Cycles identifying vertices for fault isolation in locally twisted cubes

Jheng-Cheng Chen, Chia-Jui Lai, and Chang-Hsiung Tsai

Department of Computer Science and Information Engineering
National Dong Hwa University, Shoufeng, Hualien, Taiwan 97401, R.O.C.

Abstract

A set of subgraphs $C_1, C_2, \ldots, C_k$ in a graph $G$ is said to identify the vertices if the sets \{ $j \mid v \in C_j$ \} are nonempty for all the vertices $v$ and no two are the same set. We consider the problem of minimizing $k$ when the subgraphs $C_i$ are required to be cycles. The motivation comes from fault diagnosis of multiprocessor systems. We study the cases when $G$ is the locally twisted cube.

1 Introduction

Rapid advances in semiconductor technology have made possible the design of systems with a large number of components. It is difficult to build such systems without defects. Thus fault diagnosis have become issues of great interest in design and analysis of computer. It was in this context that Preparata, Metze and Chien [17] proposed a model, now well known as the PMC model, and a framework for the diagnosis of large scale digital systems. The process of identifying faulty processors is called diagnosis of the system. In this model, every processor performs tests on its neighbors based on the communication links between them. When one processor tests another, the tester declares the tested processor to be fault-free or faulty depending on the test response; the result is always accurate if the tester is fault-free, but if the tester is faulty, the result is unreliable. The collection of all test results is called a syndrome. The problem is to identify all faulty processors subject to certain assumptions. Numerous studies have been dedicated to the PMC model [3, 4, 7, 12, 13, 16, 19, 24].

Karpovsky et al. [8] introduced the concept of identifying codes and studies several issues relating to the construction of these codes. Consider an undirected graph $G$ with vertex set $V$ and edge set $E$. A ball of radius $t \geq 1$ centered at a vertex $v$ is defined as the set of all vertices that are at distance $t$ or less from $v$. The vertex $v$ is said to cover itself and all the vertices in the ball with $v$ as the center. The identifying codes problem defined by Karprovsky et al. [8] is to find a minimum set $D$ such that every vertex in $G$ belongs to a unique set of balls of radius $t \geq 1$ centered at the vertices in $D$. The set $D$ may be viewed as a code identifying the vertices and is called an identifying code. Assume that we want to maintain a system $G$, and consider the situation in which at lost one of the processors is not working. A lot of work has been done, when the identification of malfunctioning processors is made using balls with $v$ as the center, i.e., all the processors that are within distance $r$ and reports YES/NO depending on whether it has detected a problem or not. Based on these YES/NO answers, we want to be able to tell the exact location of the malfunctioning processor or that all the processors are fine under the assumption that there is at most one malfunctioning processor. Further results for typical multiprocessor architectures have been presented, for example, in [1, 2, 5, 9, 10, 18] for hypercubes.

In [6], the idea of using cycles instead of balls is mentioned. The mathematical problem can then be formulated as follows. We send test messages and can route them through this network in any way we like. What is smallest number of messages we have to send if based on which messages safely come back we can tell which vertex is faulty? We call these the vertex identification problem. The locally twisted cube is a well-known variant of the...
classical hypercube proposed by Yang et al. [22] and has been attracting much research interest in literatures since its proposal [14, 15, 21, 20, 22]. In this paper, we study the vertex identification problem on the locally twisted cube.

The rest of this paper is organized as follows: In Section 2, a formal description of the locally twisted cube is given and some useful notations are defined, including notation for the permutation of link dimension and the reflected link label sequence. Section 3 presents the sufficient and necessary conditions for determining cycles in LTQ_n based on the reflected link label sequence approach. Using these approaches, main results are shown in Section 4. Finally, we give some conclusions.

2 Preliminaries

A topology of an interconnection network is conveniently represented by an undirected simple graph G = (V, E), where V(G) and E(G) is the vertex set and the edge set of G, respectively. For graph terminology and notation not defined here we refer the reader to [11]. A walk in a graph is a finite sequence ω : λ_0, e_1, λ_1, e_2, λ_2, . . . , λ_k, e_k, λ_k whose terms are alternately vertices and edges so, for 1 ≤ i ≤ k, the edge e_i has ends λ_{i-1} and λ_i, thus each edge e_i is immediately preceded and succeeded by the two vertices with which it is incident. In particular, a walk ω is called a path if all internal vertices, λ_i, for 1 ≤ i ≤ k - 1, of the walk ω are distinct. Both of two vertices λ_0 and λ_k are called end-vertices of the path ω. For simplicity, the path ω is also denoted by λ_0, λ_1, . . . , λ_k. If λ_0 = λ_k, then ω is called a cycle. A cycle of length l is called a l-cycle. A path (respectively, cycle) traversing each vertex of G exactly once is the Hamiltonian path (respectively, Hamiltonian cycle).

Let \{0, 1\}^n denote the set of all binary strings of length n. For two binary strings x and y \in \{0, 1\}^n, let x + y denote the (bitwise modulo 2) sum of x and y. For every integer 0 ≤ i ≤ n - 1, let b_i denote the binary string x_{n-1}x_{n-2} . . . x_0 with x_i = 1 and x_j = 0 for all j ≠ i. For every integer 2 ≤ i ≤ n - 1, let B_i denote the binary string x_{n-1}x_{n-2} . . . x_0 with x_i = 1 and x_j = 0 for all j ≠ i, i - 1. In addition, let B_1 = b_1 and B_0 = b_0. As a result, B_i = b_i for i ≤ 1 and B_i = b_{i-1} + b_{i-1} for i ≥ 2, moreover, b_1 + b_i = B_i + B_1 = 0^n where 0^n denote a string consisting of n 0s.

**Definition 1** [22] For n ≥ 2, an n-dimensional locally twisted cube, denoted by LTQ_n, is defined recursively as follow:

1. LTQ_2 is a graph consisting of four nodes labeled with 00, 01, 10, and 11 respectively, connected by four edges (00,01),(00,10),(01,11) and (10,11).

2. For n ≥ 3, LTQ_n is built from two disjoint copies of LTQ_{n-1} according to the following steps. Let 0LTQ_{n-1} (respectively, 1LTQ_{n-1}) denote the graph obtained by prefixing the label of each node in one copy of LTQ_{n-1} with 0 (respectively, 1). Each node 0x_{n-2}x_{n-3}...x_0 in 0LTQ_{n-1} is connected to the node 1(x_{n-2} + x_0)x_{n-3}...x_0 in 1LTQ_{n-1} by an edge.

Figure 2 shows examples of locally twisted cubes, LTQ_3 and LTQ_4. Either x = y + b_k or x = y + B_k for some 0 ≤ k ≤ n - 1 if vertices x and y of LTQ_n are adjacent. Therefore, we call y as the k-neighbor of x and (x, y) is labeled by k; besides, (x, y) is called to be type b if x = y + b_k and type B if x = y + B_k.

A path in LTQ_n might be specified by the source vertex and a sequence of labels detailing the edges to be traversed, for example, the path in LTQ_3 detailed as having the source vertex 000 and then following the edges labeled 0-2-1 (also denoted as [0-2-1]) is actually the path 000, 001, 111, 101, also denoted as 000[0-2-1]101, where 001 = 000 + b_0, 111 = 001 + B_2, and 101 = 111+B_1. Therefore, the sequence L = [d_0−d_1−...−d_{m-1}] is called an Link Label Sequence in LTQ_n if two adjacent labels are not identical where d_i ∈ Z_n, Z_n = \{0, 1, . . , n - 1\}, for 0 ≤
A walk, \( \omega(L, u) = \lambda_0, \lambda_1, \lambda_2, \ldots, \lambda_m \), in \( LTQ_n \) can be generated with respect to a given link label sequence \( L \) and a given vertex \( u \) as follows: \( \lambda_0 = u \), and \( \lambda_j \) is the \( d_{j-1} \)-neighbor of \( \lambda_{j-1} \) in \( LTQ_n \), where \( 0 \leq j \leq m \). Thus, this walk \( \omega(L, u) \) is also represented as \( \lambda_0[L] \lambda_m \).

An interesting approach to generate such a path in \( n \)-dimensional cube is the reflected link label sequence (as defined below), which were proposed by Zheng and Latifi [23]. Based on a reflected link label sequence, Zheng and Latifi defined a generalized Gray Code (GGC) and utilized GGCs to identify Hamiltonian paths and cycles in an \( n \)-dimensional crossed cube. From now on, we will propose a systematic approach based on reflected link label sequence to construct Hamiltonian cycles in an \( n \)-dimensional locally twisted cube.

**Definition 2** [22] For \( n \geq 1 \), let \( \pi = (d_1, d_2, \ldots, d_n) \) be a permutation over on \( Z_n \), i.e., \( \pi \in \Pi_n \), and \( \pi(i) = (d_1, d_2, \ldots, d_n) \), for \( 1 \leq i \leq n \). Then the reflected link label sequence, denoted as \( G_\pi \), defined by \( \pi \) can be generated recursively as follows:

\[
\begin{align*}
G_{\pi(1)} &= d_1, \\
G_{\pi(k)} &= G_{\pi(k-1)} - d_k - G_{\pi(k-1)}, 2 \leq k \leq n; \\
G_\pi &= G_{\pi(n)}.
\end{align*}
\]

As a result, the \( G_{\pi(k)} \) is a symmetry link label sequence in \( LTQ_n \) and hence, \( \omega(G_{\pi(k)}, u) \) comes into being a walk in \( LTQ_n \) with a given starting vertex \( u \). Subsequently, the following lemma is easily obtained and thus details of the proof are omitted.

**Lemma 1** Let \( \pi = (d_1, d_2, \ldots, d_n) \in \Pi_n \). Then the total number of \( d_i \) appearing in \( G_{\pi(k)} \) is \( 2^{k-1} \), where \( 1 \leq i \leq k \).

In addition, the following result could be concluded according to Lemma 1.

**Lemma 2** Let \( \pi = (d_1, d_2, \ldots, d_n) \in \Pi_n \). Then, the total number of \( d_i \) appearing between any two successive \( d_j \)'s in \( G_{\pi(k)} \) is \( 2^{i-j} \) if \( 1 \leq i < j \leq k-1 \).

**Proof.** Let \( k \leq n \) and \( 0 \leq i < j \leq k - 1 \). Hence \( \pi(j+1) = (d_1, d_2, \ldots, d_j, d_{j+1}) \). Recall that \( G_{\pi(j+1)} = G_{\pi(j-1)} - d_j - G_{\pi(j-1)} - d_j + G_{\pi(j-1)} \). Since \( G_{\pi(j+1)} \) is a subsequence of \( G_{\pi(k)} \) if \( j \leq k - 1 \), the total number of \( d_i \) appearing between any two successive \( d_j \)'s in \( G_{\pi(k)} \) is equal to the total number of \( d_i \) appearing between two \( d_j \)'s in \( G_{\pi(j+1)} \). By Lemma 1, the total number of \( d_i \) appearing in \( G_{\pi(j-1)} \) is \( 2^{j-1} \). Therefore, there are \( 2 \cdot 2^{(j-1)-i} = 2^{j-i} \) \( d_i \)'s between any two successive \( d_j \)'s in \( G_{\pi(k)} \), for \( 0 \leq i < j \leq k - 1 \).

Before our further works, the following definition and lemmas are necessary. For any two vertices, \( u = u_{n-1}u_{n-2} \ldots u_1u_0 \) and \( v = v_{n-1}v_{n-2} \ldots v_1v_0 \) in \( LTQ_n \), define \( Diff(u, v) = \{ i \mid u_i \neq v_i \text{ for } 0 \leq i \leq n - 1 \} \). For instance, \( Diff(0011, 1010) = \{ 0, 1, 3 \} \). Therefore, we can immediately obtain

**Lemma 3** For any link label sequence \( L \), let \( u[L]w \) and \( v[L]z \) be two walks in \( LTQ_n \). Then, \( Diff(u, v) = Diff(w, z) \) if \( u_0 = v_0 \).

### 3 Reflected link label sequence and Cycle

In this section, we firstly show the sufficient and necessary conditions for determining Hamiltonian paths and cycles in \( LTQ_n \) based on the reflected link label sequence approach.

**Lemma 4** For \( u \in V(LTQ_n) \) and \( \pi \in \Pi_n \), the walk \( \omega(G_{\pi(k)}, u) \) is a path passing through \( 2^k \) vertices in \( LTQ_n \).

**Proof.** Let \( \pi = (d_1, d_2, \ldots, d_n) \in \Pi_n \). And hence \( \pi(k) = (d_1, d_2, \ldots, d_{k+1}) \). We will claim \( \omega(G_{\pi(k)}, u) \) is a path with \( 2^k \) vertices in \( LTQ_n \) for any starting vertex \( u \), i.e., any two distinct vertices lying on \( \omega(G_{\pi(k)}, u) \) are not identical. We argue by induction on \( k \). For \( k = 1 \), the permutation \( \pi(1) = (d_1) \) is considering. Clearly, for any vertex \( u \in V(LTQ_n) \), \( \omega(G_{\pi(1)}, u) \) is an edge in \( LTQ_n \). Assume that for \( k - 1 \leq m \leq n, 2 \leq m < n \), the lemma is true.

Let \( k = m \). By definition, we have \( G_{\pi(m)} = G_{\pi(m-1)} - d_m - G_{\pi(m-1)} \). Thus \( \omega(G_{\pi(m)}, u) \) can be rewrite as \( u[G_{\pi(m-1)}], x, [d_m][y][G_{\pi(m-1)}] \). One can observe that \( \omega(G_{\pi(m-1)}, x) = x[G_{\pi(m-1)}] u \) and \( \omega(G_{\pi(m-1)}, y) = y[G_{\pi(m-1)}] v \). By the induction hypothesis, it is obvious that \( \omega(G_{\pi(m-1)}, x) \) and \( \omega(G_{\pi(m-1)}, y) \) are paths with \( 2^{m-1} \) vertices respectively. Let \( w \) be any vertex lying on the path \( x[G_{\pi(m-1)]} u \) and \( z \) be any vertex lying on the path \( y[G_{\pi(m-1)]} v \). Now we claim \( w \neq z \).

The lemma is obvious if \( d_m = 0 \) and \( n - 1 \). Thus, we only consider \( d_m \neq 0, n - 1 \). Let \( w \) be an end-vertex of the path \( x[L_1]w \) and \( z \) be an end-vertex of the path \( y[L_2]z \), where \( L_1 \) and \( L_2 \) are two sub-sequences of \( G_{\pi(m-1)} \). By Lemma 3,
Diff \((x, y) = Diff(w, z) \) if \( L_1 = L_2 \). This implies \( w \neq z \) if \( L_1 = L_2 \). From now on, we consider the case of \( L_1 \neq L_2 \). Without loss of generality, assume the length of \( L_1 \) is smaller than that of \( L_2 \). Let \( y[L_2]z \) be decomposed as \( y[L_1]w[L_3]z \). By Lemma 3, \( Diff(x, y) = Diff(w', w) \) due to \( x_0 = y_0 \). Note, \( Diff(x, y) = Diff(w', z) \) if \( z = w \). This implies \( z \neq w \) if \( Diff(x, y) \neq Diff(w', z) \).

We now claim \( Diff(x, y) \neq Diff(w', z) \). Suppose that \( Diff(x, y) = Diff(w', z) \). One can observe that \( Diff(x, y) \neq Diff(w', z) \) if there are odd number of appearances of 0 in \( L_3 \). Thus, \( L_3 \) contains even numbers of 0. Since \( x \) is the \( d_m \)-neighbour of \( y, d_m \in Diff(x, y) \). Recall \( d_m \) is not appearing in \( \pi(m - 1) \). This implies only the element \( d_m + 1 \) in \( \pi(m - 1) \) can change the value of the \( d_m \)-th bit position of \( w' \) and \( z \). Let \( d_i = d_m + 1 \) and \( d_j = 0 \) for some \( 1 \leq i, j \leq m - 1 \). Let \( s \) denote the total number of \( d_i \) in the sequence \( L_3 \). There are two possibilities with respect to the relationship of \( i \) and \( j \).

**Case 1:** \( i < j \). By Lemma 2, there are even numbers of \( d_m + 1 \) between two successive 0’s in \( G_{\pi(m - 1)} \). Since \( L_3 \) contains even numbers of 0, all edges with label \( d_m + 1 \) appearing in the path \( w'[L_3]z \) are partitioned into two subsets that one contains even numbers of one type, \( b \)-type or \( B \)-type, edges and the other contains remaining edges belonging to the other type. Since \( b_{d_m} + b_{d_m} = 0^n \) and \( B_{d_m} + B_{d_m} = 0^n, d_m \notin Diff(w', z) \) if \( s \) is an even integer; otherwise, \{\( d_m + 1 \) \} \( \subseteq Diff(w', z) \) or \{\( d_m, d_m + 1 \) \} \( \subseteq Diff(w', z) \). Since \( Diff(x, y) = \{d_m\} \) or \( Diff(x, y) = \{d_m, d_m - 1\} \), \( Diff(x, y) \neq Diff(w', z) \). A contradiction occurs.

**Case 2:** \( i > j \). By Lemma 2, there are even numbers of 0 between two successive \( (d_m + 1) \)'s in \( G_{\pi(m - 1)} \). All edges with label \( d_m + 1 \) appearing in the path \( w'[L_3]z \) are in the same type. The argument is similar to that of Case 1. This completes our inductive proof.

**Theorem 1** For \( n \geq 2 \), let \( u \in V(LTQ_n) \) and \( \pi = \langle d_1, d_2, \ldots, d_n \rangle \in \Pi_n \). Then, the walk \( \omega(G_{\pi - d_n}, u) \), generated by the reflected link label sequence \( G_{\pi} \) and the starting vertex \( u \in LTQ_n \), corresponds to a Hamiltonian cycle of \( LTQ_n \) if and only if one of the following conditions holds:

1. \( d_{n-1} \cdot d_n \neq 0 \), or
2. \( |d_{n-1} - d_n| = 1 \).

**Proof.** \((\Rightarrow)\)

**Case 1:** \( d_{n-1} \cdot d_n \neq 0 \), i.e., \( d_n \neq 0 \) and \( d_{n-1} \neq 0 \). Since \( G_\pi = G_{\pi(n-1)} - d_n - G_{\pi(n-1)} \), let \( \omega(G_\pi, u) = u[G_{\pi(n-1)}][x[d_n][y[G_{\pi(n-1)}]]v, where \( \omega(G_{\pi(n-1)}, x) = x[G_{\pi(n-1)}][u] \) and \( \omega(G_{\pi(n-1)}, y) = y[G_{\pi(n-1)}][v]. \) Since \( d_n \neq 0 \), we have \( x_0 = y_0 \), where \((x = x_{n-1}x_{n-2} \ldots x_0) \) and \((y = y_{n-1}y_{n-2} \ldots y_0) \). Thus, by Lemma 3, \( Diff(x, y) = Diff(u, v) \).

Further, by Lemma 1, \( d_{n-1} \neq 0 \) implies the total number of appearances of 0 in \( G_{\pi(n-1)} \) is even, which implies \( x_0 = y_0 = u_0 = v_0 \) where \( u = u_{n-1}u_{n-2} \ldots u_1u_0 \) and \( v = v_{n-1}v_{n-2} \ldots v_1v_0 \). Therefore, \((u, v) \) is an edge with label \( d_n \). Hence the path \( \omega(G_{\pi - d_n}, u) \) corresponds to a Hamiltonian cycle of \( LTQ_n \).

**Case 2:** \( |d_{n-1} - d_n| = 1 \). In this case, we need only to verify cases that \( d_{n-1} = j \) and \( d_n = 1 - j \) for \( j = 0, 1 \). Suppose \( d_{n-1} = 1 \) and \( d_n = 0 \). Let \( \omega(G_{\pi}, u) = u[G_{\pi(n-1)}][x[d_n][y[G_{\pi(n-1)}]]v. \) Since \( d_n = 0 \), we have \( x_0 = y_0 \). Besides, \( u_1 = u_1 \) and \( v_1 = v_1 \) because of \( d_{n-1} = 1 \). By Lemma 1, the number of \( d_i \) appearing in the sequence \( G_{\pi(n-1)} \) is even for all \( 1 \leq i \leq n - 2 \) and \( d_{n-1} \) appears once in \( G_{\pi(n-1)} \). Hence, \( Diff(u, v) = \{0\} \), which implies \((u, v) \) is an edge with label 0. Thus, the path \( \omega(G_{\pi - d_n}, u) \) corresponds to a Hamiltonian cycle of \( LTQ_n \). If \( d_{n-1} = 0 \) and \( d_n = 1 \), the argument is similar.

\((\Rightarrow)\) We argue by refutation. If \( d_{n-1} \cdot d_n = 0 \) and \( |d_{n-1} - d_n| > 1 \), we have that \( d_{n-1} = k \) and \( d_n = 0 \), or \( d_{n-1} = 0 \) and \( d_n = k \) for some \( k \geq 2 \). Suppose \( d_{n-1} = k \) and \( d_n = 0 \) for some \( k \geq 2 \). Let \( \omega(G_{\pi}, u) = u[G_{\pi(n-1)}][x[d_n][y[G_{\pi(n-1)}]][v, where \( \omega(G_{\pi(n-1)}, x) = x[G_{\pi(n-1)}][u] \) and \( \omega(G_{\pi(n-1)}, y) = y[G_{\pi(n-1)}][v. \) We have \( Diff(x, y) = \{0\} \) due to \( d_n = 0 \). Note, \( \pi(n - 1) \) is a permutation over on \( a_n \). Then, there is no 0 in \( G_{\pi(n-1)} \). Hence, 0 \( \in Diff(u, v) \). By definition \( G_{\pi(n-1)} \), the total number of \( k \), appearing in \( G_{\pi(n-1)} \) is only one. Therefore \( k - 1 \in Diff(u, v) \). Thus, since \( k \geq 1 \), \( u \) is not a neighbour of \( v \). A contradiction occurs.

If \( d_{n-1} = 0 \) and \( d_n = k \) for some \( k \geq 2 \), the argument is similar. This completes the proof of the theorem.

Following two lemmas are useful and straightforward.

**Lemma 5** For \( n \geq 2 \), let \( u \in V(LTQ_n) \) and \( \pi = \langle d_1, d_2 \rangle \). Then, the walk \( \omega(G_{\pi - d_n}, u) \) is a 4-cycle if and only if one of the following conditions holds:

1. \( d_{n-1} \cdot d_n \neq 0 \), or
2. \( |d_{n-1} - d_n| = 1 \).

**Lemma 6** Let \( u \in V(LTQ_n) \) and \( \pi = \langle d_1, d_2, \ldots, d_n \rangle \) where \( 0 \leq d_i \leq n - 1 \). The walk
\[ \omega(G_v, u) \text{ is a cycle of length } 2^m \text{ vertices in } LTQ_n \]

if the following conditions hold:

(1) \( d_{m-1} \cdot d_m \neq 0, \) or

(2) \( |d_{m-1} - d_m| = 1. \)

4 Main Results

In order to solve vertex identification problem on the locally twisted cube \( LTQ_n \), we wish to find a collection of cycles \( C_1, C_2, \ldots, C_k \) such that every vertex belongs to at least one of them and, moreover, the sets \( \{j \mid v \in C_j\} \) are all pairwise different. We denote the minimum cardinality \( k \) for \( LTQ_n \) by \( ID_v(n) \). For arbitrary sets, we have following trivial identification theorem:

**Theorem 2** A collection \( A_1, A_2, \ldots, A_k \) of subsets of an \( s \)-element set \( S \) is called identifying, if for all \( x \in S \) the sets \( \{i \mid x \in A_i\} \) are nonempty and different. Given \( s \), the smallest identify collection of subsets consists of \( \lceil \log_2(s + 1) \rceil \) subsets.

The vertex identification problem is special cases of this problem, and for the locally twisted cube \( LTQ_n \) with \( s = 2^n \) vertices we get the lower bound

\[ ID_v(n) \geq \lceil \log_2(2^n + 1) \rceil = n + 1. \]

Let \( \pi = \langle d_1, d_2, \ldots, d_m \rangle \) where \( 0 \leq d_i \leq n - 1 \). For simplicity, we use \( \Omega(\pi, u) \) to denoted the walk \( \omega([G_\pi - d_m], u) \) if \( \omega([G_\pi - d_m], u) \) is a cycle in \( LTQ_n \). Let \( 0^n \) and \( 1^n \) denote all-zero and all-one \( n \)-bit binary strings, respectively.

For \( n \geq 4 \), let \( \pi^i = \langle 0, 1, 2, \ldots, i - 1, i + 1, i + 2, \ldots, n - 1 \rangle \) where \( i \geq 2 \). By Lemma 2, we have that the total number of 0 appearing between two successive \( i + 1 \) in \( G_{\pi^i} \), is \( 2^{i-1} \) and total number of 0 appearing before \( i + 1 \) is \( 2^{i-1} \) too. Since \( G_{\pi^i} \) contains none of \( i \), the path \( \omega(G_{\pi^i}, u) \) starting any vertex \( u \) consists of \( 2^{n-1} \) vertices whose \( i \)th coordinate equals \( u_i \). Therefore, we have following useful Lemmas.

**Lemma 7** For \( n \geq 4 \), let \( \pi^i = \langle 0, 1, 2, \ldots, i - 1, i + 1, i + 2, \ldots, n - 1 \rangle \). Then, the cycle \( \Omega(\pi^i, 0^n) \) passes all vertices \( v \) with \( v_i = 0 \) where \( v = v_{n-1} v_{n-2} \ldots v_0 \) for \( i \geq 2 \).

Similarly,

**Lemma 8** For \( n \geq 4 \), let \( \pi = \langle 0, n - 1, 2, 3, \ldots, n - 2 \rangle \). Then, the cycle \( \Omega(\pi, 0^n) \) passes all vertices \( v \) with \( v_1 = 0 \) where \( v = v_{n-1} v_{n-2} \ldots v_0 \).

**Theorem 3** \( ID_v(n) = n + 1 \) for all \( n \geq 3 \)

**Proof.** The proof is divided into two cases: (1) \( n = 3 \) and (2) \( n \geq 4 \).

**Case 1.** \( n = 3 \). Let

\[
\begin{align*}
C_0 &= 000, 100, 110, 010, 000 \\
C_1 &= 000, 100, 011, 001, 000 \\
C_2 &= 000, 001, 011, 010, 000 \\
C_3 &= 100, 101, 111, 110, 000
\end{align*}
\]

Obviously, every vertex of \( LTQ_3 \) belongs to at least one of the cycles \( C_0, C_1, C_2, C_3 \) and moreover, the sets \( \{j \mid v \in C_j\} \) are all pair distinct.

**Case 2.** \( n \geq 4 \). We construct \( n + 1 \) cycles all starting from the all-zero binary string \( 0^n \). Let \( \pi = \langle 0, 1, 2, \ldots, n - 1 \rangle \). Let \( C_n = \Omega(\pi, 0^n) \). By Theorem 1, \( C_n \) is a Hamiltonian cycle of \( LTQ_n \). The \( n \) cycles \( C_0, C_1, \ldots, C_{n-1} \) are constructed as follows.

Let \( C_0 = \Omega(\pi^0, 0^n) \) and \( C_1 = \Omega(\pi^1, 0^n) \) where \( \pi^0 = \langle 1, 2, \ldots, n - 1 \rangle \) and \( \pi^1 = \langle 0, n - 1, 2, \ldots, n - 2 \rangle \), respectively. It is observed that \( C_0 \) is a cycle of length \( 2^{n-1} \) which passes all vertices whose 0th coordinate equals 0. Similarly, by Lemma 8, \( C_1 \) is a cycle consists of all vertices whose 1th coordinate equals 0. Given \( 2 \leq i \leq n - 1 \), \( \pi^i = \langle 0, 1, 2, \ldots, i - 1, i + 1, i + 2, \ldots, n - 1 \rangle \). Let \( C_i = \Omega(\pi^i, 0^n) \). By Lemma 7, \( C_i \) is a cycle of length \( 2^{n-1} \) which visits exactly once all the \( 2^{n-1} \) vertices whose \( i \)th coordinate equals 0.

Therefore, a vertex \( v \) lies in \( C_i \) if and only if \( v_i = 0 \). The cycle \( C_n \) guarantees that also the \( 1^n \) vertex lies in at least one cycle.

5 Conclusions

System level diagnosis proposed in [17] and the fault diagnosis framework based on identifying codes [8] are two powerful approaches in fault diagnosis of computer and communication systems. There are two main differences in the application of the above two frameworks in fault diagnosis. In the application of the system level diagnosis we are required to construct a test graph that satisfies certain measures of diagnosis as desired by the application. Constructing such a graph is not always easy. Also, we often need sophisticated algorithms for diagnosis using the test graph. In applying the identifying codes technique to diagnosis, we need to find a identifying code of the given graph. In certain cases an identifying code may not even exist. If such a code does not exist, we can decompose the graph into simple subgraphs.
for which identifying codes exist. Then we can place monitors on each of the subgraphs and use them to isolate faults within those subgraphs. In this paper, identifying codes can be constructed by cycle for locally twisted cubes. We solve the vertex identification problem on locally twisted cubes.

References


