and G.M. Masson, "Diagnosable systems for intermittent faults," *IEEE Trans. Comput.*, vol. C-27, no. 6, pp. 560–566, June 1978.
[8] , "Diagnosis without repair for hybrid fault situations," *IEEE Trans. Comput.*, vol. C-29, no. 6, pp. 461–470, June 1980.
[9] F.P. Preparata, G. Metze, and R.T. Chien, "On the connection assignment problem of diagnosable systems," *IEEE Trans. Electron. Comput.*, vol. EC-16, no. 6, pp. 848–854, Dec. 1967.
[10] S. Rangarajan and D. Fussell, "A probabilistic method for fault diagnosis of multiprocessor systems," *Digest of Papers*, FTCS-18, pp. 278–283, 1988.
[11] C.L. Yang and G.M. Masson, "A fault identification algorithm for t_i-diagnosable systems," *IEEE Trans. Comput.*, vol. C-35, no. 6, pp. 503–510, June 1986.

S. Lee (S'87–M'90) received the B.S.E.E. degree (with highest distinction) from the University of Kansas, Lawrence, in 1985, and the M.S.E. and Ph.D. degrees from the University of Michigan, Ann Arbor, in 1987 and 1990, respectively.

He is currently an Assistant Professor in the Department of Electronic and Electrical Engineering, Pohang University of Science and Technology, Pohang, Republic of Korea. Prior to this appointment, he was an Assistant Professor in the Department of Electrical Engineering at the University of Delaware, Newark. His research interests are in parallel and fault-tolerant computing.

Dr. Lee is a member of the IEEE Computer Society and the Tau Beta Pi Engineering Honor Society.



K. Shin (S'75–M'78–SM'83–F'92) received the B.S. degree in electronics engineering from Seoul National University, Seoul, Republic of Korea, in 1970, and the M.S. and Ph.D. degrees in electrical engineering from Cornell University, Ithaca, NY, in 1976 and 1978, respectively.

He is Professor and Director of the Real-Time Computing Laboratory, Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor. From 1978 to 1982, he was on the faculty of Rensselaer Polytechnic Institute, Troy, NY. He has held visiting positions at the U.S. Air Force Flight Dynamics Laboratory, AT&T Bell Laboratories, the Computer Science Division within the Department of Electrical Engineering and Computer Science at the University of California at Berkeley, and the International Computer Science Institute, Berkeley, CA. He also chaired the Computer Science and Engineering Division, Department of Electrical Engineering and Computer Science, University of Michigan, for three years, beginning in 1991. In 1985, he founded the Real-Time Computing Laboratory, where he and his colleagues are currently building a 19-node hexagonal mesh multicomputer, called **HARTS**, to validate various architectures and analytic results in the area of distributed real-time computing. He has also been applying the basic research results of real-time computing to manufacturing-related applications ranging from the control of robots and machine tools to the development of open architectures for manufacturing equipment and processes. Recently, he has initiated research on the open-architecture information base for machine tool controllers.

Dr. Shin has authored or coauthored over 270 technical papers (more than 120 of these in archival journals) and several book chapters in the areas of distributed real-time computing and control, fault-tolerant computing, computer architecture, robotics and automation, and intelligent manufacturing. In 1987, he received the Outstanding IEEE TRANSACTIONS ON AUTOMATIC CONTROL Paper Award for a paper on robot trajectory planning. In 1989, he also received the Research Excellence Award from the University of Michigan. He was Program Chair of the 1986 IEEE Real-Time Systems Symposium (RTSS), the General Chair of the 1987 RTSS, the Guest Editor of the 1987 special issue of IEEE TRANSACTIONS ON COMPUTERS (real-time systems), a Program Co-Chair for the 1992 International Conference on Parallel Processing, and served numerous technical program committees. He also chaired the IEEE Technical Committee on Real-Time Systems during 1991–93, served as a Distinguished Visitor of the IEEE Computer Society, and is an Editor of IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED COPMUTING, and an Area Editor of *Internation Journal of Time-Critical Computing Systems.*