Bayesian Analysis for Fault Location in Homogeneous Distributed Systems

Yu Lo Cyrus Chang*, Leslie C. Lander*, Horng-Shing Lu† and Martin T. Wells†

*Department of Computer Science
Thomas J. Watson School of Engineering and Applied Science
State University of New York, Binghamton, NY 13902-6000
(ylchang@bingsuns.cc.binghamton.edu)

†Statistics Center
Cornell University
Ithaca, NY 14853-3901

Abstract

We propose a simple and practical probabilistic comparison-based model, employing multiple incomplete test concepts, for handling fault location in distributed systems using a Bayesian analysis procedure. This approach is more practical and complete than previous ones since it does not assume any conditions such as permanently faulty units, complete tests, perfect environments, or non-malicious environments. Fault-free systems are handled without overhead, hence the test procedure may be used to monitor a functioning system. Given a system S with a specific test graph, the corresponding conditional distribution between the comparison test results (syndrome) and the fault patterns of S can be generated. To avoid the complex global Bayesian estimation process, we develop a simple bitwise Bayesian (B-) algorithm for fault location in S, which locates system failures with *linear* complexity, suitable for hard real-time systems.

Keywords: Bayesian decision rules, Distance measure, Fault location, Loss function, Probabilistic comparison model, System Diagnosis.

1. Introduction

This paper studies fault location using Bayesian inference methods based on a simple probabilistic comparison-based model. The objective of testing a system is generally twofold: first, discover whether a fault exists in the system and second, locate the fault or faults once they are known to exist. For the past few years, significant advances have been made in locating faults of a system under test, S, using a probabilistic approach [1, 3-10, 12, 17, 19, 34]. Indeed, the use of probability in fault location may be more realistic than deterministic methods because the former can accommodate the multiplicity of the random effects outlined in (ii) through (vii) below which tend to perturb the latter. However, the computational complexity involved in fault location usually increases drastically, if all the possible random effects of faults in S are considered. The situations in the following list have been a constant challenge for researchers when dealing with fault location of S. Some of the papers that address the problems are shown in parenthesis:

(i) The diagnosis results for a system may only be valid for a bounded number of faulty units (Smith [33]).
(ii) Systems may have non-permanent faults (Mallela and Mason [23]).
(iii) Faulty units may behave maliciously—they may lie about their results (Gupta and Ramakrishnan [14]).
(iv) Tests may be incomplete (Russell and Kime [28, 29]).
(v) For some test strategies, faulty units have to be assumed to be still able to execute assigned tests (Preparata *et al.* [27]).
(vi) There may be noisy environments, or errors in transmitting/receiving devices (Blount [5]).
(vii) The probability of failure may vary as the run time advances. (Chang [6] and Chang *et al.* [7-9])

In relation to item (vii), the probability of failure is assumed to be constant by Blough and Sullivan [4], Lin and Shin [19], and Berman and Pelc [3]. Furthermore, it is necessary to consider how to manage testing policies for fault location in S, where failures are dependent upon operation-time [9]. Any of these phenomena may actually occur together during the testing process. Hence, rather than trying to directly monitor and control the randomness of the system during the tests, we propose to simply apply a set of tests, gather all the results from the outputs and utilize stochastic methods to solve the fault location problem. In doing that, one should really consider a generalized approach that includes all possible random phenomena. Recently, Dahbura [12], Fussell and Rangarajan [13], Lee and Shin [18] proposed to use comparison testing in their probabilistic diagnosis strategies. The idea of performing the comparison tests among units is inspired by earlier research by Malek [22], and Chwa/Hakimi [11]. The main reason for this form of testing is that it is less intrusive and easier to compare test results among units, e.g. microprocessors, than to use some units to test others.

The generalized probabilistic approaches have so far incurred exponential computational complexity during diagnosis [1, 5, 19, 21] but more recent work in real-time fault-tolerant systems demonstrates that the complexity can be reduced to linear, see Chang *et al.*, [7, 8, 10]. A related problem is that storage for the *a priori* diagnostic data has also been exponential if all random effects are considered [1, 5, 6]. Lander and Chang [17] show that only limited quantities of data are required per test cycle and that the data can be generated on-line in polynomial time.

We may state the problem as follows: given a system S with n units or subsystems and given an observed set of outputs from S resulting from a set of tests, what is the fault pattern of the system? The failures that occur during the operation of S could be caused by one or more faulty units in S. Hence, the probability that a unit has a fault [21] should play an important part in the fault location process. Further, it should be recognized that this probability may change with the time of operation. Most of the fault location research developed for multiprocessor systems in the past two decades more or less resembles the concepts developed by the PMC model [27]. The main reason for this similarity is that the problems which have been defined in the area of system diagnosis are based on essentially the same set of simplifying assumptions, see the article by Dahbura [12]. In fact, those assumptions do not reflect the constraints of physical fault-tolerant designs too closely and various authors have attempted to modify the assumptions to correct this inconsistency, as listed above. Other approaches can be seen in [3, 4, 14, 17, 19, 27, 30, 32, 33], including work for a general multiprocessor system.

Evidently, any of the phenomena mentioned above, which have been introduced in the literature, might actually occur together, rather than in isolation, during the testing process. The only two things that we can do are to apply tests to the input and to gather the test results from the output; we have no means of governing random effects or directly monitoring the events that happen to the system during the tests. Thus, when utilizing stochastic methods to solve the fault location problem, one should really consider a generalized approach that includes all possible random phenomena. As a consequence, this paper proposes alternative methods taken from decision theory. Although a generalized solution to fault location using stochastic methods is NP-complete, we are able to design the B- (bitwise Bayesian) algorithm to locate the faults with linear run-time complexity, thus making it of interest in hard real-time applications. The reason that our solution can be so simple without sacrificing any information gathered from the tests is that we decompose the system's (global) Bayesian estimation into a bitwise Bayesian estimation, after the loss function has been introduced. The loss function that we choose is an admissible Bayesian decision rule [2] for fault location. In the light of the work of Malek [22], we use a simple probabilistic comparison-based model together with the Bayesian inference approach. We argue that a comparison-based model is more appropriate for a homogeneous system. Both techniques consider the whole fault set; thus, it is not necessary to bound the number of faulty units. In addition, the Bayesian fault location strategy also includes the faulty probability of every unit in the *a priori* information. Another Bayesian approach can be found in Lin and Shin [18], where the prior distribution of unit failures is used as the basis of the Bayesian decision rule. Their approach targets one fault at a time and uses repetitions to diagnose multiple faults.

2. Probabilistic Comparison-based Model

BASIC NOTATION

We consider an undirected test graph G(U,E) with vertex set U and edge set E. Vertices are denoted $\mathbf{u}_k$. Denote the status of unit $\mathbf{u}_k$ by ϕ_k, where ϕ_k is 1 (0) if $\mathbf{u}_k$ is faulty (fault-free), for $k = 0, 1, 2, ..., n$-1. Then, $\Phi = \phi_{n-1}\,\phi_{n-2} \cdots \phi_0$ denotes the (system) *fault pattern* of the n units. If we consider a system S with n units, we assume that one or more units may be faulty simultaneously. The faults in S will be identified by a fault pattern Φ and we denote the set of all fault patterns by $\Theta = \{\Phi_0, \Phi_1, \ldots, \Phi_{2^n-1}\}$. Each fault pattern $\Phi_j \in \Theta$ is assumed to be possible in S. A *test* is a procedure for identifying whether one of the UUTs is behaving normally or abnormally—in the comparison model—by means of the value it returns. We assume there is a test $T = \{t_1, t_2, ..., t_p\}$ consisting of p test tasks that can be applied to S. Individual tests may be *incomplete* in the sense that they may not always cause a faulty unit to return an incorrect result. This paper will be concerned with sequences $\mathrm{T} = \{T^{(k)}, k = 1, 2, ..., \tau\}$, where each $T^{(k)}$ is a collection of incomplete individual tasks t_j. Note that, Russell and Kime [28,29] suggested that it is hardly feasible to generate a complete test for the UUTs, so, as indicated by Dahbura [12], the most realistic approach is to assume tests are incomplete. Comparing tests $T^{(k)}$ and $T^{(k')}$, they may be replications or they may include different tasks belonging to the same class. The symbol $C = c_{m-1}\, c_{m-2} \cdots c_0$ denotes the global (or system) comparison pattern of m links, where c_ℓ, $\ell = m$-1, m-2, ..., 0, denotes the pairwise comparison result of ℓ-th link. When the units agree we set $c_\ell = 0$ and when the units disagree we set $c_\ell = 1$. We denote the set of all comparison patterns by $\Psi = \{C_0, C_1, \ldots, C_{2^m-1}\}$.

Suppose a given system S has n units with m comparison-test links and an undirected graph G = (U,E), where E = $\{(\mathbf{u}_i, \mathbf{u}_j) : \mathbf{u}_i, \mathbf{u}_j \in \mathrm{U}\}$ indicates the set of comparison assignments of the UUTs of the system. For example, take a complete graph of $n = 4$ units and $m = \binom{n}{2} = \binom{4}{2}$ links. The structure is displayed in Figure 1.

It is assumed that a test sequence **T**, as above, is performed by a system **S**. Each test $T^{(k)}$ consists of a set of jobs assigned to all the units $\mathbf{u}_i$. The comparison pattern observed as a result of test $T^{(k)}$ is denoted $C^{(k)}$. After observing a sequence of such patterns $\{C^{(k)} \in \Psi, k = 1,2,\ldots,\tau\}$, probabilistic analysis is applied to determine the faulty units. This analysis is described in remaining sections.

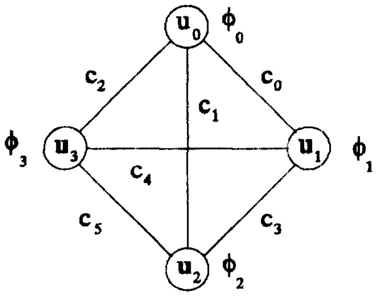

Figure 1

THE MODEL

Suppose the edge (link) c_ℓ connects $\mathbf{u}_i$ and $\mathbf{u}_j$. The behavior of the comparison test of two units $\mathbf{u}_i$ and $\mathbf{u}_j$ can be characterized and modeled using the following conditional probability test parameters:

- $p_\ell = \Pr(c_\ell = 0 \mid \phi_i = \phi_j = 0)$, the probability of agreement between fault-free units
- $q_\ell = \Pr(c_\ell = 1 \mid \phi_i \neq \phi_j)$, the probability of disagreement between a faulty and a fault-free unit, and
- $r_\ell = \Pr(c_\ell = 1 \mid \phi_i = \phi_j = 1)$, the probability of disagreement between faulty units.

To further simplify the analysis, we shall assume homogeneity. By ***homogeneous***, we mean that the UUTs are either identical or at least functionally equivalent, e.g., multiprocessor-based systems. This assumption implies symmetry in the definition of q_ℓ, i.e. $\Pr(c_\ell = 1 \mid \phi_i = 0, \phi_j = 1) = \Pr(c_\ell = 1 \mid \phi_i = 1, \phi_j = 0)$, and, further, it is reasonable to assume there are non-stochastic constants p, q, r such that $p_\ell = p$, $q_\ell = q$, $r_\ell = r$. Chang [6] and Chang *et al.*, [8,10] justifies that, in the comparison-based model, the components of C are statistically independent in the sense that $\Pr(C \mid \Phi) = \prod_{\ell=0}^{m-1} \Pr(c_\ell \mid \Phi)$. Hence, the conditional distribution of $\Pr(C|\Phi)$ can be evaluated as a function of p, q, and r. Blount [5] and Barsi [1] made a similar claim; however, the assumption is harder to justify in the PMC model, where each tester tests and decides the status of a subset of the UUTs. If c_ℓ connects $\mathbf{u}_i$ and $\mathbf{u}_j$, call c_ℓ a p-link when $\mathbf{u}_i$ and $\mathbf{u}_j$ are fault-free, a q-link when one of $\mathbf{u}_i$ and $\mathbf{u}_j$ is faulty and one is fault-free and an r-link when $\mathbf{u}_i$ and $\mathbf{u}_j$ are faulty.

2.1. ***Observation.*** If c_ℓ is a p-link for Φ, then $\Pr(c_\ell = 0 \mid \Phi) = p$ and $\Pr(c_\ell = 1 \mid \Phi) = 1 - p$. If c_ℓ is a q-link for Φ, then $\Pr(c_\ell = 1 \mid \Phi) = q$ and $\Pr(c_\ell = 0 \mid \Phi) = 1 - q$ and similarly for an r-link. Thus

$$\Pr(C \mid \Phi) = \prod_{\ell=0}^{m-1} \Pr(c_\ell \mid \Phi) = p^{n_p - d_p}(1-p)^{d_p} q^{n_q - d_q}(1-q)^{d_q} r^{n_r - d_r}(1-r)^{d_r} \quad (1)$$

where n_p, n_q and n_r are the numbers of p-, q- and r-links, respectively, $n_p + n_q + n_r = m$, and d_p, d_q and d_r are the numbers of misdiagnosed p-, q- and r-links in the comparison pattern C, respectively. Further, if we introduce the notation $\xi_p = (1-p)/p$, $\xi_q = (1-q)/q$ and $\xi_r = (1-r)/r$, then equation (1) simplifies to

$$\Pr(C \mid \Phi) = p^{n_p} q^{n_q} r^{n_r} \xi_p^{d_p} \xi_q^{d_q} \xi_r^{d_r} \quad (2)$$

An example of the use of this formula is as follows. Consider a system of four functionally identical units with a complete connection assignment. We examine one of the possible fault patterns:

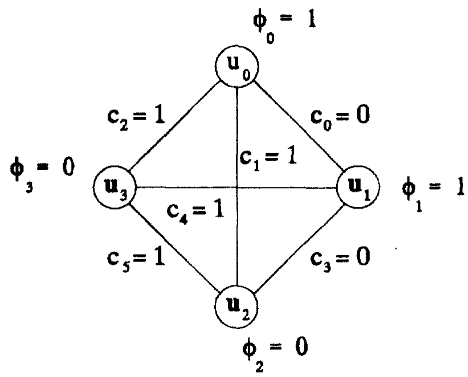

Figure 2

2.2. ***Example.*** If $\Phi = \phi_3\,\phi_2\,\phi_1\,\phi_0 = 0011$ and $C = c_5\,c_4\,c_3\,c_2\,c_1\,c_0 = 110110$, then it follows that

$$\Pr(C \mid \Phi) = \prod_{\ell=0}^{5} \Pr(c_\ell \mid \Phi) = (1-r)(1-q)q^3(1-p)$$

There are two faulty units (units $\mathbf{u}_0$ and $\mathbf{u}_1$) and three erroneous comparison results (c_0, c_3, c_5) as in Figure 2.

The objective of testing a system is to find out whether a failure exists in the system at the time of the test, and then to locate the failed unit if there is one. Hence, after the fault location process is completed, some level of repair or reconfiguration will have to be initiated.

Note that the conditional distributions $C \mid \Phi_j$ can be combined into a table listing all the values of $\Pr(C_i \mid \Phi_j)$, called the likelihood table (LT), which is a probabilistic comparison table and can be computed prior to the operation of the system. However, it is obvious that the storage size of the LT is $O(2^n * 2^m)$, e.g. refer to Table I at the end of the paper. Chang [6] develops an analytical method to avoid the need to store this enormous amount of data. The method requires $O(m)$ time to retrieve a data item or $O(\log m)$ time when prestored reference data is used. The number of data items retrieved at test time is small and has an absolute upper bound of τ, the number of tests. Example 4.7 provides an illustration.

3. Assignment of Multiple Test Sets

When diagnosing a system S, tests must be administered at some point in time. After the tests are run, the outcomes of the tests, i.e. the comparison patterns, are collected and analyzed. The whole process then turns to the fault location procedure which is intended to locate the faulty units. There is an inherently random nature to the comparison results of any given system for a variety of reasons, e.g. refer to [3-10,14,19,23,27-29,33]. It is therefore appropriate to apply the same or functionally equivalent tests many times to have more reliable data for diagnosis. Consequently, we apply to S the sequence **T** of τ tests $T^{(k)}$, defined in Section 2; then τ comparison results will form the observed output data (comparison syndrome): $\{C^{(s)} \in \Psi, s = 1, 2, \ldots, \tau\}$, where $C^{(s)}$ denotes the comparison pattern observed after the s-th test is applied. Here, each test task t in any $T^{(k)}$ is designed to cover partial functionality of the UUT, so that $T^{(k)}$ attempts to achieve pseudo-exhaustive testing [24]. The tests $T^{(k)}$ are repeated τ times, with variation of the individual tasks t within a test class, both to improve test coverage, since the tests may not be complete, and to take account of random effects in the test environment, e.g. situations (ii), (iii) and (vi) discussed in Section 1. Suppose the τ comparison results are conditionally independent, i.e.,

$$\Pr(C^{(1\ldots\tau)} \mid \Phi_j) = \prod_{s=1}^{\tau} \Pr(C^{(s)} \mid \Phi_j),$$

where $\Pr(C^{(s)} \mid \Phi_j)$ can be obtained from the comparison probabilistic table for $j = 0, 1, \ldots, 2^n-1$. Consequently, the posterior distribution can be evaluated by Bayes' theorem:

$$\begin{aligned}\Pr(\Phi_j \mid C^{(1)}C^{(2)}\ldots C^{(\tau)}) &= \frac{\Pr(C^{(1)}C^{(2)}\ldots C^{(\tau)} \mid \Phi_j)\cdot\Pr(\Phi_j)}{\Pr(C^{(1)}C^{(2)}\ldots C^{(\tau)})} \\ &= \frac{\Pr(C^{(1)}C^{(2)}\ldots C^{(\tau)} \mid \Phi_j)\cdot\Pr(\Phi_j)}{\sum_{l=0}^{2^n-1}\Pr(C^{(1)}C^{(2)}\ldots C^{(\tau)} \mid \Phi_l)\cdot\Pr(\Phi_l)}\end{aligned} \tag{3}$$

for $j = 0, 1, \ldots, 2^n - 1$.

To further simplify the notation in (3), we write $C^{(1\ldots\tau)} = C^{(1)}, C^{(2)}, \ldots, C^{(\tau)}$. As a rationale for the prior distribution on the parameter $\Phi \in \Theta$ of interest, we follow the suggestion of previous research in the field [16, 21, 25, 26, 31, 34, 35] and assume that the components of Φ, namely $\phi_{n-1}, \phi_{n-2}, \ldots, \phi_0$ are independent, identically distributed (i.i.d.), having an exponential distribution with parameter λ. This assumption is justified by the fact that all UUTs are homogeneous. The choice of the prior probability $\Pr(\Phi)$ does not affect the discussion in the next section. Hence the procedure is robust with respect to the choice of the prior distribution.

4. The Bayesian Analysis of Fault Location

In general, there are two possible ways to perform fault location using Bayesian analysis. One is analogous to classical inference methods that mostly deals with the posterior distribution. In classical inference methods, we may choose either point estimation or set estimation to perform the fault location process (see Chang [6] and Chang *et al.* [7-9]). The other introduces the idea of a loss function and turns the problem into one from decision theory. In this section the latter is studied.

We employ the Bayesian decision-theoretic approach, which enables us to estimate the fault status of each unit by doing point estimation with the choice of a reasonable loss function $\mathfrak{L}$ [2]. We claim that the distance measure is a reasonable measure for all misdiagnosed results. The loss function derived is computed bitwise from the global fault pattern. The reasons for choosing this loss function include computational efficiency and the fact that the center mean and mode of its distribution are the same.

To assist our Bayesian analysis, it is necessary to transform the likelihood table, $\Pr(C_i \mid \Phi_j)$, to a bitwise version of the likelihood table, $\Pr(C_i \mid \phi_k)$. As shown in Table II, the number of columns which was 2^n in a table such as Table I has become simply $2n$. In Section 2, we wrote ϕ_k for the fault status of unit $\mathbf{u}_k$ ($\phi_k = 0$ or 1) and now we also write $\phi_{j,k}$ for the status of unit $\mathbf{u}_k$ conditional upon the fault pattern being Φ_j, so that $\Phi_j = \phi_{j,n-1}\ldots\phi_{j,k}\ldots\phi_{j,0}$. We may extend the expressions derived in previous sections and consider the case $\phi_k = \delta$, where $\delta = 0$ or 1, to obtain:

$$\begin{aligned}\Pr(C_i \mid \phi_k = \delta) &= \frac{\Pr(C_i, \phi_k = \delta)}{\Pr(\phi_k = \delta)} \\ &= \frac{1}{\Pr(\phi_k = \delta)} \sum_{\{\Phi_j \in \Theta : \phi_{j,k} = \delta\}} \Pr(C_i, \Phi_j) \\ &= \frac{1}{\Pr(\phi_k = \delta)} \sum_{\{\Phi_j \in \Theta : \phi_{j,k} = \delta\}} \Pr(C_i \mid \Phi_j)\Pr(\Phi_j)\end{aligned} \tag{4}$$

The conditional probabilities $\Pr(C_i \mid \Phi_k)$, $k = 0,..., n-1$ in Table I are generated from a conditional probability distribution. Hence the column sums of the table are one but this is not true of the row sums. That is entirely as expected. Table II is generated by equation (4) and was validated by two different programs written separately in the C and Matlab languages. As in Table I, the tabulated values are multiplied by 10^4 for computational accuracy and readability. The shape of the distribution in each column is similar to that in Table I but the rate at which the values decrease is slower than in Table I. This observation is reasonable since the bitwise conditional distribution in Table II only specifies all the possible fault conditions of one unit, not all the units. Although this bitwise distribution seems to have a less pronounced shape than the global distribution in Table I, Table II is far smaller. Together with the bitwise method in this section, the small size of the table will reduce the complexity of the analysis markedly. Besides that, one can note that $\Pr(C_i \mid \phi_k=0)$ is not equal to $\Pr(C_i \mid \phi_k=1)$ in general. Such an equality, rarely occurring in practice, would yield an inconclusive test result and compromise the comparison steps in the diagnosis process (see step (3) of the *B*-algorithm in this section). If such an inconclusive test result were to occur it could be improved by upgrading the quality of the test tasks t or by increasing τ, the number of tests. The latter would be necessary if faults were intermittent.

POINT ESTIMATION

Use the observed comparison patterns to determine $\hat{\Phi}_{ML} \equiv \hat{\phi}_{n-1} \ldots \hat{\phi}_k \ldots \hat{\phi}_0 \in \Theta$, where $\hat{\Phi}_{ML}$ is the "generalized maximum likelihood estimate" of Φ, that is, the largest mode of the posterior distribution $\Pr(\Phi \mid C^{(1\ldots\tau)})$ (see Berger [2], p.133). This maximum likelihood estimate is expected to be unique. It is clearly $O(2^n)$ to examine all $\Phi_j \in \Theta$ to find the maximum of all the $\Pr(\Phi \mid C^{(1\ldots\tau)})$ if all Φ_j have to be examined. However, Chang and Lander [7, 8] gave a heuristic-based search algorithm to find $\hat{\Phi}_{ML}$. This algorithm has only $O(n)$ worst case complexity to locate the faults with a $1 - \alpha$ level of confidence.

SET ESTIMATION

It is possible to obtain the $100(1 - \alpha)\%$ highest posterior density (HPD) credible region for the random variable Φ, given some small real number α. To calculate the HPD we consider all subsets $\Gamma' \subset \Theta$ such that $\Pr(\Gamma' \mid C^{(1\ldots\tau)}) \geq 1 - \alpha$. Among these subsets Γ' we have to find the one with the highest density of posterior probability. This is taken to mean the subset such that MIN { $\Pr(\Phi_j \mid C^{(1\ldots\tau)}) : \Phi_j \in \Gamma'$ } is the largest, i.e. let

Table II. The simplified bitwise likelihood table of $\Pr(C_i \mid \phi_k)*10^4$ based on Table I, where $\Pr(\phi_k=0) = 0.8$, $\Pr(\phi_k=1) = 1 - 0.8 = 0.2$ for all ϕ_k (see Table I for the bit patterns for each C_i).

	$\phi_0=0$	$\phi_0=1$	$\phi_1=0$	$\phi_1=1$	$\phi_2=0$	$\phi_2=1$	$\phi_3=0$	$\phi_3=1$
C_0	3767	5	3767	5	3767	5	3767	5
C_1	208	40	208	40	218	1	218	1
C_2	208	40	218	1	208	40	218	1
C_3	12	360	101	6	101	6	101	3
C_4	208	40	218	1	218	1	208	40
C_5	12	360	101	6	101	3	101	6
C_6	12	360	101	3	101	6	101	6
C_7	2	3224	804	13	804	13	804	13
C_8	218	1	208	40	208	40	218	1
C_9	101	6	12	360	101	6	101	3
C_{10}	101	6	101	6	12	360	101	3
C_{11}	12	37	12	37	12	37	20	5
C_{12}	13	8	13	8	13	8	13	8
C_{13}	11	51	11	51	17	26	17	26
C_{14}	11	51	17	26	11	51	17	26
C_{15}	4	312	60	86	60	86	76	23
C_{16}	218	1	208	40	218	1	208	40
C_{17}	101	6	12	360	101	3	101	6
C_{18}	13	8	13	8	13	8	13	8
C_{19}	11	51	11	51	17	26	17	26
C_{20}	101	6	101	6	101	3	12	360
C_{21}	12	37	12	37	20	5	12	37
C_{22}	11	51	17	26	17	26	11	51
C_{23}	4	312	60	86	76	23	60	86
C_{24}	101	3	12	360	101	6	101	6
C_{25}	804	13	2	3224	804	13	804	13
C_{26}	17	26	11	51	11	51	17	26
C_{27}	60	86	4	312	60	86	76	23
C_{28}	17	26	11	51	17	26	11	51
C_{29}	60	86	4	312	76	23	60	86
C_{30}	52	214	52	214	52	214	52	214
C_{31}	15	674	15	674	159	99	159	99
C_{32}	218	1	218	1	208	40	208	40
C_{33}	13	8	13	8	13	8	13	8
C_{34}	101	6	101	3	12	360	101	6
C_{35}	11	51	17	26	11	51	17	26
C_{36}	101	6	101	3	101	6	12	360
C_{37}	11	51	17	26	17	26	11	51
C_{38}	12	37	20	5	12	37	12	37
C_{39}	4	312	76	23	60	86	60	86
C_{40}	101	3	101	6	12	360	101	6
C_{41}	17	26	11	51	11	51	17	26
C_{42}	804	13	804	13	2	3224	804	13
C_{43}	60	86	60	86	4	312	76	23
C_{44}	17	26	17	26	11	51	11	51
C_{45}	52	214	52	214	52	214	52	214
C_{46}	60	86	76	23	4	312	60	86
C_{47}	15	674	159	99	15	674	159	99
C_{48}	101	3	101	6	101	6	12	360
C_{49}	17	26	11	51	17	26	11	51
C_{50}	17	26	17	26	11	51	11	51
C_{51}	52	214	52	214	52	214	52	214
C_{52}	804	13	804	13	804	13	2	3224
C_{53}	60	86	60	86	76	23	4	312
C_{54}	60	86	76	23	60	86	4	312
C_{55}	15	674	159	99	159	99	15	674
C_{56}	20	5	12	37	12	37	12	37
C_{57}	76	23	4	312	60	86	60	86
C_{58}	76	23	60	86	4	312	60	86
C_{59}	159	99	15	674	15	674	159	99
C_{60}	76	23	60	86	60	86	4	312
C_{61}	159	99	15	674	159	99	15	674
C_{62}	159	99	159	99	15	674	15	674
C_{63}	49	404	49	404	49	404	49	404

$$\kappa(\Gamma') = \text{MIN} \{ \Pr(\Phi_j \mid C^{(1...\tau)}) : \Phi_j \in \Gamma' \},$$

then the HPD for Φ is the subset Γ which maximizes this κ, i.e. such that

$$\kappa(\Gamma) = \text{MAX} \{ \kappa(\Gamma') : \Pr(\Gamma' \mid C^{(1...\tau)}) \geq 1 - \alpha \}.$$

The computation cost of finding all such Γ' is exponential.

An alternative is to find a set Γ' for which the $\Pr(\Phi_j \mid C^{(1...\tau)})$ for all $\Phi_j \in \Gamma'$ are relatively large and use that Γ' as a reasonable replacement for Γ. Since $\hat{\Phi}_{ML}$ is the most likely system fault pattern, the other fault patterns $\Phi_j \in \Theta$ may be considered as misdiagnosed. Hence the number of misdiagnosed units can be used as a measure of distance of any fault pattern Φ_j from $\hat{\Phi}_{ML}$. As shown by other research found in Chang [6] and Chang and Lander [7, 8], $\Pr(\Phi_j \mid C^{(1...\tau)})$ decreases as the distance between Φ_j and $\hat{\Phi}_{ML}$. Hence we construct Γ' using only those Φ_j closest to $\hat{\Phi}_{ML}$. The distance between $\hat{\Phi}_{ML}$ and Φ can be defined as

$$d(\hat{\Phi}_{ML}, \Phi_j) \equiv \sum_{k=0}^{n-1} |\hat{\phi}_k - \phi_{j,k}|$$

so that $d(\hat{\Phi}_{ML}, \Phi_j) \in \{0, 1, ..., n\}$.

Furthermore, since $\hat{\phi}_k = 0$ or 1 and $\phi_{j,k} = 0$ or 1 only, it follows that

$$d(\hat{\Phi}_{ML}, \Phi_j) \equiv \sum_{k=0}^{n-1} |\hat{\phi}_k - \phi_{j,k}| = \sum_{k=0}^{n-1} (\hat{\phi}_k \oplus \phi_{j,k}) = \sum_{k=0}^{n-1} (\hat{\phi}_k - \phi_{j,k})^2$$

where $\oplus$ denotes the 'exclusive OR' operation.

The remaining step is to construct a $100(1 - \alpha)\%$ *credible region* for Φ which we expect approximates the HPD region. Although $\Pr(\Phi_j \mid C^{(1...\tau)})$ does not necessarily decrease as $d(\hat{\Phi}_{ML}, \Phi_j)$ increases, the fault pattern with fewer misdiagnosed links should appear more frequently. Therefore we include the Φ_j that have the smallest $d(\hat{\Phi}_{ML}, \Phi_j)$ first. In this manner, it is possible to find the *minimum* $h \in \{0, 1, 2, ..., n\}$ such that if we define $\Gamma \equiv \{\Phi_j \in \Theta : 0 \leq d(\hat{\Phi}_{ML}, \Phi_j) \leq h\}$ then $\Pr(\Gamma \mid C^{(1...\tau)}) \geq 1 - \alpha$. Thus, the region is a $100(1 - \alpha)\%$ credible region for Φ. Since Γ does not necessarily contain all $\Phi_j \in \Theta$ that have higher posterior density than all other $\Phi_{j'} \in \Gamma$, Γ cannot be assumed to be the $100(1 - \alpha)\%$ HPD credible region for Φ. However, the computation of Γ is a more efficient.

BAYESIAN DECISION THEORETIC APPROACH

We now turn to the decision theoretic approach to point estimation. A *decision rule* is a mapping from the test results $C^{(1...\tau)}$ to the fault patterns Φ_j: given a particular test result, the rule decides on a particular fault pattern. The Bayesian approach considers the true fault pattern which we denote by $\Phi = \phi_{n-1} \dots \phi_k \dots \phi_0$, although it is unknown, of course. We must then choose a reasonable loss function. In the discussion on set estimation in the previous subsection, we claimed that the distance measure is a reasonable measure of the amount of misdiagnosis in a pattern. Given the test result $C^{(1...\tau)}$ suppose an arbitrary decision rule assigns the fault pattern $\tilde{\Phi} = \tilde{\phi}_{n-1} \dots \tilde{\phi}_k \dots \tilde{\phi}_0 \in \Theta$, then consider the *loss function*:

$$\mathfrak{L}(\tilde{\Phi}, \Phi_j) \equiv d(\tilde{\Phi}, \Phi) \equiv \sum_{k=0}^{n-1} (\tilde{\phi}_k - \phi_k)^2 = \sum_{k=0}^{n-1} |\tilde{\phi}_k - \phi_k| = \sum_{k=0}^{n-1} (\tilde{\phi}_k \oplus \phi_k)$$

The statistical significance of this function is that, since ϕ_k and $\tilde{\phi}_k$ only take the values 0 and 1, it has the same properties as the square-error loss and absolute error loss. Moreover, the loss function is computationally practical since it can be evaluated by the 'exclusive OR' operation which is an efficient operation. This will shorten the unit computation time.

Given a loss function, we have to consider the fact that Φ is unknown and, in fact, every Φ_j is possible and given $C^{(1...\tau)}$, the probability of Φ being Φ_j is exactly the posterior probability $\Pr(\Phi_j \mid C^{(1...\tau)})$. We may define a risk function ρ as the expected loss given this probability distribution for Φ:

$$\rho(\tilde{\Phi}, \Phi) \equiv E[\mathfrak{L}(\tilde{\Phi}, \Phi) \mid C^{(1...\tau)}]$$

In this equation the conditional dependence is attached to the random variable Φ. The risk function may be expanded to explicit sums as follows but the sums here are over 2^n elements [6]:

$$\rho(\tilde{\Phi}, \Phi) = \frac{\sum_{\Phi \in \Theta} \mathfrak{L}(\tilde{\Phi}, \Phi) \cdot \Pr(C^{(1...\tau)} \mid \Phi) \cdot \Pr(\Phi)}{\sum_{\Phi \in \Theta} \Pr(c^{(1...\tau)} \mid \Phi) \cdot \Pr(\Phi)}$$

Now we may select the Bayesian decision rule which makes $C^{(1...\tau)}$ correspond to $\Phi_B^* = \phi_{n-1}^* \dots \phi_k^* \dots \phi_0^* \in \Theta$, the fault pattern that minimizes the risk function:

$$\rho(\Phi_B^*, \Phi) \equiv E[\mathfrak{L}(\Phi_B^*, \Phi) \mid C^{(1...\tau)}] = \sum_{k=0}^{n-1} \rho(\phi_k^*, \phi_k) \quad (5)$$

where the full derivation is given in Chang [6] and

$$\rho(\phi_k^*, \phi_k) \equiv E[(\phi_k^* \oplus \phi_k) \mid C^{(1...\tau)}]$$

We note that $\Phi_B^* = \phi_{n-1}^* \dots \phi_k^* \dots \phi_0^*$ minimizes $\rho(\Phi_B^*, \Phi)$

if and only if ϕ_k^* minimizes $\rho(\phi_k^*, \phi)$ for all $k = 0, 1, 2, \ldots, n-1$. Hence the complex global analysis of all the $\Phi \in \Theta$ is decomposed into a simple bitwise analysis. In other words, in order to compute the Φ_B^* assigned by the global Bayesian decision rule, it is sufficient to find all bitwise assignments $\phi_k^* = 0$ or 1 ($k = 0, 1, 2, \ldots, n-1$) that minimize

$$\rho(\phi_k^*, \phi_k) = E[(\phi_k^* \oplus \phi_k) \mid C^{(1\ldots\tau)}]$$

$$= \frac{\sum_{\delta=0,1} (\phi_k^* \oplus \delta) \Pr(C^{(1\ldots\tau)} \mid \phi_{j,k} = \delta) \cdot \Pr(\phi_{j,k} = \delta)}{\sum_{\delta=0,1} \Pr(C^{(1\ldots\tau)} \mid \phi_{j,k} = \delta) \cdot \Pr(\phi_{j,k} = \delta)}$$

or equivalently minimize the numerator

$$\Sigma_{\delta=0,1} (\phi_k^* \oplus \delta) \Pr(C^{(1\ldots\tau)} \mid \phi_k = \delta) \cdot \Pr(\phi_k = \delta)$$

for $\phi_k^* = 0$ or 1.

The complexity of analysis, by using this methodology, has been reduced dramatically since sums with 2^n terms are replaced by the sum of two terms! However, it may be reduced further by noting the following. Using the fact that ϕ_k^* and ϕ_k are binary variables we know that if $\phi_k^* = 0$, then $\phi_k^* \oplus \phi_k = \phi_k$. Hence

$$\rho(\phi_k^* = 0, \phi_k) = E[\phi_k \mid C^{(1\ldots\tau)}] = \Pr(\phi_k = 1 \mid C^{(1\ldots\tau)}).$$

Similarly, if $\phi^*_k = 1$, then $\phi^*_k \oplus \phi_k = (\phi_k)' = 1 - \phi_k$, so that

$$\begin{aligned}\rho(\phi_k^* = 1, \phi_k) &= E[1 - \phi_k \mid C^{(1\ldots\tau)}] \\ &= 1 - \Pr(\phi_k = 1 \mid C^{(1\ldots\tau)}) \\ &= \Pr(\phi_k = 0 \mid C^{(1\ldots\tau)}).\end{aligned}$$

Therefore, $\rho(1, \phi_k) < \rho(0, \phi_k)$ if and only if $\Pr(\phi_k = 1 \mid C^{(1\ldots\tau)}) > \Pr(\phi_k = 0 \mid C^{(1\ldots\tau)})$.

Note that $\Pr(\phi_k = 1 \mid C^{(1\ldots\tau)}) + \Pr(\phi_k = 0 \mid C^{(1\ldots\tau)}) = 1$, so $\Pr(\phi_k = 1 \mid C^{(1\ldots\tau)}) > \Pr(\phi_k = 0 \mid C^{(1\ldots\tau)})$ if and only if $\Pr(\phi_k = 1 \mid C^{(1\ldots\tau)}) > 0.5$. Furthermore, since

$$\Pr(\phi_k = 1 \mid C^{(1\ldots\tau)}) = \frac{\Pr(C^{(1\ldots\tau)} \mid \phi_k = 1) \Pr(\phi_k = 1)}{\sum_{\delta=0,1} \Pr(C^{(1\ldots\tau)} \mid \phi_k = \delta) \cdot \Pr(\phi_k = \delta)}$$

$E(\phi_k \mid C^{(1\ldots\tau)}) = \Pr(\phi_k = 1 \mid C^{(1\ldots\tau)}) > 0.5$ if and only if

$$\Pr(C^{(1\ldots\tau)} \mid \phi_k = 1) \cdot \Pr(\phi_k = 1) > \Pr(C^{(1\ldots\tau)} \mid \phi_k = 0) \cdot \Pr(\phi_k = 0).$$

The decision process outlined above may be summarized as the following bitwise version of the Bayesian decision algorithm (or B-algorithm) to perform fault location:

4.1 ***Algorithm.*** The B-algorithm for fault location: Choose the components $\phi_{n-1}^* \ldots \phi_k^* \ldots \phi_0^*$ of Φ_B^* as follows:

For each $k = n-1, \ldots, 0$:

(1) if $\Pr(C^{(1\ldots\tau)} \mid \phi_k = 1) \cdot \Pr(\phi_k = 1) >$
$\Pr(C^{(1\ldots\tau)} \mid \phi_k = 0) \cdot \Pr(\phi_k = 0)$ then choose $\phi_k^* = 1$,

(2) if $\Pr(C^{(1\ldots\tau)} \mid \phi_k = 1) \cdot \Pr(\phi_k = 1) <$
$\Pr(C^{(1\ldots\tau)} \mid \phi_k = 0) \cdot \Pr(\phi_k = 0)$ then choose $\phi_k^* = 0$,

(3) the test is inconclusive if

$$\Pr(C^{(1\ldots\tau)} \mid \phi_k = 1) \cdot \Pr(\phi_k = 1) = \Pr(C^{(1\ldots\tau)} \mid \phi_k = 0) \cdot \Pr(\phi_k = 0).$$

If a set estimation is desired, the procedures developed in the previous subsection can be applied to obtain a $100(1 - \alpha)\%$ credible region for Φ by substituting Φ_B^* for $\hat{\Phi}_{ML}$.

ANALYSIS AND DISCUSSION

4.2 ***Remark.*** The run-time complexity of the B-algorithm is $O(n)$. This use of bitwise analysis gives a dramatic improvement over previous results in the literature and is a very good result in theory and practice. From the theoretical point of view, it reduces the complexity of the Bayesian analysis; this complexity is one of the main deficiencies of the Bayesian approach. In practice, this method outperforms the other methods that have been proposed in literature since it does not really require any assumptions. The only disadvantage incurred is in the generation of the probabilistic comparison table. Again, this is a one time computation.

4.3 ***Remark.*** The comparison-based probabilistic model and the Bayesian inference algorithm make the method complete in the statistical sense, since the model together with the B-algorithm can accommodate all possible random effects. It is practical because the computations involved are simple binary operations, with linear complexity. The necessary data is directly observable during the testing process.

4.4 ***Remark.*** Since the loss function is a squared error loss, the posterior mean $E[\phi_k \mid C^{(1\ldots\tau)}]$ is the Bayes rule [2, p.161, result 3]. However, because ϕ_k^* is binary, it is reasonable to choose $\phi_k^* = 1$ if $E[\phi_k \mid C^{(1\ldots\tau)}] > 0.5$ (the same decision rule as step 1). If the loss function generalized to some kind of weighted squared error loss, then the Bayes rule could also be obtained [2, p.161, result 4]. However, the form of the Bayes rule is not as simple as the procedure proposed above. In addition, since the loss function is also the absolute error loss, the median of $\Pr(\phi_k \mid C^{(1\ldots\tau)})$ also provides a Bayes rule [2, p.162, result 5]. Again, since ϕ_k^* is binary, this is equivalent to the procedure we proposed above. If the loss function is generalized to a linear loss, the decision rule can be obtained in a similar fashion [2, p.162, result 6]. So, our proposed procedures are valid for a class of the usual loss functions. Furthermore, as ϕ_k and ϕ_k^* are binary, our suggested loss function can also be represented by the exclusive OR operation, which will save the unit computation time.

4.5 ***Remark.*** The procedures we proposed are consistent with one's intuition. Besides, the decision rule that we have proposed has the important property of positive Bayes decision rules, namely admissibility [2]. This gives extra confidence in our intuition.

4.6 ***Remark.*** The B-algorithm accommodates all possible faulty *and fault-free* systems under test, without any increase in complexity when the fault-free state is diagnosed, permitting the algorithm to be applied to monitor a system periodically. Further, the method is able to distinguish *truly* faulty units from those which appear faulty due to the imperfect environment, thus eliminating unnecessary hardware replacement or reconfiguration before the system recovery process performs rollback to a fault-free state.

An example of locating faults of a system S under diagnosis is given as follows:

4.7 ***Example.*** Consider a four-unit system S as shown in Figure 2. Suppose $\tau = 10$ and the following comparison patterns are observed: C_{40}, C_{42}, C_{42}, C_{42}, C_{40}, C_{10}, C_{34}, C_0, C_{57}, C_{42}. These patterns are sorted and counted as follows:

Patterns (sorted)	C_0	C_{10}	C_{34}	C_{40}	C_{42}	C_{57}
count	1	1	1	2	4	1

According to our B-algorithm for fault location, all we need to do is to sum $\Pr(C^{(1...10)} \mid \phi_k = 1)\cdot\Pr(\phi_k = 1)$ from Table II and compare it with the corresponding sum of $\Pr(C^{(1...10)} \mid \phi_k = 0)\cdot\Pr(\phi_k = 0)$. Repeat the same step until all the bits are done, i.e. $k = 0$ to 3, in this case. This also implies the bitwise computations are carried out only on the relevant ϕ_k column. Let $k = 0$ and first compute $\phi_0 = 1$, i.e.

$$\Pr(C^{(1...10)} \mid \phi_0 = 1)\cdot\Pr(\phi_0 = 1)$$
$$= \Pr(C_0 \mid \phi_0 = 1)\cdot\Pr(C_{10} \mid \phi_0 = 1)\cdot\Pr(C_{34} \mid \phi_0 = 1)\cdot [\Pr(C_{40} \mid \phi_0 = 1)]^2\cdot[\Pr(C_{42} \mid \phi_0 = 1)]^4\cdot \Pr(C_{57} \mid \phi_0 = 1)\cdot\Pr(\phi_0 = 1)$$
$$= 10^{-40}\cdot[\,5\cdot6\cdot6\cdot3^2\cdot13^4\cdot23]\cdot(0.2)$$

For the case of $\phi_0 = 0$, we have

$$\Pr(C^{(1...10)} \mid \phi_0 = 0)\Pr(\phi_0 = 0)$$
$$= \Pr(C_0 \mid \phi_0 = 0)\cdot\Pr(C_{10} \mid \phi_0 = 0)\cdot\Pr(C_{34} \mid \phi_0 = 0)\cdot [\Pr(C_{40} \mid \phi_0 = 0)]^2\cdot[\Pr(C_{42} \mid \phi_0 = 0)]^4\cdot \Pr(C_{57} \mid \phi_0 = 0)\cdot\Pr(\phi_0 = 0)$$
$$= 10^{-40}\cdot[\,3767\cdot101\cdot101\cdot101^2\cdot804^4\cdot76]\cdot(0.8)$$

Since $\Pr(C^{(1...10)} \mid \phi_0 = 1)\cdot\Pr(\phi_0 = 1)$ is obviously smaller than $\Pr(C^{(1...10)} \mid \phi_0 = 0)\cdot\Pr(\phi_0 = 0)$, we choose $\phi_0^* = 0$. In the next step, k is incremented by one and the process repeats. We conclude from the iterations of the B-algorithm that the fault pattern of the system is $\Phi^* = \phi_3^*\phi_2^*\phi_1^*\phi_0^* = 0100 = \Phi_4$.

5. Conclusions

By utilizing the simple probabilistic model defined, we propose a more practical Bayesian procedure for handling fault location. Since it is easier to compare the test results among units, our model is comparison-based. The approach taken in this paper is more complete than that of many authors because we do not need to assume any conditions such as permanently faulty units, complete tests, perfect environments, or non-malicious situations. It is clear that the proposed bitwise Bayes decision rule has good theoretical properties and a linear run time complexity. It is also easy to understand and operate because the B-algorithm, the decision algorithm, is consistent with one's intuition. All these benefits make this approach appealing in theory and practice. To sum up, the Bayesian decision-theoretic approach in Section 4 decomposes the enormous global calculations into simple bitwise calculations. Hence the B-algorithm provides an efficient diagnostic process to detect faults and their location on distributed system. Because the approach is based on the more complete and practical probabilistic model in Section 2, it is more realistic than the previous approaches in the literature.

Acknowledgement: The authors would like to express their appreciation to Professor Jiunn T. Hwang for his helpful discussions and suggestions.

References

[1] F. Barsi, "Probabilistic syndrome decoding in self-diagnosable digital systems," *Digital Processes*, Vol. 7, pp.33-46, 1981.

[2] J. O. Berger, *Statistical decision theory and Bayesian analysis, 2nd ed.*, Springer-Verlag, New York, 1985.

[3] P. Berman and A. Pelc, "Distributed probabilistic fault diagnosis for multiprocessor systems," *20th Symp. on Fault-Tolerant Computing*, pp. 340-346, 1990.

[4] D. M. Blough, et. al., "Fault diagnosis for sparsely interconnected multiprocessor systems," *19th Symp. on Fault-Tolerant Computing*, pp. 62-69, June 1989.

[5] M. Blount, "Probabilistic treatment of diagnosis in digital

systems," *7th Symp. on Fault-Tolerant Computing*, pp. 72-77, June 1977.

[6] Y. L. C. Chang, "Characterization and Analysis of the Random Nature of Fault Location," Ph.D. Dissertation, State University of New York at Binghamton, Oct. 1991.

[7] Y. L. C. Chang and L. Lander, "Classical inference methods for fault location in homogeneous systems," *Int. Conf. on Reliability, Quality Control, and Risk Assessment*, Int. Assoc. of Science and Tech. for Development and IEEE Reliability Soc., Nov. 4-6, 1992.

[8] Y. L. C. Chang and L. Lander, "An inference design for fault location in real-time control systems," *J. of Systems and Software*, special issue on Fault Tolerance in Real-Time Systems, Nov. 1993.

[9] Y. L. C. Chang, and L. Lander, "Management of testing policies for fault location of failures dependent upon operation-time in multiprocessor systems," *IEEE Third Int. Workshop on Responsive Computer Systems*, Sep. 29-Oct. 1, 1993.

[10] Y. L. C. Chang, L. Lander, H. Lu, and M. Wells, "Bayesian Inference for Fault Diagnosis in Real-Time Distributed Systems", *IEEE 2nd Asian Test Symp. (ATS'93)*, Nov. 16-18, 1993, Beijing, China.

[11] K. Y. Chwa and S. L. Hakimi, "Schemes for fault tolerant computing: a comparison of modularly redundant and t-diagnosable systems," *Information and Control*, Vol. 49, pp. 212-238, June 1981.

[12] A. T. Dahbura, K. K. Sabnani and L. L. King, "The comparison approach to multiprocessor fault diagnosis," *IEEE Trans. Comp.*, Vol. C-36, no. 3, pp. 373-377, 1987.

[13] D. Fussell and S. Rangarajan, "Probabilistic diagnosis of multiprocessor systems with arbitrary connectivity," *19th Symp. on Fault-Tolerant Computing*, pp. 560-565, June 1989.

[14] R. Gupta and I. V. Ramakrishnan, "System-level fault diagnosis in malicious environments," *17th Symp. on Fault-Tolerant Computing*, pp. 184-189, July 1987.

[15] S. L. Hakimi and A. T. Amin, "Characterization of the connection assignment of diagnosable systems," *IEEE Trans. Comp.*, Vol. C-23, no. 1, pp. 86-88, Jan. 1974.

[16] P. K. Lala, *Fault tolerance and testable hardware design*, Prentice-Hall, New Jersey, 1985.

[17] L. Lander and Y. L. C. Chang, "Generation and analysis of probabilistic diagnostic information in homogeneous systems," *Proc. Int. Conf. on Reliability, Quality Control and Risk Assessment*, Int. Assoc. of Science and Tech. for Development, Oct. 13-15, 1993.

[18] S. Lee and K. G. Shin, "Optimal multiple syndrome probabilistic diagnosis," *Proc. of 20th Symp. on Fault-Tolerant Computing*, pp. 324-331, 1990.

[19] T. H. Lin and K. G. Shin, "Location of a faulty module in a computing system," *IEEE Trans. Comp.*, Vol. 39, no. 2, pp. 182-194, 1990.

[20] J. Maeng and M. Malek, "A comparison connection assignment for self-diagnosis of multiprocessor systems," *11th Symp. on Fault-Tolerant Computing*, pp. 173-175, June 1981.

[21] S. Maheshwari and S. Hakimi, "On models for diagnosable systems and probabilistic fault diagnosis," *IEEE Trans. Comp.*, Vol. C-25, no. 3, pp.228-236, 1976.

[22] M. Malek, "A comparison connection assignment for diagnosis of multiprocessor systems," *10th Symp. on Fault-Tolerant Computing*, pp. 31-36, June 1980.

[23] S. Mallela and G. M. Masson, "Diagnosable systems for intermittent faults," *IEEE Trans. Comp.*, Vol. C-27, pp. 560-566, June 1978.

[24] E. J. McCluskey, "Verification testing—a pseudoexhaustive test technique," *ibid.* Vol. C-33, no. 6, pp. 541-546, June 1984.

[25] Y. W. Ng, "Reliability modeling and analysis for fault-tolerant computers," Ph.D. dissertation, Dept. of Computer Science, UCLA, 1976.

[26] D. K. Pradhan, ed., *Fault-tolerant computing: theory and techniques*, chapter 8, Vol. II, Prentice-Hall, Englewood Cliffs, NJ, 1986.

[27] F. P. Preparata, G. Metze and R. T. Chien, "On the connection assignment problem of diagnosable systems," *IEEE Trans. Comp.*, Vol. EC-16, no. 6, pp. 848-854, Dec. 1967.

[28] J. Russell and C. R. Kime, "System fault diagnosis: closure and diagnosability with repair," *IEEE Trans. Comp.*, Vol. C-24, no. 11, pp. 1078-1088, Nov. 1975.

[29] J. Russell and C. R. Kime, "System fault diagnosis: masking, exposure, and diagnosability without repair," *IEEE Trans. Comp.*, Vol C-24, no. 12, pp. 1155-1161, Dec. 1975.

[30] A. Sengupta and A. T. Dahbura, "On self-diagnosable multiprocessor systems: diagnosis by the comparison approach," *19th Symp. on Fault-Tolerant Computing*, pp. 54-61, June 1989.

[31] D. P. Siewiorek and R. S. Swarz, *The theory and practice of reliable system design*, Digital Press, Bedford, MA, 1982.

[32] L. Simoncini and A.D. Friedman, "Incomplete fault coverage in modular multiprocessor systems," *Proc. of ACM Annual Conf.*, pp. 210-216, Dec. 1978.

[33] J. E. Smith, "Universal system diagnosis algorithms," *IEEE Trans. Comp.*, Vol. C-28, no. 5, pp. 374-378, May 1979.

[34] G. Sullivan, "System-level fault diagnosability in probabilistic and weighted models," *Proc. of 17th Symp. on Fault-Tolerant Computing*, pp. 190-195, July 1987.

[35] K. S. Trivedi, *Probability and statistics with reliability, queuing, and computer science applications*, Prentice-Hall, Englewood Cliffs, NJ, 1982.

Table I. The likelihood table of $\Pr(C_i|\Phi_j)*10^4$, where p = 0.95, q = 0.9, r = 0.75.

	Φ_0 0000	Φ_1 0001	Φ_2 0010	Φ_3 0011	Φ_4 0100	Φ_5 0101	Φ_6 0110	Φ_7 0111	Φ_8 1000	Φ_9 1001	Φ_{10} 1010	Φ_{11} 1011	Φ_{12} 1100	Φ_{13} 1101	Φ_{14} 1110	Φ_{15} 1111
C_0 =000000	7351	9	9	0	9	0	0	0	9	0	0	0	0	0	0	2
C_1 =000001	387	77	77	1	0	2	2	0	0	2	2	0	0	1	1	7
C_2 =000010	387	77	0	2	77	1	2	0	0	2	0	1	2	0	1	7
C_3 =000011	20	694	4	6	4	6	19	1	0	19	0	4	0	4	13	22
C_4 =000100	387	77	0	2	0	2	0	1	77	1	2	0	2	0	1	7
C_5 =000101	20	694	4	6	0	19	0	4	4	6	19	1	0	4	13	22
C_6 =000110	20	694	0	19	4	6	0	4	4	6	0	4	19	1	13	22
C_7 =000111	1	6250	0	58	0	58	1	13	0	58	1	13	1	13	114	66
C_8 =001000	387	0	77	2	77	2	1	0	0	0	2	1	2	1	0	7
C_9 =001001	20	4	694	6	4	19	6	1	0	0	19	4	0	13	4	22
C_{10}=001010	20	4	4	19	694	6	6	1	0	0	0	13	19	4	4	22
C_{11}=001011	1	37	37	58	37	58	58	4	0	1	1	38	1	38	38	66
C_{12}=001100	20	4	4	19	4	19	0	4	4	0	19	4	19	4	4	22
C_{13}=001101	1	37	37	58	0	173	0	13	0	0	173	13	1	38	38	66
C_{14}=001110	1	37	0	173	37	58	0	13	0	0	1	38	173	13	38	66
C_{15}=001111	0	329	2	519	2	519	3	38	0	3	9	114	9	114	342	198
C_{16}=010000	387	0	77	2	0	0	2	1	77	2	1	0	2	1	0	7
C_{17}=010001	20	4	694	6	0	0	19	4	4	19	6	1	0	13	4	22
C_{18}=010010	20	4	4	19	4	0	19	4	4	19	0	4	19	4	4	22
C_{19}=010011	1	37	37	58	0	0	173	13	0	173	0	13	1	38	38	66
C_{20}=010100	20	4	4	19	0	0	0	13	694	6	6	1	19	4	4	22
C_{21}=010101	1	37	37	58	0	1	1	38	37	58	58	4	1	38	38	66
C_{22}=010110	1	37	0	173	0	0	1	38	37	58	0	13	173	13	38	66
C_{23}=010111	0	329	2	519	0	3	9	114	2	519	3	38	9	114	342	198
C_{24}=011000	20	0	694	19	4	0	6	4	4	0	6	4	19	13	1	22
C_{25}=011001	1	0	6250	58	0	1	58	13	0	1	58	13	1	114	13	66
C_{26}=011010	1	0	37	173	37	0	58	13	0	1	0	38	173	38	13	66
C_{27}=011011	0	2	329	519	2	3	519	38	0	9	3	114	9	342	114	198
C_{28}=011100	1	0	37	173	0	1	0	38	37	0	58	13	173	38	13	66
C_{29}=011101	0	2	329	519	0	9	3	114	2	3	519	38	9	342	114	198
C_{30}=011110	0	2	2	1558	2	3	3	114	2	3	3	114	1558	114	114	198
C_{31}=011111	0	17	17	4675	0	27	27	342	0	27	27	342	82	1025	1025	593
C_{32}=100000	387	0	0	0	77	2	2	1	77	2	2	1	1	0	0	7
C_{33}=100001	20	4	4	0	4	19	19	4	4	19	19	4	0	4	4	22
C_{34}=100010	20	4	0	0	694	6	19	4	4	19	0	13	6	1	4	22
C_{35}=100011	1	37	0	0	37	58	173	13	0	173	1	38	0	13	38	66
C_{36}=100100	20	4	0	0	4	19	0	13	694	6	19	4	6	1	4	22
C_{37}=100101	1	37	0	0	0	173	1	38	37	58	173	13	0	13	38	66
C_{38}=100110	1	37	0	1	37	58	1	38	37	58	1	38	58	4	38	66
C_{39}=100111	0	329	0	3	2	519	9	114	2	519	9	114	3	38	342	198
C_{40}=101000	20	0	4	0	694	19	6	4	4	0	19	13	6	4	1	22
C_{41}=101001	1	0	37	0	37	173	58	13	0	1	173	38	0	38	13	66
C_{42}=101010	1	0	0	1	6250	58	58	13	0	1	1	114	58	13	13	66
C_{43}=101011	0	2	2	3	329	519	519	38	0	9	9	342	3	114	114	198
C_{44}=101100	1	0	0	1	37	173	0	38	37	0	173	38	58	13	13	66
C_{45}=101101	0	2	2	3	2	1558	3	114	2	3	1558	114	3	114	114	198
C_{46}=101110	0	2	0	9	329	519	3	114	2	3	9	342	519	38	114	198
C_{47}=101111	0	17	0	27	17	4675	27	342	0	27	82	1025	27	342	1025	593
C_{48}=110000	20	0	4	0	4	0	19	13	694	19	6	4	6	4	1	22
C_{49}=110001	1	0	37	0	0	1	173	38	37	173	58	13	0	38	13	66
C_{50}=110010	1	0	0	1	37	0	173	38	37	173	0	38	58	13	13	66
C_{51}=110011	0	2	2	3	2	3	1558	114	2	1558	3	114	3	114	114	198
C_{52}=110100	1	0	0	1	0	1	1	114	6250	58	58	13	58	13	13	66
C_{53}=110101	0	2	2	3	0	9	9	342	329	519	519	38	3	114	114	198
C_{54}=110110	0	2	0	9	2	3	9	342	329	519	3	114	519	38	114	198
C_{55}=110111	0	17	0	27	0	27	82	1025	17	4675	27	342	27	342	1025	593
C_{56}=111000	1	0	37	1	37	1	58	38	37	1	58	38	58	38	4	66
C_{57}=111001	0	0	329	3	2	9	519	114	2	9	519	114	3	342	38	198
C_{58}=111010	0	0	2	9	329	3	519	114	2	9	3	342	519	114	38	198
C_{59}=111011	0	0	17	27	17	27	4675	342	0	82	27	1025	27	1025	342	593
C_{60}=111100	0	0	2	9	2	9	3	342	329	3	519	114	519	114	38	198
C_{61}=111101	0	0	17	27	0	82	27	1025	17	27	4675	342	27	1025	342	593
C_{62}=111110	0	0	0	82	17	27	27	1025	17	27	27	1025	4675	342	342	593
C_{63}=111111	0	1	1	246	1	246	246	3075	1	246	246	3075	246	3075	3075	1780