IEICE TRANSACTIONS

on Fundamentals of Electronics, Communications and Computer Sciences

VOL. E101-A NO. 12
DECEMBER 2018



A PUBLICATION OF THE ENGINEERING SCIENCES SOCIETY
The Institute of Electronics, Information and Communication Engineers
Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3 chome, Minato-ku, TOKYO, 105-0011 JAPAN

PAPER *Special Section on Information Theory and Its Applications*

# Probabilistic Fault Diagnosis and its Analysis in Multicomputer Systems*

**Manabu KOBAYASHI**†a), ***Member***, **Toshiyasu MATSUSHIMA**†, ***and*** **Shigeichi HIRASAWA**†, ***Fellows***

**SUMMARY** F.P. Preparata et al. have proposed a fault diagnosis model to find all faulty units in the multicomputer system by using outcomes which each unit tests some other units. In this paper, for probabilistic diagnosis models, we show an efficient diagnosis algorithm to obtain a posteriori probability that each of units is faulty given the test outcomes. Furthermore, we propose a method to analyze the diagnostic error probability of this algorithm.
***key words:*** *multicomputer systems, system-level fault diagnosis, probabilistic fault diagnosis, intermittent faults, density evolution*

## 1. Introduction

The problem of finding faulty units in the multicomputer/multiprocessor system has been investigated for a large variety of models. F.P. Preparata et al. have proposed a fault diagnosis model to find all faulty units by using outcomes that each unit independently tests some other units [1]. Then the outcome which a faulty unit tests cannot be trusted in this model. Under the assumption that the outcome which a fault-free unit tests other fault-free units can be trusted, the diagnosis algorithms to identify faulty units have been proposed in many literatures [2]–[11]. Furthermore, S. Mallela et al. have proposed the intermittent fault model that the test outcome is probabilistically incorrect when a fault-free unit tests a faulty unit [12]. A general probabilistic fault model including the intermittent fault model has been proposed by M. Blount [13]. Diagnosis algorithms for this probabilistic fault model have been proposed by D.M. Blough et al. [15], [16], S. Lee et al. [17] and M. Kobayashi et al. [18]. These models are called a directed graph model since the relations which each unit tests other units can be represented as a directed graph.

On the other hand, another model, which is called an undirected model or a comparison-based model, has been investigated in several literatures [19]–[26]. In this model, two units which are assigned to the edge of an undirected graph carry out the same job, and their outputs are compared. The diagnosis algorithm identifies faulty units by using these comparison outcomes.

Furthermore, a diagnosis method which finds all faulty units at once by using all test outcomes is called a one-step diagnosis. In contrast, a sequential diagnosis iterates to diagnose and repair after units estimated as faulty are repaired (or replaced with new units). In an adaptive diagnosis, additionally, test assignments are dynamically determined after checking the results of previous test outcomes.

In this paper, we aim at a one-step diagnosis of a probabilistic diagnosis model in a directed graph model. First, we show an efficient diagnosis algorithm which calculates a posteriori probability that each of units is faulty given the test outcomes. This algorithm is essentially identical to the Sum-Product algorithm [28] which can be used for more general case in machine learning theory and coding theory. Furthermore, we propose a method to analyze the diagnostic error probability of this algorithm. Finally, we show the results of the computer simulations and numerical results of the diagnostic error probability for the evaluation of effectiveness.

## 2. Fault Diagnosis System and Probabilistic Model

A system consists of a set of $N$ units**, and for each unit the distinct ID number from among $1, 2, \ldots, N$ is allocated for identification. Each unit has communication links to other several units and is assumed to be capable of testing faulty/fault-free status of units with communication links. Then the system is modeled by a directed graph $G = (U, E)$, where $U = \{1, 2, \ldots, N\}$ is the set of vertices representing the units, and each edge $(j, i) \in E$ represents that a unit $j$ tests a unit $i$. Faulty/fault-free status of a unit $i \in U$ will be denoted by $x_i$, where $x_i = 1$ if $i$ is faulty and $x_i = 0$ if $i$ is fault-free. For each edge $(j, i) \in E$, test outcome is represented by $s_{ji}$ such that $s_{ji} = 1$ if $i$ fails $j$'s test and $s_{ji} = 0$ if $i$ passes $j$'s test, i.e. $s_{ji} = 1$ if a unit $j$ evaluates that a unit $i$ is faulty, otherwise $s_{ji} = 0$. Note that a test outcome $s_{ji}$ may be incorrect, i.e. $x_i \neq s_{ji}$. Let $\mathbf{s}$ denote a vector whose elements consist of $s_{ji}$ for all $(j, i) \in E$, and we define $\mathbf{x} = (x_1, \ldots, x_N) \in \{0, 1\}^N$. Given a testing graph $G = (U, E)$ and all test outcomes $\mathbf{s}$, a fault diagnosis algorithm estimates $\mathbf{x}$.

For a given unit $i \in U$, let $\Gamma(i)$ denote the set of units that $i$ tests, i.e. $\Gamma(i) = \{k \in U | (i, k) \in E\}$. Inversely, let $\Gamma^{-1}(i)$ denote the set of units that test $i$, i.e. $\Gamma^{-1}(i) = \{j \in U | (j, i) \in E\}$.

**Example 1:** In Fig. 1, we show an example of the case

Manuscript received January 29, 2018.
†The authors are with Waseda University, Tokyo, 169-8555 Japan.
*This paper was partially presented at IEEE SMC 2011 [27].
a) E-mail: mkoba@waseda.jp
DOI: 10.1587/transfun.E101.A.2072

**A unit represents a computer (or a processor) in the multicomputer (or multiprocessor) system.

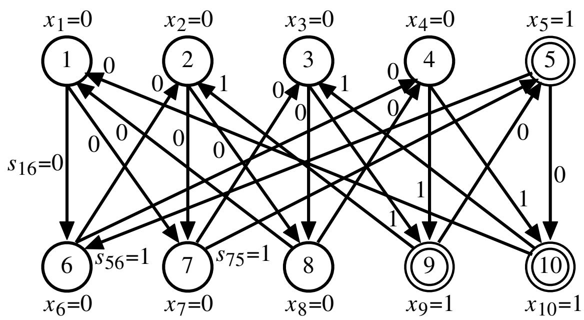


**Fig. 1** An example of the fault diagnosis system.

where units 5, 9 and 10 are faulty for $N = 10$. The number in the circle represents the unit ID. In this case, $G = (U, E)$ where $U = \{1, 2, \dots, 10\}$ and $E = \{(1, 6), (1, 7), (2, 7), (2, 8), (3, 8), (3, 9), (4, 9), (4, 10), (5, 6), (5, 10), (6, 2), (6, 4), (7, 3), (7, 5), (8, 1), (8, 4), (9, 2), (9, 5), (10, 1), (10, 3)\}$. Therefore, $\Gamma(1) = \{6, 7\}$, $\Gamma(2) = \{7, 8\}$, $\Gamma(3) = \{8, 9\}$, $\Gamma^{-1}(1) = \{8, 10\}$, $\Gamma^{-1}(2) = \{6, 9\}$, $\Gamma^{-1}(3) = \{7, 10\}$ and so on. Since only units 5, 9 and 10 are faulty, $\mathbf{x} = (x_1, x_2, \dots, x_{10}) = (0, 0, 0, 0, 1, 0, 0, 0, 1, 1)$. In Fig. 1, the label of each edge $(j, i) \in E$ represents the test outcome $s_{ji}$. Therefore, a vector of all test outcomes $\mathbf{s}$ is as follows:

$$\begin{aligned}\mathbf{s} &= (s_{16}, s_{17}, s_{27}, s_{28}, s_{38}, s_{39}, s_{49}, s_{4\,10}, s_{56}, s_{5\,10},\\ &\quad s_{62}, s_{64}, s_{73}, s_{75}, s_{81}, s_{84}, s_{92}, s_{95}, s_{10\,1}, s_{10\,3})\\ &= (0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1).\end{aligned}$$

□

In order to consider the probabilistic diagnosis, we define the probabilistic model of faults and test outcomes as follows.

**Definition 1:** For a given unit $i$, let $P(x_i)$ denote the probability mass function(PMF) of $x_i$ and we assume that $x_i$ is independent from fault statuses of other units, i.e. $P(\mathbf{x}) = \Pi_{i=1}^{N} P(x_i)$. For a given edge $(j, i) \in E$, we assume that a test outcome $s_{ji}$ depends only on fault statuses $x_j$ and $x_i$, i.e. $P(s_{ji}|\mathbf{x}) = P(s_{ji}|x_j, x_i)$. For convenience of description, we define $P_{x_j,x_i}(s_{ji}) = P(s_{ji}|x_j, x_i)$, e.g. $P_{1,0}(s_{ji}) = P(s_{ji}|x_j = 1, x_i = 0)$ and so forth. □

For the ordinary fault diagnosis model, it is assumed that $P_{0,0}(s_{ji} = 1) = 0$ for all $(j, i) \in E$. If the outcome $s_{ji}$ which a fault-free unit $j$ tests a faulty unit $i$ may be incorrect evaluation[†], the diagnosis model is called an intermittent fault model. Note that an intermittent fault model can be represented by $P_{0,1}(s_{ji} = 0) > 0$.

In this paper, we assume that the PMFs $P(x_i)$ and $P_{x_j,x_i}(s_{ji})$ are given.

There are a few probabilistic fault diagnosis algorithms in the current literature which are closely related to the diagnosis algorithms that we present in this paper. For example, probabilistic fault diagnosis algorithms have proposed by D.M. Blough et al. [15], [16] and S. Lee et al. [17][††]. The diagnosis algorithm proposed by D.M. Blough et al. [15], [16], which is called the BSM algorithm, diagnoses a unit $i$ as faulty if and only if the following Eq. is satisfied[†††]:

$$\sum_{j\in\Gamma^{-1}(i)} \{2s_{ji} - P(x_j = 1) - P(x_j = 0)P_{0,1}(s_{ji} = 1)\} > 0. \quad (1)$$

Improving Eq. (1), furthermore, S. Lee et al. have proposed the diagnosis algorithm $Opt3$ which diagnoses a unit $i$ as faulty if and only if the following Eq. is satisfied:

$$1\Big/\left\{1 + \frac{P(x_i = 0)}{P(x_i = 1)} \prod_{j\in\Gamma^{-1}(i)|s_{ji}=1} \left(\frac{\beta_{ji}}{\alpha_{ji}}\right) \times \prod_{j\in\Gamma^{-1}(i)|s_{ji}=0} \left(\frac{1-\beta_{ji}}{1-\alpha_{ji}}\right)\right\} > 0.5, \quad (2)$$

where

$$\begin{aligned}\alpha_{ji} &= P(x_j = 0)P_{0,1}(s_{ji} = 1) + P(x_j = 1)P_{1,1}(s_{ji} = 1),\\ \beta_{ji} &= P(x_j = 1)P_{1,0}(s_{ji} = 1). \end{aligned} \quad (3)$$

The diagnostic error probability of $Opt3$ is less than or equal to that of the BSM algorithm [17]. Furthermore, S. Lee et al. have proposed the improved algorithm $Opt2$ which uses the relation $P_{0,0}(s_{ji} = 1) = 0$. The worst-case complexities of $Opt3$ and $Opt2$ are $O(|E|)$ and $O(N^2)$, respectively [17].

## 3. Maximum A Posteriori (MAP) Diagnosis Algorithm

We assume that a testing graph in this section satisfies that $(i, j) \notin E$ for all $(j, i) \in E$, i.e. if $j \in U$ tests $i \in U$ then $i$ doesn't test $j$. Let $\tilde{\mathcal{G}}$ denote a set of such testing graphs.

**Definition 2:** For given two graphs $G_1 = (U_1, E_1)$ and $G_2 = (U_2, E_2)$, the union of $G_1$ and $G_2$ is denoted by $G_1 \cup G_2 = (U_1 \cup U_2, E_1 \cup E_2)$.

For a given testing graph $G = (U, E) \in \tilde{\mathcal{G}}$ and $(j, i) \in E$, let $g_{ji}$ denote a sub-graph which consists of units $i, j$ and only one edge $(j, i)$, i.e. $g_{ji} = (\{i, j\}, \{(j, i)\})$. Given a unit $j \in U$, let a testing sub-graph $G_j^{(0)} = G_{j\backslash i}^{(0)} = (\{j\}, \emptyset)$.

For a given testing graph $G = (U, E)$ and any $l \geq 1$, let testing sub-graphs $G_i^{(l)}$ and $G_{i\backslash h}^{(l)}$ be defined recursively as follows:

$$G_{i\backslash h}^{(l)} = \bigcup_{j\in\Gamma^{-1}(i)\backslash\{h\}} \left(G_{j\backslash i}^{(l-1)} \cup g_{ji}\right) \cup \bigcup_{k\in\Gamma(i)\backslash\{h\}} \left(G_{k\backslash i}^{(l-1)} \cup g_{ik}\right), \quad (4)$$

$$G_i^{(l)} = \bigcup_{j\in\Gamma^{-1}(i)} \left(G_{j\backslash i}^{(l-1)} \cup g_{ji}\right) \cup \bigcup_{k\in\Gamma(i)} \left(G_{k\backslash i}^{(l-1)} \cup g_{ik}\right). \quad (5)$$

□

[†]That is, $s_{ji} = 0$ despite of $x_i = 1$.

[††]Note that these algorithms assume that $P_{0,0}(s_{ji} = 1) = 0$.

[†††]That is, the BSM algorithm outputs $\hat{x}_i = 1$.

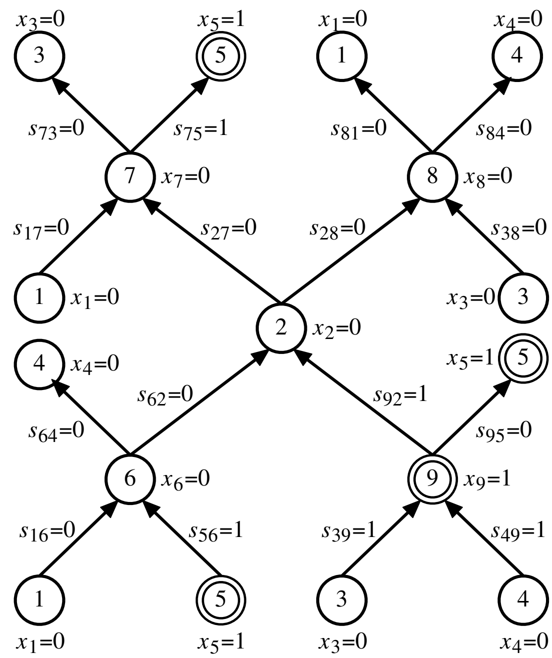


**Fig. 2** A tree expansion of a testing sub-graph $G_2^{(2)}$.

**Definition 3:** Given a constant $L$, let $\mathcal{G}^{(L)}$ denote a set of testing graphs $G \in \tilde{\mathcal{G}}$ such that there is no cycles when we regard all directed edges of $G_i^{(L)}$ as undirected ones for all $i \in U$. In other words, for $G \in \mathcal{G}^{(L)}$ and each $i \in U$, $G_i^{(L)}$ is a tree structure. □

**Example 2:** We consider a testing graph $G$ of Example 1. From Definition 2, $G_{j\backslash i}^{(0)} = (\{j\}, \emptyset)$ for all $j \in U$. For example, from Eq. (5) $G_1^{(1)} = (\{1, 6, 7, 8, 10\}, \{(1, 6), (1, 7), (8, 1), (10, 1)\})$, $G_2^{(1)} = (\{2, 6, 7, 8, 9\}, \{(2, 7), (2, 8), (6, 2), (9, 2)\})$ and so forth. From Eq. (4) $G_{2\backslash 6}^{(1)} = (\{2, 7, 8, 9\}, \{(2, 7), (2, 8), (9, 2)\})$, $G_{2\backslash 7}^{(1)} = (\{2, 6, 8, 10\}, \{(2, 8), (6, 2), (9, 2)\})$, and so forth. As a result, for $G$ of Fig. 1 from Definition 3 it follows that $G \in \mathcal{G}^{(1)}$ since for each $i \in U$ there is no cycles when we regard all directed edges of $G_i^{(1)}$ as undirected ones.

Next, $G_2^{(2)} = \left(G_{6\backslash 2}^{(1)} \cup g_{62}\right) \cup \left(G_{7\backslash 2}^{(1)} \cup g_{27}\right) \cup \left(G_{8\backslash 2}^{(1)} \cup g_{28}\right) \cup \left(G_{9\backslash 2}^{(1)} \cup g_{92}\right)$, $G_{2\backslash 6}^{(2)} = \left(G_{7\backslash 2}^{(1)} \cup g_{27}\right) \cup \left(G_{8\backslash 2}^{(1)} \cup g_{28}\right) \cup \left(G_{9\backslash 2}^{(1)} \cup g_{92}\right)$ and so on.

In Fig. 2, a tree expansion of a testing sub-graph $G_2^{(2)}$ is shown. There are some same units in this figure. This implies that $G_2^{(2)}$ has some cycles by regarding as undirected edges. Therefore, $G \notin \mathcal{G}^{(2)}$. □

**Definition 4:** Given $l \geq 1$ and $G_i^{(l)} = (U_i^{(l)}, E_i^{(l)})$ for $G = (U, E)$, let $\mathbf{x}_i^{(l)}$ denote a vector whose elements consist of $x_h$ for all $h \in U_i^{(l)}$, i.e. $\mathbf{x}_i^{(l)} = (x_h)_{h \in U_i^{(l)}}$. Let $\mathbf{s}_i^{(l)}$ denote a vector whose elements consist of test outcomes $s_{kh}$ for all $(k, h) \in E_i^{(l)}$, i.e. $\mathbf{s}_i^{(l)} = (s_{kh})_{(k,h) \in E_i^{(l)}}$. Similarly, given $G_{i\backslash j}^{(l)} = (U_{i\backslash j}^{(l)}, E_{i\backslash j}^{(l)})$, $\mathbf{x}_{i\backslash j}^{(l)} = (x_h)_{h \in U_{i\backslash j}^{(l)}}$ and $\mathbf{s}_{i\backslash j}^{(l)} = (s_{kh})_{(k,h) \in E_{i\backslash j}^{(l)}}$. Let $\mathbf{s}_i^{(0)}$ and $\mathbf{s}_{i\backslash j}^{(0)}$ be the empty sequence, respectively. □

**Theorem 1:** Given $G = (U, E) \in \mathcal{G}^{(L)}$, for any $i \in U$ and $l < L$ the following equation holds.

$$\ln \frac{P(x_i = 0|\mathbf{s}_i^{(l+1)})}{P(x_i = 1|\mathbf{s}_i^{(l+1)})} = \ln \frac{P(x_i = 0)}{P(x_i = 1)} + \sum_{j \in \Gamma^{-1}(i)} \ln \frac{\sum_{x_j} P(s_{ji}|x_j, x_i = 0) P(x_j|\mathbf{s}_{j\backslash i}^{(l)})}{\sum_{x_j} P(s_{ji}|x_j, x_i = 1) P(x_j|\mathbf{s}_{j\backslash i}^{(l)})} + \sum_{k \in \Gamma(i)} \ln \frac{\sum_{x_k} P(s_{ik}|x_i = 0, x_k) P(x_k|\mathbf{s}_{k\backslash i}^{(l)})}{\sum_{x_k} P(s_{ik}|x_i = 1, x_k) P(x_k|\mathbf{s}_{k\backslash i}^{(l)})}. \quad (6)$$

**Proof:** It is shown in Appendix A. □

$P(x_j|\mathbf{s}_{j\backslash i}^{(l)})$ must be calculated to obtain $P(x_i|\mathbf{s}_i^{(l+1)})$ in Theorem 1. The following theorem is used for this purpose.

**Theorem 2:** Given $G = (U, E) \in \mathcal{G}^{(L)}$, the following equation holds for any $i \in U$, $h \in \Gamma^{-1}(i) \cup \Gamma(i)$ and $l < L$.

$$\ln \frac{P(x_i = 0|\mathbf{s}_{i\backslash h}^{(l+1)})}{P(x_i = 1|\mathbf{s}_{i\backslash h}^{(l+1)})} = \ln \frac{P(x_i = 0)}{P(x_i = 1)} + \sum_{j \in \Gamma^{-1}(i)\backslash\{h\}} \ln \frac{\sum_{x_j} P(s_{ji}|x_j, x_i = 0) P(x_j|\mathbf{s}_{j\backslash i}^{(l)})}{\sum_{x_j} P(s_{ji}|x_j, x_i = 1) P(x_j|\mathbf{s}_{j\backslash i}^{(l)})} + \sum_{k \in \Gamma(i)\backslash\{h\}} \ln \frac{\sum_{x_k} P(s_{ik}|x_i = 0, x_k) P(x_k|\mathbf{s}_{k\backslash i}^{(l)})}{\sum_{x_k} P(s_{ik}|x_i = 1, x_k) P(x_k|\mathbf{s}_{k\backslash i}^{(l)})}. \quad (7)$$

**Proof:** This can be proved in almost the same way as Theorem 1. □

Theorem 2 implies that the log posteriori probability ratio $\ln \frac{P(x_i=0|\mathbf{s}_{i\backslash h}^{(l+1)})}{P(x_i=1|\mathbf{s}_{i\backslash h}^{(l+1)})}$ can be efficiently calculated for any $h \in \Gamma^{-1}(i) \cup \Gamma(i)$ from Eq. (7) as soon as $P(x_j|\mathbf{s}_{j\backslash i}^{(l)})$ are calculated for any $i \in U$ and all $j \in \Gamma^{-1}(i) \cup \Gamma(i)$. Using Theorems 1 and 2, the diagnosis algorithm which calculates $\ln \frac{P(x_i=0|\mathbf{s}_i^{(l+1)})}{P(x_i=1|\mathbf{s}_i^{(l+1)})}$ efficiently is as follows, where $l_{\max}$ is a constant which implies the iteration number of the algorithm.

[MAP Diagnosis Algorithm(MAPDA) ]

1) Let $l := 0$. For all $i \in U$ and $j \in \Gamma^{-1}(i)$, $k \in \Gamma(i)$,

$$a_{ik}^{(0)} = c_{ji}^{(0)} = t_i^{(0)} := \ln \frac{P(x_i = 0)}{P(x_i = 1)}. \quad (8)$$

2) For all $i \in U$, execute the following steps:

i) For all $j \in \Gamma^{-1}(i)$ and $k \in \Gamma(i)$, $b_{ji}^{(l+1)}$ and $d_{ik}^{(l+1)}$ are calculated by the following formulas.

$$b_{ji}^{(l+1)} := \ln \frac{P_{0,0}(s_{ji})\exp(a_{ji}^{(l)}) + P_{1,0}(s_{ji})}{P_{0,1}(s_{ji})\exp(a_{ji}^{(l)}) + P_{1,1}(s_{ji})}. \quad (9)$$

$$d_{ik}^{(l+1)} := \ln \frac{P_{0,0}(s_{ik})\exp(c_{ik}^{(l)}) + P_{0,1}(s_{ik})}{P_{1,0}(s_{ik})\exp(c_{ik}^{(l)}) + P_{1,1}(s_{ik})}. \quad (10)$$

ii) $t_i^{(l+1)}$ is calculated by the following.

$$t_i^{(l+1)} := t_i^{(0)} + \sum_{j\in\Gamma^{-1}(i)} b_{ji}^{(l+1)} + \sum_{k\in\Gamma(i)} d_{ik}^{(l+1)}. \quad (11)$$

iii) For all $j \in \Gamma^{-1}(i)$ and $k \in \Gamma(i)$, $a_{ik}^{(l+1)}$ and $c_{ji}^{(l+1)}$ are updated by the following formulas.

$$a_{ik}^{(l+1)} := t_i^{(l+1)} - d_{ik}^{(l+1)}. \quad (12)$$

$$c_{ji}^{(l+1)} := t_i^{(l+1)} - b_{ji}^{(l+1)}. \quad (13)$$

3) If $l+1 = l_{\max}$, then output Eq. (14) and halt; otherwise set $l := l+1$ and go to 2).

$$\hat{x}_i := \begin{cases} 0, & t_i^{(l_{\max})} \geq 0, \\ 1, & \textit{otherwise}. \end{cases} \quad (14)$$

□

The complexity of this algorithm is $O(|E|)$ since $l_{\max}$ is a constant and the following holds.

$$\sum_{i\in U}(|\Gamma^{-1}(i)| + |\Gamma(i)|) = 2|E|. \quad (15)$$

**Theorem 3:** Given $G \in \mathcal{G}^{(L)}$, for any $0 \leq l \leq L$ of the MAPDA

$$\ln \frac{P(x_i = 0|\mathbf{s}_{i\backslash h}^{(l)})}{P(x_i = 1|\mathbf{s}_{i\backslash h}^{(l)})} = \begin{cases} a_{ih}^{(l)}, & i \in \Gamma^{-1}(h), \\ c_{hi}^{(l)}, & i \in \Gamma(h). \end{cases} \quad (16)$$

**Proof:** It is shown in Appendix B. □

From this theorem, the following theorem holds for the MAPDA.

**Theorem 4:** Given $G \in \mathcal{G}^{(L)}$, the following Eq. holds for any $0 \leq l \leq L$.

$$t_i^{(l)} = \ln \frac{P(x_i = 0|\mathbf{s}_i^{(l)})}{P(x_i = 1|\mathbf{s}_i^{(l)})}. \quad (17)$$

**Proof:** If $l = 0$, Eq. (17) holds from the step 1) of the MAPDA since $\mathbf{s}_i^{(0)}$ is empty sequence from Definition 4.

If $l > 0$, then Eq. (17) holds from Theorems 1, 3 and the step 2) of the MAPDA. □

If $G \in \mathcal{G}^{(L)}$ and $l_{\max} \leq L$, then for the step 3) of the MAPDA the following Eq. holds from Theorem 4.

$$\hat{x}_i = \arg \max_{x_i\in\{0,1\}} P(x_i|\mathbf{s}_i^{(l_{\max})}). \quad (18)$$

This implies that the MAPDA estimates $\hat{x}_i$ by the maximum a posteriori probability.

It is obviously possible to use the MAPDA for the case where $l_{\max} > L$. In this case, Eq. (17) doesn't hold exactly. However, $t_i^{(l)}$ can be considered as an approximated value of the right-hand side of Eq. (17).

Next, we describe the ranges of $b_{ji}^{(l+1)}$ and $d_{ji}^{(l+1)}$ of Eqs. (9) and (10), respectively. Given $s_{ji}$ and $s_{ik}$, we put $B_1, B_2, D_1$ and $D_2$ as follows:

$$B_1 = \lim_{a_{ji}^{(l+1)}\to\infty} b_{ji}^{(l+1)} = \ln \frac{P_{0,0}(s_{ji}) + P_{1,0}(s_{ji})}{P_{0,1}(s_{ji}) + P_{1,1}(s_{ji})}, \quad (19)$$

$$B_2 = \lim_{a_{ji}^{(l+1)}\to-\infty} b_{ji}^{(l+1)} = \ln \frac{P_{0,0}(s_{ji})}{P_{0,1}(s_{ji})}, \quad (20)$$

$$D_1 = \lim_{c_{ik}^{(l+1)}\to\infty} d_{ik}^{(l+1)} = \ln \frac{P_{0,0}(s_{ik}) + P_{0,1}(s_{ik})}{P_{1,0}(s_{ik}) + P_{1,1}(s_{ik})}, \quad (21)$$

$$D_2 = \lim_{c_{ik}^{(l+1)}\to-\infty} d_{ik}^{(l+1)} = \ln \frac{P_{0,0}(s_{ik})}{P_{1,0}(s_{ik})}. \quad (22)$$

Then, the following inequalities hold from the simple calculation.

$$\begin{cases} B_1 < b_{ji}^{(l+1)} < B_2 & \text{if } P_{1,0}(s_{ji})P_{0,1}(s_{ji}) < P_{0,0}(s_{ji})P_{1,1}(s_{ji}) \\ B_1 = b_{ji}^{(l+1)} = B_2 & \text{if } P_{1,0}(s_{ji})P_{0,1}(s_{ji}) = P_{0,0}(s_{ji})P_{1,1}(s_{ji}) \\ B_1 > b_{ji}^{(l+1)} > B_2 & \text{if } P_{1,0}(s_{ji})P_{0,1}(s_{ji}) > P_{0,0}(s_{ji})P_{1,1}(s_{ji}) \end{cases} \quad (23)$$

$$\begin{cases} D_1 < d_{ik}^{(l+1)} < D_2 & \text{if } P_{1,0}(s_{ik})P_{0,1}(s_{ik}) < P_{0,0}(s_{ik})P_{1,1}(s_{ik}) \\ D_1 = d_{ik}^{(l+1)} = D_2 & \text{if } P_{1,0}(s_{ik})P_{0,1}(s_{ik}) = P_{0,0}(s_{ik})P_{1,1}(s_{ik}) \\ D_1 > d_{ik}^{(l+1)} > D_2 & \text{if } P_{1,0}(s_{ik})P_{0,1}(s_{ik}) > P_{0,0}(s_{ik})P_{1,1}(s_{ik}) \end{cases} \quad (24)$$

Note that if $P_{1,0}(s_{ji})P_{0,1}(s_{ji}) = P_{0,0}(s_{ji})P_{1,1}(s_{ji})$, then $b_{ji}^{(l+1)}$ is a constant independent of the value of $a_{ji}^{(l+1)}$. $d_{ik}^{(l+1)}$ is also the same.

Here, we discuss the difference with our previous method [18] to calculate the maximum a posteriori of a testing sub-graph as shown in Fig. 3. A testing sub-graph $\tilde{G}_i^{(L)} = (\tilde{U}_i^{(L)}, \tilde{E}_i^{(L)})$ in Fig. 3 is defined as a directed sub-graph of depth $L$ that reaches unit $i$ for given testing graph $G$. Comparing with $G_i^{(L)} = (U_i^{(L)}, E_i^{(L)})$ of Fig. 2, it is obvious that $\tilde{E}_i^{(L)} \subset E_i^{(L)}$. Therefore, we can expect to improve the diagnostic error probability of the previous method [18]. In Table 1, we show the construction of the sub-graph used in the fault diagnosis and corresponding diagnosis method.

The MAPDA is essentially identical to the Sum-Product algorithm [28] which can be used for more general case in machine learning theory. In this paper, we directly obtained the diagnosis algorithm by deriving Theorems 1 and 2 from the structure of the testing graph instead of deriving the Sum-Product algorithm by considering the factor graph [28] of the

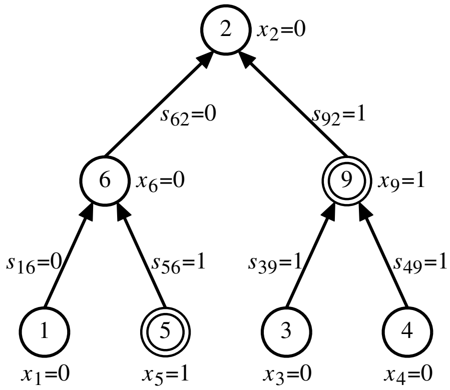


**Fig. 3** A testing sub-graph $\tilde{G}_2^{(2)}$ of [18].

**Table 1** Sub-graph and property used in fault diagnosis and corresponding diagnosis method.

| Used Sub-Graph and Property | Probabilistic Fault Diagnosis Method |
|---|---|
| $\tilde{G}_i^{(1)}$ | BSM[15,16], Opt3[17] |
| $\tilde{G}_i^{(1)}$ and $P_{0,0}(s_{ji}=1)=0$ | Opt2[17] |
| $\tilde{G}_i^{(L)}$ | MAPDA[18] for $\tilde{G}_i^{(L)}$ |
| $G_i^{(L)}$ | This study |

diagnosis system.

The evaluation of the diagnostic error probability using the computer simulation is shown in Sect. 5.

## 4. Analysis of MAP Diagnosis Algorithm

In coding theory, T.J. Richardson et al. have proposed the density evolution method to calculate the decoding error probability of low-density parity-check(LDPC) codes [29]. The decoding algorithm of LDPC codes is also a special case of the Sum-Product algorithm. In this section, we show a method, which is an application of the density evolution for the diagnosis system, to calculate the diagnostic error probability of the MAPDA. Since the MAPDA is different from the decoding algorithm of LDPC codes, we will formulate the density evolution method for the MAPDA from the start.

In this section, we assume that $G \in \mathcal{G}^{(L)}$ and $l_{\max} \leq L$ for the MAPDA[†].

**Definition 5:** Capital letters $X_i$ and $S_{ji}$ represent the random variables corresponding to $x_i$ and $s_{ji}$, respectively. Similarly, $A_{ji}^{(l)}$, $B_{ji}^{(l)}$, $C_{ji}^{(l)}$, $D_{ji}^{(l)}$ and $T_i^{(l)}$ represent the random variables corresponding to $a_{ji}^{(l)}$, $b_{ji}^{(l)}$, $c_{ji}^{(l)}$, $d_{ji}^{(l)}$ and $t_i^{(l)}$ in the MAPDA.

In general, let $f_Y(y)$ denote the probability density function(PDF) of a random variable $Y$. Similarly, let $f_{Y|Z}(y|z)$ denote the conditional PDF of a random variable $Y$ given a random variable $Z$. For simplicity, we may omit random variables of the PDF in the absence of confusion, e.g. $f(y)$ and $f(y|z)$. □

[†] In this paper, we treat the case where a testing graph $G$ is given, though the random graph ensemble is often used for an analysis of LDPC codes with large code length. This is because the number of units $N$ is a constant given by the fault diagnosis system, and $N$ is not large compared with the code length of the LDPC codes.

First, we show the PDF for Eq. (8) in the MAPDA. Letting $\tilde{t}_i = \ln \frac{P(x_i=0)}{P(x_i=1)}$, the variables of Eq. (8) are necessarily identical to $\tilde{t}_i$. Using the Dirac delta function $\delta$, therefore, the PDFs of these variables are as follows.

$$f_{A_{ik}^{(0)}|X_i}(y|x) = f_{C_{ji}^{(0)}|X_i}(y|x) = f_{T_i^{(0)}|X_i}(y|x) = \delta(y-\tilde{t}_i). \quad (25)$$

Next, we consider the PDFs for Eqs. (9) and (10). If $P_{1,0}(s_{ji})P_{0,1}(s_{ji}) = P_{0,0}(s_{ji})P_{1,1}(s_{ji})$, then $b_{ji}^{(l+1)}$ is a constant. Using $B_1$ of Eq. (19), in this case, it holds that

$$f_{B_{ji}^{(l+1)}|S_{ji},X_j,X_i}(b_{ji}^{(l+1)}|s_{ji},x_j,x_i) = \delta(b_{ji}^{(l+1)} - B_1). \quad (26)$$

Similarly, if $P_{1,0}(s_{ik})P_{0,1}(s_{ik}) = P_{0,0}(s_{ik})P_{1,1}(s_{ik})$, then

$$f_{D_{ik}^{(l+1)}|S_{ik},X_i,X_k}(d_{ik}^{(l+1)}|s_{ik},x_i,x_k) = \delta(d_{ik}^{(l+1)} - D_1). \quad (27)$$

Next, we consider the case where $b_{ji}^{(l+1)}$ and $d_{ik}^{(l+1)}$ are not constant. Let $g_A(b_{ji}^{(l+1)}, s_{ji})$ denote the function which calculates $a_{ji}^{(l)}$ from $b_{ji}^{(l+1)}$ and $s_{ji}$ for Eq. (9), i.e. $a_{ji}^{(l)} = g_A(b_{ji}^{(l+1)}, s_{ji})$. Similarly, let $g_C(d_{ik}^{(l+1)}, s_{ik})$ denote the function which calculates $c_{ik}^{(l)}$ from $d_{ik}^{(l+1)}$ and $s_{ik}$ for Eq. (10), i.e. $c_{ik}^{(l)} = g_C(d_{ik}^{(l+1)}, s_{ik})$. Then the following equations hold from Eqs. (9) and (10).

$$g_A(b_{ji}^{(l+1)}, s_{ji}) = \ln \frac{P_{1,0}(s_{ji}) - e^{b_{ji}^{(l+1)}} P_{1,1}(s_{ji})}{e^{b_{ji}^{(l+1)}} P_{0,1}(s_{ji}) - P_{0,0}(s_{ji})}, \quad (28)$$

$$g_C(d_{ik}^{(l+1)}, s_{ik}) = \ln \frac{P_{0,1}(s_{ik}) - e^{d_{ik}^{(l+1)}} P_{1,1}(s_{ik})}{e^{d_{ik}^{(l+1)}} P_{1,0}(s_{ik}) - P_{0,0}(s_{ik})}. \quad (29)$$

Note that the functions $g_A$ and $g_C$ are bijections given $s_{ji}$ and $s_{ik}$, respectively. Given $s \in \{0,1\}$, let $g'_A(b,s)$ and $g'_C(d,s)$ denote the derivatives with respect to $b$ and $d$ for the functions $g_A(b,s), b \in (-\infty,\infty)$, and $g_C(d,s), d \in (-\infty,\infty)$, respectively. Furthermore, note that $A_{ji}^{(l)}$ and $X_i$ are conditionally independent given $S_{ji}$. Similarly, $C_{ik}^{(l)}$ and $X_i$ are conditionally independent given $S_{ik}$. Using change of variable, therefore, the following equations hold.

$$\begin{aligned} &f_{B_{ji}^{(l+1)}|S_{ji},X_j,X_i}(b_{ji}^{(l+1)}|s_{ji},x_j,x_i) \\ &= f_{A_{ji}^{(l)}|X_j}(g_A(b_{ji}^{(l+1)}, s_{ji})|x_j) \left| g'_A(b_{ji}^{(l+1)}, s_{ji}) \right|, \quad (30) \end{aligned}$$

$$\begin{aligned} &f_{D_{ik}^{(l+1)}|S_{ik},X_i,X_k}(d_{ik}^{(l+1)}|s_{ik},x_i,x_k) \\ &= f_{C_{ik}^{(l)}|X_k}(g_C(d_{ik}^{(l+1)}, s_{ik})|x_k) \left| g'_C(d_{ik}^{(l+1)}, s_{ik}) \right|, \quad (31) \end{aligned}$$

where $g'_A(b,s)$ and $g'_C(d,s)$ are given in Eqs. (32) and (33), respectively. As a result, given $f(a_{ji}^{(l)}|x_j)$ and $f(c_{ik}^{(l)}|x_k)$, $f(b_{ji}^{(l+1)}|x_i)$ and $f(d_{ik}^{(l+1)}|x_i)$ can be calculated as follows by using Eqs. (30) and (31), respectively.

$$g'_A(b, s_{ji}) = e^b \left(P_{1,1}(s_{ji})P_{0,0}(s_{ji}) - P_{1,0}(s_{ji})P_{0,1}(s_{ji})\right) / \left\{\left(e^b P_{0,1}(s_{ji}) - P_{0,0}(s_{ji})\right)\left(P_{1,0}(s_{ji}) - e^b P_{1,1}(s_{ji})\right)\right\}, \tag{32}$$

$$g'_C(d, s_{ik}) = e^d \left(P_{1,1}(s_{ik})P_{0,0}(s_{ik}) - P_{0,1}(s_{ik})P_{1,0}(s_{ik})\right) / \left\{\left(e^d P_{1,0}(s_{ik}) - P_{0,0}(s_{ik})\right)\left(P_{0,1}(s_{ik}) - e^d P_{1,1}(s_{ik})\right)\right\}. \tag{33}$$

$$f(b_{ji}^{(l+1)}|x_i) = \sum_{x_j \in \{0,1\}} \sum_{s_{ji} \in \{0,1\}} P(x_j) \times P(s_{ji}|x_j, x_i) f(b_{ji}^{(l+1)}|s_{ji}, x_j, x_i), \tag{34}$$

$$f(d_{ik}^{(l+1)}|x_i) = \sum_{x_k \in \{0,1\}} \sum_{s_{ik} \in \{0,1\}} P(x_k) \times P(s_{ik}|x_i, x_k) f(d_{ik}^{(l+1)}|s_{ik}, x_i, x_k). \tag{35}$$

Next, we consider to calculate the PDF $f(t_i^{(l+1)}|x_i)$ from $f(b_{ji}^{(l+1)}|x_i)$ and $f(d_{ik}^{(l+1)}|x_k)$.

**Definition 6:** Let $f_{Y_1} * f_{Y_2}$ denote the resulting PDF of the convolution of two PDFs $f_{Y_1}(y)$ and $f_{Y_2}(y)$. Given a number of PDFs $f_{Y_i}(y), i \in I$, let $\prod_{i \in I} * f_{Y_i}$ denote the result of the convolution of these PDFs. □

Note that the PDF of the sum of two independent random variables $Y_1$ and $Y_2$ can be calculated as the convolution $f_{Y_1} * f_{Y_2}$. From Definition 6 and Eq. (11), it follows that

$$f_{T_i^{(l+1)}|X_i} = f_{T_i^{(0)}|X_i} * \left(\prod_{j \in \Gamma^{-1}(i)} * \ f_{B_{ji}^{(l+1)}|X_i}\right) * \left(\prod_{k \in \Gamma(i)} * \ f_{D_{ik}^{(l+1)}|X_i}\right). \tag{36}$$

From (11), (12) and (13), similarly, it follows that

$$f_{A_{ik}^{(l+1)}|X_i} = f_{T_i^{(0)}|X_i} * \left(\prod_{j' \in \Gamma^{-1}(i)} * \ f_{B_{j'i}^{(l+1)}|X_i}\right) * \left(\prod_{k' \in \Gamma(i) \setminus \{k\}} * \ f_{D_{ik'}^{(l+1)}|X_i}\right), \tag{37}$$

$$f_{C_{ji}^{(l+1)}|X_i} = f_{T_i^{(0)}|X_i} * \left(\prod_{j' \in \Gamma^{-1}(i) \setminus \{j\}} * \ f_{B_{j'i}^{(l+1)}|X_i}\right) * \left(\prod_{k' \in \Gamma(i)} * \ f_{D_{ik'}^{(l+1)}|X_i}\right). \tag{38}$$

Therefore, the PDFs corresponding to the variables in the MAPDA can be calculated sequentially from above. As a result, the diagnostic error probability $P_{\text{UER},i}$ of a unit $i \in U$, which is a probability that a unit $i$ is incorrectly diagnosed by the MAPDA, is calculated as follows.

$$P_{\text{UER},i} = \int_{-\infty}^{0} f_{T_i^{(l_{\max})}|X_i}(t|x_i = 0) P(x_i = 0) dt + \int_{0}^{\infty} f_{T_i^{(l_{\max})}|X_i}(t|x_i = 1) P(x_i = 1) dt. \tag{39}$$

From above argument, a proposed density evolution algorithm to calculate the diagnostic error probability is as follows.

[Density Evolution(DE) for MAP Diagnosis Algorithm ]

1) Let $l := 0$ and set Eq. (25) for all $i \in U$ and $j \in \Gamma^{-1}(i)$, $k \in \Gamma(i)$.
2) For all $i \in U$, execute the following steps.
   i) For all $j \in \Gamma^{-1}(i)$ and $k \in \Gamma(i)$, calculate Eqs. (34) and (35).
   ii) Calculate Eq. (36).
   iii) For all $j \in \Gamma^{-1}(i)$ and $k \in \Gamma(i)$, calculate Eqs. (37) and (38).
3) If $l + 1 = l_{\max}$, then output $P_{\text{UER},i}$ in Eq. (39) for all $i \in U$ and halt; otherwise set $l := l + 1$ and go to 2).

□

Here, we consider the complexity of the DE for the MAPDA. In the DE, it is difficult to calculate the PDFs exactly since the PDFs have continuous variables. Therefore, using quantization of the continuous random variables, we can calculate approximated PDFs by treating as the discrete random variables. Let $Q$ denote the number of quantized values of the random variables $A_{ji}^{(l)}$, $B_{ji}^{(l)}$, $C_{ji}^{(l)}$, $D_{ji}^{(l)}$ and $T_i^{(l)}$. Then the complexity of the convolution of two PDFs is $O(Q \ln Q)$. The dominant complexity is the calculation of step 2)-iii) in the DE for the MAPDA. Since the complexity to calculate Eqs. (37) and (38) is at most $O((|\Gamma^{-1}(i)| + |\Gamma(i)|)Q \ln Q)$, the total complexity of the DE for the MAPDA is at most $O(|E|NQ \ln Q)$. Note that the complexity of this algorithm doesn't depend on the PMFs $P(x_i = 1)$ and $P(s_{ji}|x_j, x_i)$. Therefore, the DE for the MAPDA can calculate the diagnostic error probability much faster than Monte Carlo simulation.

Next, we consider the case where the fault probabilities $P(x_i = 1)$ are identical for all $i$ and the probabilities $P_{x,x'}(s_{ji} = 1)$ of test outcomes are identical for all edges $(j, i) \in E$. Let $k$ be the unit with the smallest number of adjacent units, that is

$$k = \arg\min_i \{|\Gamma^{-1}(i)| + |\Gamma(i)|\}. \tag{40}$$

If $k$ and all units of $\Gamma^{-1}(k) \cup \Gamma(k)$ are faulty, then it is the most difficult case where MAPDA correctly diagnose the unit $k$ since the reliability of test outcomes of fault units is low†. This suggests that a unit with the

† $P(x_i = 1) < 0.5$, $P_{1,1}(s_{ji} = 1) \le P_{0,1}(s_{ji} = 1)$ and $P_{1,0}(s_{ji} = 0) < P_{0,0}(s_{ji} = 0)$ are assumed for the conventional settings of the fault diagnosis problem.

smallest number of adjacent units increases the diagnostic error probability. Therefore, it is reasonable to set $|\Gamma^{-1}(i)| = |\Gamma(i)| = m$ for all $i \in U$. Then $f_{B_{ji}^{(l)}|X_i}$ are identical for any $i$ and $j$. $f_{D_{ik}^{(l)}|X_i}$ is also the same. Therefore, it holds that $f_{A_{ji}^{(l)}|X_j} = f_{A_{j'i'}^{(l)}|X_{j'}}$, $f_{C_{ji}^{(l)}|X_i} = f_{C_{j'i'}^{(l)}|X_{i'}}$ and $f_{T_i^{(l)}|X_i} = f_{T_{i'}^{(l)}|X_{i'}}$ for all $(j,i), (j',i') \in E$. In this case, we can significantly reduce the amout of computational complexity and memory to obtain PDFs.

## 5. Simulation Results

In this section, we show the results of the computer simulations for the evaluation of the MAPDA. And using the DE for the MAPDA, a numerical analysis of the diagnostic error probability is also shown.

For simplicity of the evaluation, we set $|\Gamma^{-1}(i)| = |\Gamma(i)| = m$ for all $i \in U$, where $m$ is a constant. Furthermore, we assume that the fault probabilities of all units and the probabilities of test outcomes of all edges are identical, i.e. $P(x_i = 1) = P(x_j = 1)$ for any $i, j \in U$, and $P_{x,x'}(s_{ji} = 1) = P_{x,x'}(s_{lk} = 1)$ for any $(j,i), (l,k) \in E$ and $x, x' \in \{0,1\}$.

To construct the testing graph $G \in \mathcal{G}^{(1)}$ for the simulation, we consider the adjacency matrix $A = [A_{ji}]$ of the directed graph $G = (U, E) \in \mathcal{G}^{(1)}$, where

$$A_{ji} = \begin{cases} 1 & (j,i) \in E, \\ 0 & \text{otherwise}. \end{cases} \tag{41}$$

From the condition of $|\Gamma^{-1}(i)| = |\Gamma(i)| = m$, it must hold that

$$\sum_j A_{ji} = m \quad \text{for all } i, \tag{42}$$

$$\sum_i A_{ji} = m \quad \text{for all } j. \tag{43}$$

If $A_{ji} = 1$ for some $j$ and $i$, then it holds that $A_{ij} = 0$ since $G \in \mathcal{G}^{(1)}$. Using these properties, we show the algorithm to construct $G \in \mathcal{G}^{(1)}$ randomly.

[Construction Algorithm of $G$]

1) Set $A_{ji} := 0$ for all $i$ and $j$; $W_j := 0$ for all $j$; $\tilde{i} := 1$;
2) Choose $J$ randomly such that $|J| = m$ and

$$J \subset \{j \neq \tilde{i} | W_j < m\} \setminus \{j < \tilde{i} | A_{\tilde{i}j} = 1\}. \tag{44}$$

If there is not such $J$, then go to step 1), otherwise set $A_{j\tilde{i}} := 1$ and $W_j$++ for all $j \in J$.
3) If $\tilde{i} = N$, then stop the algorithm, otherwise $\tilde{i}$++ and go to step 2). □

When this algorithm stop successfully, it holds that

$$\sum_i A_{ji} = W_j = m \quad \text{for all } j, \tag{45}$$

since $|E| = mN$ and $|E|$ can be divided by $m$. Then, it holds that $G \in \mathcal{G}^{(1)\dagger}$.

Figure 4 shows the result of the case where $N = |U| = 100$, $m = 3$, $P_{0,0}(0) = 1$, $P_{0,1}(0) = 0$, $P_{1,0}(0) = 0.5$ and $P_{1,1}(0) = 0.5$. The horizontal and the vertical axes of Fig. 4 imply fault probability $P(x_i = 1)$ and the diagnostic error probability per unit $P_{\text{UER}}$, respectively. Dotted lines represent the results of Monte Carlo simulation. In the figure, the result of MAPDA for each $l_{\max} = 1, 2, 3$ is represented as MAPDA($l_{\max}$). For comparison, the simulation results of *Opt*2 and *Opt*3, which are the diagnosis algorithms proposed by Lee et al. [17], are shown as LS *Opt*2 and LS *Opt*3, respectively. Solid lines represent the numerical results of the DE for the MAPDA. Note that these lines present the diagnostic error probability of the MAPDA for the case where the testing graph is $G \in \mathcal{G}^{(L)}$, $L \geq l_{\max}$. Then we set the number of the quantized values as $Q = 2^{15}$. Similarly, Fig. 5 shows the result of the case where $N = 100$, $m = 7$, $P_{0,0}(0) = 1$, $P_{0,1}(0) = 0.5$, $P_{1,0}(0) = 0.5$ and $P_{1,1}(0) = 0.5$.

As $l_{\max}$ increases, $P_{\text{UER}}$ of the MAPDA($l_{\max}$) decreases from Figs. 4 and 5. This implies that the MAPDA($l_{\max}$), $l_{\max} > 1$, is effective even when the testing graph $G \in \mathcal{G}^{(1)}$ is used$^{\dagger\dagger}$. Furthermore, the results of the MAPDA for $l_{\max} = 1$ are superior to those of the LS *Opt*3. Similarly, the results of the MAPDA for $l_{\max} \geq 2$ are superior to those of the LS *Opt*2.

Next, the results of Monte Carlo simulation are almost identical to those of the DE for the MAPDA although $G \in \mathcal{G}^{(1)}$ is used in the simulation. This implies that MAPDA($l_{\max}$) for $G \in \mathcal{G}^{(L)}$, $l_{\max} > L$, is a good estimator of $\ln \frac{P(x_i=0|\mathbf{s}_i^{(l_{\max})})}{P(x_i=1|\mathbf{s}_i^{(l_{\max})})}$. Note that we cannot engage that the $P_{\text{UER}}$ of the MAPDA for $l_{\max} > 1$ and $G \in \mathcal{G}^{(1)}$ is close to that of the DE for the MAPDA when $P(x_i = 1)$ is smaller than Figs. 4 and 5. Therefore, the construction method of $G \in \mathcal{G}^{(L)}$, $L > 1$, is an important subject for study.

Finally, we show the comparison with our previous method [18] for a testing sub-graph $\tilde{G}_i^{(l_{\max})}$. Fig. 6 shows the result of the case where $N = |U| = 100$, $m = 3$, $P_{0,0}(0) = 1$, $P_{0,1}(0) = 0$, $P_{1,0}(0) = 0.5$ and $P_{1,1}(0) = 0.5$. Similarly, Fig. 7 shows the result of the case where $N = 100$, $m = 7$, $P_{0,0}(0) = 1$, $P_{0,1}(0) = 0.5$, $P_{1,0}(0) = 0.5$ and $P_{1,1}(0) = 0.5$. From Figs. 4~7, $P_{\text{UER}}$ of a proposed method is significantly smaller than that of the previous method. Our previous method uses only the information on the branches $\tilde{E}_i^{(l_{\max})}$ that reaches unit $i$. On the otherhand, a proposed method effectively uses more information on the branches $E_i^{(l_{\max})}$ such as Fig. 2. This makes the difference of $P_{\text{UER}}$.

$^{\dagger}$This construction method of $G \in \mathcal{G}^{(1)}$ is simple and fast if the total number of units $N$ is not small. However, it is difficult to construct $G \in \mathcal{G}^{(L)}$ for $L > 1$ by this method. Therefore, we use only $G \in \mathcal{G}^{(1)}$ for the simulation.

$^{\dagger\dagger}$MAPDA($l_{\max} = 1$) estimates $\hat{x}_i$ by Eq. (18) from Theorem 4 since $G \in \mathcal{G}^{(1)}$. On the other hand, MAPDA($l_{\max}$), $l_{\max} = 2, 3$, estimates $\hat{x}_i$ by the *approximate value* of $\ln \frac{P(x_i=0|\mathbf{s}_i^{(l_{\max})})}{P(x_i=1|\mathbf{s}_i^{(l_{\max})})}$ for these simulations.

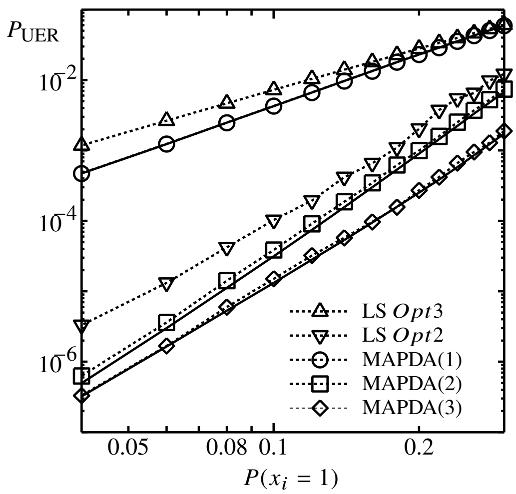


**Fig. 4** Diagnostic error probability per unit, where $m = 3$, $P_{0,0}(0) = 1$, $P_{0,1}(0) = 0$, $P_{1,0}(0) = 0.5$ and $P_{1,1}(0) = 0.5$.

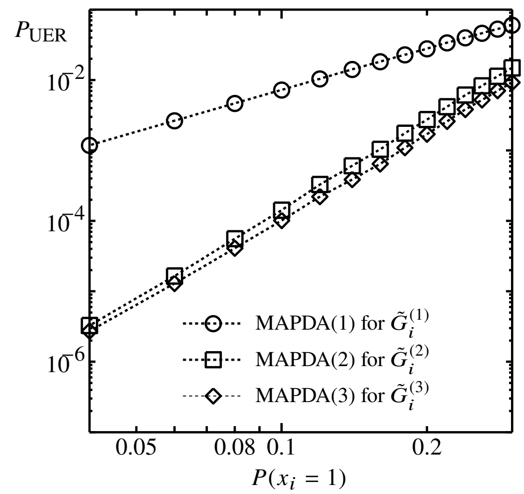


**Fig. 6** Diagnostic error probability for $\tilde{G}_i^{(l_{max})}$, where $m = 3$, $P_{0,0}(0) = 1$, $P_{0,1}(0) = 0$, $P_{1,0}(0) = 0.5$ and $P_{1,1}(0) = 0.5$.

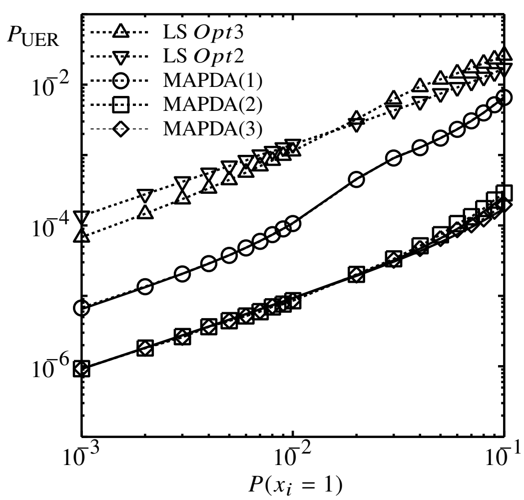


**Fig. 5** Diagnostic error probability per unit, where $m = 7$, $P_{0,0}(0) = 1$, $P_{0,1}(0) = 0.5$, $P_{1,0}(0) = 0.5$ and $P_{1,1}(0) = 0.5$.

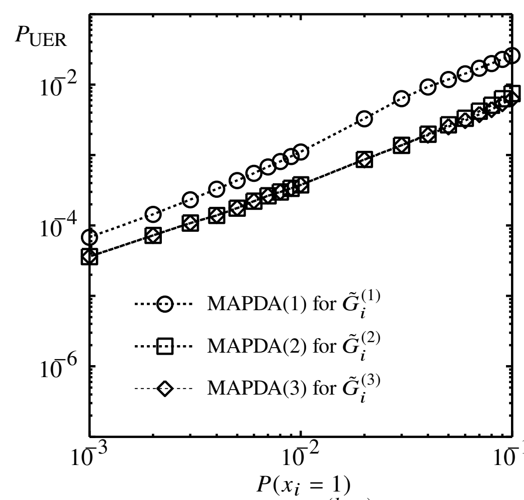


**Fig. 7** Diagnostic error probability for $\tilde{G}_i^{(l_{max})}$, where $m = 7$, $P_{0,0}(0) = 1$, $P_{0,1}(0) = 0.5$, $P_{1,0}(0) = 0.5$ and $P_{1,1}(0) = 0.5$.

## 6. Conclusion

In this paper, for probabilistic diagnosis models, we showed the MAPDA to calculate a posteriori probability that each of units is faulty given the test outcomes. Furthermore, we proposed the DE to analyze the diagnostic error probability of the MAPDA. As a result, the results of Monte Carlo simulation are almost identical to those of the DE for the MAPDA even when $G \in \mathcal{G}^{(1)}$ is used for the MAPDA. If $G \in \mathcal{G}^{(L)}$ can be constructed for a small constant $L$, then the diagnostic error probability of MAPDA for $l_{max} \leq L$ is exactly identical to that of the DE. To develop the method to construct $G \in \mathcal{G}^{(L)}, L > 1$, is needed in the future.

Here, we consider the case where the fault probabilities $P(x_i = 1)$ are different depending on $i$. If the number of units $N$ is too large, it is difficult to use the proposed analysis method described in Section 4 since the required memory and computational complexity become too large. Given the distribution of the fault probabilities, then, we may be able to reduce the required memory and complexity by considering the ensemble of random graphs like as modern coding theory [30]. Furthermore, it may be possible to analyze the optimal numbers of $|\Gamma^{-1}(i)|$ and $|\Gamma(i)|$ according to $P(x_i = 1)$. These are important future works.

## Acknowledgments

This work was supported in part by Grants-in-Aid for Scientific Research from KAKENHI (16K00491).

### References

[1] F.P. Preparata, G. Metze, and R.T. Chien, "On the connection assignment of diagnosable systems," IEEE Trans. Electron. Comput., vol.16, no.6, pp.848–854, Dec. 1967.
[2] T. Kohda, "A simple discriminator for identifying faults in highly structured diagnosable systems," J. Circuit. Syst. Comp., vol.4, no.3, pp.255–277, Sept. 1994.
[3] A. Pelc, "Optimal diagnosis of heterogeneous systems with random faults," IEEE Trans. Comput., vol.47, no.3, pp.298–304, March 1998.

[4] Z. Dafang, X. Gaogang, and M. Yinghua, "Node grouping in system-level fault diagnosis," J. Comput. Sci. Technol., vol.16, no.5, pp.474–479, Sept. 2001.
[5] T. Araki and Y. Shibata, "Optimal diagnosable systems on cayley graphs," IEICE Trans. Fundamentals, vol.E85-A, no.2, pp.455–462, Feb. 2002.
[6] K. Yoshida, Y. Sujaku, and T. Kohda, "A recursive procedure for designing optimal d-matched digraphs," IEICE Trans. Fundamentals, vol.E86-A, no.5, pp.1266–1274, May 2003.
[7] A. Okashita, T. Araki, and Y. Shibata, "An optimal adaptive diagnosis of butterfly networks," IEICE Trans. Fundamentals, vol.E86-A, no.5, pp.1008–1018, May 2003.
[8] T. Araki and Y. Shibata, "(t, k)-diagnosable system: A generalization of the PMC models," IEEE Trans. Comput., vol.52, no.7, pp.971–975, July 2003.
[9] T. Yamada, T. Ohtsuka, A. Watanabe, and S. Ueno, "On sequential diagnosis of multiprocessor systems," Discrete Appl. Math., vol.146, no.3, pp.311–342, March 2005.
[10] M. Elhadef, K. Abrougui, S. Das, and A. Nayak, "A parallel probabilistic system-level fault diagnosis approach for large multiprocessor systems," Parallel Process. Lett., vol.16, no.1, pp.63–79, March 2006.
[11] R. Ahlswede and H. Aydinian, "On diagnosability of large multiprocessor networks," Discrete Appl. Math., vol.156, no.18, pp.3464–3474, Nov. 2008.
[12] S. Mallela and G.M. Masson, "Diagnosable systems for intermittent faults," IEEE Trans. Comput., vol.C-27, no.6, pp.560–566, June 1978.
[13] M. Blount, "Probabilistic treatment of diagnosis in digital systems," Digest of the 7th International Symposium on Fault-Tolerant Computing, pp.72–77, IEEE Computer Society Press, Los Angeles, CA, June 1977.
[14] C.L. Yang and G.M. Masson, "A new measure for hybrid fault diagnosability," IEEE Trans. Comput., vol.C-36, no.3, pp.378–383, March 1987.
[15] D.M. Blough, G.F. Sullivan, and G.M. Masson, "Efficient diagnosis of multiprocessor systems under probabilistic models," IEEE Trans. Comput., vol.41, no.9, pp.1126–1136, Sept. 1992.
[16] D.M. Blough, G.F. Sullivan, and G.M. Masson, "Intermittent fault diagnosis in multiprocessor systems," IEEE Trans. Comput., vol.41, no.11, pp.1430–1441, Nov. 1992.
[17] S. Lee and K.G. Shin, "Optimal and efficient probabilistic distributed diagnosis schemes," IEEE Trans. Comput., vol.42, no.7, pp.882–886, July 1993.
[18] M. Kobayashi, T. Takabatake, S. Amano, and S. Hirasawa, "Probabilistic fault diagnosis and its analysis in large scale multiprocessor systems," IEICE Trans. Inf & Syst. (Japanese Edition), vol.J93-D, no.8, pp.1602–1613, Aug. 2010.
[19] 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, March 1987.
[20] S. Lee and K.G. Shin, "On probabilistic diagnosis of multiprocessor systems using multiple syndromes," IEEE Trans. Parallel Distrib. Syst., vol.5, no.6, pp.630–638, June 1994.
[21] K. Huang, V.K. Agarwal, and K. Thulasiraman, "Diagnosis of clustered faults and wafer testing," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol.17, no.2, pp.136–148, Feb. 1998.
[22] Q. Tang and X. Song, "Diagnosis of parallel computers with arbitrary connectivity," IEEE Trans. Comput., vol.48, no.7, pp.757–761, July 1999.
[23] Q. Tang, X. Song, and Y. Wang, "Diagnosis of clustered faults for identical degree topologies," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol.18, no.8, pp.1192–1201, Aug. 1999.
[24] T. Araki and Y. Shibata, "Efficient diagnosis algorithms on butterfly networks under the comparison approach," IEICE Trans. Fundamentals, vol.E85-A, no.4, pp.842–848, April 2002.
[25] T. Araki and Y. Shibata, "Diagnosability of butterfly networks under the comparison approach," IEICE Trans. Fundamentals, vol.E85-A, no.5, pp.1152–1160, May 2002.
[26] X. Lu, J. Li, and C.-J. Seo, "Probabilistic diagnosis of clustered faults for shared structures," Math. Comput. Model., vol.49, no.3-4, pp.623–634, Feb. 2009.
[27] M. Kobayashi, T. Takabatake, T. Matsushima, and S. Hirasawa, "Probabilistic fault diagnosis and its analysis in multicomputer systems," Proc. 2011 IEEE International Conference on Systems, Man, and Cybernetics, pp.1205–1211, Anchorage, Alaska, Oct. 2011.
[28] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
[29] T.J. Richardson and R.L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," IEEE Trans. Inf. Theory, vol.47, no.2, pp.599–618, Feb. 2001.
[30] T.J. Richardson and R.L. Urbanke, Modern Coding Theory, Cambridge University Press, 2008.

## Appendix A: Proof of Theorem 1

Using Definitions 1, 4 and the marginal PMF, the following equation holds for $x \in \{0, 1\}$.

$$P(x_i = x|\mathbf{s}_i^{(l+1)}) = \sum_{\mathbf{x}\in\{0,1\}^N|x_i=x} P(\mathbf{x}|\mathbf{s}_i^{(l+1)})$$
$$= \sum_{\mathbf{x}\in\{0,1\}^N|x_i=x} \frac{P(\mathbf{s}_i^{(l+1)}|\mathbf{x})P(\mathbf{x})}{P(\mathbf{s}_i^{(l+1)})}$$
$$= \sum_{\mathbf{x}_i^{(l+1)}|x_i=x} \frac{P(\mathbf{s}_i^{(l+1)}|\mathbf{x}_i^{(l+1)})P(\mathbf{x}_i^{(l+1)})}{P(\mathbf{s}_i^{(l+1)})}. \quad \text{(A· 1)}$$

From Definitions 1, 2 and 4, it follows that

$$P(\mathbf{s}_i^{(l+1)}|\mathbf{x}_i^{(l+1)})P(\mathbf{x}_i^{(l+1)})$$
$$= P(\mathbf{x}_i^{(l+1)}) \prod_{j\in\Gamma^{-1}(i)} P(\mathbf{s}_{j\backslash i}^{(l)}, s_{ji}|\mathbf{x}_{j\backslash i}^{(l)}, x_i) \times$$
$$\prod_{k\in\Gamma(i)} P(\mathbf{s}_{k\backslash i}^{(l)}, s_{ik}|\mathbf{x}_{k\backslash i}^{(l)}, x_i)$$
$$= P(\mathbf{x}_i^{(l+1)}) \prod_{j\in\Gamma^{-1}(i)} P(\mathbf{s}_{j\backslash i}^{(l)}|\mathbf{x}_{j\backslash i}^{(l)})P(s_{ji}|\mathbf{x}_{j\backslash i}^{(l)}, x_i) \times$$
$$\prod_{k\in\Gamma(i)} P(\mathbf{s}_{k\backslash i}^{(l)}|\mathbf{x}_{k\backslash i}^{(l)})P(s_{ik}|\mathbf{x}_{k\backslash i}^{(l)}, x_i)$$
$$= P(x_i) \prod_{j\in\Gamma^{-1}(i)} P(\mathbf{s}_{j\backslash i}^{(l)}, \mathbf{x}_{j\backslash i}^{(l)})P(s_{ji}|x_j, x_i) \times$$
$$\prod_{k\in\Gamma(i)} P(\mathbf{s}_{k\backslash i}^{(l)}, \mathbf{x}_{k\backslash i}^{(l)})P(s_{ik}|x_i, x_k). \quad \text{(A· 2)}$$

Noting that for $x \in \{0, 1\}$

$$\sum_{\mathbf{x}_i^{(l+1)}|x_i=x} \prod_{j\in\Gamma^{-1}(i)} P(\mathbf{s}_{j\backslash i}^{(l)}, \mathbf{x}_{j\backslash i}^{(l)})P(s_{ji}|x_j, x_i)$$
$$\prod_{k\in\Gamma(i)} P(\mathbf{s}_{k\backslash i}^{(l)}, \mathbf{x}_{k\backslash i}^{(l)})P(s_{ik}|x_i, x_k)$$
$$= \prod_{j\in\Gamma^{-1}(i)} \left\{ \sum_{x_j} P(\mathbf{s}_{j\backslash i}^{(l)}, x_j)P(s_{ji}|x_j, x_i = x) \right\} \times$$

$$\prod_{k\in\Gamma(i)}\left\{\sum_{x_k} P(\mathbf{s}_{k\setminus i}^{(l)}, x_k)P(s_{ik}|x_i = x, x_k)\right\}, \quad \text{(A· 3)}$$

it follows that

$$\begin{aligned}
&\ln\frac{P(x_i=0|\mathbf{s}_i^{(l+1)})}{P(x_i=1|\mathbf{s}_i^{(l+1)})} = \ln\frac{P(x_i=0)}{P(x_i=1)}\\
&+\ln\frac{\prod_{j\in\Gamma^{-1}(i)}\sum_{x_j}P(s_{ji}|x_j,x_i=0)P(x_j|\mathbf{s}_{j\setminus i}^{(l)})}{\prod_{j\in\Gamma^{-1}(i)}\sum_{x_j}P(s_{ji}|x_j,x_i=1)P(x_j|\mathbf{s}_{j\setminus i}^{(l)})}\\
&+\ln\frac{\prod_{k\in\Gamma(i)}\sum_{x_k}P(s_{ik}|x_i=0,x_k)P(x_k|\mathbf{s}_{k\setminus i}^{(l)})}{\prod_{k\in\Gamma(i)}\sum_{x_k}P(s_{ik}|x_i=1,x_k)P(x_k|\mathbf{s}_{k\setminus i}^{(l)})}\\
&=\ln\frac{P(x_i=0)}{P(x_i=1)}+\sum_{j\in\Gamma^{-1}(i)}\ln\frac{\sum_{x_j}P(s_{ji}|x_j,x_i=0)P(x_j|\mathbf{s}_{j\setminus i}^{(l)})}{\sum_{x_j}P(s_{ji}|x_j,x_i=1)P(x_j|\mathbf{s}_{j\setminus i}^{(l)})}\\
&+\sum_{k\in\Gamma(i)}\ln\frac{\sum_{x_k}P(s_{ik}|x_i=0,x_k)P(x_k|\mathbf{s}_{k\setminus i}^{(l)})}{\sum_{x_k}P(s_{ik}|x_i=1,x_k)P(x_k|\mathbf{s}_{k\setminus i}^{(l)})}. \quad \text{(A· 4)}
\end{aligned}$$

Therefore, the theorem follows. □

## Appendix B: Proof of Theorem 3

We will use the inductive method. $U_{i\setminus h}^{(0)} = \{i\}$, $E_{i\setminus h}^{(0)} = \emptyset$ and $\mathbf{s}_{i\setminus h}^{(0)}$ is empty sequence from Definition 4. Therefore, Eq. (16) holds for the step 1) of the MAPDA when $l = 0$.

Assuming Eq. (16) holds for certain $l$, from Eq. (16) it follows that

$$P(x_j=0|\mathbf{s}_{j\setminus i}^{(l)}) = \begin{cases}\exp(a_{ji}^{(l)})/\left(1+\exp(a_{ji}^{(l)})\right), & j\in\Gamma^{-1}(i),\\ \exp(c_{ij}^{(l)})/\left(1+\exp(c_{ij}^{(l)})\right), & j\in\Gamma(i),\end{cases} \quad \text{(A· 5)}$$

$$P(x_j=1|\mathbf{s}_{j\setminus i}^{(l)}) = 1-P(x_j=0|\mathbf{s}_{j\setminus i}^{(l)}). \quad \text{(A· 6)}$$

Substituting Eqs. (A· 5) and (A· 6) to the right-hand side of Eq. (7), from Eqs. (9) and (10) for all $j\in\Gamma^{-1}(i)$ and $k\in\Gamma(i)$ it follows that

$$\ln\frac{\sum_{x_j}P(s_{ji}|x_j,x_i=0)P(x_j|\mathbf{s}_{j\setminus i}^{(l)})}{\sum_{x_j}P(s_{ji}|x_j,x_i=1)P(x_j|\mathbf{s}_{j\setminus i}^{(l)})} = b_{ji}^{(l+1)}, \quad \text{(A· 7)}$$

$$\ln\frac{\sum_{x_k}P(s_{ik}|x_i=0,x_k)P(x_k|\mathbf{s}_{k\setminus i}^{(l)})}{\sum_{x_k}P(s_{ik}|x_i=1,x_k)P(x_k|\mathbf{s}_{k\setminus i}^{(l)})} = d_{ik}^{(l+1)}. \quad \text{(A· 8)}$$

As a result, Eq. (16) holds for $l+1$ from Eq. (7) and the step 2) of MAPDA, and the theorem follows. □

**Manabu Kobayashi** was born in Yokohama, Japan, on Oct. 30, 1971. He received the B.E. degree, M.E. degree and Dr.E. degree in Industrial and Management Systems Engineering form Waseda University, Tokyo, Japan, in 1994, 1996 and 2000, respectively. From 1998 to 2001, he was a research associate in Industrial and Management Systems Engineering at Waseda University. From 2001 to 2018, he was with the Department of Information Science at Shonan Institute of Technology. He is currently a professor of Center for Data Science at Waseda University. His research interests are information theory and machine learning theory. He is a member of the Information Processing Society of Japan and IEEE.

**Toshiyasu Matsushima** was born in Tokyo, Japan, on Nov. 26, 1955. He received the B.E. degree, M.E. degree and Dr.E. degree in Industrial and Management Systems Engineering from Waseda University, Tokyo, Japan, in 1978, 1980 and 1991, respectively. From 1980 to 1986, he was with Nippon Electric Corporation, Kanagawa, Japan. From 1986 to 1992, he was a lecturer at Department of Management Information, Yokohama College of Commerce. From 1993, he was an associate professor and from 1996 to 2007 was a professor of School of Science and Engineering, Waseda University, Tokyo, Japan. Since 2007, he has been a professor of School of Fundamental Science and Engineering, Waseda University. His research interests are information theory and its application, statistics and artificial intelligence. He is an IEICE Fellow, and a member of the Japan Society for Quality Control, the Japan Industrial Management Association, the Japan Society for Artificial Intelligence and IEEE.

**Shigeichi Hirasawa** was born in Kobe, Japan, on Oct. 2, 1938. He received the B.S. degree in mathematics and the B.E. degree in electrical communication engineering from Waseda University, Tokyo, Japan, in 1961 and 1963, respectively, and the Dr.E. degree in electrical communication engineering from Osaka University, Osaka, Japan, in 1975. From 1963 to 1981, he was with the Mitsubishi Electric Corporation, Hyogo, Japan. rom 1981 to 2007, he was a professor of School of Science and Engineering and since 2007, has been a professor of School of Creative Science and Engineering, Waseda University, Tokyo, Japan. In 1979, he was a Visiting Scholar in the Computer Science Department at the University of California, Los Angels (CSD, UCLA), CA. He was a Visiting Researcher at the Hungarian Academy of Science, Hungary, in 1985, and at the University of Trieste, Italy, in 1986. In 2002, he was also a Visiting Faculty at CSD, UCLA. From 1987 to 1989, he was the Chairman of Technical Group on Information Theory of IEICE. He received the 1993 Achievement Award, and the 1993 Kobayashi-Memorial Achievement Award from IEICE. In 1996, he was the President of the Society of Information Theory and Its Applications (Soc. of ITA). His research interests are information theory and its applications, and information processing systems. He is an IEEE Life Fellow, an IEICE Fellow, and a member of the Operations Research Society of Japan, the Information Processing Society of Japan, the Japan Industrial Management Association, and Informs.