Vertex Identifying Codes for Fault Isolation in Communication Networks *†‡

Krishnaiyan Thulasiraman[a], Min Xu[a,b], Ying Xiao[c], and Xiao-Dong Hu[b]

[a] School of Computer Science, University of Oklahoma, Norman, Oklahoma, 73019, USA

[b] Institute of Applied Mathematics, Chinese Academy of Sciences, Beijing, 100080, China

[c] Packet Design Inc., Palo Alto, California, 94304, USA

Abstract 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, testing and tolerance have become issues of great interest in the design and analysis of computer and communication systems. It was in this context that Preparata, Metze and Chien [1] at the University of Illinois proposed a model, now well known as the PMC model, and a framework for the diagnosis of large scale digital systems. Recently, Karpovsky et al. [5] proposed the concept of identifying codes and its application in the detection/isolation of faults in communication systems. In this paper we give a broad survey of certain important results in this area along with our recent work on a generalization of the identifying codes concept and an approximation algorithm.

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, testing and tolerance have become issues of great interest

*The work was supported in part by the US National Science Foundation under the ITR grant ECS 0426831.

†The work was supported in part by NNSF of China No. 10626053, No. 70221001 and No. 60373012.

‡The work was supported in part by China Postdoctoral Science Foundation.

in the design and analysis of computer and communication systems. It was in this context that Preparata, Metze and Chien [1] at the University of Illinois proposed a model, now well known as the PMC model, and a framework for the diagnosis of large scale digital systems.

In the PMC model for the diagnosis of multiprocessor systems, the problem is formulated by what is called a test graph. Each vertex in the test graph represents a processor. It is assumed that each processor tests a subset of the remaining processors. A link directed from the vertex representing processor p_i to the vertex representing processor p_j indicates that processor p_i has tested p_j. If p_i tests p_j as fault free then a bit 0 is associated with the corresponding link $(p_i,\ p_j)$; otherwise a 1 bit is associated with this link. The test result arising from a faulty testing processor could be arbitrary, that is , unreliable. The collection of all test results (0 or 1) is called a syndrome. The problem is to identify all faulty processors subject to certain assumptions. In the last forty years or so since the introduction of this approach to diagnosis, several fundamental results have been reported on various generalizations and related diagnosis algorithms. For examples, see [2]-[4] and the reference therein.

Recently Karpovsky et al. [5] introduced the concept of identifying codes and studied 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 Karpovsky et al. [5] 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. Two important applications have triggered considerable research on the identifying codes problem. One of these is the problem of diagnosing faulty processors in a multiprocessor system [5]. Another application is robust location detection in emergency sensor networks [6]. Next we briefly describe the application of identifying codes in fault diagnosis.

Consider a communication network modeled as an undirected graph G. Each vertex in the graph represents a processor and each edge represents the communication link connecting the processors represented by its end vertices. Some of the processors could become faulty. To simplify the presentation let us assume that at most one processor could become faulty at any given time. Assume that a processor, when it becomes faulty, can trigger an alarm placed on an adjacent processor. We would like to place alarms on certain processors that will facilitate unique identification of the faulty processors. We would also like to place alarms on as few processors as possible. If D is a minimum identifying code for the case $t = 1$, then placing alarms on the processors represented by the vertices in the set D will help us to uniquely identify the faulty processor. Notice that we only need to consider $t = 1$ because if $t > 1$ is desired, we can proceed with G^t, the tth power of G.

Karpovsky et al. [5] have studied the identifying codes selection problem extensively and have established bounds on the cardinality of the identifying codes. They have shown how to construct the identifying codes for specific topologies such as binary cubes and trees. Several problems closely related to the identifying codes problem have been studied in the literature. Some of these may be found in [7], [8].

This paper is organized as follows. In Section 2, a formal definition of the identifying codes problem is first given. This is followed by a review of results on this problem for certain specific topologies, namely, cycles, trees and hypercubes. In Section 3, a generalization of the identifying codes problem and an approximation algorithm for this problem are discussed.

2 Identifying Codes Problem and Construction of Codes

Let $G = (V, E)$ be a simple, connected, undirected graph and $r \geq 1$ be an integer. Given a vertex $x \in V$, we define $B_r(x) = \{y : d(x, y) \leq r\}$ where $d(x, y)$ denotes the distance of the shortest path between x and y in G. For a subset C of V and any vertex set $X \subseteq V$, we define $I_r(X) = I_r(C; X) = (\cup_{x \in X} B_r(x)) \cap C$ and we say that C r-covers X if $I_r(X) \neq \emptyset$. We say that a subset C, r-separates two distinct vertex sets X and Y if and only if $I_r(X) \neq I_r(Y)$. For a graph G, let S be a collection of all the vertex sets of $V(G)$ with cardinality at most l, then an $(r, \leq l)$-identifying code of G is a set $C \subseteq V$ which r-covers all the vertex set in S and r-separates any pair of distinct vertex set in S.

The identifying codes problem is to find the minimum cardinality of such a code, which we denote by $M_r^{(\leq l)}(G)$. For convenience, we use $M_r(G)$ for $l = 1$.

The problem of determining an identifying code with minimum cardinality in a graph has been proved to be NP-Complete [9]. Many researchers have focused on the study of identifying codes in some restricted classes of networks, for example [10]-[14]. In this section, we survey results on this problem for the case of cycles, trees and hypercubes.

2.1 Identifying Codes for Cycles

A cycle C_n for $n \geq 3$ denotes a graph $(V(C_n), E(C_n))$ where $V(C_n) = \{v_i : 0 \leq i \leq n-1\}$ and $E(C_n) = \{v_i v_{i+1} : 0 \leq i \leq n-1\}$, and the subscripts are taken modulo n. We will present some known results about the r-identifying code of C_n.

When n is even, Bertrand *et al.* give the following theorem in [11].

Theorem 2.1 (Bertrand *et al.* [11]) For all $r \geq 1$, we have $M_r(C_{2r+2}) = 2r+1$ and $M_r(C_n) = \frac{n}{2}$ for $n \geq 2r+4$ even. Furthermore, the identifying code can be selected as $C' = V(C_n) \setminus \{v_0\}$ and $C'' = \{v_i : i \text{ is odd}\}$ for $n = 2r+2$ and $n \geq 2r+4$, respectively. ∎

In [13], Gravier *et al.* consider the problem with odd n. They get the following result.

Theorem 2.2 (Gravier *et al.* [13]) For all $r \geq 1$ and $n \geq 2r+3$ odd, we have

$$\frac{n+1}{2} + \frac{\gcd(2r+1, n) - 1}{2} \leq M_r(C_n) \leq \frac{n+1}{2} + r.$$

∎

For some subcases, they get the theorems below.

Theorem 2.3(Gravier *et al.* [13]) Let $r \geq 1$, n be an odd integer such that $3r+2 \leq n \leq 4r+1$, then $M_r(C_n) = \frac{n+1}{2} + \frac{\gcd(2r+1,n)-1}{2}$. ∎

Theorem 2.4(Gravier *et al.* [13]) Let $r \geq 1$, n be an odd integer such that $\gcd(2r+1,n) = 1$, and $4r+5 \leq n \leq 8r+1$. Then $M_r(C_n) = \frac{n+1}{2}$. ∎

Theorem 2.5(Gravier *et al.* [13]) Let $r \geq 1$, n be an odd integer such that $n \geq 3r+2$ and $\gcd(2r+1,n) \neq 1$. Then $M_r(C_n) = \frac{n+1}{2} + \frac{\gcd(2r+1,n)-1}{2}$. ∎

Theorem 2.6(Gravier *et al.* [13]) $M_1(C_5) = 3$; $M_1(C_n) = \frac{n+1}{2}$ for all $n \geq 7$, n odd; $M_r(C_n) = \lfloor \frac{2n}{3} \rfloor$ for all $r \geq 1$, $n = 2r+3$; $M_r(C_{4r+3}) = 2r+3$. ∎

Recently, we studied in [14] the r-identifying code about the cycles C_n for odd n with $\gcd(2r+1,n) = 1$ and established the following results.

Theorem 2.7(Xu *et al.* [14]) Let $r \geq 1$, n be an odd integer such that $n \geq 3r+2$, and $\gcd(2r+1,n) = 1$. If $n = 2m(2r+1)+x$ for $m \geq 1$ and $2 \leq x \leq 2r$ or $n = (2m+1)(2r+1)+y$ for $m \geq 1$, $1 \leq y \leq 2r-1$ or $m = 0$, $r+1 \leq y \leq 2r-1$, then $M_r(C_n) = \frac{n+1}{2}$. ∎

Theorem 2.8(Xu *et al.* [14]) Let $r \geq 1$, n be an odd integer such that $n \geq 3r+2$, and $\gcd(2r+1,n) = 1$. If $n = 2m(2r+1)+x$ for $m \geq 1$ and $x = 1$ or $n = (2m+1)(2r+1)+y$ for $m \geq 1$, $y = 2r$, then $M_r(C_n) = \frac{n+1}{2} + 1$. ∎

Remark 2.9 In Theorem 2.3, 2.4, 2.5, 2.7, we can define an r-identifying code with minimum cardinality as $C = \bigcup_{j=0}^{a-1} \bigcup_{i=0}^{\frac{n}{a}-1} \{v_{i(2r+1)+j} : i \text{ is even}\}$, where $a = \gcd(2r+1,n)$.

Table 1 shows all of the results about the r-identifying code of cycle C_n.

	$n =$ $2r+3$	$2r+5 \leq n$ $< 3r+2$	$3r+2 \leq n$ $\leq 8r+1$	$n > 8r+1$	
				$\gcd(2r+1,n) \neq 1$	$\gcd(2r+1,n) = 1$
odd n	√	?	√	√	√
even n	√				

Table 1

From Table 1, we can see that, when $2r+5 \leq n \leq 3r+1$, the r-identifying code of C_n with minimum cardinality for odd n is not determined yet.

2.2 Identifying Codes for Trees

Tree is a connected network without cycle. We select a vertex in the tree as root. The tree with root is called a rooted-tree. In a rooted-tree any vertex of one degree, unless it is the root, is called a leaf. Each node is either a leaf or an internal vertex in a rooted-tree. The maximum length from root to all leaves is called the height of the rooted-tree. For a rooted-tree, if its root is of degree k and all other internal vertices are of degree at most $k+1$, then we call this rooted-tree a k-ary tree.

In [10], Bertrand *et al.* give a lower bound about the tree with n vertices.

Theorem 2.10(Bertrand *et al.* [10]) For a tree T with n vertices, $M_1(T) \geq \frac{3(n+1)}{7}$. ∎

Karpovsky *et al.* consider the p-ary tree in [5].

Theorem 2.11(Karpovsky *et al.* [5]) For a p-ary tree T with height l ($l \geq 3$), we have

$$p^{l-3}(p^2+1) \leq M_1(T) \leq \begin{cases} \frac{p^{l+1}-1}{p^2-1}, & \text{if } l \text{ is odd} \\ \frac{p^{l+1}-p}{p^2-1}, & \text{if } l \text{ is even.} \end{cases}$$

∎

Corollary 2.12(Karpovsky *et al.* [5]) For p-ary trees T with height $l=3$, $M_1(T) = p^2+1$, while for a p-ary tree T' with height $l=4$, $M_1(T') = p(p^2+1)$. ∎

They also consider the case when $r > 1$. But they proved that the vertices in a tree are not identifiable if $r > 1$.

Theorem 2.13(Karpovsky *et al.* [5]) It is not possible to uniquely identify the vertices of a p-ary l-height tree for $r > 1$. ∎

For the complete binary trees, Bertrand *et al.* get the following result in [10].

Theorem 2.14(Bertrand *et al.* [10]) For a complete binary tree T with 2^h-1 vertices, $M_1(T) = \lceil 20(2^h-1)/31 \rceil$. ∎

A path P_n is a special tree where $V(P_n) = \{v_1, v_2, \ldots, v_n\}$ and $E(P_n) = \{(v_i, v_{i+1}) : i = 1, 2, \ldots, n-1\}$. In [11], Bertrand *et al.* get the following results.

Theorem 2.15(Bertrand *et al.* [11]) For a path P_n, $M_1(P_n) = \frac{n+1}{2}$ if $n \geq 1$ is odd and $M_1(P_n) = \frac{n}{2} + 1$ if $n \geq 4$ is even. Furthermore, the identifying code can be selected as $C' = \{1, 3, \ldots, n\}$ and $C'' = \{1, 3, \ldots, n-7, n-5, n-3, n-2, n-1\}$ for odd n and even n, respectively. ∎

Theorem 2.16(Bertrand *et al.* [11]) For $r \geq 2$, $n \geq 2r+1$, $M_r(P_n) \geq \lceil \frac{n+1}{2} \rceil$. ∎

Theorem 2.17(Bertrand *et al.* [11]) Let k be a nonnegative integer. For any fixed $r \geq 2$, and for $n = (4r+2)k+1$, $M_r(P_n) \leq \frac{n+1}{2}$. ∎

2.3 Identifying Codes for Hypercubes

The hypercube Q_n is a popular interconnection network. In Q_n, $V(Q_n) = \{(x_1, x_2, \ldots, x_n) : x_i \in \{0,1\},\ i = 1, 2, \ldots, n\}$ and two vertices $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots, y_n) \in V(Q_n)$ are linked by an edge if and only if $x - y = (0, 0, \ldots, \pm 1, 0, \ldots, 0)$. Karpovsky *et al.* [5] first obtained some lower bounds on $M_1(Q_n)$.

Theorem 2.18(Karpovsky *et al.* [5]) For an n-dimensional hypercube Q_n, $n \geq 3$, $M_1(Q_n) \geq \frac{n \times 2^{n+1}}{n(n+1)+2}$. ∎

Let $K(n,q)$ be the size of an optimal code C^* in $V(Q_n)$ satisfying that every vertex in $V(Q_n)$ is at distance at most q from a codeword of C^*. An upper bound on $M_1(Q_n)$ follows from the theorem below.

Theorem 2.19(Karpovsky *et al.* [5]) Let C^* be an optimal code of Q_n and covering radius 2, i.e., C^* has $K(n,2)$ codewords. Then, for $r = 1$, a code C identifying vertices in the n-dimensional hypercube can be selected as $C = \{w : \exists\, v \in C^*,\ d(v,w) = 1\}$ ($d(v,w)$ is Hamming distance between v and w). ∎

Corollary 2.20(Karpovsky *et al.* [5]) The number of codewords in an optimal identifying code with $r = 1$ for a n-dimension hypercube Q_n ($n \geq 3$) is upper-bounded by $M_1(Q_n) \leq nK(n,2)$. ∎

Theorem 2.19 can be extended to the case $r > 1$ as follows.

Theorem 2.21(Karpovsky *et al.* [5]) For any given $r < \frac{n}{2}$, a code C for identifying vertices in the n-dimensional hypercube ($n > 2$) can be obtained by selecting as codewords all vertices at distance exactly r from the codewords of a optimal code C^* which has covering radius $2r$, i.e., $C = \{x : \exists\, u \in C^*,\ d(x,u) = r\}$. ∎

Corollary 2.22(Karpovsky *et al.* [5]) For $r < \frac{n}{2}$, the number of codewords required for identifying vertices in a hypercube is upper bounded by $M_r(Q_n) \leq K(n, 2r)\binom{n}{r}$. ∎

Corollary 2.23(Karpovsky *et al.* [5]) The number of codewords required for a hypercube with $(4s+1)t$ dimensions using balls of radius st is upper-bounded by $M_{st}(Q_{(4s+1)t}) \leq \binom{(4s+1)t}{st}2^t$. ∎

A code $C \subseteq V(Q_n)$ is called a u-fold r-covering if every vertex in $V(Q_n)$ is r-covered by at least u codewords of C. Let us denote by $K(n, r, u)$ the smallest size of a u-fold r-covering in hypercube Q_n.

Theorem 2.24(Laihonen [15]) Let $l \geq 3$. A code $C \subseteq V(Q_n)$ is $(1, \leq l)$ identifying code of Q_n if and only if it is a $(2l-1)$-fold 1-covering. In particular $M_1^{\leq l}(Q_n) = K(n, 1, 2l-1) \geq \lceil (2l-1)\frac{2^n}{n+1} \rceil$. ∎

Corollary 2.25(Laihonen [15]) Let $l \geq 3$. Then $M_1^{\leq l}(Q_n) = (2l-1)\frac{2^n}{n+1}$ if and only if there are integers $i \geq 0$, $u_0 > 0$ such that $u_0|(2l-1)$ and $2l-1 \leq 2^i u_0$ and $n = u_0 2^i - 1$. ∎

3 d-Identifying Codes Problem

Karpovsky et al. [5] have shown that unique identification of vertices may not always be possible for certain topologies. In other words, triggering of alarms on a set of processors could mean that one of several candidate processors could be faulty. Once such a set of possible faulty processors has been identified then testing each processor in this set will identify the faulty processor. This motivates the generalization of the identifying codes problem to d-identifying codes problem defined below [16]. This generalization is similar to the introduction of t/s diagnosable systems that generalize the t-diagnosable systems introduced by Preparata, Metze and Chien [1]. An introduction to t-diagnosable systems and their generalization may be found in [2], [3].

3.1 Definition of the d-Identifying Codes Problem

Consider an undirected graph $G(V, E)$ with each vertex $v \in V$ associated with an integer cost $c(v) > 0$ and an integer weight $w(v) > 0$.

Let $N[v]$ be the set of vertices containing v and all its neighbors. For a subset of vertices $S \subseteq V$, define the cost and weight of S as

$$c(S) = \sum_{v \in S} c(v) \text{ and } w(S) = \sum_{v \in S} w(v).$$

Two vertices $u, v \in V$ are distinguished by vertex w iff $|N[w] \cap \{u, v\}| = 1$. A set of vertices $D \subseteq V$ is called an identifying set if (1) every unordered vertex pair (u, v) is distinguished by some vertex in D and (2) D is a dominating set of G, i.e., each vertex in G is adjacent to at least one vertex in D (we will relax this requirement later).

Given $D \subseteq V$, define $I_D(v) = N[v] \cap D$ and an equivalence relation $u \equiv v$ iff $I_D(u) = I_D(v)$. The equivalence relation partitions V into equivalence classes $V_D = \{S_1, S_2 \ldots, S_l\}$ such that $u, v \in S_i \iff I_D(u) = I_D(v)$.

For any $D \subseteq V$, let V_D be the equivalence classes induced by D. If D is a dominating set of G and $d \geq \max\{w(S_1), w(S_2) \ldots, w(S_l)\}$, then D is called a d-identifying code of G. The d-identifying codes problem is to find a d-identifying code $D \subseteq V$ with minimum cost.

Note that if $d = 1$ then the d-identifying codes problem reduces to the identifying codes problem if the vertex costs and weights are equal to unity. Also, whereas the cost of the d-identifying code is a measure of the cost of installing alarms, the value of d is a measure of the degree of uncertainty in the identification of faulty processors. Since the value of d is also a measure of the expenses involved in testing each processor in an equivalence class, d has to be set at a small value.

The identifying code must be a dominating set. However we can drop this requirement after a simple transformation of the graph, i.e., adding a new isolated vertex with weight d and a very big cost such that any cost aware algorithm will not include this vertex in the solution set. Thus it will be the only vertex not adjacent to the identifying set. So we will ignore the dominating set condition for the simplicity of presentation.

3.2 An Approximation Algorithm for the d-Identifying Codes Problem

We now present a greedy approximation algorithm for the d-identifying codes problem. This algorithm, presented in [16], was inspired by a heuristic for the minimum probe selection problem [17] based on ideas from information theory. In the following $H(V_s)$ is called the entropy defined on V_s which is the set of equivalence classes induced by S. Similarly, $I(V_s, v) = H(V_s) - H(V_{S+v})$ is called the information content of $v \in V - S$ with respect to S. The definition of a required entropy function will be presented later. The framework of this greedy algorithm without specific entropy function is applicable to any class of d-identifying codes problems whose detailed specifications can be hidden in the definition of the entropy. Based on the framework of this greedy algorithm, one only needs to focus on the design of the entropy for other variations of the d-identifying codes problem, e.g., the strong identification codes problem [8].

Algorithm 1 Greedy Algorithm

```
1: Initialize D = ∅
2: while H(D) > 0 do
3:     Select vertex v* = arg max_{v∈V−D} I(V_D; v)/c(v)
4:     D ← D ∪ {v*}.
5: end while
```

The time complexity of the above greedy algorithm is $O(n^2 T_H(n))$, where T_H is the time complexity function of the algorithm computing $H(\cdot)$. The following theorem is the main result on the approximation ratio of the greedy algorithm.

Theorem 3.1(Xiao *et al.* [16]) Denote V_D as the set of equivalence classes induced by $D \subseteq V$. Suppose an entropy function $H(\cdot)$ satisfies the following conditions:

(a) $H(V_D) = 0$ for any d-identifying set D,

(b) If $H(V_S) \neq 0$, then $H(V_S) \geq 1$, and

(c) $I(V_S; v) \geq I(V_{S+u}; v)$ for all $u \neq v, S \subseteq V$, then the greedy algorithm returns a d-identifying code D such that $c(D)/c(D^*) < \ln[H(V_\emptyset)] + 1$ (recall that by definition $V_\emptyset = V$), where $D^* = \{v_1^*, v_2^* \dots, v_{|D^*|}^*\}$ is the minimum d-identifying code. ∎

Corollary 3.2(Xiao *et al.* [16]) Let $G(V, E)$ be a graph with n vertices with equal cost which are labeled such that $I(V_\emptyset; v_1) \geq I(V_\emptyset; v_2) \cdots \geq I(V_\emptyset; v_n)$. Then the optimal cost of the minimum d-identifying set, $OPT_d(G) \geq K$, where K is the smallest integer such that $\sum_{i=1}^{K} I(V_\emptyset; v_i) \geq H(V_\emptyset)$. ∎

Given $d \geq 2$, let the function $f_d(n)$ defined be as follows.

$$f_d(n) = \begin{cases} n \lg(n/d), n \geq d \\ 0, otherwise. \end{cases} \tag{1}$$

Theorem 3.3(Xiao *et al.* [16]) Using the entropy $H_d(V_D) = \sum_{S \in V_D} f_d(w(S))$ with $f_d(\cdot)$ as defined above, the greedy algorithm guarantees the approximation ratio of $1 + \ln d + \ln(|V| \lg |V|)$. ∎

Corollary 3.4(Xiao *et al.* [16]) For the identifying codes problem, the greedy algorithm guarantees the approximation ratio of $1 + \ln |V| + \ln(\lg |V|)$. ∎

In fact, we can show that the function $f_d(n)$ defined in (1) is optimal in asymptotic sense, i.e., the approximation ratio based on Theorem 3.1 cannot be improved by finding better entropy.

Lemma 3.5(Xiao *et al.* [16]) $f(n) \geq \Theta(n \lg n)$. ∎

3.3 Hardness of the d-Identifying Codes Problem

Lemma 3.6(Xiao *et al.* [16]) Assume that the identifying codes problem is feasible. For any fixed $d \geq 2$, if there exists a polynomial time ϕ-approximation algorithm for the d-identifying codes problem, there also exists a polynomial time ϕ-approximation algorithm for the identifying codes problem. ∎

Lemma 3.6 means that for any fixed d, the d-identifying codes problem is at least as hard as the identifying codes problem in term of approximability. Thus, with an application of the results in [7], we have the following theorem.

Theorem 3.7(Xiao *et al.* [16]) For any given $d \geq 1$, the d-identifying codes problem with unit vertex costs and weights is not approximable within $(1 - \epsilon) \ln |V|$ unless $NP \in DTIME(n^{\lg \lg n})$. ∎

In view of Corollary 3.4, we can see that the approximation ratio of the greedy algorithm is quite tight for the d-identifying codes problem where the vertex costs and weights are 1. Furthermore, we can expect that the approximation ratio is also very tight for general d-identifying codes problem as in the special case.

In [16] the authors reported a probabilistic analysis on random graphs assuming that vertex costs and weights are all equal. It has been shown that a d-identifying set of certain cardinality exists with very high probability. It has also been shown that a d-identifying code of cardinality smaller than this number does not exist with a high probability.

4 Summary

System level diagnosis proposed in [1] and the fault diagnosis framework based on identifying codes [5] are two powerful approaches of importance in fault diagnosis, testing and tolerance studies of computer and communication systems. References [2]-[4] point to several advances in the system level diagnosis area. In this paper we have given a broad survey of certain key results on the identifying codes problem, its generalization and related algorithms.

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. Since identifying codes can be constructed for graphs such as trees, cycles and hypercubes as discussed in this paper, the diagnosis approach based on identifying codes presents an interesting and alternative approach to the system level diagnosis approach.

References

[1] F. Preparata, G. Metze, and R. Chien, On the connection assignment problem of diagnosiable systems. *IEEE Trans. on Electronic Computers*, 16, (1967), 848-854.

[2] A. Das and K. Thulasiraman, Diagnosis of t/s-diagnosable systems. In *International Workshop on Graph-Theoretic Concepts in Computer Science*, (1990), 193-205.

[3] A. Das, K. Thulasiraman, and V. Agarwal, Diagnosis of t/(t+1)-diagnosable systems. *SIAM J. Comput.*, 23(5), (1994), 895-905.

[4] A. Das, K. Thulasiraman, V. K. Agarwal and K. B. Lakshmanan, Multiprocessor Fault Diagnosis Under Local Constraints. *IEEE Trans. on Computers*, 42(8), (1993), 984-988.

[5] M. Karpovsky, K. Chakrabarty, and L. Levitin, On a new class of codes for identifying vertices in graphs. *IEEE Trans. on Information Theory*, 44(2), (1998), 599-611.

[6] S. Ray, R. Ungrangsi, F. Pellegrini, A. Trachtenberg, and D. Starobinski, Robust location detection in emergency sensor networks. *INFOCOM*, 2003.

[7] Bjarni V. Halldórsson, Magnús M. Halldórsson, and R. Ravi, On the approximability of the minimum test collection problem. In *ESA*, (2001), 158-169.

[8] Tero Laihonen, Optimal codes for strong identification. *European Journal of Combinatorics*, 23(3), (2002), 307-313.

[9] I. Charon, O. Hudry, A. Lobstein, Minimizing the size of an identifying or locating-dominating code in a graph is NP-hard. *Theoretical Computer Science,* 290(3), (2003), 2109-2120.

[10] N. Bertrand, I. Charon, O. Hudry and A. Lobstein, 1-Identifying codes on trees. *Australasian Journal of Combinatorics,* 31, (2005), 21-35.

[11] N. Bertrand, I. Charon, O. Hudry and A. Lobstein, Identifying and locating-dominating codes on chains and cycles. *European Journal of Combinatorics,* 25(7), (2004), 969-987.

[12] G. Cohen, I. Honkala, S. Litsyn, and A. Lobstein, *Covering codes*. Amsterdam, The Netherlands: Elsevier, 1997.

[13] S. Gravier, J. Moncel and A. Semri, Identifying codes of cycles. *European Journal of Combinatorics,* 27(5), (2006), 767-776.

[14] M. Xu, K. Thulasiraman and X.-D. Hu, The identifying codes of cycles with odd order. *Manuscript*, 2006.

[15] T. K. Laihonen, Sequences of Optimal identifying codes. *IEEE Transaction on Information Theory,* 48(3), (2002), 774-776.

[16] Y. Xiao, C. Hadjicostis, and K.Thulasiraman, The d-Identifying Codes Problem for Vertex Identification in Graphs: Probablistic Analysis and an Approximation Algorithm. In *COCOON 2006 (12th Annual International Computing and Combinatorics Conference)*, Taipei, August 2006.

[17] M. Brodie, I. Rish, and S. Ma, Optimizing probe selection for fault localization. In *International Workshop on Distributed Systems: Operations and Management*, 2001.