On Embedding Hamiltonian Cycles in Crossed Cubes

Dajin Wang, *Member*, *IEEE Computer Society*

Abstract—We study the embedding of Hamiltonian cycle in the *Crossed Cube*, which is a prominent variant of the classical hypercube, obtained by crossing some straight links of a hypercube, and has been attracting much research interest in literatures since its proposal. We will show that due to the loss of link-topology regularity, generating Hamiltonian cycles in a crossed cube is a more complicated procedure than in its original counterpart. The paper studies how the crossed links affect an otherwise succinct process to generate a host of well-structured Hamiltonian cycles traversing all nodes. The condition for generating these Hamiltonian cycles in a crossed cube is proposed. An algorithm is presented that works out a Hamiltonian cycle for a given *link permutation*. The useful properties revealed and the algorithm proposed in this paper can find their way when system designers evaluate a candidate network's competence and suitability, balancing regularity and other performance criteria, in choosing an interconnection network.

Index Terms—Crossed cube, embedding, Hamiltonian cycles, interconnection architectures, network topology.

1 Introduction

The hypercube structure is a well-known interconnection model. As an important topology to interconnect multiprocessor systems, it has been proven to possess many attractive properties, and multiprocessor computers built with hypercube structure have been in existence for a long time. Since its introduction many years ago, numerous variants of hypercube have been proposed. One of the most notable among them, the *crossed cube*, was first proposed by Efe [6] and has attracted much attention in literatures [4], [7], [9], [19], [20], [24]. An n-dimensional crossed cube, denoted as CQ_n, is derived by "crossing" some links in an n-dimensional hypercube (n-cube for short). With exactly same hardware cost as hypercube, it has been shown that such a simple variation gains important benefits such as greatly reduced diameter.

There are many parallel and distributed algorithms developed using such regular data structure as linear arrays, rings, trees, and meshes. Their implementation on a hypercube-type interconnection network very often requires that a specific topology be mapped into the network. Of those commonly used topologies, Hamiltonian cycle is frequently formed among the nodes of the network. Being able to embed a Hamiltonian cycle in an interconnection network is very important: Besides its use in application algorithms, many primitive network tasks such as multicasting and broadcasting rely on establishing a path that traverses every node. For example, a Hamiltonian path can be used in dual-path and multipath multicast routing algorithms, so that congestions incurred by traditionally tree-based multicast are alleviated and deadlocks avoided [3], [21].

The Hamiltonianicity of various interconnection architectures has attracted the interest of many researchers, and interesting results have been continuing to appear in literatures [2], [11], [13], [14], [16], [17], [18]. In this paper, we study the problem of embedding a family of *regularly structured* Hamiltonian cycles in a crossed cube. Since the crossed cube shows performance improvement over a regular hypercube in many aspects, we are interested in knowing whether it has the comparable capability in terms of structure embedding—specifically, in this paper, the embedding of Hamiltonian cycles. It needs to be pointed out that we are only considering a family of Hamiltonian cycles that can be systematically constructed, characterized by the permutation of link dimensions (*link permutation* for short). The total number $h(n)$ of Hamiltonian cycles in a regular n-dimensional hypercube happens to be huge, and many of them cannot be constructed in a systematic way. The exact $h(n)$ has not been established for large n. For example, $h(3)$ is known to be 6, but $h(4)$ jumps to 1,344. For larger n, only bounds on $h(n)$ are known [5], [12].

As will be shown in the paper, the partial loss of the highly regular highly symmetrical topology in the hypercube does affect the network's capability in terms of Hamiltonian embedding. We will show that due to this loss of regularity in link-topology, an otherwise unified, systematic procedure to generate Hamiltonian cycles for all link permutations is now more complicated, and not every permutation can generate a Hamiltonian cycle as it would in the hypercube. The main contribution of this paper is a characterization of link permutations in a crossed cube that can facilitate Hamiltonian cycles. For those Hamiltonian facilitating permutations, we propose an algorithm that works out a well-structured Hamiltonian cycle.

It is worth noting that crossed cube and hypercube are structurally different. In fact, the very notion of "dimension" is not as well defined in crossed cube as in hypercube. However, crossed cube (and many other hypercube variants) originated from hypercube. Many ideas and techniques based on hypercube can still find their use in these variants. The results of this paper show that the notion of dimension, although somewhat blurred in crossed cube, can still be well exploited when we develop algorithms on crossed cube.

- *The author is with the Department of Computer Science, Montclair State University, Upper Montclair, NJ 07043. E-mail: wang@pegasus.montclair.edu.*

Manuscript received 20 Nov. 2006; revised 14 May 2007; accepted 13 June 2007; published online 28 June 2007.
Recommended for acceptance by B. Parhami.
For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number TPDS-0370-1106.
Digital Object Identifier no. 10.1109/TPDS.2007.70729.

Authorized licensed use limited to: Rutgers University. Downloaded on May 12, 2009 at 11:39 from IEEE Xplore. Restrictions apply.

The rest of this paper is organized as follows: In Section 2, we give a formal description of the crossed cube and define notations used in this paper, including notation for crossed links and link permutation; Section 2 also briefly reviews work in past literatures that are related to the work in this paper. Section 3.1 presents a succinct algorithm ($CQ\_HAMIL\_P_0$) that generates a Hamiltonian cycle in a crossed cube for a special link permutation. Through the presentation and proof of $CQ\_HAMIL\_P_0$, important insights are made that will afford hints for a more general version of the algorithm. In Section 3.2, we use an example to demonstrate an important implication of $CQ\_HAMIL\_P_0$. That is, at each round of $CQ\_HAMIL\_P_0$, the structure formed by the remaining links is actually a κ-connected subcrossed-cube for $n-1 \geq \kappa \geq 2$. In Section 4, we present the algorithm (CQ_HAMIL) to generate a Hamiltonian cycle with any link permutation in a crossed cube, provided that the permutation is Hamiltonian facilitating. In Section 5, we research the relationship of a link permutation and its Hamiltonianness and give a characterization of link permutations that will/will not facilitate Hamiltonian cycles. Section 6 gives some concluding remarks.

2 Preliminaries and Previous Work

2.1 Crossed Cube

Definition 1 [6]. *Let* $R = \{(00,00), (10,10), (01,11), (11,01)\}$. *An n-dimensional crossed cube, denoted* CQ_n, *is recursively defined as follows:*

1. $n = 1$. *CQ_1 consists of two nodes, labeled as "0" and "1," respectively, and they are connected by a link.*
2. $n = 2$. *CQ_2 consists of four nodes, labeled as "00," "01," "10," and "11"; any two nodes whose labels differ at exactly one bit are connected by a link. That is, the four links are {00, 01}, {00, 10}, {01, 11}, and {10, 11}.*
3. *For* $n \geq 3$, *CQ_n consists of two identical* $(n-1)$*-dimensional crossed cubes,* CQ^0_{n-1}, *and* CQ^1_{n-1}:
   - ***Node labeling.*** *The nodes of* CQ^0_{n-1} *are labeled* $0u_{n-1}\dots u_1$, *where* $u_{n-1}\dots u_1$ *are the labels of* CQ_{n-1} *nodes; the nodes of* CQ^1_{n-1} *are labeled* $1v_{n-1}\dots v_1$, *where* $v_{n-1}\dots v_1$ *are the labels of* CQ_{n-1} *nodes.*
   - ***New links between*** CQ^0_{n-1} ***and*** CQ^1_{n-1}. *If* $n-1$ *is even, then nodes* $(0u_{n-1}\dots u_1, 1v_{n-1}\dots v_1)$ *have a link if*

$$(u_{2i}u_{2i-1}, v_{2i}v_{2i-1}) \in R, \quad 1 \leq i \leq \frac{n-1}{2}. \tag{1}$$

If $n-1$ *is odd, then nodes* $(0u_{n-1}\dots u_1, 1v_{n-1}\dots v_1)$ *have a link if*

$$u_{n-1} = v_{n-1} \quad \text{and} \quad (u_{2i}u_{2i-1}, v_{2i}v_{2i-1}) \in R, \quad 1 \leq i \leq \frac{n\text{-}2}{2}. \tag{2}$$

To uniquely represent a link l in CQ_n, note that two nodes connected by a straight link have exactly one bit different (same as in the hypercube), whereas two nodes connected by a crossed link have multiple different bits. We denote a link l_i of dimension i (dim-i for short) using the two nodes it connects. We use an "x" or "$\bar{x}$" to represent a bit at which the two connected nodes have different (that is, complementary) bits. We use capital "X" at the "major bit" i for links of dim-i.

Definition 2. *Let* l_i *be the* dim-i *link that links two nodes* $u = u_n u_{n-1}\dots u_1$ *and* $v = v_n v_{n-1}\dots v_1$, *then* $l_i = e_n e_{n-1}\dots e_i \dots e_1$, *such that:*

1. *For* $i+1 \leq j \leq n$, $e_j = u_j = v_j$.
2. $e_i = X$.
3. *For* $1 \leq k \leq \lfloor \frac{i-1}{2} \rfloor$,
   - *if* $(u_{2k}u_{2k-1}, v_{2k}v_{2k-1}) \in \{(00,00), (10,10)\}$, *then* $e_{2k}e_{2k-1} = u_{2k}u_{2k-1} = v_{2k}v_{2k-1}$ *("straight component" of a link);*
   - *if* $(u_{2k}u_{2k-1}, v_{2k}v_{2k-1}) = (01,11)$ *and* $u_i = 0$, *then* $e_{2k}e_{2k-1} = x1$; *if* $(u_{2k}u_{2k-1}, v_{2k}v_{2k-1}) = (01,11)$ *and* $u_i = 1$, *then* $e_{2k}e_{2k-1} = \bar{x}1$ *("cross component" of a link); and*
   - *if* $(u_{2k}u_{2k-1}, v_{2k}v_{2k-1}) = (11,01)$ *and* $u_i = 0$, *then* $e_{2k}e_{2k-1} = \bar{x}1$; *if* $(u_{2k}u_{2k-1}, v_{2k}v_{2k-1}) = (11,01)$ *and* $u_i = 1$, *then* $e_{2k}e_{2k-1} = x1$ *("cross component" of a link).*
4. *For the case of* $(i-1)$ *being odd,* $e_{i-1} = u_{i-1} = v_{i-1}$.

If a link representation contains any cross component, it is said to be a crossed link. As an example, for a CQ_4, the links are denoted as follows (with the nodes they connect):

$$\begin{aligned}
\text{dim-1}: 0000 &\xleftrightarrow{000X} 0001, 0010 \xleftrightarrow{001X} 0011,\\
0100 &\xleftrightarrow{010X} 0101, 0110 \xleftrightarrow{011X} 0111,\\
1000 &\xleftrightarrow{100X} 1001, 1010 \xleftrightarrow{101X} 1011,\\
1100 &\xleftrightarrow{110X} 1101, 1110 \xleftrightarrow{111X} 1111.\\
\text{dim-2}: 0000 &\xleftrightarrow{00X0} 0010, 0001 \xleftrightarrow{00X1} 0011,\\
0100 &\xleftrightarrow{01X0} 0110, 0101 \xleftrightarrow{01X1} 0111,\\
1000 &\xleftrightarrow{10X0} 1010, 1001 \xleftrightarrow{10X1} 1011,\\
1100 &\xleftrightarrow{11X0} 1110, 1101 \xleftrightarrow{11X1} 1111.\\
\text{dim-3}: 0000 &\xleftrightarrow{0X00} 0100, 0001 \xleftrightarrow{0Xx1} 0111,\\
0010 &\xleftrightarrow{0X10} 0110, 0011 \xleftrightarrow{0X\bar{x}1} 0101,\\
1000 &\xleftrightarrow{1X00} 1100, 1001 \xleftrightarrow{1Xx1} 1111,\\
1010 &\xleftrightarrow{1X10} 1110, 1011 \xleftrightarrow{1X\bar{x}1} 1101.\\
\text{dim-4}: 0000 &\xleftrightarrow{X000} 1000, 0001 \xleftrightarrow{X0x1} 1011,\\
0010 &\xleftrightarrow{X010} 1010, 0011 \xleftrightarrow{X0\bar{x}1} 1001,\\
0100 &\xleftrightarrow{X100} 1100, 0101 \xleftrightarrow{X1x1} 1111,\\
0110 &\xleftrightarrow{X110} 1110, 0111 \xleftrightarrow{X1\bar{x}1} 1101.
\end{aligned}$$

2.2 Link Permutation

The host of Hamiltonian cycles concerned in this paper takes a well-structured format. Each one of them assigns the needed links in a well-patterned manner. For example, it may use all (that is, 2^{n-1}) links of dimension n, half (that is, 2^{n-2}) of the

Authorized licensed use limited to: Rutgers University. Downloaded on May 12, 2009 at 11:39 from IEEE Xplore. Restrictions apply.

links of dimension 1, and 1/4 (that is, 2^{n-3}) of the links of dimension 2, and so on. This assignment of links to dimensions is called the *permutation* of a Hamiltonian cycle. Formally, we use

$$P : [1..n] \mapsto [1..n-1]$$

to denote a permutation, where $P(i) = j$ represents that 2^j links of dimension i will be used in forming the Hamiltonian ring. Since a Hamiltonian ring traverses all nodes, we must have $\sum_{i=1}^{n} P(i) = 2^n$.

Clearly, there are $n!$ different permutations in total. In a regular hypercube, all permutations can generate a Hamiltonian ring, which is of essentially the same structure. By renaming the links, a unified procedure can be used to produce all of them. In the crossed cube, however, the crossing of links makes some permutations unable to form a Hamiltonian cycle, as will be shown later.

2.3 A Brief Review of Related Previous Work

Since its proposal, various properties of crossed cube have been studied by many researchers. In [1], Abuelrub and Bettayeb studied the embedding of rings of various lengths into a node-faulty crossed cube with up to 2^{n-3} faulty nodes. Using the idea first proposed by Efe and Fernandez [8] that *edge congestion* should be an important parameter for analyzing various properties of graphs, Chang et al. showed in [4] that crossed cube and hypercube have actually the same edge congestion. In [9], Fan showed that a CQ_n is n-diagnosable under a major diagnosis model—the comparison diagnosis model. In their most recent work, Fan et al. [10] studied the problem of optimal embedding of paths of different lengths between any two nodes in crossed cubes. It was proved that in a CQ_n, the paths of all lengths between $\lceil \frac{n+1}{2} \rceil + 1$ and $2^n - 1$ can be optimally (that is, with a dilation of 1) embedded between any two distinct nodes. Kulasinghe and Bettayeb [20] showed that the (2^n-1)-node complete binary tree can be embedded into the CQ_n with dilation 1.

The fault-tolerant cycle embedding of CQ_n has been separately studied in [15] and [23]. Both works considered faulty CQ_n with hybrid node and/or edge faults and proved that any cycle of length l $(4 \le l \le |V(CQ_n| - f_v)$ can be embedded into a faulty CQ_n with dilation 1, where f_v is the number of faulty nodes, and the total number of faults (nodes and edges) is less than $n-2$. In [24], Zheng and Latifi introduced the notion of *reflected link label sequence* and used it to define a *Generalized Gray Code* (GGC), which was used to determine whether a given sequence of links form a Hamiltonian path or cycle in a CQ_n. A scheme was also proposed to embed a cycle of arbitrary length into a CQ_n.

3 Hamiltonian Cycle Generation for a Special Link Permutation

To make some insightful observations that can help us devise an algorithm that works out a Hamiltonian cycle in a crossed cube, we will first make use of a known algorithm to generate a Hamiltonian cycle for a special permutation. The algorithm was originally used to generate Hamiltonian cycles in a regular hypercube [22]. Renamed as $CQ\_HAMIL\_P_0$ in this paper, we will show that it can be directly used to produce a Hamiltonian cycle for the given permutation.

For the purpose of our algorithm, we list the $n2^{n-1}$ links of CQ_n in a specific pattern, that is, the $n2^{n-1}$ are listed in n columns, with column i containing all links of dim-i. The first column lists the 2^{n-1} links in the following "increasing" order:

$$\begin{array}{c} 000\ldots00X \\ 000\ldots01X \\ \cdots\cdots\cdots \\ \underbrace{111\ldots11X}_{n}. \end{array}$$

The ith column, $2 \le i \le n$, is obtained by first left-rotating one bit, including the X-bit, for all links in the $(i-1)$th column. For example, for CQ_4, we first get

$dim-4$	$dim-3$	$dim-2$	$dim-1$
$X000$	$0X00$	$00X0$	$000X$
$X001$	$1X00$	$01X0$	$001X$
$X010$	$0X01$	$10X0$	$010X$
$X011$	$1X01$	$11X0$	$011X$
$X100$	$0X10$	$00X1$	$100X$
$X101$	$1X10$	$01X1$	$101X$
$X110$	$0X11$	$10X1$	$110X$
$X111$	$1X11$	$11X1$	$111X$

Then, according to the structure rule of the crossed cube, for each dim-i, $i \ge 3$, perform the following bit-turning operation: For $1 \le k \le \lfloor \frac{i-1}{2} \rfloor$, turn "$0_{2k}1_{2k-1}$" into "$x_{2k}1_{2k-1}$"; turn "$1_{2k}1_{2k-1}$" into "$\bar{x}_{2k}1_{2k-1}$." The resulting link listing of CQ_4 is given below:

$dim-4$	$dim-3$	$dim-2$	$dim-1$
$X000$	$0X00$	$00X0$	$000X$
$X0x1$	$1X00$	$01X0$	$001X$
$X010$	$0Xx1$	$10X0$	$010X$
$X0\bar{x}1$	$1Xx1$	$11X0$	$011X$
$X100$	$0X10$	$00X1$	$100X$
$X1x1$	$1X10$	$01X1$	$101X$
$X110$	$0X\bar{x}1$	$10X1$	$110X$
$X1\bar{x}1$	$1X\bar{x}1$	$11X1$	$111X$

The listing of all links of CQ_5 is

$dim-5$	$dim-4$	$dim-3$	$dim-2$	$dim-1$
$X0000$	$0X000$	$00X00$	$000X0$	$0000X$
$X00x1$	$1X000$	$01X00$	$001X0$	$0001X$
$X0010$	$0X0x1$	$10X00$	$010X0$	$0010X$
$X00\bar{x}1$	$1X0x1$	$11X00$	$011X0$	$0011X$
$Xx100$	$0X010$	$00Xx1$	$100X0$	$0100X$
$Xx1x1$	$1X010$	$01Xx1$	$101X0$	$0101X$
$Xx110$	$0X0\bar{x}1$	$10Xx1$	$110X0$	$0110X$
$Xx1\bar{x}1$	$1X0\bar{x}1$	$11Xx1$	$111X0$	$0111X$
$X1000$	$0X100$	$00X10$	$000X1$	$1000X$
$X10x1$	$1X100$	$01X10$	$001X1$	$1001X$
$X1010$	$0X1x1$	$10X10$	$010X1$	$1010X$
$X10\bar{x}1$	$1X1x1$	$11X10$	$011X1$	$1011X$
$X\bar{x}100$	$0X110$	$00X\bar{x}1$	$100X1$	$1100X$
$X\bar{x}1x1$	$1X110$	$01X\bar{x}1$	$101X1$	$1101X$
$X\bar{x}110$	$0X1\bar{x}1$	$10X\bar{x}1$	$110X1$	$1110X$
$X\bar{x}1\bar{x}1$	$1X1\bar{x}1$	$11X\bar{x}1$	$111X1$	$1111X$

Using this listing of links, the algorithm $CQ\_HAMIL\_P_0$ presented below constructs a Hamiltonian cycle by selectively removing $(n-2)2^{n-1}$ links from a CQ_n, so that the remaining $2 \cdot 2^{n-1} = 2^n$ links just construct a Hamiltonian cycle, which is effectively a Hamiltonian cycle for the following link permutation:

Authorized licensed use limited to: Rutgers University. Downloaded on May 12, 2009 at 11:39 from IEEE Xplore. Restrictions apply.

CQ_4

dim-4	*dim*-3	*dim*-2	*dim*-1
$X000$	$0X00$		
$X0x1$			
$X010$			
$X0\bar{x}1$			
$X100$	$0X10$	$00X1$	$100X$
$X1x1$		$01X1$	$101X$
$X110$			$110X$
$X1\bar{x}1$			$111X$

CQ_5

dim-5	*dim*-4	*dim*-3	*dim*-2	*dim*-1
$X0000$	$0X000$			
$X00x1$				
$X0010$				
$X00\bar{x}1$				
$Xx100$				
$Xx1x1$				
$Xx110$				
$Xx1\bar{x}1$				
$X1000$	$0X100$	$00X10$	$000X1$	$1000X$
$X10x1$		$01X10$	$001X1$	$1001X$
$X1010$			$010X1$	$1010X$
$X10\bar{x}1$			$011X1$	$1011X$
$X\bar{x}100$				$1100X$
$X\bar{x}1x1$				$1101X$
$X\bar{x}110$				$1110X$
$X\bar{x}1\bar{x}1$				$1111X$

Fig. 1. Links forming Hamiltonian cycles in CQ_4 and CQ_5. Note that the link assignment in both cases is as specified in P_0.

$$P_0(i) = \begin{cases} 1 & i = n-1 \\ n-1-(i \bmod n) & \text{otherwise.} \end{cases}$$

3.1 Hamiltonian Cycle Generation for P_0

$CQ\_HAMIL\_P_0$ takes as input the complete link-set of a CQ_n, listed in n columns, as shown above. The 2^{n-1} links in each column are referred as the first link, second link, ..., in top-down order.

ALGORITHM $CQ\_HAMIL\_P_0$

{Purpose: Remove $(n-2)2^{n-1}$ links from a CQ_n to induce a Hamiltonian cycle}
Input: The complete link set of CQ_n listed in n columns
Output: The remaining 2^n links forming a Hamiltonian cycle

```
for (j = 1 to n-2)
{
    at column j,
        for i = 1 to 2^{n-j-1}
        { remove the ith link }
    at column j+1,
        for i = 2^{n-j-2} + 1 to 2^{n-2}
        { remove the ith link }
    at column j+1,
        for i = 2^{n-2} + 1 + 2^{n-j-2} to 2^{n-1}
        { remove the ith link }
}                                   /* end_for */
```

Applying $CQ\_HAMIL\_P_0$ to a 4-cube and a 5-cube, respectively, the Hamiltonian link sets generated by $CQ\_HAMIL\_P_0$ are shown below in Fig. 1. Fig. 2 shows the Hamiltonian cycle $CQ\_HAMIL\_P_0$ generates for CQ_4. Note that both Hamiltonian link sets in Fig. 1 implement permutation P_0.

In the remainder of this section, we will first prove a property of the remaining links after running $CQ\_HAMIL\_P_0$. We then prove that these links indeed form a Hamiltonian cycle traversing all 2^n nodes.

Lemma 1. *Of the remaining links after $CQ\_HAMIL\_P_0$, for any two links l_i of* dim-i *and l_j of* dim-j*, where $i \neq j$ and $i, j \in \{1, 2, \ldots, n-1\}$, l_i and l_j do not share a same node.*

Proof. In each round of Algorithm $CQ\_HAMIL\_P_0$, exactly 2^{n-1} links are removed. Among them, 2^{n-j-1} links are removed from dimension j and $2^{n-1} - 2^{n-j-1}$ from dimension $j+1$. After $n-2$ rounds, the remaining links follow the pattern described below:

- **Upper half links.** In dim-1 through dim-$(n-2)$, all links in the upper half are removed; in dim-$(n-1)$,

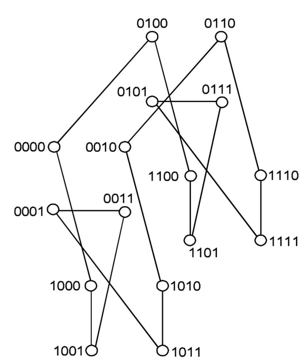
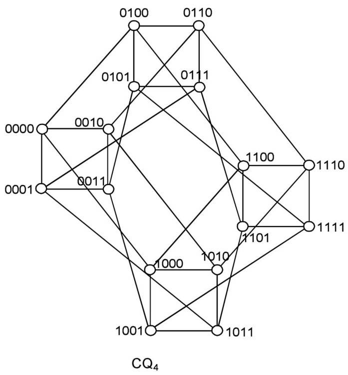


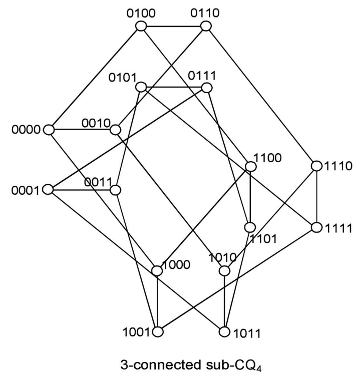
Fig. 2. Hamiltonian cycle induced from CQ_4.

Authorized licensed use limited to: Rutgers University. Downloaded on May 12, 2009 at 11:39 from IEEE Xplore. Restrictions apply.



dim-4	*dim*-3	*dim*-2	*dim*-1
$X000$	$0X00$	$00X0$	$000X$
$X0x1$	$1X00$	$01X0$	$001X$
$X010$	$0Xx1$	$10X0$	$010X$
$X0\bar{x}1$	$1Xx1$	$11X0$	$011X$
$X100$	$0X10$	$00X1$	$100X$
$X1x1$	$1X10$	$01X1$	$101X$
$X110$	$0X\bar{x}1$	$10X1$	$110X$
$X1\bar{x}1$	$1X\bar{x}1$	$11X1$	$111X$

Fig. 3. A CQ_4 and its link set.



dim-4	*dim*-3	*dim*-2	*dim*-1
$X000$	$0X00$	$00X0$	
$X0x1$	$1X00$	$01X0$	
$X010$	$0Xx1$		
$X0\bar{x}1$	$1Xx1$		
$X100$	$0X10$	$00X1$	$100X$
$X1x1$	$1X10$	$01X1$	$101X$
$X110$	$0X\bar{x}1$		$110X$
$X1\bar{x}1$	$1X\bar{x}1$		$111X$

Fig. 4. The CQ_4 after running $CQ\_HAMIL\_P_0$ for $j = 1$. 2^{n-1} links have been removed. The graph is 3-connected.

only the first link $(0X00\ldots0)$ remains in the upper half.

- **Lower half links.** The third **for**-loop in $CQ\_HAMIL\_P_0$ removes links in the lower half. The first round removes 2^{n-3} links from the bottom of dim-2; the second round removes $2^{n-3} + 2^{n-4} = 3 \cdot 2^{n-4}$ links from the bottom of dim-3; the third $2^{n-3} + 2^{n-4} + 2^{n-5} = 7 \cdot 2^{n-5}$ links from dim-4, and so on. The layout of the remaining lower half links (upto dim-$(n-1)$) looks like an upside-down staircase.

Two example results for CQ_4 and CQ_5 are shown in Fig. 1. Because of the way the links are originally listed, the following properties of the remaining links are noticed (refer to examples in Fig. 1 to help make the observation): All links in dim-1 have their bit-n being "1," whereas all links in dim-i, $2 \leq i \leq n-1$, have their bit-n being "0." Therefore, dim-1 links share no nodes with any dim-i links, $2 \leq i \leq n-1$.

Links in dim-j, $2 \leq j \leq n-2$: All links in dim-j have their bit-$(j-1)$ being "1," whereas all links in dim-i, $j+1 \leq i \leq n-1$, have their bit-$(j-1)$ being "0." Therefore, dim-j links share no nodes with any dim-i links, $j+1 \leq i \leq n-1$. □

Theorem 1. *Algorithm $CQ\_HAMIL\_P_0$ generates a Hamiltonian cycle that traverses all 2^n nodes of a CQ_n.*

Proof. The proof for Theorem 1 is provided in the Appendix. □

$CQ\_HAMIL\_P_0$ works out a Hamiltonian cycle for a special permutation, P_0. In a regular hypercube, the same procedure can be used to generate a Hamiltonian cycle for *all* permutations. However, as pointed out earlier, in a CQ_n, not all permutations can generate a Hamiltonian cycle. In Section 4, we will present an extended more complex version of $CQ\_HAMIL\_P_0$ that can produce a cycle for *any* link permutation, provided the permutation is Hamiltonian facilitating. Then, in Section 5, we will characterize the permutations that can or cannot generate a Hamiltonian cycle in CQ_n.

3.2 A More General Implication of $CQ\_HAMIL\_P_0$

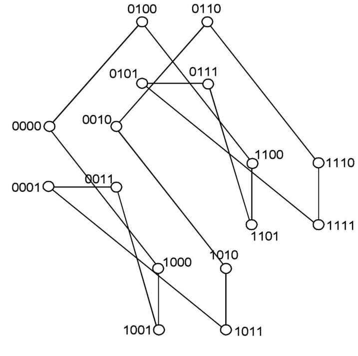
The **for**-loop of $CQ\_HAMIL\_P_0$ runs $(n-2)$ rounds to arrive at a Hamiltonian cycle *for a particular link permutation* (that is, P_0). The cycle can also be viewed as a 2-connected subnetwork of CQ_n. It is worth pointing out a more general implication of $CQ\_HAMIL\_P_0$. That is, at *each round* of the **for**-loop of $CQ\_HAMIL\_P_0$, the structure formed by the remaining links is actually a κ-connected subnetwork, $n-1 \geq \kappa \geq 2$, which contains all 2^n nodes of CQ_n and preserves the symmetrical structure. That makes the Hamiltonian cycle generated by $CQ\_HAMIL\_P_0$ a special case of a more general procedure.

However, as has been emphasized, $CQ\_HAMIL\_P_0$ only generates one particular cycle for one particular link permutation. The objective of this paper is to characterize *all* link permutations that facilitate Hamiltonianness, and devise an algorithm to generate Hamiltonian cycles for *any* facilitating permutation. In this section, we will just demonstrate this general implication of $CQ\_HAMIL\_P_0$ through an example CQ_4. A formal proof that it holds true for CQ_n of any n is another major undertaking and is beyond the range of this paper.

Before applying $CQ\_HAMIL\_P_0$, CQ_4 and its complete link set are shown in Fig. 3.

After the first round of the **for**-loop of $CQ\_HAMIL\_P_0$ (that is, $j = 1$), the remaining link set and the corresponding incomplete CQ_4 are shown in Fig. 4. It is easy to verify that it is a 3-connected graph.

When $j = 2$, another 2^{n-1} links are removed by $CQ\_HAMIL\_P_0$, resulting in the link set and its corresponding incomplete CQ_4 shown in Fig. 5. The remaining links form a cycle that traverses all nodes, and the graph is 2-connected.



dim-4	*dim*-3	*dim*-2	*dim*-1
$X000$	$0X00$		
$X0x1$			
$X010$			
$X0\bar{x}1$			
$X100$	$0X10$	$00X1$	$100X$
$X1x1$		$01X1$	$101X$
$X110$			$110X$
$X1\bar{x}1$			$111X$

Fig. 5. The CQ_4 after running $CQ\_HAMIL\_P_0$ for $j = 2$. $2 \times 2^{n-1}$ links have been removed. The graph is 2-connected.

Authorized licensed use limited to: Rutgers University. Downloaded on May 12, 2009 at 11:39 from IEEE Xplore. Restrictions apply.

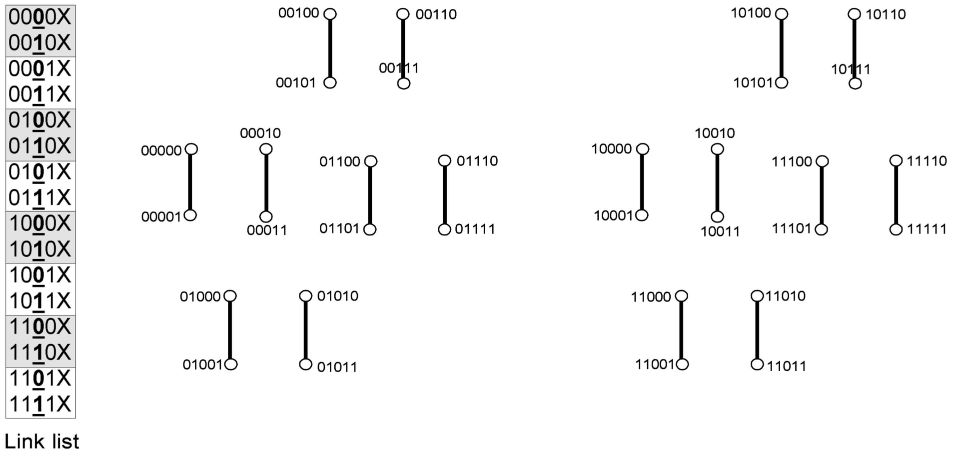


Fig. 6. The 16 links of dim-1. They form 16 disjoint 2-node subcubes.

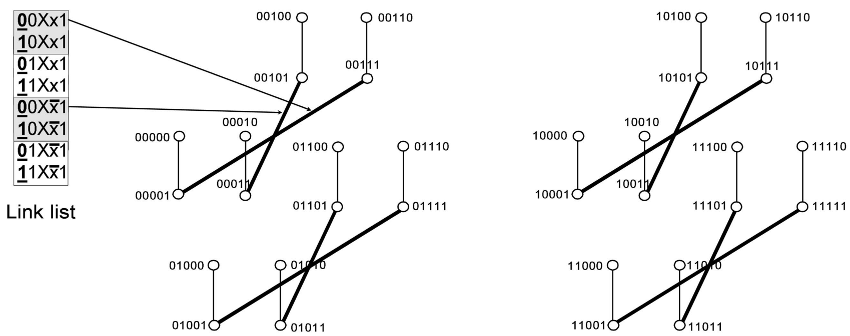


Fig. 7. The eight (crossed) links of dim-3 are shown in bold lines. They merge the 2-node subcubes into disjoint 4-node sub-Hamiltonian paths.

4 The Algorithm to Generate a Hamiltonian Cycle with Arbitrary Permutations

The succinctness and simple form of $CQ\_HAMIL\_P_0$ is due to the special link permutation P_0. If permutations are more "arbitrary" than P_0, the Hamiltonian cycle generation becomes less concise a procedure. In this section, we will present an algorithm that produces a Hamiltonian cycle with any permutation that facilitates Hamiltonianness.

We name the algorithm CQ_HAMIL, naturally after $CQ\_HAMIL\_P_0$. The main idea of the algorithm is to emulate the steps of $CQ\_HAMIL\_P_0$, that is, adding links of a particular dimension at a time, so that bigger and bigger disjoint sub-Hamiltonian paths are formed, as demonstrated in the proof of Theorem 1. The choice/listing of links in each dimension will follow a pattern similar to that in Fig. 18 (in the Appendix, the proof for Theorem 1). For an easy grasp of the points of algorithm CQ_HAMIL, we will proceed in a somewhat unconventional order: We will first go over the steps of CQ_HAMIL with an example permutation; we will then describe the algorithm in a formal way.

4.1 An Example

Let us take an example of CQ_5 and suppose the permutation P of

$$P(1) = 4 \text{ (All 16 links)},$$
$$P(3) = 3 \text{ (8 links)},$$
$$P(5) = 2 \text{ (4 links)},$$
$$P(2) = 1 \text{ (2 links)},$$
$$P(4) = 1 \text{ (2 links)}.$$

4.1.1 dim-1 Links

According to permutation P, all links of dim-1 are to be used. They will form 16 disjoint 2-node subcubes, as illustrated in Fig. 6. The used links are listed on the left in a special pattern: Since dim-3 links are to be chosen next, we group the current links in such a way that two links that differ only at bit-3 are put together. By "merging" these two links into one, we obtain a link that we need in the next step.

4.1.2 dim-3 Links

Refer to that in Fig. 7. The dim-3 links are obtained from the link-set of the previous step. First, at bit-1, change all "X" to "1." Then, merge all pair of links, which differ only at bit-3. Lastly, turn certain bits (to the right of bit-3) into either x or $\bar{x}$, according to the link representation defined in Definition 2. For example, merging links "$00\underline{0}0X$"and "$00\underline{1}0X$," we first obtain "$00X01$," and then, we turn the "01" right to bit-3 into "$x1$," finally, obtaining a needed link $\mathbf{00}\boldsymbol{Xx}\mathbf{1}$, as pointed out in Fig. 7. Similarly, merging links "$00\underline{0}1X$"and "$00\underline{1}1X$," we obtain "$00X11$"; then by turning the "11" right to bit-3 into "$\bar{x}1$," we get a needed link $\mathbf{00}\boldsymbol{X\bar{x}}\mathbf{1}$, also pointed out in Fig. 7.

Authorized licensed use limited to: Rutgers University. Downloaded on May 12, 2009 at 11:39 from IEEE Xplore. Restrictions apply.

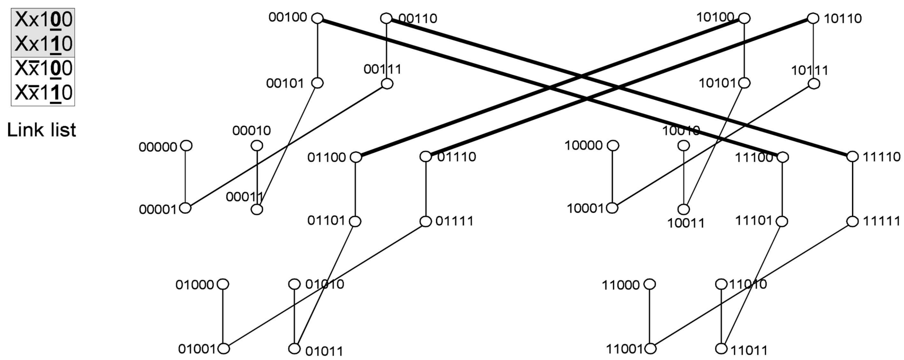


Fig. 8. The four links of dim-5 are shown in bold lines. They merge the 4-node subcubes into disjoint 8-node sub-Hamiltonian paths.

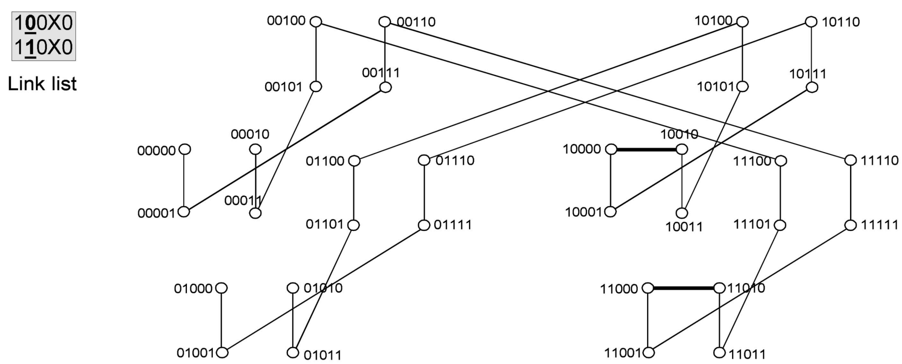


Fig. 9. The two links of dim-2 are shown in bold lines, merging the 8-node subcubes into disjoint 16-node sub-Hamiltonian paths.

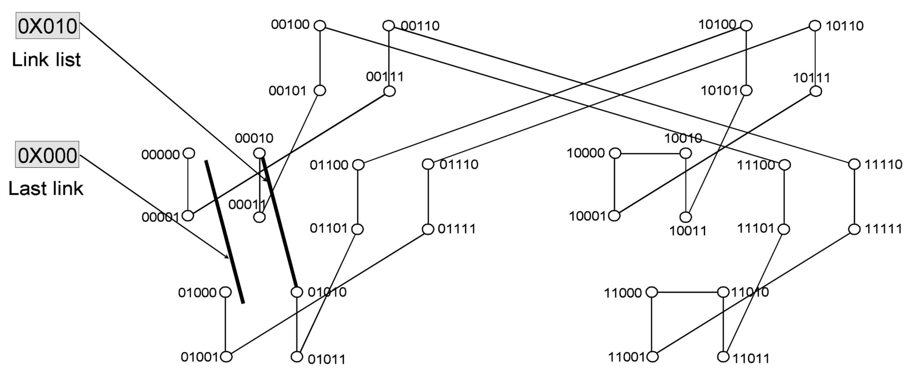


Fig. 10. Link $0X010$ will give an all-node Hamiltonian path; adding $0X000$ completes the construction of a Hamiltonian cycle.

All consecutive pairs in Fig. 6 are merged, as described above to generate the eight links in dim-3. We put together two links that differ only at bit-5, since the next dimension of links to be added is of dim-5.

4.1.3 dim-5 Links

Similar merge procedure as in the last step is performed. First, for crossed links, change back those $x/\bar{x}$ bits. Then, merge the pair of links that are grouped together.

Rule of bit-change in the merged links:

- The bit of immediately previous dimension (that is, dim-3) is changed from "X" to "1" ($X \rightarrow 1$).
- The bit that is "1" in last step (that is, dim-1) now assumes "0" ($1 \rightarrow 0$).
- The bit that is "0" in last step remains "0" ($0 \rightarrow 0$).

After the bit-change, turn certain bits (to the right of bit-5) into either x or $\bar{x}$ due to the crossed link effect.

Four links of dim-5 are generated, as shown in Fig. 8. Links that differ only at bit-2 are put together because the next dimension of links to be added is of dim-2.

4.1.4 dim-2 Links

The same procedure as in the last step. Two links of dim-2 added, as shown in Fig. 9.

4.1.5 dim-4 Links

Merging the two links from previous step produces link $0X010$, which will complete a 32-node Hamiltonian path, as can be seen in Fig. 10. We need one more "last link" for a Hamiltonian cycle. That last link is obtained by switching "1" to "0" at the dimension of the previous step. In this example, it is bit-2. Thus, the last link is $0X000$, as illustrated in Fig. 10.

4.1.6 Algorithm Correctness

A complete proof would be very similar to that of Theorem 1. Therefore, we will just highlight the main

Authorized licensed use limited to: Rutgers University. Downloaded on May 12, 2009 at 11:39 from IEEE Xplore. Restrictions apply.

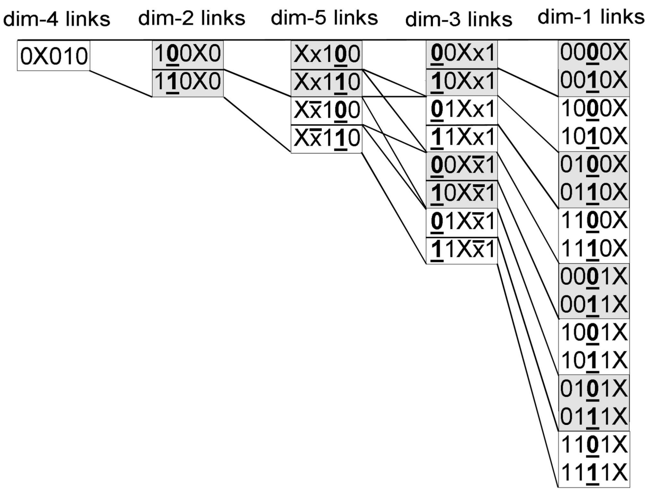


Fig. 11. Merging to produce links for the next dimension in the permutation.

points. Refer to Fig. 11, and compare it with Fig. 18. It can be observed that the two link-merge procedures are performing essentially the same operations, except in different order of dimensions. Dim-1 (dim-3, dim-5, etc.) links in Fig. 11 play the roles of dim-n (dim-1, dim-2, etc.) links in the proof of Theorem 1. Dim-1 links first form the 16 2-node subcubes (also sub-Hamiltonian paths). Then, the links of subsequent dimensions are added, one dimension at a time. It is fairly obvious to see that the new links (that is, links of next dimension) are chosen/grouped to guarantee the following critical properties:

- The newly added dimension of links does not share nodes with links of any previous dimensions, except that of dim-1 (property of Lemma 1). This is guaranteed by bit-change rules, as stated in the step of dim-5 **links** ("$X \rightarrow 1$," "$1 \rightarrow 0$," and "$0 \rightarrow 0$" rules).
- Every newly added link merges two sub-Hamiltonian paths to form a bigger (size doubled) sub-Hamiltonian path.

Maintaining the above two properties, an essentially the same procedure as demonstrated in the proof of Theorem 1 will be effected; the final round will produce an all-node Hamiltonian path; and finally, the "last link" completes the Hamiltonian cycle.

4.2 Algorithm Description

Having gone through the steps of CQ_HAMIL with an example, we now describe the algorithm in general terms. As we will show in Section 5, not all permutations can facilitate a Hamiltonian cycle. Therefore, the assumption of the algorithm is that the input permutation is indeed Hamiltonian facilitating.

ALGORITHM CQ_HAMIL

The algorithm specifies the links at each dimension of the given permutation. Let the Hamiltonian facilitating permutation P be

$$\{P(d_1) = n{-}1, P(d_2) = n{-}2, P(d_3) = n{-}3, \cdots\cdots, P(d_{n-1}) = 1, P(d_n) = 1\}.$$

dim-d_1 **links**

All 2^{n-1} links of dim-d_1 are used. To prepare for the next dimension in the permutation (dim-d_2), we match links in such a way that two links are put together if they differ only at bit-d_2. There may be two different cases:

- Two links that are exactly the same except at bit-d_2, where they have "0" or "1," respectively, are put together.
- For $d_2 < d_1$, d_2 even, and bit-$(d_2 - 1) = 1$, two links that are exactly the same except at bit-d_2, where they have "x" or "$\bar{x}$," respectively, are put together.

dim-d_2 **links**

Every pair of put-together links from last step will generate one dim-d_2 link by the following procedure **merge**$(d_1 \rightarrow d_2)$.

merge$(d_1 \rightarrow d_2)$
For *all pairs* of dim-d_1 links, do {

1. take either link of the pair; set bit-d_2 to "X";
2. set bit-d_1 to "1";
3. for all bits $e_{2k}e_{2k-1}$ such that $d_2 > 2k$,
   if $e_{2k}e_{2k-1} =$ "01", set $e_{2k}e_{2k-1}$ to "$x1$";
   if $e_{2k}e_{2k-1} =$ "11", set $e_{2k}e_{2k-1}$ to "$\bar{x}1$";
4. for all bits $e_{2k}e_{2k-1}$ such that $2k - 1 > d_2$,
   if $e_{2k}e_{2k-1} =$ "$x1$", set $e_{2k}e_{2k-1}$ back to "01";
   if $e_{2k}e_{2k-1} =$ "$\bar{x}1$", set $e_{2k}e_{2k-1}$ back to "11."

}

The result of running **merge**$(d_1 \rightarrow d_2)$ is the set of dim-d_2 links needed to construct the Hamiltonian cycle for the given permutation. They are grouped together according to bit-d_3, the same way as described in the previous step.

The **merge** operation for the subsequent dimensions is almost the same as that of dim-d_2, except at step 2.

dim-d_i **links** $(3 \leq i \leq n{-}1)$
merge$(d_{i-1} \rightarrow d_i)$
For *all pairs* of dim-d_{i-1} links, do {

1. take either link of the pair; set bit-d_i to "X";
2. a. set bit-d_{i-1} to "1";
   b. set bits $d_{i-2}, d_{i-3}, \ldots, d_1$ to "0";
3. for all bits $e_{2k}e_{2k-1}$ such that $d_i > 2k$,
   if $e_{2k}e_{2k-1} =$ "01", set $e_{2k}e_{2k-1}$ to "$x1$";
   if $e_{2k}e_{2k-1} =$ "11", set $e_{2k}e_{2k-1}$ to "$\bar{x}1$";
4. for all bits $e_{2k}e_{2k-1}$ such that $2k - 1 > d_i$,
   if $e_{2k}e_{2k-1} =$ "$x1$", set $e_{2k}e_{2k-1}$ back to "01";
   if $e_{2k}e_{2k-1} =$ "$\bar{x}1$", set $e_{2k}e_{2k-1}$ back to "11."

}

merge$(d_{i-1} \rightarrow d_i)$ produces dim-d_i Hamiltonian links, which are then grouped together according to bit-d_{i+1}.

Authorized licensed use limited to: Rutgers University. Downloaded on May 12, 2009 at 11:39 from IEEE Xplore. Restrictions apply.

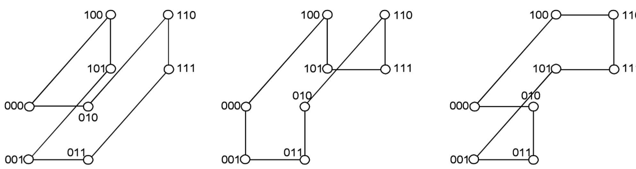


Fig. 12. In a regular three-dimensional hypercube, three different permutations and their respective Hamiltonian cycles. By turning them around, they can become of exactly the same structure.

$dim\text{-}d_n$ **links**

merge$(d_{n-1} \rightarrow d_n)$ will give one $dim\text{-}d_n$ Hamiltonian link, which must be of the form

$$0\ldots0\mathbf{X_{d_n}}0\ldots01_{\mathbf{d_{n-1}}}0\ldots0 \quad \text{or} \quad 0\ldots01_{\mathbf{d_{n-1}}}0\ldots0\mathbf{X_{d_n}}0\ldots0$$

That is, "X" at bit-d_n, and "1" at bit-d_{n-1}. The "Last link" to complete the Hamiltonian cycle is then, respectively

$$0\ldots0\mathbf{X_{d_n}}0\ldots0\mathbf{0_{d_{n-1}}}0\ldots0 \quad \text{or} \quad 0\ldots0\mathbf{0_{d_{n-1}}}0\ldots0\mathbf{X_{d_n}}0\ldots0.$$

5 Hamiltonian Facilitating Permutations in Crossed Cubes

In this section, we will characterize link permutations that facilitate Hamiltonian cycles in a crossed cube CQ_n.

In a regular hypercube, all permutations of links can produce a Hamiltonian cycle, and their structures are essentially the same. $CQ\_HAMIL\_P_0$ would be applied to any given permutation; all that needs to be done before applying $CQ\_HAMIL\_P_0$ is to rename the nodes (and therefore, the links) to accommodate to the given permutation. In Fig. 12, cycles are shown using three different permutations.

In a crossed cube, however, due to the loss of some regularity in link topology, not all permutations can facilitate a Hamiltonian cycle. For a simple example CQ_3, the Hamiltonian cycle produced by $CQ\_HAMIL\_P_0$ is as shown in Figs. 13a and 13b. If you would like to use all links of dim-1 (that is, for a permutation P such that $P(1) = 2$), a Hamiltonian cycle can still be generated (Fig. 13c). However, if you would like to use all links of dim-2 (that is, for a permutation P such that $P(2) = 2$), then no matter how you arrange the links in other dimensions, a Hamiltonian cycle will not be formed, as can be observed in Fig. 13d.

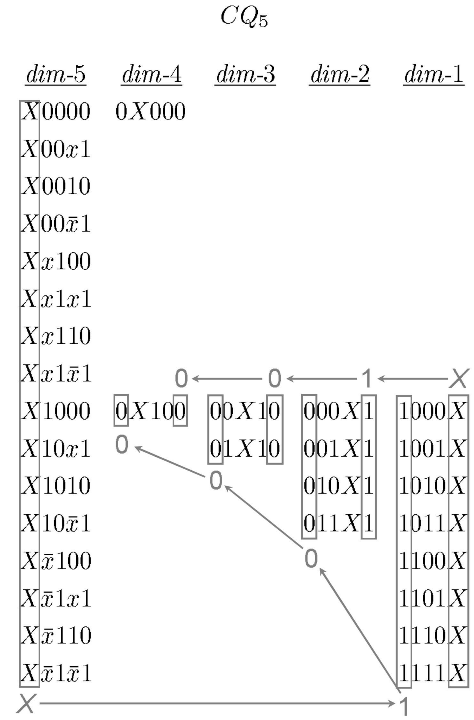


Fig. 14. The pattern of links in a Hamiltonian facilitating permutation.

Therefore, what is characteristic of a permutation that makes it Hamiltonian facilitating? The most important property of a Hamiltonian facilitating permutation is that of Lemma 1, where the links in $\dim\text{-}p_2$ through $\dim\text{-}p_n$ are not touching each other. That means, every one of them touches only with $\dim\text{-}p_1$ links at both ends.

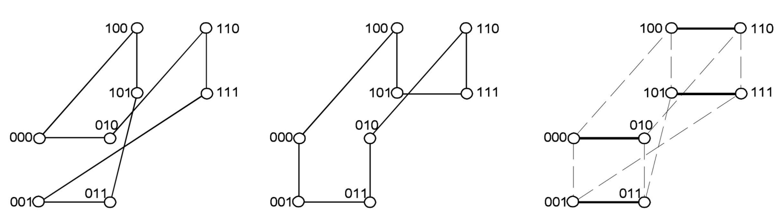
Take the example of CQ_5 in Fig. 1, where $d_1 = n, d_2 = 1, d_3 = 2, \ldots$ Since $P(n) = n-1$, which means that all $\dim\text{-}n$ links are present, each node in dim-1 through $\dim\text{-}(n-1)$ will have exactly degree 2, a necessary condition for Hamiltonianness. We refer to the link set of CQ_5 in Fig. 14 to characterize its pattern.

The permutation of the link set in Fig. 14 is

$$\begin{aligned} P_0(5) &= 4 \text{ (All 16 links)}, \\ P_0(1) &= 3 \text{ (8 links)}, \\ P_0(2) &= 2 \text{ (4 links)}, \\ P_0(3) &= 1 \text{ (2 links)}, \\ P_0(4) &= 1 \text{ (2 links)}. \end{aligned}$$

dim-3	dim-2	dim-1
$X00$	$0X0$	
$Xx1$		
$X10$	$0X1$	$10X$
$X\bar{x}1$		$11X$

(a)



(b) (c) (d)

Fig. 13. (a) and (b) The Hamiltonian cycle in CQ_3 produced by $CQ\_HAMIL\_P_0$, (c) using all links of dim-1, a Hamiltonian cycle can still be formed, and (d) using all links of dim-2, a Hamiltonian cycle will not be formed no matter how you arrange the remaining links.

Authorized licensed use limited to: Rutgers University. Downloaded on May 12, 2009 at 11:39 from IEEE Xplore. Restrictions apply.

In dim-5, bit-5 is of X, as it should be. In the following dimensions 1, 2, 3, and 4, bit-5 is of 1, 0, 0, and 0, respectively. Therefore, in this Hamiltonian-facilitating link set, the change of bit-5 takes the pattern of $(X \to 1 \to 0 \to 0 \to 0)$ for all links. In dim-1, the dimension that has the second-most links, bit-1 is of X; and then, in the following dimensions 2, 3, and 4, it is of 1, 0, and 0. The change of bit-1 takes the pattern of $(X \to 1 \to 0 \to 0)$. To list them all:

$$\begin{aligned}
&\text{bit}-5: && (X \to 1 \to 0 \to 0 \to 0),\\
&\text{bit}-1: && (X \to 1 \to 0 \to 0),\\
&\text{bit}-2: && (X \to 1 \to 0),\\
&\text{bit}-3: && (X \to 1)\\
& && (\text{not counting the special ``last link'' } 0X000),\\
&\text{bit}-4: && (X).
\end{aligned}$$

In the example of CQ_5 in Fig. 11, with which we went over algorithm CQ_HAMIL, the exactly same patter also demonstrates

$$\begin{aligned}
&\text{bit}-1: && (X \to 1 \to 0 \to 0 \to 0),\\
&\text{bit}-3: && (X \to 1 \to 0 \to 0),\\
&\text{bit}-5: && (X \to 1 \to 0),\\
&\text{bit}-2: && (X \to 1),\\
&\text{bit}-4: && (X).
\end{aligned}$$

It is this bit-pattern of $(X \to 1 \to 0 \to 0 \to \cdots \to 0)$ of links that guarantees the property of Lemma 1; for otherwise, if some 0-bit were turned to a "x" (or 1-bit to "$\bar{x}$") due to crossed link, it would imply that there would be a node touched by more than two links (consequently, there would be another node touched by less than two links), violating the property of Lemma 1.

Observing the small example of CQ_3 again, for the Hamiltonian cycle in Fig. 13c, the link set is

$\underline{dim-3}$	$\underline{dim-2}$	$\underline{dim-1}$
$X10$	$0X1$	$00X$
	$1X1$	$01X$
$X00$		$10X$
		$11X$

$$\begin{aligned}
&\text{bit}-1: && (X \to 1 \to 0)\\
&\text{bit}-2: && (X \to 1)\\
&\text{bit}-3: && (X)
\end{aligned}$$

which preserves the required bit pattern.

However, if we try to list the links in Fig. 13d, making an effort to keep the pattern, we would have a situation either in Fig. 15a or in 15b. In both cases, the x- or $\bar{x}$-bit of a crossed link will cause the violation of the pattern of $(X \to 1 \to 0 \to \cdots \to 0)$, which will then cause the violation of Lemma 1.

Theorem 2. *In a CQ_n, let $d_1, d_2, \ldots, d_n$ represent the n dimensions. Let a permutation P be such that*

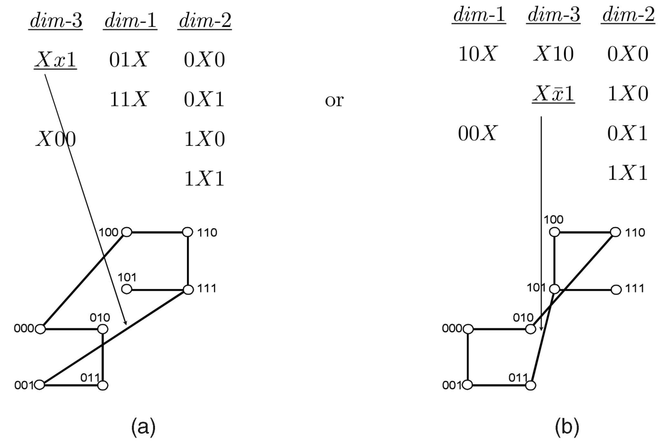


Fig. 15. (a) The bit-pattern at bit-2 is now $(X \to 1 \to x)$. Crossed link "$Xx1$" (underlined) causes the violation of the $(X \to 1 \to 0)$ bit-pattern, causing node "111" to have degree 3, "101" to have degree 1. (b) The bit-pattern at bit-2 now contains $(X \to \bar{x} \to 0)$, caused by crossed link "$X\bar{x}1$."

$$\begin{aligned}
P(d_1) &= n-1\ (2^{n-1}(all)\ links\ of\ dim - d_1),\\
P(d_2) &= n-2\ (2^{n-2}(half)\ links\ of\ dim - d_2),\\
&\ldots\ldots\\
P(d_p) &= n-p\ (2^{n-p}\ links\ of\ dim - d_p),\\
&\ldots\ldots\\
P(d_q) &= n-q\ (2^{n-q}\ links\ of\ dim - d_q),\\
&\ldots\ldots\\
P(d_r) &= n-r\ (2^{n-r}\ links\ of\ dim - d_r),\\
&\ldots\ldots\\
P(d_{n-1}) &= 1\ (2\ links\ of\ dim - d_{n-1}),\\
P(d_n) &= 1\ (2\ links\ of\ dim - d_n).
\end{aligned}$$

If P is such that $p < q < r$, and

1. *$d_p = 2k$ for some k, $d_q = 2k-1$, $d_r > d_p$, and $r = q+1$; or*
2. *$d_p = 2k$ for some k, $d_r = 2k-1$, and $d_q > d_p$,*

then P will not *facilitate a Hamiltonian cycle.*

Before proving it, let us check out Theorem 2 with the small examples in Fig. 15. The permutation P_a in Fig. 15a is $\{P_a(2) = 2, P_a(1) = 1, P_a(3) = 1\}$; we have $d_p = 2 (= 2 \cdot 1)$, $d_q = 1 (= 2 \cdot 1 - 1)$, $3 = d_r > d_p = 2$, and d_r immediately follows d_q in P_a (condition 1 of Theorem 2). The permutation P_b in Fig. 15b is $\{P_b(2) = 2, P_b(3) = 1, P_b(1) = 1\}$; we have $d_p = 2, d_r = 1$, and $3 = d_q > d_p = 2$ (condition 2 of Theorem 2).

Proof. We will show that the conditions in Theorem 2 will result in either a bit-pattern of $(X \to 1 \cdots \to x \to \cdots)$, or $(X \to \bar{x} \to \cdots)$ at bit-d_p.

1. $d_p = 2k$, $d_q = 2k-1$, $d_r > d_p$, **and** $r = q+1$.

   If $d_p = 2k$ and $d_q = 2k-1$, bits d_p and d_q are adjacent in link representation. The permutation P determines that the order of the link merge is $d_p \to \cdots \to d_q \to d_r$. When $\dim\text{-}d_p$ links are generated, because $d_r > d_p$, we will have a link with the following representation:

$$\ldots e_{d_r} \ldots X_{d_p} 0_{d_q} \ldots,$$

Authorized licensed use limited to: Rutgers University. Downloaded on May 12, 2009 at 11:39 from IEEE Xplore. Restrictions apply.

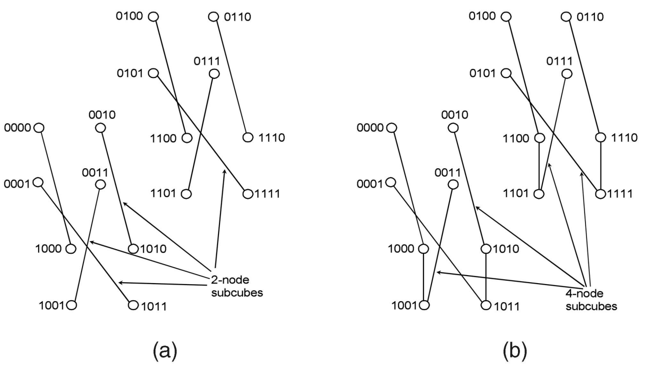


Fig. 16. (a) $\mathrm{Dim}\text{-}n$ links ($n = 4$ here) connect the corresponding nodes so that there are now 2^{n-1} 2-node subcubes; (b) Dim-1 links being added. There are now 2^{n-2} isolated, 4-node subcubes.

where $e_{d_r} \in \{0, 1\}$. According to algorithm CQ_HAMIL, bit-d_p assumes $1_{d_p} \rightarrow 0_{d_p} \rightarrow 0_{d_p} \rightarrow \cdots$ in the subsequent link merges. Therefore, by the time of $\dim\text{-}d_q$ link merge, we have a link with the following representation:

$$\ldots e_{d_r} \ldots 0_{d_p} X_{d_q} \ldots .$$

Since $r = q + 1$, the immediate next link merge is for $\dim\text{-}d_r$. According to algorithm CQ_HAMIL, one link will be

$$\ldots X_{d_r} \ldots x_{d_p} 1_{d_q} \ldots .$$

The "x_{d_p}" contravenes the Hamiltonian facilitating pattern and, as we pointed out before, will cause the violation of the property of Lemma 1.

2. $d_p = 2k$, $d_r = 2k - 1$, **and** $d_q > d_p$.

Since $d_p = 2k$ and $d_r = 2k - 1$, bits d_p and d_r are adjacent in link representation. Also, we have $d_q > d_p$. Therefore, when generating $\dim\text{-}d_p$ links, we must have a link as follows:

$$\ldots e_{d_q} \ldots X_{d_p} 1_{d_r} \ldots .$$

We make two cases and show that in each case, the Hamiltonian facilitating pattern is violated.

Case 1. $q = p + 1$.

$q = p + 1$ means that $\dim\text{-}d_q$ is the immediate next dimension to generate links. According to the algorithm, the link generated out of "$\ldots e_{d_q} \ldots X_{d_p} 1_{d_r} \ldots$" would be

$$\ldots X_{d_q} \ldots \bar{x}_{d_p} 1_{d_r} \ldots ,$$

giving $(X_{d_p} \rightarrow \bar{x}_{d_p} \rightarrow \cdots)$ at bit-d_p.

Case 2. $q > p + 1$.

If $q > .p + 1$, there will be at least one other dimension before $\dim\text{-}d_q$ to generate links. According to the algorithm, links generated out of "$\ldots e_{d_q} \ldots X_{d_p} 1_{d_r} \ldots$" would include

$$[\ldots e_{d_q} \ldots 1_{d_p} 1_{d_r} \ldots] \rightarrow [\ldots e_{d_q} \ldots 0_{d_p} 1_{d_r} \ldots] \rightarrow \cdots \rightarrow [\ldots X_{d_q} \ldots x_{d_p} 1_{d_r} \ldots],$$

giving $(X_{d_p} \rightarrow 1_{d_p} \cdots \rightarrow x_{d_p} \rightarrow \cdots)$ at bit-d_p. □

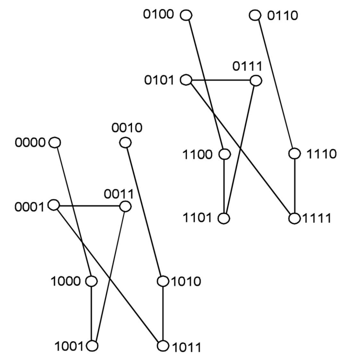


Fig. 17. Dim-2 links being added. There are now 2^{n-3} isolated, 8-node subcubes.

6 CONCLUSION

Numerous variants of hypercube have been proposed and studied. Most of them were conceived to bring about improvement in some aspects of their original counterpart by discounting its high regularity. A good understanding of the implication of this loss of structural regularity is a worthy undertaking from both architectural and algorithmic points of view. In this paper, we did a thorough study of the problem of embedding Hamiltonian cycle in crossed cubes. Crossed cube is one of the most prominent variants of hypercube, which has been shown to outperform hypercube in some important aspects. Due to the loss of link-topology regularity in crossed cube, producing Hamiltonian cycles in a crossed cube is more intricate of a process than in a regular hypercube. What is more, not all link permutations can facilitate a Hamiltonian cycle as in the case of regular hypercube. In this paper, we have proposed the characterization of Hamiltonian facilitating permutations in a crossed cube. An algorithm was presented that generates a Hamiltonian cycle for a given Hamiltonian facilitating permutation.

APPENDIX

Proof for Theorem 1. After running $CQ\_HAMIL\_P_0$, there are 2^n links remaining: All 2^{n-1} $\dim\text{-}n$ links, half of dim-1 links (2^{n-2}), 1/4 of dim-2 links (2^{n-3}), ..., and, finally, 2 $\dim\text{-}(n-2)$ links and 2 $\dim\text{-}(n-1)$ links. (Fig. 1 shows examples for CQ_4 and CQ_5. The pattern is very clear to see and easy to extend to CQ_n of any n.) Lemma 1 proves that none of the 2^{n-1} $\dim\text{-}i$ links $(1 \leq i \leq n-1)$ are touching each other, which means every one of them touches only with $\dim\text{-}n$ links at both ends. That means that each node will have exactly degree 2, a necessary condition for Hamiltonianness.

We then proceed by examining the effect of adding all remaining links after running $CQ\_HAMIL\_P_0$ column by column. The procedure will be described in general terms. However, whenever visualization helps make observation, we refer to examples in Fig. 1 and a graph of CQ_4. It is easy to extend the observation to CQ_n of any n.

Before adding any links, all 2^n nodes can be viewed as isolated, trivial one-node subcubes. The result of $CQ\_HAMIL\_P_0$ preserves all 2^{n-1} links in $\dim\text{-}n$. The effect of adding these links is to connect the corresponding nodes in the "left subcube" (nodes whose addresses start with "0") and "right subcube" (nodes whose addresses start with "1"), so that there are now 2^{n-1}

Authorized licensed use limited to: Rutgers University. Downloaded on May 12, 2009 at 11:39 from IEEE Xplore. Restrictions apply.

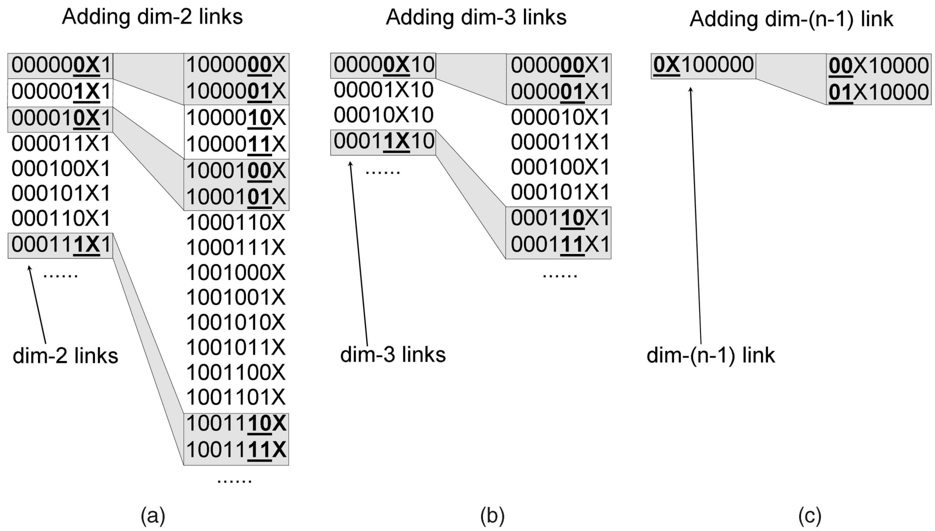


Fig. 18. (a) When each dim-2 link is added, it merges two previously isolated 4-node subcubes. (b) Each dim-3 link merges two previously isolated 8-node subcubes. (c) The one $\mathrm{dim}\text{-}(n{-}1)$ link $\underline{0X}100000$ will merge the two subcubes into one 2^n-node subcube with no ring formed. The result is a Hamiltonian path traversing all nodes.

isolated, 2-node subcubes. An example for CQ_4 is illustrated in Fig. 16a. Next, adding the 2^{n-2} links of dim-1. Note that all remaining dim-1 links are from the "right subcube." Adding these will result in 2^{n-2} isolated, 4-node subcubes, as shown in Fig. 16b for the example CQ_4. It is worth pointing out that all dim-1 links belong to different subcubes.

Next, 2^{n-3} links of dim-2 (all in the left subcube) are added. Note that from now on, added links are *all* from the left cube (the "0"-cube). The result is shown in Fig. 17 for the example CQ_4.

It is necessary at this point to summarize what has been done and to generalize what is effected as each column of links are added. Fig. 18 illustrates the patterns. When each dim-2 link is added (Fig. 18a), it makes a single-link connection for two previously isolated 4-node subcubes. For example, link $0000\underline{0X}1$ connects two subcubes, which links $10000\underline{00}X$ and $10000\underline{01}X$ belong to, respectively. Note that *no rings will be formed* during that process, because every time a new dim-2 link is added, it connects a *new* pair of 4-node subcubes. At the end of this round, the previously 2^{n-2} isolated, 4-node subcubes are merged into 2^{n-3} isolated, 8-node subcubes, with no rings formed.

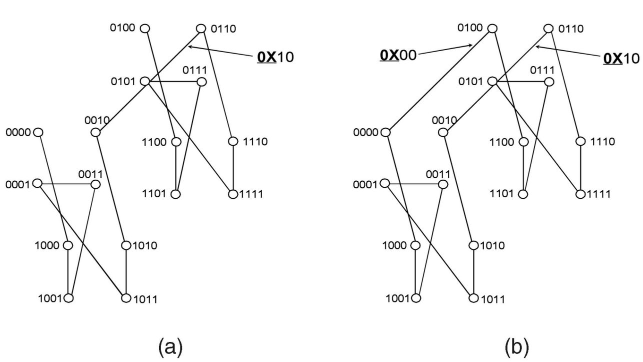


Fig. 19. (a) Adding link $\underline{0X}100000$ makes a single-link connection for the two $2^{n-1}\text{-node subcubes}$, producing a Hamiltonian path traversing all nodes; (b) Adding link $\underline{0X}000000$ completes the Hamiltonian cycle.

Dim-3 links are added following exactly the same pattern (Fig. 18b). At the end, the 2^{n-3} isolated, 8-node subcubes are merged into 2^{n-4} isolated, 16-node subcubes, with no rings formed.

The above process repeats. On each iteration, the number of isolated, identical subcubes is reduced by half. Before $\mathrm{dim}\text{-}(n{-}1)$ links are added, there are just $2\ (= 2^{n-(n-1)})$ identical subcubes, each having 2^{n-1} nodes. An example for CQ_4 can be seen in Fig. 17. The addition of one $\mathrm{dim}\text{-}(n{-}1)$ link $\underline{0X}100000$ (Figs. 18c and 19a) will merge the two subcubes into one 2^n-node subcube, *with no ring formed yet*. The result is now a Hamiltonian path traversing all nodes, with nodes 00000000 and 01000000 "open ended." The addition of the "last link" $0X000000$ of $\mathrm{dim}\text{-}(n{-}1)$ will complete the Hamiltonian cycle (Fig. 19b). □

REFERENCES

[1] E. Abuelrub and S. Bettayeb, "Embedding Rings into Faulty Twisted Hypercubes," *Computers and Artificial Intelligence*, vol. 16, pp. 425-441, 1997.

[2] M.M. Bae and B. Bose, "Edge Disjoint Hamiltonian Cycles in k-ary n-cubes and Hypercubes," *IEEE Trans. Computers*, vol. 52, no. 10, pp. 1271-1284, Oct. 2003.

[3] R.V. Boppana, S. Chalasani, and C.S. Raghavendra, "Resource Deadlock and Performance of Wormhole Multicast Routing Algorithms," *IEEE Trans. Parallel and Distributed Systems*, vol. 9, no. 6, pp. 535-549, June 1998.

[4] C.-P. Chang, T.-Y. Sung, and L.-H. Hsu, "Edge Congestion and Topological Properties of Crossed Cube," *IEEE Trans. Parallel and Distributed Systems*, vol. 11, no. 1, pp. 64-80, Jan. 2000.

[5] E. Dixon and S. Goodman, "On the Number of Hamiltonian Circuits in the n-Cube," *Proc. Am. Math. Soc.*, pp. 500-504, 1975.

[6] K. Efe, "The Crossed Cube Architecture for Parallel Computing," *IEEE Trans. Parallel and Distributed Systems*, vol. 3, no. 5, pp. 513-524, Sept. 1992.

[7] K. Efe, P.K. Blackwell, W. Slough, and T. Shiau, "Topological Properties of the Crossed Cube Architecture," *Parallel Computing*, vol. 20, pp. 1763-1775, 1994.

[8] K. Efe and A. Fernandez, "Products of Networks with Logarithmic Diameter and Fixed Degree," *IEEE Trans. Parallel and Distributed Systems*, vol. 6, no. 9, pp. 963-975, Sept. 1995.

Authorized licensed use limited to: Rutgers University. Downloaded on May 12, 2009 at 11:39 from IEEE Xplore. Restrictions apply.

[9] J. Fan, "Diagnosability of Crossed Cubes under the Comparison Diagnosis Model," *IEEE Trans. Parallel and Distributed Systems,* vol. 13, no. 10, pp. 1099-1104, Oct. 2002.

[10] J. Fan, X. Lin, and X. Jia, "Optimal Path Embedding in Crossed Cubes," *IEEE Trans. Parallel and Distributed Systems,* vol. 16, no. 12, pp. 1190-1200, Dec. 2005.

[11] J.-S. Fu, "Hamiltonicity of the WK-Recursive Network with and without Faulty Nodes," *IEEE Trans. Parallel and Distributed Systems,* vol. 16, no. 9, pp. 853-865, Sept. 2005.

[12] F. Harary, "A Survey of Hypercube Graphs," *Computers and Math. Applications,* 1989.

[13] S.-Y. Hsieh, C.-W. Ho, and G.-H. Chen, "Fault-Free Hamiltonian Cycles in Faulty Arrangement Graphs," *IEEE Trans. Parallel and Distributed Systems,* vol. 10, no. 3, pp. 223-237, Mar. 1999.

[14] H.-C. Hsu, T.-K. Li, J.J.M. Tan, and L.-H. Hsu, "Fault Hamiltonicity and Fault Hamiltonian Connectivity of the Arrangement Graphs," *IEEE Trans. Computers,* vol. 53, no. 1, pp. 39-53, Jan. 2004.

[15] W.-T. Huang, W.-K. Chen, and C.-H. Chen, "On the Fault-Tolerant Pancyclicity of Crossed Cubes," *Proc. Ninth Int'l Conf. Parallel and Distributed Systems,* p. 483, 2002.

[16] W.-T. Huang, Y.-C. Chuang, J.M. Tan, and L.-H. Hsu, "On the Fault-Tolerant Hamiltonicity of Faulty Crossed Cubes," *IEICE Trans. Fundamentals,* vol. E85-A, no. 6, pp. 1359-1370, June 2002.

[17] W.-T. Huang, J.J.M. Tan, C.-N. Hung, and L.-H. Hsu, "Fault-Tolerant Hamiltonicity of Twisted Cubes," *J. Parallel and Distributes Computing,* vol. 62, pp. 591-604, 2002.

[18] R.-S. Lo and G.-H. Chen, "Embedding Hamiltonian Paths in Faulty Arrangement Graphs with the Backtracking Method," *IEEE Trans. Parallel and Distributed Systems,* vol. 12, no. 2, pp. 209-222, Feb. 2001.

[19] P. Kulasinghe, "Connectivity of the Crossed Cube," *Information Processing Letters,* vol. 61, pp. 221-226, Feb. 1997.

[20] P. Kulasinghe and S. Bettayeb, "Embedding Binary Trees into Crossed Cube," *IEEE Trans. Computers,* vol. 44, no. 7, pp. 923-929, July 1995.

[21] X. Lin, P.K. Mckinley, and L.M. Ni, "Deadlock-Free Multicast Wormhole Routing in 2D Mesh Multicomputers," *IEEE Trans. Parallel and Distributed Systems,* vol. 5, no. 8, pp. 793-804, Aug. 1994.

[22] D. Wang, "Embedding Hamiltonian Cycles into Folded Hypercubes with Faulty Links," *J. Parallel and Distributed Systems,* vol. 61, no. 4, pp. 545-564, 2001.

[23] M.-C. Yang, T.-K. Li, J.M. Tan, and L.-H. Hsu, "Fault-Tolerant Cycle-Embedding of Crossed Cubes," *Information Processing Letters,* vol. 88, no. 4, pp. 149-154, Nov. 2003.

[24] S.Q. Zheng and S. Latifi, "Optimal Simulation of Linear Multiprocessor Architectures on Multiply-Twisted Cube Using Generalized Gray Code," *IEEE Trans. Parallel and Distributed Systems,* vol. 7, no. 6, pp. 612-619, June 1996.

Dajin Wang received the BEng degree from the Shanghai University of Science and Technology, Shanghai, in 1982, and the MS and PhD degrees in computer science from the Stevens Institute of Technology in 1986 and 1990, respectively. Since then, he has been with the faculty of the Department of Computer Science, Montclair State University, Upper Montclair, New Jersey, where he is currently a professor. His main research interests include fault-tolerant computing, parallel processing, algorithmic robotics, and wireless ad hoc and sensor networks. He is a member of the IEEE Computer Society.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.

Authorized licensed use limited to: Rutgers University. Downloaded on May 12, 2009 at 11:39 from IEEE Xplore. Restrictions apply.