Info to Proc. Letter Dec 96

On the Covering of Vertices for Fault Diagnosis in Hypercubes*

Mark G. Karpovsky Krishnendu Chakrabarty
Lev B. Levitin Dimiter R. Avresky

Department of Electrical and Computer Engineering
Boston University
44 Cummington Street
Boston, MA 02215

December 1996

Contact author: Krishnendu Chakrabarty
Phone: (617) 353-1235
FAX: (617) 353-6440
Email: kchakrab@hilsa.bu.edu

ABSTRACT

We investigate the optimal covering of vertices by Hamming balls of radius t in a hypercube $\mathcal{Z}_2^n$ such that any vertex in $\mathcal{Z}_2^n$ can be uniquely identified by examining the vertices that cover it. Given $\mathcal{Z}_2^n$ and an integer $t \geq 1$, we find a (minimal) set C of vertices such that every vertex in $\mathcal{Z}_2^n$ belongs to a unique set of balls of radius t centered at the vertices in C. This is useful in diagnosing processor faults in hypercube-based multiprocessor systems.

Keywords: Coding theory, distributed systems, fault diagnosis, fault tolerance, multiprocessors.

*This research was supported in part by the National Science Foundation under grant no. MIP 9630096, by NATO under grant no. 910411, and by a start-up grant from Boston University's College of Engineering.

1 Introduction

A hypercube computer is a multiprocessor system with $N = 2^n$ processors interconnected as an n-dimensional binary cube. Each processor P_i constitutes a node of the cube and is a self-contained computer with its own CPU and local memory. Each P_i also has direct communication paths to n other neighbor processors through the edges of the cube. The processors are labeled with n-bit vectors such that two processors are adjacent if and only the Hamming distance between their n-bit labels is one. An example of a commercial hypercube computer is the NCUBE/ten, which is a 10-dimensional system developed by NCUBE Corporation [7, 8].

We investigate the problem of covering the vertices of a hypercube $\mathcal{Z}_2^n$ such that we can uniquely identify any vertex in $\mathcal{Z}_2^n$ by examining the vertices that cover it. We define a *ball* of radius t centered on a vertex v to be the set of vertices of $\mathcal{Z}_2^n$ that are at distance at most t from v. (The distance between vertices v_i and v_j is the number of edges in the shortest path between v_i and v_j.) The vertex v is then said to *cover* itself and every other vertex in its ball. Given $\mathcal{Z}_2^n$ and an integer $t \geq 1$, we determine a (minimal) set $\mathcal{C}$ of vertices such that every vertex of $\mathcal{Z}_2^n$ belongs to a unique set of balls of radius t centered at the vertices in $\mathcal{C}$. We view $\mathcal{C}$ as a covering code such that all vertices in it are codewords.

An application of our results lies in fault diagnosis of hypercube-based multiprocessor systems. The goal of fault diagnosis is to locate faulty processors. Specific software routines are executed on certain selected processors to carry out diagnosis. The selection is done by generating the code $\mathcal{C}$ that allows for unique identification of processors. Every processor corresponding to a codeword vertex tests itelf and all its neighboring processors. This corresponds to the use of balls of radius one centered at the codewords, i.e. $t = 1$. Hence an optimal code minimizes the amount of overhead required to implement fault diagnosis.

2 Vertex covering

Let $M_t(n)$ be the minimum number of codewords required to identify every vertex uniquely in a n-dimensional hypercube when balls of radius t are used. We first derive the following lower bounds on $M_t(n)$:

Theorem 1 *For an n-dimensional hypercube with $N = 2^n$ vertices,*

1. $M_t(n) \geq \lceil \log_2(N+1) \rceil$.
2. $M_t(n) \geq \dfrac{2N}{1 + \sum_{j=0}^{t} \binom{n}{j}}$.

Proof: The first lower bound follows from the fact that there are $N+1$ cases (N different vertices, and the selection of no vertex—no vertex is identified) to be distinguished. Therefore, the information can be encoded in a minimum of $\lceil \log_2(N+1) \rceil$ bits.

To prove the second bound, consider a matrix with $M_t(n)$ rows and N columns. An entry $x_{i,j}$ in this matrix is one if vertex i covers vertex j. The number of ones in this matrix is $M_t(n) \cdot V(t)$, where $V(t) = \sum_{j=0}^{t} \binom{n}{j}$ is the number of vertices that are at distance at most t from any vertex v in the hypercube. The definition of $M_t(n)$ implies that the maximum number of columns with a single one is $M_t(n)$; an additional $M_t(n) \cdot V(t) - M_t(n)$ ones are present in the matrix. Since every other column of the matrix has at least two ones, we have

$$M_t(n) + \frac{M_t(n) \cdot V(t) - M_t(n)}{2} \ \geq \ N,$$

which yields

$$M_t(n) \ \geq \ \frac{2N}{1 + V(t)} = \frac{2N}{1 + \sum_{j=0}^{t} \binom{n}{j}}. \tag{1}$$

□

We next turn to the problem of generating codewords to cover the vertices in $\mathcal{Z}_2^n$. This will lead to upper bounds on $M_t(n)$.

Let $K(n,q)$ be the size of a minimal code[1] C^* of length n with covering radius q, i.e., every noncodeword is at Hamming distance at most q from a codeword of C^* [3, 4].

[1] A minimal code is one that has a minimum number of codewords.

Theorem 2 *For any given $t < n/2$, the code C for identifying vertices in the n-dimensional binary cube ($n > 2$) can be obtained by selecting as codewords all vertices at distance t from the codewords of a minimal code $C^\star$ which has covering radius $2t$, i.e. $C = \{x \mid \exists u \in C^\star, d(x,u) = t\}$.*

Proof: We first make the following observation: if vertices v_1 and v_2 are such that there is at least one ball of radius $2t$ centered at a vertex in $C^\star$ to which v_1 (v_2) belongs but v_2 (v_1) does not belong, then v_1 and v_2 can be distinguished using codewords from C. We therefore need to prove that a set of vertices can be distinguished even if they always belong to the same ball centered at a vertex $u \in C^\star$.

Without loss of generality, let $u = \underbrace{00\ldots0}_{n} \in C^\star$. All vertices with t ones now belong to C and serve as codewords for identifying a vertex. Given two vertices v_1 and v_2 that are in the same ball centered at u, we show that we can always find a codeword $x \in C^\star$ such that x covers v_1 but not v_2. Let $\overline{v_i}$ be the bitwise complement of v_i, and let $v_i v_j$ ($v_i + v_j$) be the component-wise AND (OR) of v_i and v_j, and $v_i \leq v_j$ if $v_i v_j = v_j$. Let $w(v_1) = l_1$ and $w(v_2) = l_2$, where $w(v)$ is the weight of any given vertex v. Assume, without loss of generality, that $l_1 \geq l_2$. It follows therefore that $l_2 \geq 1$ because otherwise both v_1 and v_2 will be the all-0 vertex u.

We choose $x \in C$ such that either $x \leq v_1\overline{v_2}$ or $v_1\overline{v_2} \leq x$, and $S_1 = w(xv_1) = \lceil l/2 \rceil$. Then, $d(x, v_1) = t + l_1 - 2S_1 \leq t$. We next consider the vector $z = \tilde{1}\overline{(v_1 + v_2)}$. Obviously, $w(z) = n - l_1 - l_2 + k$, where $k = w(v_1v_2)$. Choose $y = x\overline{(xv_1)}$ so that either $y \leq z$, or $z \leq y$. The remaining nonzero components of x can be chosen arbitrarily. Note that $w(x(v_1\overline{v_2})) = \min\{t, l_1 - k\}$, and $w(xz) = \min\{t - \min\{t, l_1 - k\}, n - l_1 - l_2 + k\}$. Therefore,

$$\begin{aligned} S_2 &= w(xv_2) \\ &= \max\{0, t - \min\{t, l_1 - k\} - \min\{t - \min\{t, l_1 - k\}, n - l_1 - l_2 + k\}\} \\ &= \max\{0, t - \min\{t, l_1 - k\} - (n - l_1 - l_2 + k)\} \\ &= \max\{0, (t - l_1 + k) - (n - l_1 - l_2 + k)\} \;=\; \max\{0, t + l_2 - n\} \end{aligned}$$

Now, the distance between x and v_2 is given by

$$d(x, v_2) \;=\; t + l_2 - 2S_2$$

$$\begin{aligned} &= t + l_2 - 2 \cdot \max\{0, t + l_2 - n\} \\ &= t + l_2 - \max\{0, 2(t + l_2 - n)\} \\ &= t + l_2 + \min\{0, 2(n - t - l_2)\} \\ &= \min\{t + l_2, (n - 2t) + (n - l_2) + t\} \ \geq \ \min\{t + l_2, t + 1\} \end{aligned}$$

if $n > 2t$. Since $l_2 \geq 1$, this implies that $d(x, v_2) > t$. □

Corollary 1 *For $t < n/2$, the number of codewords required for identifying vertices, i.e. covering every vertex with a unique subset of codewords, in a binary cube is given by*

$$M_t(n) \ \leq \ K(n, 2t)\binom{n}{t}. \tag{2}$$

Exact values (for small n) as well as bounds on $K(n, 2t)$ are available in the literature (e.g. [2]).

Example: Consider a 5-dimensional binary cube. The best code with covering radius two is $C^* = \{00000, 11111\}$. Therefore, the set of ten codewords $\{10000, \ldots, 11110\}$ can be selected as the identifying code $\mathcal{C}$. In this example, the number of codewords is 10. From the lower bound (1), we get $M_1(5) \geq 64/7$, i.e. $m \geq 10$. Therefore, the number of codewords in the identifying code for this example is minimal and $M_1(5) = 10$.

We next estimate the ratio r_n between the upper bound and the lower bound on $M_t(n)$, the number of codewords. From (1) and (2), and using the result $K(n_1 + n_2, t_1 + t_2) \leq K(n_1, t_1)K(n_2, t_2)$ from [2] we get,

$$\frac{2^{n+1}}{1 + \sum_{i=0}^{t}\binom{n}{i}} \leq M_t(n) \leq \binom{n}{t}\{K(n/2t, 1)\}^{2t}$$

Furthermore, since $K(n, 1) \leq \frac{2^n}{2^{\lfloor \log_2(n+1) \rfloor}}$, it follows that

$$\frac{2^{n+1}}{1 + \sum_{i=0}^{t}\binom{n}{i}} \leq M_t(n) \leq \binom{n}{t}\frac{2^n}{2^{2t\lfloor \log_2(n/2t+1) \rfloor}}.$$

If $n/2t+1=2^s$, then using $\sum_{i=0}^{t}\binom{n}{i} \sim n^t/t!$ for $n\to\infty$ and $t/n\to 0$, we get[2]

$$\frac{2^{n+1}}{1+n^t/t!} \lesssim M_t(n) \lesssim \{\frac{2t\cdot 2^{n/2t}}{n}\}^{2t}\frac{n^t}{t!}.$$

The ratio r_∞ of the upper bound to the lower bound ($n\to\infty$) is given by

$$\begin{aligned} r_\infty &= \frac{(n^t/t!)(2t/n)^{2t}2^n(n^t/t!)}{2\cdot 2^n} \\ &= 2^{2t-1}\cdot\frac{t^{2t}}{(t!)^2} \end{aligned}$$

For $t=1$, we have $r_\infty=2$ as before, while for $t=2$, $r_\infty=32$.

For the special case of $n=(4s+1)t$, $s\geq 1$, we have the following corollary, which follows from the fact that $K(r(2s+1),rs)\leq\{K(2s+1,s)\}^r=2^r$.

Corollary 2 *The number of codewords required for a binary cube with $(4s+1)t$ dimensions using balls of radius st is given by*

$$M_{st}((4s+1)t) \;\leq\; \binom{(4s+1)t}{st}2^t.$$

As special cases, for $s=1$, we have $M_t(5t)\leq\binom{5t}{t}2^t$, and for $s=2$, we have $M_{2t}(9t)\leq\binom{9t}{2t}2^t$.

Another solution to the identifying code construction problem for an n-dimensional hypercube is obtained by selecting codewords separately for its two constituent $(n-1)$-dimensional cubes. This "divide and conquer" approach, which implies that $M_1(n)\leq 2M_1(n-1)$, often gives better results than the construction method using $K(n,2)$ for small values of n; see Table 1. (Note that for $n=3$ and $n=4$, we achieved the lower bound on $M_1(n)$ using ad hoc construction methods.)

A generalization of the "divide and conquer" approach is given by the following theorem.

Theorem 3 *The number of codewords required to identify vertices, i.e. cover every vertex with a unique subset of codewords, in an n-dimensional cube is given by*

$$M_t(n)\leq M_s(a)\cdot M_{t-s}(n-a),$$

[2] $a(n)\sim b(n) \leftrightarrow \lim_{n\to\infty}\frac{a(n)}{b(n)}=1.$

n	Lower bound bound on $M_1(n)$	$M_1(n)$ (using $K(n,2)$)	$M_1(n)$ (divide and conquer)
3	4*	6	6
4	6*	8	8
5	10*	10	12
6	16	24	20
7	29	49	40
10	177	300	320
16	7282	14336	20480

* Lower bound attained by ad hoc construction.

Table 1: Number of codewords required for identifying vertices in hypercubes.

n	$V(2)$	Lower bound bound on $M_2(n)$	Upper bound on $K(n,4)$	Upper bound on $M_2(n)$ using $K(n,4)$	Upper bound on on $M_2(n)$ using (3)
3	7	—	—	—	7
4	11	5	—	—	9
5	16	6	—	—	12
6	22	7	—	—	16
8	37	14	2	56	36
12	79	104	12	792	400
16	137	950	64	7680	6400
20	211	9893	512	97220	90000

Table 2: Number of codewords in a binary cube for $t = 2$.

where $0 \leq t < n$, $0 \leq s \leq t$ *and* $1 \leq a \leq n-1$.

Proof: Let $x = x_1x_2$ and $y = y_1y_2$ be vectors of length n, where x_1 (y_1) and x_2 (y_2) are of length a and $n-a$, respectively. Let $v^x = v_1^x v_2^x$ be a vector of length n such that v_1^x (v_2^x) covers x_1 (x_2) but not y_1 (y_2) with a ball of radius s ($t-s$) centered at it. Then $d(v_1^x, x_1) \leq s$ and $d(v_2^x, x_2) \leq t-s$, and this implies that $d(v^x, x) = d(v_1^x, x_1) + d(v_2^x, x_2) \leq t$. Hence v^x covers x with a ball of radius t. Now, $d(v_1^x, y_1) > s$ and $d(v_2^x, y_2) > t-s$, which implies that $d(v^x, y) > t$. Thus v^x does not cover y with a ball of radius t. Therefore, the identifying code $C(n,t)$ for an n-dimensional cube can be constructed using the identifying codes for the smaller a and $n-a$ dimensions in the following way: $C(n,t) = \{xy | x \in C(a,s), y \in C(n-a.t-s)\}$ and $M_t \leq |C(n,t)|$. □

Corollary 3 *As a special case of Theorem 3, we have*

$$M_t(n) = 2^a \cdot M_t(n-a).$$

Proof: From Theorem 3, we have $M_t(n) = M_0(a) \cdot M_t(n-a) = 2^a \cdot M_t^{(p)}(n-a)$. (When $t = 0$, every vertex in the a-dimensional cube must be selected as a codeword.) □

Corollary 4 *For any $t < n$, we have*

$$M_t(n) \;\; \leq \;\; M_{\lfloor t/2 \rfloor}(\lfloor /2 \rfloor) \cdot M_{\lceil t/2 \rceil}(\lceil n/2 \rceil). \tag{3}$$

Although it may be intuitively expected that the number of codewords required for identification decreases as t increases, this is not necessarily the case. For example, $M_2(3) = 7$ but $M_1(3) = 4$. Table 2 shows $M_n(2)$ for various values of n. Theorem 3 and Corollaries 3–4 can be easily generalized to p-ary ($p > 2$) n-dimensional cubes. Other topologies such as meshes and trees can be considered similarly.

3 Applications

An application of the results of Section 2 lies in the diagnosis of faults in hypercube multiprocessor systems. Traditional diagnosis techniques model the multiprocessor system as a digraph, termed the test graph, whose vertices denote processors and an edge or test link (p_i, p_j) from processor p_i to p_j indicates that p_i tests p_j. A test link between p_i and p_j is labeled 1 (0) if p_i determines p_j to be faulty (fault-free) [9]. A collection of 0-1 values on the test links is referred to as a syndrome and a central host locates a faulty processor from the syndrome information. The number of bits in the syndrome equals the number of test links in the test graph; this can be extremely large in systems with thousands of processors, and can easily lead to traffic congestion when the syndrome is communicated to the host.

To carry out efficient fault diagnosis, we determine a covering code on the vertices (processors) such that every processor is covered by a unique set of codewords. We refer to the codewords as monitors. Every monitor tests itelf and all its neighboring processors and sends a single bit value 1 (0) to the host if it detects (does not detect) the presence of a fault in its ball. The number of bits in the syndrome is therefore equal to the number of monitors. Monitors must be selected such that by using balls of radius one ($t = 1$) centered at the monitors, we can diagnose processor faults in the system. An important design objective therefore is to minimize the number of monitors. In addition to minimizing the syndrome length, this offers another important advantage. Since the test program has to reside on the local memory of every monitor processor, optimal monitor selection also minimizes the amount of memory required to store the test program.

References

[1] K. Chakrabarty, M. G. Karpovsky and L. B. Levitin. Fault isolation and diagnosis in multiprocessor systems with point-to-point connections. To appear in *Fault-Tolerant Parallel and Distributed Systems*, D. R. Avresky, (ed.), Kluwer Academic Publishers, 1997.

[2] G. D. Cohen et al. Covering radius 1985–1994. Tech. report, Department Informatique, Ecole Nationale Superieure des Telecommunications, France, 1994.

[3] G. D. Cohen, M. G. Karpovsky, H. F. Mattson, Jr. and J. R. Shatz. Covering radius survey and recent results. *IEEE Transactions on Information Theory*, vol. IT-31, pp. 328–344, May 1985.

[4] G. D. Cohen, A. C. Lobstein and N. J. A. Sloane. Further results on the covering radius of codes. *IEEE Transactions on Information Theory*, vol. IT-32, pp. 680–694, September 1986.

[5] W. J. Dally et al. The message-driven processor: a multicomputer processing node with efficient mechanisms. *IEEE Micro*, vol. 12, pp. 23–39, April 1992.

[6] S. L. Hakimi and A. T. Amin. Characterization of the connection assignment of diagnosable systems. *IEEE Transactions on Computers*, vol. 23, pp. 86–88, January 1974.

[7] J. P. Hayes et al. A microprocessor-based hypercube supercomputer. *IEEE Micro*, vol. 6, pp. 6–17, October 1986.

[8] D. Jurasek, W. Richardson and D. Wilde. A multiprocessor design in custom VLSI. *VLSI Systems Design*, pp. 26–30, June 1986.

[9] G. M. Masson, G. M. Blough and G. F. Sullivan. System diagnosis. In *Fault-Tolerant Computer System Design*, D. K. Pradhan (ed.), Prentice-Hall, New Jersey, 1996.