An Algorithmic Approach to Conditional-Fault Local Diagnosis of Regular Multiprocessor Interconnected Systems under the PMC Model

Cheng-Kuan Lin, Tzu-Liang Kung, and Jimmy J.M. Tan

Abstract—System-level diagnosis is a crucial subject for maintaining the reliability of multiprocessor interconnected systems. Consider a system composed of N independent processors, each of which tests a subset of the others. Under the PMC diagnosis model, Dahbura and Masson [10] proposed an $O(N^{2.5})$ algorithm to identify the set of faulty processors in a t-diagnosable system, in which at most t processors are permanently faulty. In this paper, we establish some sufficient conditions so that a t-regular system can be conditionally $(2t-1)$-diagnosable, provided every fault-free processor has at least one fault-free neighbor. Because any t-regular system is no more than t-diagnosable, the approached diagnostic capability is nearly double the classical one-step diagnosability. Furthermore, a correct and complete method is given which exploits these conditions and the presented branch-of-tree architecture to determine the fault status of any single processor. The proposed method has time complexity $O(t^2)$, and thus can diagnose the whole system in time $O(t^2N)$. In short, not only could the diagnostic capability be proved theoretically, but also it is feasible from an algorithmic perspective.

Index Terms—Diagnosis, diagnosability, reliability, PMC model, graph, multiprocessor, algorithm

1 Introduction

The past decade has witnessed remarkable advances in microprocessor technology for the development of high-speed multiprocessor systems. As networking infrastructure evolves, the vision of using the Internet as one large heterogeneous parallel and distributed computing environment has taken shape. For instance, cloud computing has become more and more popular recently in the area of information technology. In essence, cloud computing is a paradigm that allows one to access a remote data center or applications that actually reside at a location other than a local computer. Although not everyone agrees on what cloud computing is, it brings an Internet user applications on the go, a convenient way of viewing, manipulating, and sharing data. Hence, cloud computing can be thought of as a highly cooperative computing scheme that shares through the Internet hardware resources, software applications, and databases.

The multiprocessor system is one typical application of massive parallel computing. In a multiprocessor system, the reliability of each computing and/or storage unit is crucial because even a few malfunctions may make system service unreliable. Whenever units are found faulty, they should be replaced with fault-free ones in order to guarantee that the system continues operating properly. Identifying all the faulty units in a system is known as *system-level diagnosis*. Preparata et al. [31] distinguished two types of self-diagnosable systems: one-step diagnosable systems and sequentially diagnosable systems. A system is said to be *one-step t-diagnosable* if all its faulty units can be precisely pointed out by one application of some diagnostic process provided that the total number of faulty units does not exceed t; a system is *sequentially t-diagnosable* if at least one faulty unit can be identified provided that the total number of faulty units does not exceed t. In this paper, we focus on one-step diagnosis only. The maximum number of faulty units that can be correctly identified is known as the *one-step diagnosability* of a system. In other words, the one-step diagnosability of a system G is just equal to the maximum integer t such that G is one-step t-diagnosable.

System-level diagnosis has been widely addressed by many researchers [7], [10], [11], [13], [14], [15], [16], [18], [19], [23], [26], [31], [33]. One classic approach to this problem, called the PMC diagnosis model (or PMC model for short), was first proposed by Preparata et al. [31]. In this approach, each unit is tested by some other units in the system. Following the PMC model, Hakimi and Amin [14] proved that a system is one-step t-diagnosable if it is t-connected and has at least $2t+1$ units. They also pointed out that the problem could be approached with 0-1 integer programming and gave a sufficient and necessary condition for verifying whether or not a system is one-step t-diagnosable under the PMC model. Dahbura and Masson [10] presented an $O(N^{2.5})$ fault identification algorithm, where N denotes the total number of units in a system. A recent paper [24] addressed the problem of determining the one-step diagnosability of a big family of interconnection networks called (1,2)-Matching Composition Networks, each of which is constructed by connecting two graphs via one or two perfect matchings. Many famous network

- C.-K. Lin and J.J.M. Tan are with the Department of Computer Science, National Chiao Tung University, 1001 University Road, Hsinchu 30010, Taiwan. E-mail: {cklin, jmtan}@cs.nctu.edu.tw.
- T.-L. Kung is with the Department of Computer Science and Information Engineering, Asia University, 500 Lioufeng Road, Wufeng, Taichung 41354, Taiwan. E-mail: tlkung@asia.edu.tw.

Manuscript received 28 Jan. 2011; revised 24 Oct. 2011; accepted 9 Dec. 2011. Recommended for acceptance by W. Najjar.
For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number TC-2011-01-0065. Digital Object Identifier no. 10.1109/TC.2011.249.

topologies, such as hypercubes, folded hypercubes, crossed cubes, twisted cubes, locally twisted cubes, augmented cubes, recursive circulants, hyper-Petersen networks, etc., belong to this family.

The one-step diagnosability of a system G is upper bounded by the minimum degree of G, intuitively because it is unlikely to determine whether or not a unit is faulty if all its neighboring units happen to be faulty simultaneously. For most practical systems that are sparsely connected, only a small number of faulty units can be recognized under the PMC model. Therefore, it has long been an intriguing issue to discover some measure that can better reflect fault patterns in real systems. For example, Somani and Peleg [35] proposed the t/k-diagnosability to evaluate an alternative approach to system-level diagnosis by allowing a few upper bounded number of units to be diagnosed incorrectly; Das et al. [11] studied fault diagnosis under local constraints; Lai et al. [23] proposed a new measure of diagnostic capability, called conditional diagnosability, by restricting that for each unit in a system, all its neighboring units do not fail at the same time. Under the PMC model, Zhu [37] discussed the conditional diagnosability of the BC networks; Xu et al. [36] investigated the conditional diagnosability with respect to a class of matching composition networks. More recently, Chang and Hsieh [5] considered the conditional diagnosability of augmented cubes. However, those studies were purely theoretical and did not provide any diagnosis procedure, so it is not clear how to identify faulty units efficiently in such kind of situation. In this paper, we somewhat relax that condition imposed in [5], [23], [20], [36], [37] and assume instead that every fault-free unit should have at least one fault-free neighbor. Under this new condition, not only can the diagnostic capability be proved theoretically, but also it is feasible from an algorithmic point of view.

A variety of methods have been developed to achieve system-level diagnosis for various interconnected structures. For example, Chessa and Maestrini [8] introduced a correct and almost complete diagnosis method for square grids. Later, Caruso et al. [2], [3], [4] presented two correct and almost complete diagnosis algorithms, called EDARS and NDA, respectively. Recently, Mánik and Gramatová [27], [28] proposed the Boolean formalization of the PMC model for the syndrome-decoding process. When this approach is applied to regular systems, the computation time of fault diagnosis can be significantly reduced. In addition, Somani and Agarwal [34] developed a distributed diagnosis algorithm for regular systems based on the concept of local diagnosis; Masuyama and Miyoshi [29] presented a nonadaptive distributed system-level diagnosis method for computer networks.

Generally speaking, the design of parallel algorithms depends on complete or incomplete mapping of tasks to specific architectures such as rings, paths, trees, meshes, and so on. If all units of the architecture are fault-free, procedures can operate properly even though there exist many faulty units in the remaining part of the system. Thus, such a kernel architecture plays a key role in the development of parallel computing systems. To this effect, Hsu and Tan [19] presented a novel measure of diagnostic capability, known as local diagnosability. This paper aims to extend that previous study to be capable of diagnosing conditionally faulty systems, in which every fault-free unit has at least one fault-free neighboring unit. In order to achieve this purpose, we design a kernel branch-of-tree (BOT) architecture and propose a fault identification method based on it. For a k-regular interconnected systems composed of N units, the running time for diagnosing any single unit u can be bounded by $O(k^2)$ if there exists a branch-of-tree architecture rooted at u. So, all the faulty units can be identified in time $O(k^2N)$, provided a branch-of-tree rooted at each unit can also be built in time $O(k^2)$.

The rest of this paper is organized as follows: Section 2 provides a preliminary background for system-level diagnosis and graph-theoretic terminology. Section 3 introduces local diagnosis. The conditional-fault local diagnosis is studied in Section 4. Some examples and simulation results are shown in Section 5. Finally, we draw a conclusion in Section 6.

2 Preliminaries

The underlying topology of a multiprocessor interconnected system is usually modeled as a graph, whose node set and edge set represent the set of all processors and the set of all communication links between processors, respectively. Throughout this paper graphs are finite, simple, and unless specified otherwise, undirected. Some important graph-theoretic definitions and notations have to be introduced in advance. For those not defined here, however, we follow the standard terminology given in [6], [21].

An undirected graph G is an ordered pair (V, E), where V is a nonempty set, and E is a subset of $\{\{u, v\} \mid u \in V \text{ and } v \in V\}$.[1] The set V is called the *node set* of G, and the set E is called the *edge set* of G. For convenience, we denote the node set and the edge set of G by $V(G)$ and $E(G)$, respectively. Two nodes, u and v, in graph G are *adjacent* if $\{u, v\} \in E(G)$; we say u is a *neighbor* of v, and vice versa. The degree of a node v in G, denoted by $deg_G(v)$, is the number of edges incident to v. The neighborhood of node v, denoted by $N_G(v)$, is the set of nodes adjacent to v. For a set $S \subset V$, the notation $G - S$ represents the graph obtained by removing every node in S from G and deleting those edges incident to at least one node in S. A graph H is a *subgraph* of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. The *components* of a graph G are its maximal connected subgraphs. A component is trivial if it has no edge; otherwise, it is nontrivial.

The PMC model [31] allows adjacent units to execute tests on each other. A testing unit u_i specifies some test sequence for a tested unit u_j and receives a response sequence from u_j. The testing unit outputs a test outcome $a_{i,j} = 1$ if the response sequence mismatches the expected one; otherwise, $a_{i,j} = 0$. Let an undirected graph $G = (V, E)$ denote the underlying topology of a multiprocessor system. For any two adjacent nodes $u, v \in V$, the ordered pair (u, v) represents the *test* that processor u diagnoses processor v. In this scenario, u is a *tester*, and v is a *testee*. The outcome of a test (u, v) is 1 (respectively, 0) if u evaluates v to be faulty

1. We denote a directed arc from u to v and an undirected edge between u and v by (u, v) and $\{u, v\}$, respectively.

TABLE 1
Invalidation Rule of the PMC Model

tester	testee	test outcome
fault-free	fault-free	0
fault-free	faulty	1
faulty	fault-free	0 or 1
faulty	faulty	0 or 1

(respectively, fault-free). The notation $u \xrightarrow{\gamma} v$ means that u tests v with outcome γ. Because faults considered here are permanent, the outcome of a test is *reliable* if and only if the tester is fault-free. Table 1 summarizes the invalidation rule of the PMC model.

The *test assignment* for a system $G = (V, E)$ is modeled as a directed graph $T = (V, L)$, where $(u, v) \in L$ and $(v, u) \in L$ if and only if $\{u, v\} \in E$. The collection of all test outcomes from the test assignment T is called a *syndrome*. Formally, a syndrome of T is a mapping $\sigma : L \to \{0, 1\}$. A set F of faulty nodes in G is called a *faulty set*. It is noticed that F can be any subset of V. The process of identifying all faulty nodes is said to be the system-level diagnosis. Furthermore, the maximum number of faulty nodes that can be correctly identified is called the *one-step diagnosability* of G, denoted by $\tau(G)$.

For any given syndrome σ collected from a test assignment $T = (V, L)$, a subset of nodes $F \subseteq V$ is said to be *consistent* with σ if for any arc $(u, v) \in L$ with $u \in V - F$, then $\sigma(u, v) = 1$ if and only if $v \in F$. This corresponds to the assumption that fault-free testers always give correct test results, but faulty testers do not. Therefore, a given set F of faulty nodes may be consistent with different syndromes. Let $\sigma(F)$ denote the set of all possible syndromes with which the faulty set F can be consistent. Then, two distinct faulty sets $F_1, F_2 \subset V$ are *distinguishable* if $\sigma(F_1) \cap \sigma(F_2) = \emptyset$; otherwise, F_1 and F_2 are *indistinguishable*. Dahbura and Masson [10] indicated that a system G is one-step t-diagnosable if and only if for any two distinct faulty sets $F_1, F_2 \subset V(G)$ with $|F_1| \leq t$ and $|F_2| \leq t$, F_1 and F_2 are distinguishable. Lemma 1 is one of the most important characterization of distinguishable faulty sets.

Lemma 1 ([10]). *Let $G = (V, E)$ be a graph. For any two distinct faulty sets $F_1, F_2 \subset V$, then F_1 and F_2 are distinguishable if and only if there exists a node $u \in V - (F_1 \cup F_2)$ and a node $v \in F_1 \triangle F_2$ such that $\{u, v\} \in E$, where $F_1 \triangle F_2 = (F_1 - F_2) \cup (F_2 - F_1)$ denotes the* symmetric difference *between F_1 and F_2.*

3 Local Diagnosis

Let G be a graph and v denote any one of its nodes. The main purpose of local diagnosis is to determine whether or not v is faulty. Obviously, the sets $N_G(v)$ and $\{v\} \cup N_G(v)$ are indistinguishable, so the one-step diagnosability of G must be upper bounded by the minimum degree of G. Instead of addressing such a traditional measurement of diagnosability, Hsu and Tan [19] defined the *local diagnosability* $\tau_G(v)$ of node v in graph G, which is the maximum

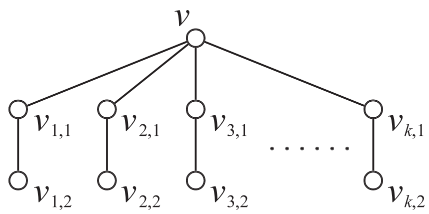


Fig. 1. The extending star $\mathbb{T}_G(v; k)$ consists of $2k + 1$ nodes and $2k$ edges.

positive integer t such that G is locally t-diagnosable at v. It was shown that $\tau(G) = \min\{\tau_G(v) \mid v \in V(G)\}$. Suppose that k is an integer greater than or equal to 1. An *extending star of order k rooted at node v* is defined to be the subgraph of G, denoted by $\mathbb{T}_G(v; k) = (V(v; k), E(v; k))$, where $V(v; k) = \{v\} \cup \{v_{i,j} \mid 1 \leq i \leq k, 1 \leq j \leq 2\}$ and $E(v; k) = \{\{v, v_{i,1}\}, \{v_{i,1}, v_{i,2}\} \mid 1 \leq i \leq k\}$. An extending star of order k is said to be of *full order* if $k = deg_G(u)$. See Fig. 1 for illustration.

Based on the extending star architecture, we design a polynomial-time algorithm, named Local-Diagnosis (**LD**, abbreviated for short), to determine the fault status of its root node [22].

Algorithm 1: **LD**($\mathbb{T}_G(v; t)$)

Input: $\mathbb{T}_G(v; t)$, an extending star of order t rooted at node v in graph G.
Output: A boolean variable, whose value is 0 or 1 if v is fault-free or faulty, respectively.
begin
  $n_{0,0} \leftarrow |\{1 \leq i \leq t \mid (\sigma(v_{i,1}, v), \sigma(v_{i,2}, v_{i,1})) = (0, 0)\}|$;
  $n_{1,0} \leftarrow |\{1 \leq i \leq t \mid (\sigma(v_{i,1}, v), \sigma(v_{i,2}, v_{i,1})) = (1, 0)\}|$;
  if $n_{0,0} \geq n_{1,0}$ **then**
    return 0;
  else
    return 1;
end

Theorem 1. *Let $\mathbb{T}_G(v; t)$ be an extending star of order t rooted at node v in a graph G. The algorithm **LD**($\mathbb{T}_G(v; t)$) correctly identifies the fault status of node v if the total number of faulty nodes in G does not exceed t.*

Proof. Let $n_{i,j} = |\{1 \leq k \leq t \mid (\sigma(v_{k,1}, v), \sigma(v_{k,2}, v_{k,1})) = (i, j)\}|$ for any $i \in \{0, 1\}$ and $j \in \{0, 1\}$. We have $t = n_{0,0} + n_{1,0} + n_{0,1} + n_{1,1}$ to prove this theorem by contradiction.

First, we assume that v is faulty and $n_{0,0} \geq n_{1,0}$. Then, the total number of faulty nodes in G amounts to, at the least, $2n_{0,0} + n_{0,1} + n_{1,1} + 1 \geq (n_{0,0} + n_{1,0}) + n_{0,1} + n_{1,1} + 1 = t + 1$. This contradicts the assumption that the total number of faulty nodes in G does not exceed t. Thus, v is fault-free if $n_{0,0} \geq n_{1,0}$.

Second, we consider that v is fault-free and $n_{0,0} < n_{1,0}$. Then, the total number of faulty nodes in G amounts to, at the least, $2n_{1,0} + n_{0,1} + n_{1,1} > (n_{0,0} + n_{1,0}) + n_{0,1} + n_{1,1} = t$. Again, this contradicts the assumption that the total number of faulty nodes in G does not exceed t. Hence, v is faulty if $n_{0,0} < n_{1,0}$.

Therefore, the proposed algorithm correctly diagnoses the root node v. □

Similar to the statement and the proof of Theorem 1, we have the following result:

Corollary 1. *Let $\mathbb{T}_G(v; t)$ be an extending star of order t rooted at node v in a graph G. The algorithm **LD**($\mathbb{T}_G(v; t)$) correctly identifies the fault status of node v if the total number of faulty nodes in $\mathbb{T}_G(v; t)$ does not exceed t.*

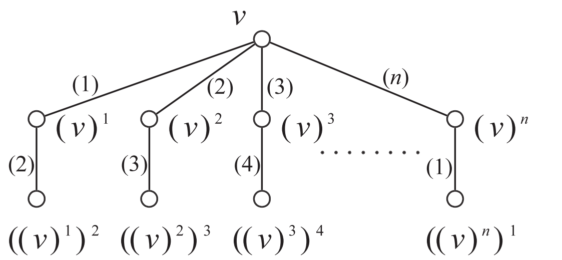


Fig. 2. An extending star of full order rooted at any node v in Q_n.

The extending star is a simple architecture that can be embedded in many interconnection networks, such as hypercubes [32], crossed cubes [12], Möbius cubes [9], star graphs [1], etc. Among various kinds of network topologies, the hypercube is one of the most popular networks for parallel and distributed computation. Not only is it ideally suited to both special-purpose and general-purpose tasks, but also it can efficiently simulate many other networks [21], [25]. Hence, we show how to construct an extending star of full order rooted at any node in the hypercube.

Let $v = b_n \cdots b_i \cdots b_1$ be an n-bit binary string. For $1 \leq i \leq n$, we use $(v)^i$ to denote the binary string $b_n \cdots \bar{b_i} \cdots b_1$. Moreover, we use $[v]_i$ to denote the ith bit b_i of v. The n-dimensional hypercube (or n-cube for short), denoted by Q_n, consists of 2^n nodes and $n2^{n-1}$ edges. Each node corresponds to an n-bit binary string. Two nodes, u and v, are adjacent if and only if $v = (u)^i$ for some i. We say that nodes u and $(u)^i$ are linked by an (i)-edge. An n-cube can be constructed recursively. Let $Q_n^{(0)}$ and $Q_n^{(1)}$ denote two disjoint subgraphs of Q_n induced by node subsets $\{v \in V(Q_n) \mid [v]_n = 0\}$ and $\{v \in V(Q_n) \mid [v]_n = 1\}$, respectively. For $n \geq 2$, $Q_n^{(0)}$ and $Q_n^{(1)}$ are isomorphic to Q_{n-1}. Then, an extending star of full order rooted at node v can be formed by the graph $\mathbb{T}_{Q_n}(v; n)$, whose node set and edge set are

$$\{v, (v)^n, ((v)^n)^1\} \cup \left(\bigcup_{i=1}^{n-1} \{(v)^i, ((v)^i)^{i+1}\}\right) \text{ and } \{\{v, (v)^n\},$$
$$\{(v)^n, ((v)^n)^1\}\} \cup \left(\bigcup_{i=1}^{n-1} \{\{v, (v)^i\}, \{(v)^i, ((v)^i)^{i+1}\}\}\right),$$

respectively. See Fig. 2 for illustration.

We now measure the time complexity of the proposed algorithm. For many practical systems of N nodes, the degree of each node is in the order of $\log N$, and extending stars of full order rooted at each node can be embedded in time $O(\log N)$. For example, both the n-cube and n-dimensional crossed cube have $N = 2^n$ nodes, and the degree of each node is $n = \log N$. Under the PMC model we assume that the time for a node to test another one is a constant c. Then, the running time of the **LD** procedure is $2c \log N = O(\log N)$. In general, the time complexity is $O(k)$ for a k-regular interconnected system.

4 Conditional-Fault Local Diagnosis

A typical approach to the evaluation of one-step diagnosability usually assumes that probabilities of device failures are statistically independent. Not only does this assumption ignore the hidden correlation between device failures, but also it does not take the system size into account. Therefore, Najjar and Gaudiot [30] defined the *fault resilience* as the maximum number of failures that can be sustained while the network remains connected with a high probability. Their simulation showed that the hypercube's fault resilience increases from 25 to 33 percent as its dimensionality increases from 4 to 10. In particular, the 10-cube remains connected with probability 0.99 even when 33 percent of its nodes are injured.

Suppose that G is a graph whose one-step diagnosability is equal to t. Then, G is one-step t-diagnosable but not $(t+1)$-diagnosable. As stated in Section 1, the one-step diagnosability is upper bounded by the minimum degree. For many t-diagnosable interconnected systems, the only case stopping them from being $(t+1)$-diagnosable is usually that some node happens to have no fault-free neighbor. For example, members in the cube family are so. A system is known to be *strongly t-diagnosable* if it is one-step t-diagnosable and can achieve $(t+1)$-diagnosability, except for the case where a node's neighbors are all faulty. Recently, Hsieh and Chuang [17] studied the strong diagnosability of regular networks and product networks under the PMC model. Such results are raising an intriguing question: How large is the maximum integer t such that G remains t-diagnosable when every fault-free node has at least one fault-free neighbor?

4.1 Conditional Diagnosis

In this paper, a set F of faulty nodes in a graph G is called *conditionally faulty* if $N_G(v) \not\subseteq F$ for every node $v \in V(G) - F$. A graph is conditionally faulty if its faulty nodes form a conditionally faulty set.

Definition 1. *A graph G is said to be* conditionally t-diagnosable *if, for any two distinct conditionally faulty sets F_1 and F_2 of G with $|F_1| \leq t$ and $|F_2| \leq t$, F_1 and F_2 are distinguishable.*

The maximum number of faulty nodes that can be correctly identified with one application of some diagnostic process in a conditionally faulty system G is said to be the *conditional diagnosability* of G, denoted by $\tau^c(G)$. We propose the following concepts.

Definition 2. *A graph G is conditionally t-diagnosable locally at node $v \in V(G)$ if F_1 and F_2 are distinguishable for any two different conditionally faulty sets $F_1, F_2 \subset V(G)$ such that $v \in F_1 \triangle F_2$, $|F_1| \leq t$, and $|F_2| \leq t$.*

Definition 3. *Let G be a graph, and let v be any node in G. The* conditionally local diagnosability *of node v in graph G, denoted by $\tau_G^c(v)$, is defined to be the maximum integer t such that G is conditionally t-diagnosable locally at node v.*

The next two theorems clarify the relationship between conditional diagnosability and conditionally local diagnosability.

Theorem 2. *A graph G is conditionally t-diagnosable if and only if it is conditionally t-diagnosable locally at each node.*

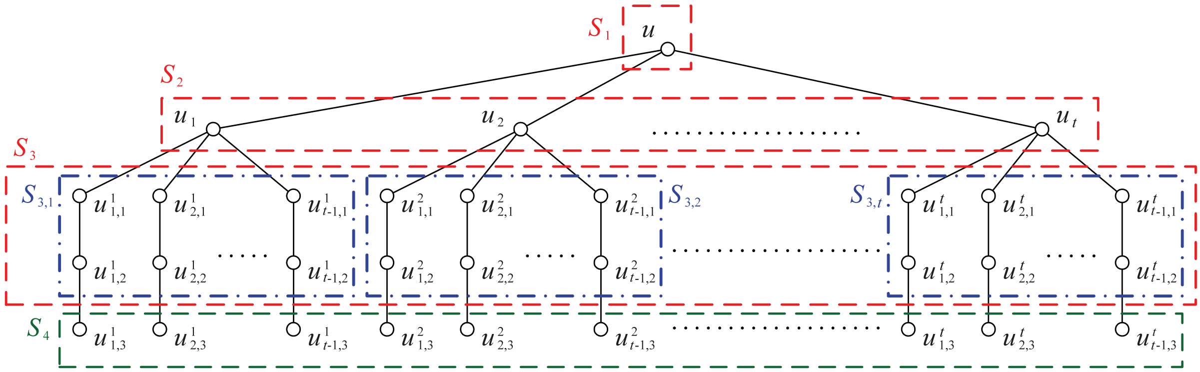


Fig. 3. The branch-of-tree of order t rooted at node u, in which $|S_4| = t^2 - t$ and $S_{3,i} \cap S_{3,j} = \emptyset$ whenever $i \neq j$.

Proof. To prove the necessity, we assume that G is conditionally t-diagnosable. Then, we have to show that G is conditionally t-diagnosable locally at each node. If not, G is not conditionally t-diagnosable locally at some node $v \in V(G)$. By Definition 2, there exists an indistinguishable pair F_1, F_2 of conditionally faulty sets in G, where $|F_1| \leq t$, $|F_2| \leq t$, and $v \in F_1 \triangle F_2$. By Definition 1, G is not conditionally t-diagnosable, contradicting the initial assumption. As a result, the necessary condition follows.

To prove the sufficiency, we assume that G is conditionally t-diagnosable locally at each node. Then, we need to show that G is conditionally t-diagnosable. Suppose, by contradiction, that G is not conditionally t-diagnosable. By Definition 1, there exist indistinguishable faulty sets F_1 and F_2 in G with $|F_1| \leq t$ and $|F_2| \leq t$. Since $F_1 \neq F_2$, there is some node v in $F_1 \triangle F_2$. By Definition 2, G is not conditionally t-diagnosable locally at node v, contradicting the initial assumption. Consequently, the sufficient condition holds. □

Theorem 3. *Let G denote the underlying topology of an interconnected system. Then, $\tau^c(G) = \min\{\tau^c_G(v) \mid v \in V(G)\}$.*

Proof. For convenience, let $t = \min\{\tau^c_G(v) \mid v \in V(G)\}$. That is, G is conditionally t-diagnosable locally at every node. By Theorem 2, G is conditionally t-diagnosable too. Since $t = \min\{\tau^c_G(v) \mid v \in V(G)\}$, G is not conditionally $(t+1)$-diagnosable locally at some node $u \in V(G)$. It still follows from Theorem 2 that G is not conditionally $(t+1)$-diagnosable. □

4.2 The Branch-of-Tree Architecture

In this section we present an architecture, named branch-of-tree, which helps with identifying the fault status of a given node in a conditionally faulty system.

Definition 4. *Let u be any node in a graph G, and let t be any positive integer with $t \geq 2$. We set $S_1 = \{u\}$, $S_2 = \{u_i \mid 1 \leq i \leq t\}$, $S_3 = \bigcup_{i=1}^{t} S_{3,i}$, where $S_{3,i} = \{u^i_{j,k} \mid 1 \leq j \leq t-1$ and $1 \leq k \leq 2\}$ for every $1 \leq i \leq t$, and $S_4 = \bigcup_{i=1}^{t}\bigcup_{j=1}^{t-1}\{u^i_{j,3}\}$. Let $\mathbb{B}_G(u;t) = (V(u;t), E(u;t))$ be a subgraph of G with $V(u;t) = S_1 \cup S_2 \cup S_3 \cup S_4$ and $E(u;t) = \{\{u, u_i\} \mid 1 \leq i \leq t\} \cup \{\{u_i, u^i_{j,1}\} \mid 1 \leq i \leq t$ and $1 \leq j \leq t-1\} \cup \{\{u^i_{j,k}, u^i_{j,k+1}\} \mid 1 \leq i \leq t, 1 \leq j \leq t-1,$ and $1 \leq k \leq 2\}$. Then, $\mathbb{B}_G(u;t)$ is called a* branch-of-tree of order t rooted at *node u if all of the following conditions hold:*

1. *$S_i \cap S_j = \emptyset$ for every $i \neq j$,*
2. *$|S_2| = t$,*
3. *$|S_{3,i}| = 2t - 2$ for every $1 \leq i \leq t$,*
4. *$|S_{3,i} \cap S_{3,j}| \leq 1$ for every two distinct elements i and j with $1 \leq i \leq t$ and $1 \leq j \leq t$, and*
5. *$|S_4| \geq 1$.*

The branch-of-tree is not strictly a tree. Fig. 3 illustrates the branch-of-tree of order t rooted at node u with $|S_4| = t^2 - t$ and $S_{3,i} \cap S_{3,j} = \emptyset$ for every two distinct elements i and j, $1 \leq i \leq t$ and $1 \leq j \leq t$. Fig. 4 illustrates two different branch-of-trees of order 4.

Theorem 4. *Let G be a graph and $u \in V(G)$ denote a node. Suppose that the degree t of node u is at least 4; i.e., $t \geq 4$. Then, G is conditionally $(2t-1)$-diagnosable locally at node u if it contains a branch-of-tree of order t rooted at u as a subgraph.*

Proof. Suppose that G contains a branch-of-tree of order t rooted at u, $\mathbb{B}_G(u;t)$, as a subgraph. Let F_1 and F_2 be any two distinct conditionally faulty sets such that $|F_1| \leq 2t-1$, $|F_2| \leq 2t-1$, and $u \in F_1 \triangle F_2$. It suffices to prove that F_1 and F_2 are distinguishable.

Without loss of generality, we assume that $u \in F_1$. For convenience, let $F = F_1 \cap F_2$ and $p = |F|$. Obviously, we have $0 \leq p \leq 2t-2$. Because both F_1 and F_2 are conditionally faulty, their intersection F is conditionally faulty too. Since $u \notin F$, we have $|N_G(u) \cap F| \leq \min\{t-1, p\}$.

Case 1: Suppose that $p = 2t-2$. Since $u \in F_1$, we have $|F_1| = 2t-1$. Obviously, u is the sole node in $F_1 - F_2$. We claim that u has a neighbor outside $F_1 \cup F_2$; that is, $N_G(u) - (F_1 \cup F_2)$ is not empty. If not, then $N_G(u) \subseteq F_2$. However, F_2 is a conditionally faulty set not containing node u, so it cannot cover the set of all neighbors of u. Therefore, we obtain a contradiction, and the claim holds. By Lemma 1, F_1 and F_2 are distinguishable.

Case 2: Suppose that $p \leq 2t-3$. Applying Lemma 1, it suffices to show that the component C_u of $\mathbb{B}_G(u;t) - F$, which u belongs to, contains at least $2(2t-1-p)+1 = 4t-2p-1$ nodes. We set $\mathbb{T}_G(u_i; t-1)$ to be the subgraph of $\mathbb{B}_G(u;t)$ induced by $\{u_i\} \cup \{u^i_{j,k} \mid 1 \leq j \leq t-1$ and $1 \leq k \leq 2\}$. It is noted that $\mathbb{T}_G(u_i; t-1)$ has $2(t-1)+1 = 2t-1$ nodes. Let $r = |N_G(u) \cap F|$. Without loss of generality, we assume that $\{u_i \mid r+1 \leq i \leq t\} \cap F = \emptyset$.

Subcase 2.1: Assume that $r \leq t-2$. Obviously, we have $F \cap V(\mathbb{B}_G(u;t)) - \bigcup_{i \leq r}\{u_i\} \subset S_3 \cup S_4$. Denote $F \cap V(\mathbb{B}_G(u;t)) - \bigcup_{i \leq r}\{u_i\}$ by F'. When $u^{i_1}_{j_1,k_1} = u^{i_2}_{j_2,k_2} \in F'$ for $r+1 \leq i_1 < i_2 \leq t$, $1 \leq j_1, j_2 \leq t-1$, and $1 \leq k_1$,

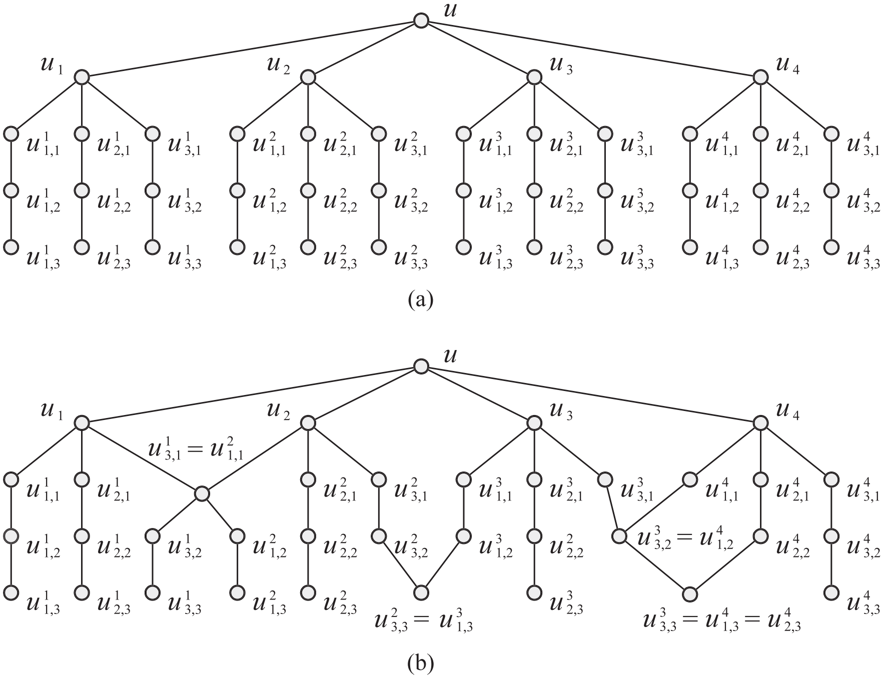


Fig. 4. (a) A branch-of-tree of order 4 rooted at node u consists 41 nodes and 40 edges. (b) A branch-of-tree of order 4 rooted at node u consists 36 nodes and 39 edges.

$k_2 \leq 2$, maybe none of $A = \{u^{i_1}_{j_1,1}, u^{i_1}_{j_1,2}\} \cup \{u^{i_2}_{j_2,1}, u^{i_2}_{j_2,2}\}$ is in $\mathbb{B}_G(u;t) - F$. Since $|S_{3,i} \cap S_{3,j}| \leq 1$ for any two distinct integers $i, j \in \{1, 2, \ldots, t\}$, we have $|A| = 3$ if $u^{i_1}_{j_1,k_1} = u^{i_2}_{j_2,k_2}$, and then

$$|V(\mathbb{B}_G(u;t) - \{v\})| \geq |\{u\} \cup \bigcup_{i=r+1}^{t} V(\mathbb{T}_G(u_i; t-1))| - 3$$

for every $v \in F'$. Hence, C_u has at least

$$\begin{aligned}&|\{u\} \cup \bigcup_{i=r+1}^{t} V(\mathbb{T}_G(u_i; t-1))| - 3|F'| \geq |\{u\}| \\ &\quad + \sum_{i=r+1}^{t} |V(\mathbb{T}_G(u_i; t-1))| - \binom{t-r}{2} - 3|F'| \geq 1 \\ &\quad + (t-r)(2t-1) - \binom{t-r}{2} - 3(p-r)\end{aligned}$$

nodes. Comparing this with $4t - 2p - 1$, we can compute their difference as follows:

$$\begin{aligned}\Delta &= \left[1 + (t-r)(2t-1) - \binom{t-r}{2} - 3(p-r)\right] \\ &\quad - (4t - 2p - 1) \\ &\geq \frac{1}{2}(3t^2 - 13t + 10 - r^2 - 2rt + 7r) \ (\because p \leq 2t-3) \\ &= \frac{1}{2}[3t^2 - 13t + 10 - r(r + 2t - 7)] \\ &\geq \frac{1}{2}[3t^2 - 13t + 10 - (t-2)(t-2+2t-7)] \ (\because r \leq t-2) \\ &= t - 4.\end{aligned}$$

Therefore, we have $\Delta \geq 0$ if $t \geq 4$.

Subcase 2.2: Assume that $r = t - 1$. That is, u_t is the sole neighbor of u in $\mathbb{B}_G(u;t) - F$. Let $L_j = \{u^t_{j,1}, u^t_{j,2}\}$ for $1 \leq j \leq t-1$. If $\{u^t_{1,3}, u^t_{2,3}, \ldots, u^t_{t-1,3}\} \cap F = \emptyset$, there exist at least $(t-1) - (p-r) = t - p + r - 1$ distinct integers, $1 \leq j_1 < \cdots < j_{t-p+r-1} \leq t-1$, such that

$$F \cap \bigcup_{h=1}^{t-p+r-1} L_{j_h} = \emptyset.$$

So, C_u contains

$$\{u\} \cup \{u_t\} \cup \bigcup_{h=1}^{t-p+r-1} L_{j_h} \cup \bigcup_{h=1}^{t-p+r-1} \{u^t_{j_h,3}\}.$$

Accordingly, C_u has at least $1 + 1 + 2(t-p+r-1) + 1 = 2t - 2p + 2r + 1$ nodes. Otherwise, we have $\{u^t_{1,3}, u^t_{2,3}, \ldots, u^t_{t-1,3}\} \cap F \neq \emptyset$, so there exist at least $t - p + r$ integers, $1 \leq j_1 < \cdots < j_{t-p+r} \leq t-1$, such that

$$F \cap \bigcup_{h=1}^{t-p+r} L_{j_h} = \emptyset.$$

Thus, C_u contains $\{u\} \cup \{u_t\} \cup \bigcup_{h=1}^{t-p+r} L_{j_h}$. Accordingly, C_u has at least $1 + 1 + 2(t-p+r) = 2t - 2p + 2r + 2$ nodes. Comparing them with $4t - 2p - 1$, we can derive the inequality as follows:

$$\begin{aligned}\Delta &= \min\{2t - 2p + 2r + 1, 2t - 2p + 2r + 2\} - (4t - 2p - 1) \\ &= (2t - 2p + 2r + 1) - (4t - 2p - 1) \\ &= (4t - 2p - 1) - (4t - 2p - 1) \\ &= 0.\end{aligned}$$

In either subcase, component C_u contains at least $4t - 2p - 1$ nodes. This implies that some node $x \in V(G) - (F_1 \cup F_2)$ is adjacent to a node $y \in F_1 \triangle F_2$. By Lemma 1, F_1 and F_2 are distinguishable. □

4.3 The Fault Identification Algorithm

By means of collecting syndromes from branch-of-tree $\mathbb{B}_G(u;t)$, we design an efficient algorithm, namely Local-Diagnosis-Under-Conditional-Faults (**LDUCF**), which can identify the fault status of node u in a conditionally faulty system G. It is noticed that the following notations u_i and $u^i_{j,k}$ are the same as those in Definition 4.

Algorithm 2: **LDUCF**$(\mathbb{B}_G(u;t))$

Input: $\mathbb{B}_G(u;t)$, a branch-of-tree of order t rooted at node u in graph G.
Output: A boolean variable, whose value is 0 or 1 if u is fault-free of faulty, respectively.
begin
  $\mathcal{D} \leftarrow \emptyset$;
  for $i = 1$ *to* t **do**
    Set $\mathbb{T}_G(u_i; t-1)$ to be the subgraph of $\mathbb{B}_G(u;t)$ induced by the set $\{u_i\} \cup \{u^i_{j,k} \mid 1 \le j \le t-1 \text{ and } 1 \le k \le 2\}$;
    $n^i_{0,0} \leftarrow |\{1 \le j \le t-1 \mid (\sigma(u^i_{j,1}, u_i), \sigma(u^i_{j,2}, u^i_{j,1})) = (0,0)\}|$;
    $n^i_{1,0} \leftarrow |\{1 \le j \le t-1 \mid (\sigma(u^i_{j,1}, u_i), \sigma(u^i_{j,2}, u^i_{j,1})) = (1,0)\}|$;
    if LD$(\mathbb{T}_G(u_i; t-1)) = 0$ **then**
      $\mathcal{D} \leftarrow \mathcal{D} \cup \{u_i\}$;
  if $|\mathcal{D}| \ge 3$ **then**
    return VOTE$(u, \mathcal{D})$;
  else if $|\mathcal{D}| = 2$ **then**
    Let u_p and u_q be two nodes in $\mathcal{D}$;
    if $n^p_{0,0} - n^p_{1,0} \ge n^q_{0,0} - n^q_{1,0}$ **then**
      return $\sigma(u_p, u)$;
    else
      return $\sigma(u_q, u)$;
  else if $|\mathcal{D}| = 1$ **then**
    return VOTE$(u, \mathcal{D})$;
  else
    if $n^i_{1,0} - n^i_{0,0} \ge 2$ *for every* $1 \le i \le t$ **then**
      return 1;
    else
      Let p be an integer such that $n^p_{1,0} - n^p_{0,0} = 1$;
      Let $r \in \{1, 2, \cdots, t-1\}$ such that $(\sigma(u^p_{r,1}, u_p), \sigma(u^p_{r,2}, u^p_{r,1})) = (1,0)$;
      if $\sigma(u^p_{r,3}, u^p_{r,2}) = 1$ **then**
        return $\sigma(u_p, u)$;
      else
        return 1;
end

Algorithm 3: **VOTE**$(u, \mathcal{D})$

Input: A node u and a subset $\mathcal{D}$ of the set of u's neighboring nodes.
Output: A boolean variable, whose value is 0 or 1 if u is fault-free or faulty, respectively.
begin
  if $|\mathcal{D}| = 1$ **then**
    Let v be the node in $\mathcal{D}$;
    return $\sigma(v, u)$;
  else
    $n_0 \leftarrow |\{v \in \mathcal{D} \mid \sigma(v,u) = 0\}|$;
    $n_1 \leftarrow |\{v \in \mathcal{D} \mid \sigma(v,u) = 1\}|$;
    if $n_0 \ge n_1$ **then**
      return 0;
    else
      return 1;
end

Some notation used in the analysis of **LDUCF** algorithm is introduced below. For any positive integer $t \ge 4$, let G be a conditionally faulty t-regular graph, and let $\mathbb{B}_G(u;t)$ be a branch-of-tree of order t rooted at node u in graph G. Suppose that $\mathbb{B}_G(u;t)$ contains at most $2t-1$ faulty nodes. We set $\mathbb{T}_G(u_i;t-1)$ to be the subgraph of $\mathbb{B}_G(u;t)$ induced by $\{u_i\} \cup \{u^i_{j,k} \mid 1 \le j \le t-1 \text{ and } 1 \le k \le 2\}$, define $\mathcal{D}$ to be the set $\cup_{1 \le i \le t}\{u_i \mid \mathbf{LD}(\mathbb{T}_G(u_i;t-1)) = 0\}$, and set $S_{3,l} = \{u^l_{j,k} \mid 1 \le j \le t-1 \text{ and } 1 \le k \le 2\}$ for every $1 \le l \le t$. It is noticed that $|S_{3,i} \cap S_{3,j}| \le 1$ for any two distinct i and j, $1 \le i, j \le t$. Moreover, let $n^i_{\alpha,\beta} = |\{1 \le j \le t-1 \mid (\sigma(u^i_{j,1}, u_i), \sigma(u^i_{j,2}, u^i_{j,1})) = (\alpha, \beta)\}|$, where $(\alpha,\beta) \in \{(0,0),(0,1),(1,0),(1,1)\}$. For $1 \le i \le t$, let f_i be 0 or 1 if node u_i is fault-free or faulty, respectively, and let $\hat{f}_i = \mathbf{LD}(\mathbb{T}_G(u_i;t-1))$. Finally, we use $F(H)$ to denote the set of all faulty nodes in graph H.

Lemma 2. *Suppose that $t \ge 4$. Then, $\sum_{i=1}^t |\hat{f}_i - f_i| \le 2$.*

Proof. Suppose, by contradiction, that $\sum_{i=1}^t |\hat{f}_i - f_i| \ge 3$. That is, there exist at least three distinct integers $1 \le p_1, p_2, p_3 \le t$ such that $\hat{f}_{p_k} \ne f_{p_k}$ for $k = 1,2,3$. By Corollary 1, we have $|F(\mathbb{T}_G(u_{p_k};t-1))| \ge t$ for $k = 1,2,3$. Since $|S_{3,i} \cap S_{3,j}| \le 1$ for any two distinct i and j, $1 \le i,j \le t$, we obtain $|F(\mathbb{B}_G(u;t))| \ge \sum_{k=1}^3 |F(\mathbb{T}_G(u_{p_k}; t-1))| - (|S_{3,p_1} \cap S_{3,p_2}| + |S_{3,p_1} \cap S_{3,p_3}| + |S_{3,p_2} \cap S_{3,p_3}|) \ge 3t-3 > 2t-1$ for $t \ge 4$. This contradicts the condition that $\mathbb{B}_G(u;t)$ has at most $2t-1$ faulty nodes. Thus, the proof is completed. □

Lemma 3. *Suppose that $2 \le |\mathcal{D}| \le t-1$ for any $t \ge 4$. Then, the set $\mathcal{D}$ contains at most one faulty node in it.*

Proof. It follows from Lemma 2 that $\sum_{i=1}^t |\hat{f}_i - f_i| \le 2$. Suppose, by contradiction, that there are two faulty nodes, say u_i and u_j, in $\mathcal{D}$. Hence, every of $N_G(u) - \mathcal{D}$ is really faulty nodes. By Corollary 1, we have $|F(\mathbb{T}_G(u_i;t-1))| \ge t$ and $|F(\mathbb{T}_G(u_j;t-1))| \ge t$. Since $|S_{3,i} \cap S_{3,j}| \le 1$, we obtain

$$|F(\mathbb{B}_G(u;t))| \ge |F(\mathbb{T}_G(u_i;t-1))| + |F(\mathbb{T}_G(u_j;t-1))| - |S_{3,i} \cap S_{3,j}| + |N_G(u) - \mathcal{D}| \ge 2t-1+(t-|\mathcal{D}|) \ge 2t-1+[t-(t-1)] = 2t > 2t-1,$$

contradicting the condition that $\mathbb{B}_G(u;t)$ has at most $2t-1$ faulty nodes. Therefore, the lemma holds. □

Lemma 4. *Suppose that $|\mathcal{D}| \le 1$ and $t \ge 4$. Then, $\sum_{i=1}^t |\hat{f}_i - f_i| \le 1$.*

Proof. By Lemma 2, we know that $\sum_{i=1}^t |\hat{f}_i - f_i| \le 2$. Suppose that $\sum_{i=1}^t |\hat{f}_i - f_i| = 2$. That is, there exist two distinct integers $1 \le p, q \le t$ such that $\hat{f}_p \ne f_p$ and $\hat{f}_q \ne f_q$. By Corollary 1, we have $|F(\mathbb{T}_G(u_p;t-1))| \ge t$ and $|F(\mathbb{T}_G(u_q;t-1))| \ge t$. Since $|S_{3,p} \cap S_{3,q}| \le 1$, we derive $|F(\mathbb{B}_G(u;t))| \ge 2t-1+(t-|\mathcal{D}|-2) = 2t-3+(t-|\mathcal{D}|) \ge 3t-4 > 2t-1$ for $t \ge 4$. By contradiction, this lemma follows. □

We analyze the **LDUCF** procedure step by step.

4.3.1 $|\mathcal{D}| \ge 3$

The following theorem is drawn from Lemmas 2 and 3.

Theorem 5. *Suppose that $|\mathcal{D}| \ge 3$ and $t \ge 4$. Let $n_0 = |\{v \in \mathcal{D} \mid \sigma(v,u) = 0\}|$ and $n_1 = |\{v \in \mathcal{D} \mid \sigma(v,u) = 1\}|$. Then, u is fault-free if and only if $n_0 \ge n_1$.*

Proof. Let m_0 and m_1 denote the numbers of fault-free nodes and faulty nodes in $\mathcal{D}$, respectively. It follows from Lemmas 2 and 3 that $m_0 \ge m_1$.

Case 1: $m_0 > m_1$. Because fault-free testers always make reliable diagnosis, we have either $n_0 > n_1$ or $n_0 < n_1$. If u is fault-free, then we have $n_0 \ge m_0 > m_1 \ge n_1$; otherwise, we have $n_1 \ge m_0 > m_1 \ge n_0$. By contraposition, u is fault-free or faulty if $n_0 > n_1$ or $n_0 < n_1$, respectively.

Case 2: $m_0 = m_1$. In this case, $\mathcal{D}$ has exactly two faulty nodes, say u_1 and u_2. Then, we have $2t-1 \ge |F(\mathbb{T}_G(u_1; t-1))| + |F(\mathbb{T}_G(u_2;t-1))| - |S_{3,1} \cap S_{3,2}| \ge 2t-1$. That is, all faulty nodes are in $\mathbb{T}_G(u_1;t-1)$ and $\mathbb{T}_G(u_2;t-1)$. Hence, u must be fault-free and $n_0 \ge m_0 = m_1 \ge n_1$. □

4.3.2 $|\mathcal{D}| = 2$

We have the following theorem.

Theorem 6. *Suppose that $\mathcal{D} = \{u_p, u_q\}$ with $n^p_{0,0} - n^p_{1,0} \geq n^q_{0,0} - n^q_{1,0}$. Then, node u_p is fault-free if $t \geq 4$.*

Proof. Without loss of generality, we assume that $p = 1$ and $q = 2$. Suppose, by contradiction, that node u_1 is faulty. By Lemma 3, node u_2 is fault-free. Therefore, it follows from Lemma 2 that $\sum_{3 \leq i \leq t} |\hat{f}_i - f_i| \leq 1$.

Case 1: Suppose that $\sum_{i=3}^{t} |\hat{f}_i - f_i| = 0$. Thus, nodes $u_3, u_4, \ldots, u_t$ are faulty. Accordingly, we have

$$\begin{aligned}|F(\mathbb{B}_G(u;t))| &\geq |F(\mathbb{T}_G(u_1;t-1))| + |F(\mathbb{T}_G(u_2;t-1))| \\ &\quad - |S_{3,1} \cap S_{3,2}| + |\{u_3, u_4, \ldots, u_t\}| \\ &\geq \left(1 + 2n^1_{0,0} + n^1_{0,1} + n^1_{1,1}\right) + \left(2n^2_{1,0} + n^2_{0,1} + n^2_{1,1}\right) \\ &\quad - 1 + (t-2) \geq \left(n^1_{0,0} + n^1_{0,1} + n^1_{1,0} + n^1_{1,1}\right) \\ &\quad + \left(n^2_{0,0} + n^2_{0,1} + n^2_{1,0} + n^2_{1,1}\right) + (t-2) \\ &= (t-1) + (t-1) + (t-2) = 3t - 4 \\ &> 2t - 1 \text{ for } t \geq 4.\end{aligned}$$

Case 2: Suppose that $\sum_{i=3}^{t} |\hat{f}_i - f_i| = 1$. Let r be the integer in $\{3, 4, \ldots, t\}$ such that $\hat{f}_r \neq f_r$. Thus, we have

$$\begin{aligned}|F(\mathbb{B}_G(u;t))| &\geq |F(\mathbb{T}_G(u_1;t-1))| + |F(\mathbb{T}_G(u_r;t-1))| \\ &\quad - |S_{3,1} \cap S_{3,r}| + |\{u_3, u_4, \ldots, u_t\} - \{u_r\}| \\ &\geq t + t - 1 + (t-3) = 3t - 4 > 2t - 1 \text{ for } t \geq 4.\end{aligned}$$

In either case, the number of faulty nodes exceeds $2t - 1$. Hence, the theorem holds. □

4.3.3 $|\mathcal{D}| = 1$

Suppose that $\mathcal{D} = \{u_p\}$. We claim that node u_p is fault-free. If not, then $|\hat{f}_p - f_p| = 1$. Because Lemma 4 ensures that $\sum_{i=1}^{t} |\hat{f}_i - f_i| \leq 1$, nodes $u_1, u_2, \ldots, u_t$ are all faulty. That is, node u has no fault-free neighbors. According to the definition of conditional faults, the node u must be faulty too. Then, we have that $|F(\mathbb{B}_G(u;t))| \geq |F(\mathbb{T}_G(u_p;t-1))| + |\{u\} \cup N_G(u) - \{u_p\}| \geq t + t = 2t$, violating the condition that $|F(\mathbb{B}_G(u;t))| \leq 2t - 1$. By contradiction, u_p has to be fault-free.

4.3.4 $|\mathcal{D}| = 0$

Suppose that $n^p_{1,0} - n^p_{0,0} \geq 2$ for any $1 \leq p \leq t$. We claim that node u_p is faulty. Without loss of generality, we assume that $p = 1$. If u_1 is fault-free, then it follows from Lemma 4 that nodes $u_2, u_3, \ldots, u_t$ are faulty. Accordingly, we have that $|F(\mathbb{B}_G(u;t))| \geq |F(\mathbb{T}_G(u_1;t-1))| + |\{u_2, u_3, \ldots, u_t\}| \geq 2n^1_{1,0} + n^1_{0,1} + n^1_{1,1} + (t-1) \geq n^1_{0,0} + n^1_{0,1} + n^1_{1,0} + n^1_{1,1} + 2 + (t-1) = (t-1) + 2 + (t-1) = 2t > 2t - 1$. By contradiction, the claim holds.

Lemma 5. *Suppose that $|\mathcal{D}| = 0$ and there exists an index p in $\{1, 2, \ldots, t\}$ such that $n^p_{1,0} - n^p_{0,0} = 1$. Then, $n^i_{1,0} - n^i_{0,0} \geq 2$ for every $i \in \{1, 2, \ldots, t\} - \{p\}$ if $t \geq 4$.*

Proof. We assume, by contradiction, that $n^q_{1,0} - n^q_{0,0} = 1$ for some $q \in \{1, 2, \ldots, t\} - \{p\}$. By Lemma 4, we have $\sum_{i=1}^{t} |\hat{f}_i - f_i| \leq 1$. Hence, at most one of u_p and u_q is fault-free.

Case 1: Either u_p or u_q is fault-free. Without loss of generality, we assume that u_p is fault-free. Then,

$$\begin{aligned}|F(\mathbb{B}_G(u;t))| &\geq |F(\mathbb{T}_G(u_p;t-1))| + |F(\mathbb{T}_G(u_q;t-1))| \\ &\quad - |S_{3,p} \cap S_{3,q}| + |N_G(u) - \{u_p, u_q\}| \\ &\geq \left(n^p_{0,1} + 2n^p_{1,0} + n^p_{1,1}\right) + \left(2n^q_{0,0} + n^q_{0,1} + n^q_{1,1} + 1\right) \\ &\quad - 1 + (t-2) = \left(n^p_{0,0} + n^p_{0,1} + n^p_{1,0} + n^p_{1,1} + 1\right) \\ &\quad + \left(n^q_{0,0} + n^q_{0,1} + n^q_{1,0} + n^q_{1,1}\right) - 1 + (t-2) \\ &= 3t - 4 > 2t - 1 \text{ for } t \geq 4.\end{aligned}$$

Case 2: Both u_p and u_q are faulty.

Subcase 2.1: Suppose that $\sum_{i=1}^{t} |\hat{f}_i - f_i| = 0$. That is, u has no fault-free neighbors. According to the definition of conditional faults, u is faulty too. Hence,

$$\begin{aligned}|F(\mathbb{B}_G(u;t))| &\geq |F(\mathbb{T}_G(u_p;t-1))| + |F(\mathbb{T}_G(u_q;t-1))| \\ &\quad - |S_{3,p} \cap S_{3,q}| + |\{u\} \cup N_G(u) - \{u_p, u_q\}| \\ &\geq \left(1 + 2n^p_{0,0} + n^p_{0,1} + n^p_{1,1}\right) + \left(1 + 2n^q_{0,0} + n^q_{0,1}\right. \\ &\quad \left. + n^q_{1,1}\right) - 1 + (t-1) = \left(n^p_{0,0} + n^p_{0,1} + n^p_{1,0}\right. \\ &\quad \left. + n^p_{1,1}\right) + \left(n^q_{0,0} + n^q_{0,1} + n^q_{1,0} + n^q_{1,1}\right) - 1 \\ &\quad + (t-1) = 3t - 4 > 2t - 1 \text{ for } t \geq 4.\end{aligned}$$

Subcase 2.2: Suppose that $\sum_{i=1}^{t} |\hat{f}_i - f_i| = 1$. Let $r \in \{1, 2, \ldots, t\} - \{p, q\}$ such that $\hat{f}_r \neq f_r$. That is, u_r is fault-free. Then, we can estimate the number of faulty nodes as follows:

$$\begin{aligned}|F(\mathbb{B}_G(u;t))| &\geq |F(\mathbb{T}_G(u_p;t-1))| + |F(\mathbb{T}_G(u_q;t-1))| \\ &\quad + |F(\mathbb{T}_G(u_r;t-1))| - |S_{3,p} \cap S_{3,q}| \\ &\quad - |S_{3,p} \cap S_{3,r}| - |S_{3,r} \cap S_{3,q}| + |N_G(u) \\ &\quad - \{u_p, u_q, u_r\}| \geq \left(1 + 2n^p_{0,0} + n^p_{0,1} + n^p_{1,1}\right) \\ &\quad + \left(1 + 2n^q_{0,0} + n^q_{0,1} + n^q_{1,1}\right) + t - 3 + (t-3) \\ &= \left(n^p_{0,0} + n^p_{0,1} + n^p_{1,0} + n^p_{1,1}\right) + \left(n^q_{0,0} + n^q_{0,1}\right. \\ &\quad \left. + n^q_{1,0} + n^q_{1,1}\right) + 2t - 6 = (t-1) + (t-1) \\ &\quad + (2t-6) = 4t - 8 > 2t - 1 \text{ for } t \geq 4.\end{aligned}$$

By contradiction, this lemma holds. □

Theorem 7. *Suppose that $|\mathcal{D}| = 0$ and there exists an index p in $\{1, 2, \ldots, t\}$ such that $n^p_{1,0} - n^p_{0,0} = 1$. Let r be an index in $\{1, 2, \ldots, t-1\}$ such that $(\sigma(u^p_{r,1}, u_p), \sigma(u^p_{r,2}, u^p_{r,1})) = (1, 0)$. Then, u_p is faulty if and only if $\sigma(u^p_{r,3}, u^p_{r,2}) = 0$.*

Proof. It follows from Lemma 5 that $n^i_{1,0} - n^i_{0,0} \geq 2$ for every $i \in \{1, 2, \ldots, t\} - \{p\}$. Thus, node u_i is faulty for $i \in \{1, 2, \ldots, t\} - \{p\}$. It suffices to consider the following two cases:

Case 1: Suppose that $\sigma(u^p_{r,3}, u^p_{r,2}) = 0$. We claim that node u_p is faulty. If not, then $u^p_{r,1}$, $u^p_{r,2}$, and $u^p_{r,3}$ are faulty. Thus, we have

$$\begin{aligned}|F(\mathbb{B}_G(u;t))| &\geq |F(\mathbb{T}_G(u_p;t-1))| + |\{u^p_{r,3}\}| + |N_G(u) \\ &\quad - \{u_p\}| \geq (2n^p_{1,0} + n^p_{0,1} + n^p_{1,1}) + 1 + (t-1) \\ &= (n^p_{0,0} + n^p_{1,0} + n^p_{0,1} + n^p_{1,1}) + 2 + (t-1) \\ &= (t-1) + 2 + (t-1) = 2t > 2t - 1.\end{aligned}$$

By contradiction, u_p is faulty if $\sigma(u^p_{r,3}, u^p_{r,2}) = 0$.

Case 2: Suppose that $\sigma(u^p_{r,3}, u^p_{r,2}) = 1$. We claim that u_p is fault-free. If not, u has no fault-free neighbors. By the definition of conditional faults, u is faulty too. Since

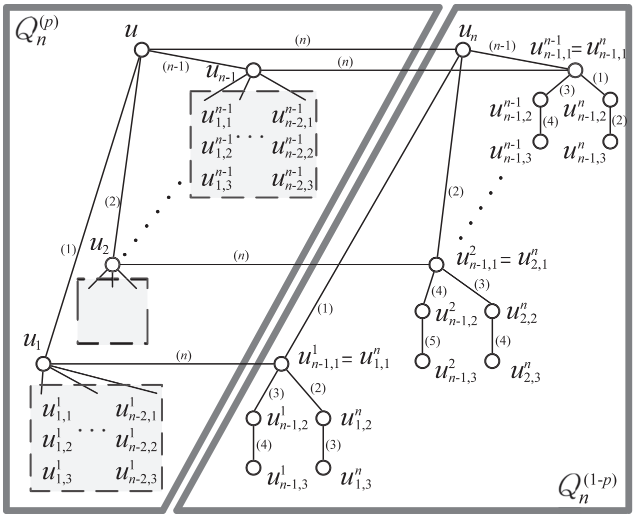


Fig. 5. The branch-of-tree rooted at any node u in $Q_n^{(p)}$, $p \in \{0, 1\}$. The label (i), $1 \le i \le n$, denotes an (i)-edge.

$\sigma(u_{r,3}^p, u_{r,2}^p) = 1$, $u_{r,3}^p$ and/or $u_{r,2}^p$ must be faulty. If $u_{r,3}^p$ is faulty, we have

$$|F(\mathbb{B}_G(u;t))| \ge |F(\mathbb{T}_G(u_p; t-1))| + |\{u_{r,3}^p\}| + |\{u\} \cup N_G(u) - \{u_p\}| \ge \left(1 + 2n_{0,0}^p + n_{0,1}^p + n_{1,1}^p\right) + 1 + t = 2t.$$

If $u_{r,2}^p$ is faulty, then

$$\begin{aligned} |F(\mathbb{B}_G(u;t))| &\ge |F(\mathbb{T}_G(u_p; t-1))| + |\{u\} \cup N_G(u) - \{u_p\}| \\ &\ge \left(1 + 2n_{0,0}^p + n_{0,1}^p + n_{1,1}^p + \left|\{u_{r,2}^p\}\right|\right) + t \\ &= \left(n_{0,0}^p + n_{0,1}^p + n_{1,0}^p + n_{1,1}^p + 1\right) + t = 2t. \end{aligned}$$

By contradiction, u_p is fault-free if $\sigma(u_{r,3}^p, u_{r,2}^p) = 1$. □

In short, the **LDUCF** procedure makes a correct diagnosis if $\mathbb{B}_G(u;t)$ has at most $2t - 1$ faulty nodes for $t \ge 4$.

5 Examples and Simulation Results

In this section, we show how to construct branch-of-tree architectures in some well-known interconnected systems and give examples to explain the **LDUCF** procedure.

5.1 Construction of Branch-of-Trees

The proposed algorithm can be applied to diagnose many interconnection networks such as the hypercube, the star graph, the torus/mesh, etc. Hypercube's popularity stems from its topological properties, so it is ideally suited to a variety of parallel and distributed computation tasks. For this reason, we first show how to construct a branch-of-tree architecture in the hypercube.

First of all, we use the Initial-Branch-of-Tree (**INIBOT**) procedure (Algorithm 4) to construct a branch-of-tree in Q_8. It is noticed the symbol $\oplus$ denotes the bitwise XOR operation. Next, we propose the Branch-of-Tree recursive algorithm (Algorithm 5), which is able to construct a branch-of-tree rooted at any node in Q_n for $n \ge 8$. See Fig. 5 for illustration. Because Q_n is n-regular, the time complexity of **BOT** algorithm is $O(n^2)$.

Algorithm 4: INIBOT(u)

Input: A node u in 8-cube Q_8.
Output: $\mathbb{B}_{Q_8}(u;8) = (V, E)$, a branch-of-tree in Q_8.
begin
  for $i = 1$ *to* 8 **do**
    $u_i \leftarrow (u)^i$;
    for $j = 1$ *to* 7 **do**
      for $k = 1$ *to* 3 **do**
        Set $x_{j,k}^i$ as that in TABLE II;
        $u_{j,k}^i \leftarrow x_{j,k}^i \oplus u$;
  $V \leftarrow \{u\} \cup \{u_i \mid 1 \le i \le 8\} \cup \{u_{j,k}^i \mid 1 \le i \le 8, 1 \le j \le 7, \text{ and } 1 \le k \le 3\}$;
  $E \leftarrow \bigcup_{i=1}^{8} \left(\{u, u_i\} \cup \left(\bigcup_{j=1}^{7} \{\{u_i, u_{j,1}^i\}, \{u_{j,1}^i, u_{j,2}^i\}, \{u_{j,2}^i, u_{j,3}^i\}\}\right)\right)$;
  return (V, E);
end

Algorithm 5: BOT(n, Q_n, u)

Input: A positive integer $n \ge 8$, the n-cube Q_n, and a node u in Q_n.
Output: A branch-of-tree, $\mathbb{B}_{Q_n}(u;n) = (V, E)$, in Q_n.
begin
  if $n = 8$ **then**
    return INIBOT(u);
  else
    $p \leftarrow [u]_n$;
    $\mathbb{B}_{Q_n^{(p)}}(u; n-1) \leftarrow$ **BOT**$(n-1, Q_n^{(p)}, u)$;
    for $i = 1$ *to* $n - 1$ **do**
      if $i \le n - 4$ **then**
        $\langle u_{n-1,1}^i, u_{n-1,2}^i, u_{n-1,3}^i \rangle \leftarrow$ **TRI**$(u_i, n, i+2, i+3)$;
      else
        $\langle u_{n-1,1}^i, u_{n-1,2}^i, u_{n-1,3}^i \rangle \leftarrow$ **TRI**$(u_i, n, i+4-n, i+5-n)$;
    $u_n \leftarrow (u)^n$;
    for $j = 1$ *to* $n - 1$ **do**
      $\langle u_{j,1}^n, u_{j,2}^n, u_{j,3}^n \rangle \leftarrow$ **NEXT**(n, u_n, j);
    $V \leftarrow \{u\} \cup \{u_i, u_{j,k}^i \mid 1 \le i \le n, 1 \le j \le n-1, \text{ and } 1 \le k \le 3\}$;
    $E \leftarrow \bigcup_{i=1}^{n} \left(\{u, u_i\} \cup \left(\bigcup_{j=1}^{n-1} \{\{u_i, u_{j,1}^i\}, \{u_{j,1}^i, u_{j,2}^i\}, \{u_{j,2}^i, u_{j,3}^i\}\}\right)\right)$;
    return (V, E);
end

Algorithm 6: TRI(x, i, j, k)

Input: A node x in Q_n, and three positive integers $i, j, k \le n$.
Output: A sequence of three nodes in Q_n.
begin
  return $\langle (x)^i, ((x)^i)^j, (((x)^i)^j)^k \rangle$;
end

Algorithm 7: NEXT(n, y, j)

Input: A positive integer n, a node y of Q_n, and a positive integer j less than n.
Output: A sequence of three nodes in Q_n.
begin
  if $j \le n - 3$ **then**
    return TRI$(y, j, j+1, j+2)$;
  else if $j = n - 2$ **then**
    return TRI$(y, j, j+1, 1)$;
  else if $j = n - 1$ **then**
    return TRI$(y, j, 1, 2)$;
end

Theorem 8. *Let u be any node of Q_n for $n \ge 8$. The* **BOT** *algorithm (Algorithm 5) builds a branch-of-tree of order n rooted at u.*

Proof. The proof proceeds by induction on n. Because Q_n is node-transitive [32], we assume that $u = 0^n$. For $n = 8$, nodes u, $(u)^i$ with $1 \le i \le 8$, and those listed in Table 2 form a branch-of-tree of order 8. For $n \ge 9$, the inductive hypothesis is that the **BOT** algorithm builds a branch-of-tree $\mathbb{B}_{Q_n^{(0)}}(u; n-1)$ in $Q_n^{(0)}$. Then, $\mathbb{B}_{Q_n^{(0)}}(u; n-1)$ is augmented to complete the construction. Fig. 5 illustrates the whole branch-of-tree of order n rooted at u. □

The star graph [1] is an attractive alternative to the hypercube. Table 3 shows a branch-of-tree in the 5-dimensional star graph, rooted at node $u = 12{,}345$. We adopt it as an induction base to construct branch-of-trees recursively in the star graph.

Let n be a positive integer. The *n-dimensional star graph*, denoted by S_n, is a graph whose node set consists of all

TABLE 2
The Nodes $x, x_i, x^i_{j,k}$ in Q_8 for $1 \le i \le 8$, $1 \le j \le 7$, and $1 \le k \le 3$

$x = 00000000$	x_i							
	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$	$i=8$
	00000001	00000010	00000100	00001000	00010000	00100000	01000000	10000000
	$x^i_{j,k}$							
	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$	$i=8$
$j=1$ & $k=1$	00000011	00000011	00000101	00001001	00010001	00100001	01000001	10000001
$j=1$ & $k=2$	00000111	00001011	00010101	00101001	01010001	00100011	01000011	10000101
$j=1$ & $k=3$	00001111	00011011	00110101	01101001	01010011	01100011	01000111	10001101
$j=2$ & $k=1$	00000101	00000110	00000110	00001010	00010010	00100010	01000010	10000010
$j=2$ & $k=2$	00001101	00001110	00010110	00101010	00010011	10100010	01000110	10010010
$j=2$ & $k=3$	00011101	00011110	00110110	01101010	00110011	10100110	11000110	10010011
$j=3$ & $k=1$	00001001	00001010	00001100	00001100	00010100	00100100	01000100	10000100
$j=3$ & $k=2$	00011001	00011010	00011100	00101100	01010100	00100110	01001100	10001100
$j=3$ & $k=3$	00111001	00111010	00111100	00101110	01010101	01100110	01001101	10011100
$j=4$ & $k=1$	00010001	00010010	00010100	00011000	00011000	00101000	01001000	10001000
$j=4$ & $k=2$	00110001	00110010	00110100	00111000	01011000	10101000	01001010	10011000
$j=4$ & $k=3$	01110001	10110010	01110100	01111000	01011100	10101001	01011010	10011010
$j=5$ & $k=1$	00100001	00100010	00100100	00101000	00110000	00110000	01010000	10010000
$j=5$ & $k=2$	01100001	01100010	00100101	01101000	01110000	10110000	01010010	10010100
$j=5$ & $k=3$	11100001	11100010	01100101	11101000	01110010	10110100	11010010	10010101
$j=6$ & $k=1$	01000001	01000010	01000100	01001000	01010000	01100000	01100000	10100000
$j=6$ & $k=2$	01000101	11000010	11000100	01001001	11010000	11100000	01100100	10100100
$j=6$ & $k=3$	11000101	11000011	11001100	01011001	11110000	11100100	01101100	10101100
$j=7$ & $k=1$	10000001	10000010	10000100	10001000	10010000	10100000	11000000	11000000
$j=7$ & $k=2$	10001001	10000011	10000110	10001010	10010001	10100001	11001000	11000001
$j=7$ & $k=3$	10011001	10000111	10001110	10101010	11010001	10100011	11001010	11001001

permutations of $\{1, 2, \ldots, n\}$. Each node is uniquely assigned a permutation $x_1x_2 \ldots x_n$ and is adjacent to $(n-1)$ nodes $x_ix_2 \ldots x_{i-1}x_1x_{i+1} \ldots x_n$ for $2 \le i \le n$, which are obtained by a transposition of the first digit with the ith one. Consequently, there are $n!$ nodes in an n-dimensional star graph, and each node has degree $n-1$.

For any node $u \in V(S_n)$, its *i-neighbor*, denoted by $(u)^i$, is just the node obtained by a transposition of the first digit with the ith one. For convenience of description, we say that nodes u and $(u)^i$ are adjacent to each other with an (i)-edge. For any $1 \le i \le n$, let V_i denote a subset of permutations of $\{1, 2, \ldots, n\}$, whose elements have symbol i in the nth digit. Clearly, we have $V(S_n) = \bigcup_{i=1}^{n} V_i$. Moreover, it was shown in [1] that the subgraph of S_n induced by V_i is isomorphic to an $(n-1)$-dimensional star graph S_{n-1}. We denote this subgraph by $S_n^{\{i\}}$. By this recursive structure of star graphs, it is easy to derive the following theorem.

TABLE 3
The Nodes u, u_i, and $u^i_{j,k}$ in the 5-Dimensional Star Graph for $1 \le i \le 4$, $1 \le j \le 3$, and $1 \le k \le 3$

	$u = 12345$			
	u_i			
	$i=1$	$i=2$	$i=3$	$i=4$
	21345	32145	42315	52341
	$u^i_{j,k}$			
	$i=1$	$i=2$	$i=3$	$i=4$
$j=1$ & $k=1$	31245	23145	24315	25341
$j=1$ & $k=2$	41235	13245	34215	35241
$j=1$ & $k=3$	51234	43215	14235	45231
$j=2$ & $k=1$	41325	42135	32415	32541
$j=2$ & $k=2$	51324	52134	12435	42531
$j=2$ & $k=3$	15324	25134	52431	12534
$j=3$ & $k=1$	51342	52143	52314	42351
$j=3$ & $k=2$	15342	25143	25314	12354
$j=3$ & $k=3$	35142	15243	35214	21354

Theorem 9. *Let $u = x_1x_2 \cdots x_n$ be any node of S_n for $n \ge 5$. Then, there exists a branch-of-tree of order $n-1$ rooted at u in S_n. See Fig. 6a for illustration.*

Mesh is another popular network, whose topology is much more closely knitted in the sense that neighbors of a unit tend to share more neighbors. A branch-of-tree can be easily embedded in a 2D mesh. See Fig. 6b for illustration.

5.2 Examples of LDUCF

We take mesh as example to explain the **LDUCF** algorithm. Fig. 7 illustrates a branch-of-tree, with two test assignments and resulting syndromes, in the 2D mesh. In Fig. 7a, we assume that

$$F_1 = \{u_1, u_2, u^1_{1,1} = u^2_{3,1}, u^1_{2,1}, u^1_{3,2}, u^2_{1,2}, u^2_{2,1}\},$$

is a set of seven faulty nodes. The goal is to identify the fault status of node u. After the beginning for-loop of **LDUCF**, the set $\mathcal{D}$ turns out to be $\{u_1, u_2, u_3, u_4\}$. Since $|\mathcal{D}| \ge 3$, the procedure runs to enter the subroutine $\mathbf{VOTE}(u, \mathcal{D})$, in which $n_0 = n_1 = 2$ is computed. Finally, the output 0 is returned. That is, node u is fault-free. In Fig. 7b, we assign that

$$F_2 = \{u_2, u_3, u_4, u^1_{1,1}, u^1_{1,2}, u^1_{2,1}, u^1_{3,1} = u^4_{1,1}\},$$

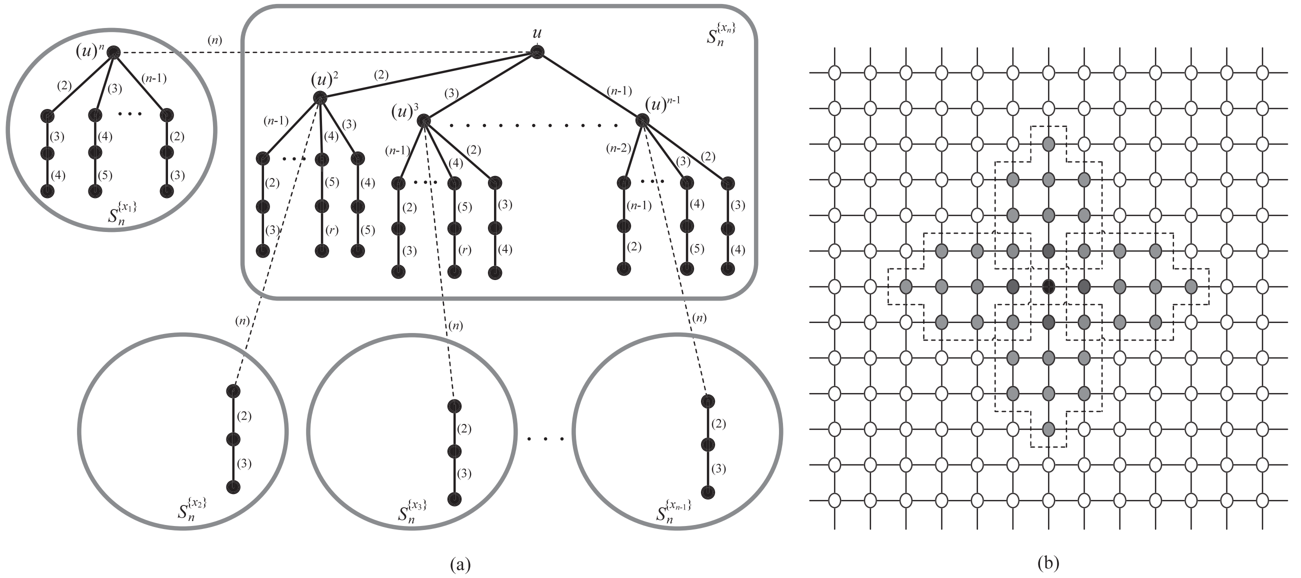


Fig. 6. (a) The branch-of-tree rooted at any node $u = x_1x_2 \cdots x_n$ in S_n, in which $r = 2$ if $n = 6$, and $r = 6$ if $n \geq 7$. (b) A branch-of-tree in the 2D mesh.

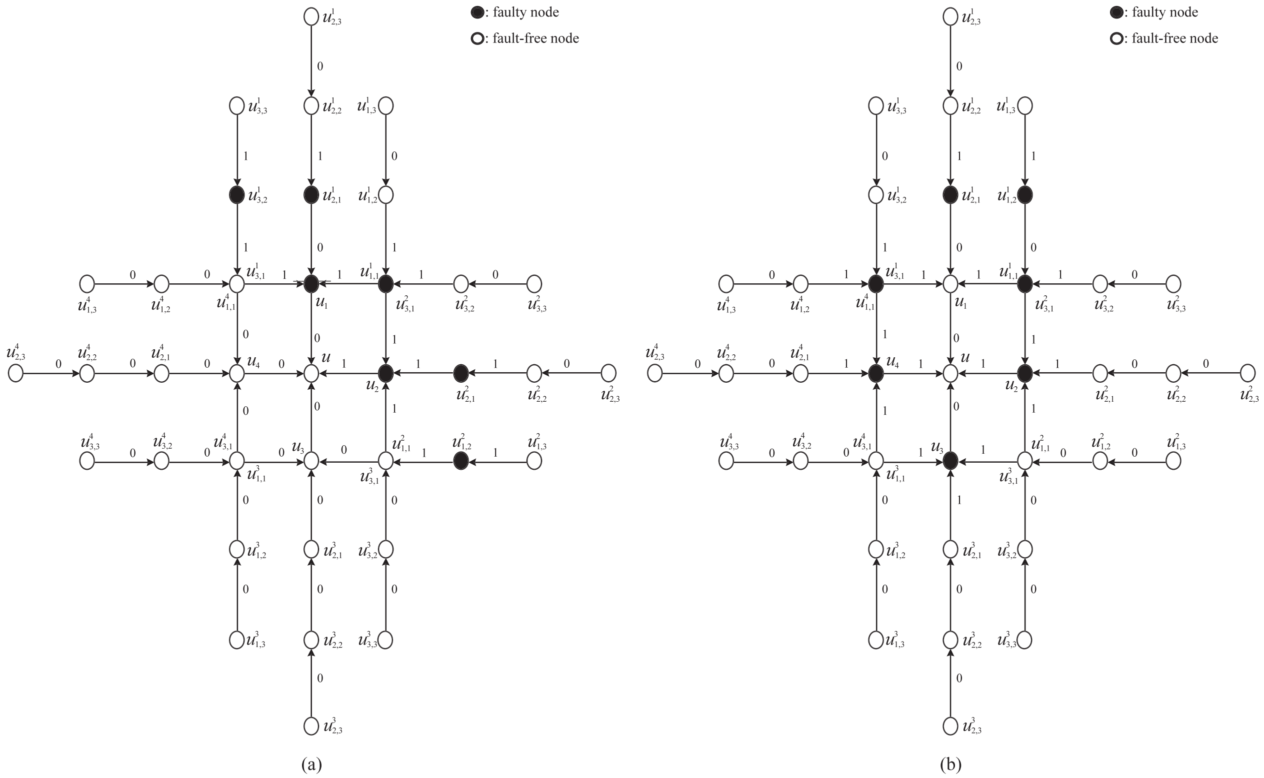


Fig. 7. Two test assignments and resulting syndromes of a branch-of-tree in the 2D mesh.

is a set of seven faulty nodes. In the beginning for-loop of **LDUCF**, $\mathcal{D} = \emptyset$ is determined. Therefore, the procedure runs to enter the else-block and have $n^1_{1,0} - n^1_{0,0} = 1$, $n^2_{1,0} - n^2_{0,0} = 2$, $n^3_{1,0} - n^3_{0,0} = 3$, and $n^4_{1,0} - n^4_{0,0} = 2$. Furthermore, since $p = 1$, $r = 1$ and $\sigma(u^1_{1,3}, u^1_{1,2}) = 1$, the test outcome $\sigma(u_1, u) = 0$ is returned. That is, node u is fault-free.

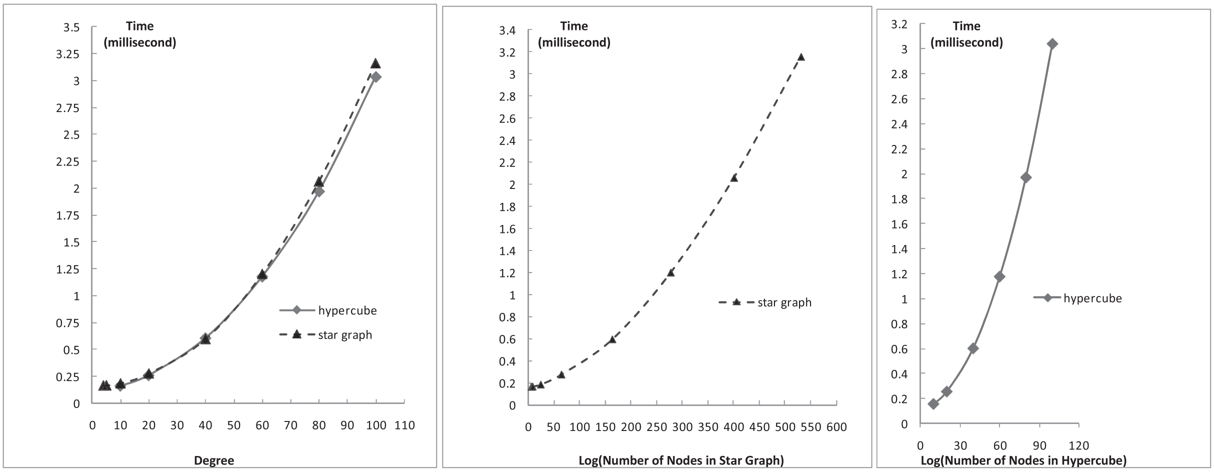
As shown above, a branch-of-tree rooted at $x = 00000000$ in Q_8 is listed in Table 2. In the second example, we assign

$$F_3 = \{x, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x^2_{1,1}, x^2_{2,1}, x^2_{3,1}, x^2_{4,1}, x^2_{5,1}, x^2_{6,1}, x^2_{7,1}\},$$

to be the set of 15 faulty nodes. For the sake of simplicity, we assume that faulty testers always report incorrect test outputs. Then, **LDUCF** procedure can determine that $\mathcal{D} = \{x_1, x_2\}$ with $n^1_{0,0} - n^1_{1,0} = 7$ and $n^2_{0,0} - n^2_{1,0} = 0$ after its beginning for-loop. Because of $n^1_{0,0} - n^1_{1,0} > n^2_{0,0} - n^2_{1,0}$, the final output $\sigma(x_1, x) = 1$ is returned; that is, node x is faulty.

5.3 Simulation

The numerical simulation is provided to confirm the practical time consuming of **LDUCF** algorithm. With respect to the hypercube and the star graph of various sizes, we carry out a round of simulation by randomly



(a) (b) (c)

Fig. 8. The time consuming of **LDUCF** algorithm over the hypercube and the star graph of various sizes.

assigning a conditionally faulty set of $2t-1$ nodes in the branch-of-tree of order t rooted at any node v for 10,000 times and compute the average time for identifying the fault status of v. Then, such a round of simulation has to be repeated 30 times to obtain the overall average. The hardware and software configuration include:

1. Intel Core 2 Quad CPU Q8300 2.5GHz,
2. 4 GB DRAM,
3. 64-bit Windows 7 OS, and
4. C++ Programming Language in Microsoft Visual Studio 2005.

The simulation results are shown in Fig. 8. One can see the elapsing time is proportional to the square of node degree, and the star graph is much more sparsely connected than the hypercube.

6 Conclusions

The multiprocessor system is a typical representative of massive parallel and distributed computing and has a variety of applications. To better reflect the impact of fault patterns on system-level diagnosis, many researchers have taken conditional diagnosability into account. However, those previous works are only of a theoretical nature. Instead, we relax the addressed fault condition to require that every fault-free unit has at least one fault-free neighbor. Under this new condition, not only can the diagnostic capability be proved theoretically, but also it is achieved in an algorithmic point of view. We establish some sufficient conditions so that a k-regular interconnected system is conditionally $(2k-1)$-diagnosable. Moreover, we design an $O(k^2)$ fault identification method, provided that there exists a branch-of-tree architecture rooted at each unit and the time for any unit to test another one is a constant. Our future research is devoted to connecting the practice and theoretical foundations of conditional-fault diagnosis for various diagnostic models.

Acknowledgments

The corresponding author, Prof. Tzu-Liang Kung, would like to thank the colleagues who provided valuable comments to an earlier draft and helped in revising this paper. In particular, the gratitude goes to Prof. Timothy Williams, Asia University, for his kindly help with language editing. The authors are also thankful to the anonymous referees and the editors for their thorough review. Their comments and suggestions help us to improve the present manuscript significantly.

References

[1] S.B. Akers and B. Krishnamurthy, "A Group-Theoretic Model for Symmetric Interconnection Networks," *IEEE Trans. Computers,* vol. 38, no. 4, pp. 555-566, Apr. 1989.
[2] A. Caruso, S. Chessa, P. Maestrini, and P. Santi, "Evaluation of a Diagnosis Algorithm for Regular Structures," *IEEE Trans. Computers,* vol. 51, no. 7, pp. 850-865, July 2002.
[3] A. Caruso, S. Chessa, P. Maestrini, and P. Santi, "Fault-Diagnosis of Grid Structures," *Theoretical Computer Science,* vol. 290, no. 2, pp. 1149-1174, Jan. 2003.
[4] A. Caruso, L. Albini, and P. Maestrini, "A New Diagnosis Algorithm for Regular Interconnected Structures," pp. 264-281, 2003.
[5] N.-W. Chang and S.-Y. Hsieh, "Conditional Diagnosability of Augmented Cubes under the PMC Model," *IEEE Trans. Dependable and Secure Computing,* vol. 9, no. 1, pp. 46-60, http://doi.ieeecomputersociety.org/10.1109/TDSC.2010.59, Jan./Feb. 2012.
[6] G. Chartrand and O.R. Ollermann, *Applied and Algorithmic Graph Theory.* McGraw-Hill, 1993.
[7] C.-A. Chen and S.-Y. Hsieh, "(t,k)-Diagnosis for Component Composition Graphs under the MM^* Model," *IEEE Trans. Computers,* vol. 60, no. 12, pp. 1704-1717, http://doi.ieeecomputersociety.org/10.1109/TC.2010.2011, Dec. 2011.
[8] S. Chessa and P. Maestrini, "Correct and Almost Complete Diagnosis of Processor Grids," *IEEE Trans. Computers,* vol. 50, no. 10, pp. 1095-1102, Oct. 2001.
[9] P. Cull and S.M. Larson, "The Möbius Cubes," *IEEE Trans. Computers,* vol. 44, no. 5, pp. 647-659, May 1995.
[10] A. Dahbura and G. Masson, "An $O(N^{2.5})$ Fault Identification Algorithm for Diagnosable Systems," *IEEE Trans. Computers,* vol. C-33, no. 6, pp. 486-492, June 1984.
[11] A. Das, K. Thulasiraman, V.K. Agarwal, and K.B. Lakshmanan, "Multiprocessor Fault Diagnosis under Local Constraints," *IEEE Trans. Computers,* vol. 42, no. 8, pp. 984-988, Aug. 1993.
[12] K. Efe, "A Variation on the Hypercube with Lower Diameter," *IEEE Trans. Computers,* vol. 40, no. 11, pp. 1312-1316, Nov. 1991.
[13] A.D. Friedman and L. Simoncini, "System-Level Fault Diagnosis," *The Computer J.,* vol. 13, no. 3, pp. 47-53, Mar. 1980.
[14] S.L. Hakimi and A.T. Amin, "Characterization of Connection Assignment of Diagnosable Systems," *IEEE Trans. Computers,* vol. C-23, no. 1, pp. 86-88, Jan. 1974.

[15] S.-Y. Hsieh and Y.-S. Chen, “Strongly Diagnosable Product Networks under the Comparison Diagnosis Model,” *IEEE Trans. Computers,* vol. 57, no. 6, pp. 721-732, June 2008.
[16] S.-Y. Hsieh and Y.-S. Chen, “Strongly Diagnosable Systems under the Comparison Diagnosis Model,” *IEEE Trans. Computers,* vol. 57, no. 12, pp. 1720-1725, Dec. 2008.
[17] S.-Y. Hsieh and T.-Y. Chuang, “The Strong Diagnosability of Regular Networks and Product Networks under the PMC Model,” *IEEE Trans. Parallel and Distributed Systems,* vol. 20, no. 3, pp. 367-378, Mar. 2009.
[18] S.-Y. Hsieh and C.-W. Lee, “Diagnosability of Two-Matching Composition Networks under the MM^* Model,” *IEEE Trans. Dependable and Secure Computing,* vol. 8, no. 2, pp. 246-255, Mar./Apr. 2011.
[19] G.-H. Hsu and J.J.M. Tan, “A Local Diagnosability Measure for Multiprocessor Systems,” *IEEE Trans. Parallel and Distributed Systems,* vol. 18, no. 5, pp. 598-607, May 2007.
[20] G.-H. Hsu, C.-F. Chiang, L.-M. Shih, L.-H. Hsu, and J.J.M. Tan, “Conditional Diagnosability of Hypercubes under the Comparison Diagnosis Model,” *J. Systems Architecture,* vol. 55, no. 2, pp. 140-146, Feb. 2009.
[21] L.-H. Hsu and C.-K. Lin, *Graph Theory and Interconnection Networks.* CRC Press, 2008.
[22] T.-L. Kung, H.-C. Chen, and J.J.M. Tan, “On the Faulty Sensor Identification Algorithm of Wireless Sensor Networks under the PMC Diagnosis Model,” *Proc. Sixth Int'l Conf. Networked Computing and Advanced Information Management (NCM '10),* pp. 657-661, Aug. 2010.
[23] P.-L. Lai, J.J.M. Tan, C.-P. Chang, and L.-H. Hsu, “Conditional Diagnosability Measures for Large Multiprocessor Systems,” *IEEE Trans. Computers,* vol. 54, no. 2, pp. 165-175, Feb. 2005.
[24] C.-W. Lee and S.-Y. Hsieh, “Determining the Diagnosability of (1,2)-Matching Composition Networks and Its Applications,” *IEEE Trans. Dependable and Secure Computing,* vol. 8, no. 3, pp. 353-362, May/June 2011.
[25] F.T. Leighton, *Introduction to Parallel Algorithms and Architectures: Arrays · Trees · Hypercubes.* Morgan Kaufmann, 1992.
[26] C.-K. Lin, S.-L. Peng, J.J.M. Tan, T.-L. Kung, and L.-H. Hsu, “The Diagnosability of g-Good-Neighbor Conditional-Faulty Hypercube Under PMC Model,” *Proc. Int'l Conf. Parallel and Distributed Processing Techniques and Applications (PDPTA '10),* vol. 2, pp. 494-499, July 2010.
[27] M. Mánik and E. Gramatová, “Boolean Formalisation of the PMC Model for Faulty Units Diagnosis in Regular Multi-Processor Systems,” *Proc. IEEE 11th Workshop Design and Diagnostics of Electronic Circuits and Systems (DDECS),* pp. 1-2, Apr. 2008.
[28] M. Mánik and E. Gramatová, “Diagnosis of Faulty Units in Regular Graphs under the PMC Model,” *Proc. 12th Int'l Symp. Design and Diagnostics of Electronic Circuits and Systems (DDECS),* pp. 202-205, Apr. 2009.
[29] H. Masuyama and T. Miyoshi, “A Non-Adaptive Distributed System-Level Diagnosis Method for Computer Networks,” *Proc. Int'l Conf. Comm. Systems and Networks,* 2003.
[30] W. Najjar and J.L. Gaudiot, “Network Resilience: A Measure of Network Fault Tolerance,” *IEEE Trans. Computers,* vol. 39, no. 2, pp. 174-181, Feb. 1990.
[31] F.P. Preparata, G. Metze, and R.T. Chien, “On the Connection Assignment Problem of Diagnosis Systems,” *IEEE Trans. Electronic Computers,* vol. EC-16, no. 12, pp. 848-854, Dec. 1967.
[32] Y. Saad and M.H. Shultz, “Topological Properties of Hypercubes,” *IEEE Trans. Computers,* vol. 37, no. 7, pp. 867-872, July 1988.
[33] A.K. Somani, V.K. Agarwal, and D. Avis, “A Generalized Theory for System Level Diagnosis,” *IEEE Trans. Computers,* vol. C-36, no. 5, pp. 538-546, May 1987.
[34] A.K. Somani and V.K. Agarwal, “Distributed Diagnosis Algorithm for Regular Interconnected Systems,” *IEEE Trans. Parallel and Distributed Systems,* vol. 42, no. 7, pp. 899-906, July 1992.
[35] A.K. Somani and O. Peleg, “On Diagnosability of Large Fault Sets in Regular Topology-Based Computer Systems,” *IEEE Trans. Computers,* vol. 45, no. 8, pp. 892-903, Aug. 1996.
[36] M. Xu, K. Thulasiraman, and X.-D. Xu, “Conditional Diagnosability of Matching Composition Networks under the PMC Model,” *IEEE Trans. Circuits and Systems-II: Express Briefs,* vol. 56, no. 11, pp. 875-879, Nov. 2009.
[37] Q. Zhu, “On Conditional Diagnosability and Reliability of the BC Networks,” *The J. Supercomputing,* vol. 45, no. 2, pp. 173-184, Aug. 2008.

Cheng-Kuan Lin received the BS degree in applied mathematics from Chinese Culture University, Taiwan, Republic of China, in 2000, the MS degree in mathematics from the National Central University in 2002, and the PhD degree in computer science from the National Chiao Tung University in 2011. He is now a postdoctoral researcher in the Department of Computer Science, National Chiao Tung University. His research interests include interconnection networks, algorithms, and graph theory.

Tzu-Liang Kung received the BS degree in industrial administration from the National Taiwan University in 1997, the MS degree in statistics from the National Chiao Tung University, Taiwan, in 2001, and the PhD degree in computer science from the National Chiao Tung University in 2009. From 2001 to 2004, he served as a senior engineer at the Behavior Design Corporation, Taiwan. He is currently an assistant professor in the Department of Computer Science and Information Engineering, Asia University. His research interests include multivariate data analysis, machine translation, natural language processing, interconnected systems, fault-tolerant computing, algorithm design, and wireless networks.

Jimmy J.M. Tan received the BS and MS degrees in mathematics from the National Taiwan University in 1970 and 1973, respectively, and the PhD degree from Carleton University, Ottawa, in 1981. He has been on the faculty of the Department of Computer Science, National Chiao Tung University, since 1983. His research interests include design and analysis of algorithms, combinatorial optimization, and interconnection networks.