Gray Codes for Torus and Edge Disjoint Hamiltonian Cycles*

Myung M. Bae
Scalable Powerparallel Systems,
MS/P963, IBM Corp.,
Poughkeepsie, NY 12601
myungbae@us.ibm.com

Bella Bose
Dept. of Computer Science
Oregon State University,
Corvallis, OR 97331-3902
bose@cs.orst.edu

Abstract

Lee distance Gray codes for k-ary n-cubes and torus networks are presented. Using these Lee distance Gray codes, it is further shown how to directly generate edge disjoint Hamiltonian cycles for a class of k-ary n-cubes, 2-D tori, and hypercubes.

1. Introduction

A multicomputer system consists of multiple nodes that communicate by exchanging messages through an interconnection network. At a minimum, each node normally has one or more processing elements, a local memory, and a communication module. A popular topology for the interconnection network is the torus. Also called a wrap-around mesh or a toroidal mesh, this topology includes the k-ary n-cube which is an n-dimensional torus with the restriction that each dimension is of the same size, k, and the hypercube, which is a k-ary n-cube with k = 2; a mesh is a subgraph of a torus.

Several parallel machines, both commercial and experimental, have been designed with a toroidal interconnection network. Included among these machines are the following: Cray T3D and T3E (3D torus) [1], the Mosaic (k-ary n-cube) [2], the iWrap (torus) [3], the Tera Parallel Computer (torus) [4].

Some topological properties of Torus and k-ary n-cubes based on Lee distance are given in [5, 6]. This paper presents some results related to edge disjoint Hamiltonian cycles in these networks. Although the existence of disjoint Hamiltonian cycles in the cross product of various graphs has been discussed in literature [13, 7, 8, 9, 12, 10, 11], a straightforward way of generating such cycles is not clear. Here, first we design Lee distance Gray codes and then using these codes show how to decompose a higher dimension torus to edge disjoint lower dimensional tori - in particular, edge disjoint Hamiltonian cycles.

Figure 1. Two disjoint Hamiltonian cycles in $C_3 \times C_3$

For example, Figure 1 gives two edge disjoint cycles (solid and dotted lines) in $C_3 \times C_3$. Similarly, a $C_3 \times C_3 \times C_3 \times C_3$ is decomposed into two edge disjoint tori of size $C_6 \times C_6$ in Figure 2(a) and Figure 2(b). The solid and dotted lines in these two figures give a total of four disjoint Hamiltonian cycles in $C_3 \times C_3 \times C_3 \times C_3$.

The rest of the paper is organized as follows. Section 2 gives some preliminaries. Lee distance Gray codes (for torus network) are described in Section 3. Section 4 discusses some results on edge disjoint Hamiltonian cycles. Section 5 gives similar results for hypercubes. Section 6 is the conclusion of this paper.

2. Preliminaries

2.1. Lee Distance and Torus

Let $A = a_{n-1}a_{n-2}\cdots a_0$ be a $n$-dimensional mixed radix vector over $Z_K$, where $K = k_{n-1} \times k_{n-2} \times \cdots \times k_0$, i.e., all $x_i \in Z_{k_i}$ for $i = 0, 1, \cdots, n - 1$. The Lee weight of $A$ in mixed radix notation is defined as $W_L(A) = \sum_{i=0}^{n-1} |t_i|$, where $|t_i| = \min(a_i, k_i - a_i)$, for $i = 0, 1, \cdots, n - 1$. The Lee distance between two vectors $A$ and $B$ is denoted by $D_L(A, B)$ and is defined...
to be \( W_L(A - B). \) That is, the Lee distance between two vectors is the Lee weight of their digit-wise difference. In other words, \( D_L(A, B) = \sum_{i=0}^{n-1} \min(a_i - b_i, b_i - a_i) \), where \( a_i - b_i \) and \( b_i - a_i \) are mod \( k_i \) operations. For example, when \( K = 4 \times 6 \times 3, W_L(321) = \min(3, 4 - 3) + \min(2, 6 - 2) + \min(1, 3 - 1) = 1 + 2 + 1 = 4, \) and \( D_L(123 - 321) = W_L(202) = 3. \)

Let \( D_H(A, B) \) be the Hamming distance between two vectors \( A \) and \( B \), i.e. the number of position in which \( A \) and \( B \) differ. Then \( D_L(A, B) = D_H(A, B) \) when \( k_i = 2 \) or 3, for all \( i \), and \( D_L(A, B) \geq D_H(A, B) \) when \( k_i > 3 \) for some \( i \). In the rest of the paper, it is assumed that \( k_i \geq 3. \)

Just as Hamming distance may be used to define the binary hypercube graph, \( Q_n \), the generalized hypercube graph [14], and twisted hypercube graph [15], Lee distance may be used to define the \( k \)-ary \( n \)-cube graph, \( C_k^n \), and the \( n \)-dimensional torus, \( T_{k_1, k_2, \ldots, k_n} \).

A \( k \)-ary \( n \)-cube graph \( (C_k^n) \) and an \( n \)-dimensional torus \( (T_{k_1, k_2, \ldots, k_n}) \) are \( 2n \)-regular graphs containing \( k^n \) and \( k_1 k_2 \cdots k_n \) nodes respectively. Each node in a \( C_k^n \) is labeled with a distinct \( n \)-digit radix-\( k \) vector while each node in a \( T_{k_1, k_2, \ldots, k_n} \) is labeled with a distinct \( n \)-digit mixed radix vector. In this paper, node labels will be written as \( (a_{n-1} a_{n-2} \cdots a_0) \) or as \( a_{n-1} a_{n-2} \cdots a_0 \) rather than the \( n \)-tuples \( (a_{n-1}, a_{n-2}, \ldots, a_0) \) when there is no confusion. If \( u \) and \( v \) are two nodes in the graph, then there is an edge between them if \( D_L(\{u, v\}) = 1 \). From the definition of Lee distance, it can be seen that every node in a \( C_k^n \) or a \( T_{k_1, k_2, \ldots, k_n} \) shares an edge with two nodes in every dimension, resulting in a regular graph of degree \( 2n \). In addition, the shortest path between any two vectors, \( u \) and \( v \) has length \( D_L(u, v) \). Note that a \( C_k^n \) is an \( n \)-dimensional hypercube, \( Q_n \), when \( k = 2. \)

### 3. Lee Distance Gray Codes

Many algorithms can be solved efficiently by embedding a Hamiltonian cycle or a Hamiltonian path within torus network. This section addresses the embedding problem by presenting four methods of constructing a Lee distance Gray code. For each method, let \( R = (r_{n-1} r_{n-2} \cdots r_0) \) be a number in radix notation, and let \( G = (g_{n-1} g_{n-2} \cdots g_0) \) be the Gray code presentation given by \( f_i \), i.e. \( G = f_i(R) \). The first three methods are given in [5, 6] and so the proofs are omitted. However, the methods are useful in generating disjoint Hamiltonian cycles and so they are briefly described here.

#### 3.1. Single Radix Codes

First, assume there are \( n \) dimensions, each having the same number of processors, \( k \), where \( k \geq 3. \) Each processor node is labeled with a distinct \( n \)-digit, radix \( k \) vector \( (r_{n-1} r_{n-2} \cdots r_0) \), where \( r_i \leq k \) for \( 0 \leq i \leq n - 1. \) Two nodes, \( A = (a_{n-1} a_{n-2} \cdots a_0) \) and \( B = (b_{n-1} b_{n-2} \cdots b_0) \), are adjacent if the Lee distance between them, \( D_L(A, B) \), is one.

Two methods are given below for constructing a Gray code based on the assumption of the previous paragraph.

---

**Figure 2. Decomposition of** \( C_2 \times C_2 \times C_3 \times C_2 \) **into two edge disjoint** \( C_0 \times C_0 \) **and four edge disjoint Hamiltonian cycles.**

2.2. Cross Product of Graphs

Given graphs \( G_1 = (V_1, E_1) \) and \( G_2 = (V_2, E_2) \), define the cross product of \( G_1 \) and \( G_2 \), denoted by \( G_1 \times G_2 \) [16], as the graph \( G = (V, E) \), where

\[
V = \{(u, v) | u \in V_1, v \in V_2 \}, \quad \text{and} \quad E = \{(u_1, v_1), (u_2, v_2) | (u_1, u_2) \in E_1 \text{ and } v_1 = v_2 \},
\]

or \( (u_1 = u_2 \text{ and } v_1 \neq v_2) \), or \( (u_1 \neq u_2 \text{ and } v_1 = v_2) \).

A cycle of length \( k \) is denoted by \( C_k \), and each node in \( C_k \) is labeled with a radix \( k \) number, \( 0 \cdots k - 1. \) There is an edge between vertices \( u \) and \( v \) if \( D_L(u, v) = 1. \)

A \( k \)-ary \( n \)-cube \( (C_k^n) \) and an \( n \)-dimensional torus \( (T_{k_1, k_2, \ldots, k_n}) \) can alternately be defined as the product of cycles as follows.

\[
C_k^n = C_k \times C_k \times \cdots \times C_k \quad \text{or} \quad n \text{ times}
\]

\[
T_{k_1, k_2, \ldots, k_n} = C_{k_1} \times C_{k_2} \times \cdots \times C_{k_n}
\]

The above definition demonstrates a useful topological property of a \( C_k^n \); a \( C_k^n \) can be recursively defined in terms of smaller \( k \)-ary cubes.

\[
C_k^n = \begin{cases} 
C_k, & \text{if } n = 1 \\
C_k \times C_k^{n-1}, & \text{if } n > 1
\end{cases}
\]
Method 1 \[ f_i(r_{n-1}r_{n-2} \cdots r_0) = r_{n-1}(r_{n-2} - r_{n-1}) \cdots (r_0 - r_1) \]

Method 2 \[ This \ method \ produces \ a \ Hamiltonian \ cycle \ if \ k \ is \ even, \ and \ a \ Hamiltonian \ path \ if \ k \ is \ odd. \ Let \ \overrightarrow{r} = k - 1 - r_i, \ and \ let \ g_{n-1} = r_{n-1}. \ Then \ for \ i = n - 2, \cdots, 0, \ if \ k \ is \ even \ then \]

\[ g_i = \begin{cases} r_i, & \text{if } r_{i+1} \text{ is even} \\ \overrightarrow{r}, & \text{otherwise} \end{cases} \]

or, if \( k \) is odd, \( \overrightarrow{r}' = \sum_{j=i+1}^{n-1} r_j, \) and

\[ g_i = \begin{cases} r_i, & \text{if } \overrightarrow{r}' \text{ is even} \\ \overrightarrow{r}, & \text{otherwise} \end{cases} \]

3.2. Mixed Radix Codes

In many cases, however, the number of processors per dimension varies. Let \( K = k_{n-1}k_{n-2} \cdots k_0 \) be an \( n \)-dimensional vector where \( k_i \) is the radix of dimension \( i \) and \( k_i \geq 3 \) for \( 0 \leq i \leq n - 1. \) In this case, Method 3 gives a Gray code design resulting in a Hamiltonian cycle if \( k_i \) is even for at least one value of \( i. \) If each \( k_i \) is odd, the resulting Gray code produces a Hamiltonian path. Method 4 produces a Hamiltonian cycle if all \( k_i \)'s are odd.

Let each processor node be labeled with a distinct \( n \)-digit vector \( R = (r_{n-1}r_{n-2} \cdots r_0), \) where \( 0 \leq r_i \leq k_i - 1 \) for \( i = 0, 1, \cdots, n - 1. \) Vector \( R \) is said to be in mixed-radix notation, and the integer value of \( R \) is given by

\[ I(R) = r_0 + r_1 k_0 + r_2 k_0 k_1 + \cdots + r_{n-1}k_0k_1 \cdots k_{n-2} = \sum_{i=1}^{n-1} \left( r_i \prod_{j=0}^{i-1} k_j \right) + r_0 \]

Method 3 \[ Assume \ that \ at \ least \ one \ of \ the \ k_i \'s \ is \ even. \ Without \ loss \ of \ generality, \ assume \ that \ the \ dimensions \ are \ ordered \ so \ that \ if \ k_i \ is \ even \ and \ k_j \ is \ odd, \ then \ i > j. \ Let \ l \ be \ the \ index \ of \ the \ lowest \ even \ dimension. \ That \ is, \ the \ dimensions \ are \ ordered \ as \ follows.

\[ k_{n-1} \cdots k_l \ \overline{k_l} \cdots k_0 \]

Now, letting \( \overrightarrow{r} = k_l - 1 - r_i, \) and \( \overrightarrow{r}_l' = \sum_{j=i+1}^{l-1} r_j, \) \( f_l \) is defined as follows.

\[ g_{n-1} = r_{n-1}, \] and

for \( i = n - 2 \) down to \( l: \)

\[ g_i = \begin{cases} r_i, & \text{if } r_{i+1} \text{ is even} \\ \overrightarrow{r}, & \text{otherwise} \end{cases} \]

for \( i = l - 1 \) down to 0:

\[ g_i = \begin{cases} r_i, & \text{if } \overrightarrow{r}_l' \text{ is even} \\ \overrightarrow{r}_l, & \text{otherwise} \end{cases} \]

Method 4 \[ Assume \ that \ k_i \ is \ odd \ for \ 0 \leq i \leq n - 1, \ and \ that \ the \ dimensions \ are \ ordered \ such \ that \ k_{n-1} \geq k_{n-2} \geq \cdots \geq k_0. \ Also \ define \]

\[ r_{\overrightarrow{l}} = \begin{cases} r_i, & \text{if } r_{i+1} \text{ is odd} \\ k_i - 1 - r_i, & \text{otherwise} \end{cases} \]

Now, \( f_l, \) which produces a Gray code yielding a Hamiltonian cycle, is defined as follows.

\[ g_{n-1} = r_{n-1}, \] and

for \( i \leq i \leq n - 2: \)

\[ g_i = \begin{cases} (r_i - r_{i+1}) \mod k_i, & \text{if } r_{i+1} < k_i \\ \overrightarrow{r}_l, & \text{otherwise} \end{cases} \]

Figure 3(a) shows the Hamiltonian cycles \( C_5 \times C_3 \) using Method 4 (the rest of the edges form the other Hamiltonian cycle).

\[ Note: \ When \ all \ k_i \'s, \ i = 0, 1, \cdots, n - 1, \ are \ even, \ a \ similar \ Gray \ code \ described \ below \ gives \ a \ Hamiltonian \ cycle. \ Again \ assume \ k_{n-1} \geq k_{n-2} \geq \cdots \geq k_0. \]

Define

\[ r_{\overrightarrow{l}} = \begin{cases} r_i, & \text{if } r_{i+1} \text{ is even} \\ k_i - 1 - r_i, & \text{otherwise} \end{cases} \]

Then

\[ g_{n-1} = r_{n-1}, \] and

for \( i \leq i \leq n - 2: \)

\[ g_i = \begin{cases} r_i - r_{i+1}, & \text{if } r_{i+1} < k_i \\ \overrightarrow{r}_l, & \text{otherwise} \end{cases} \]

Figure 3(b) shows the Hamiltonian cycle in \( C_6 \times C_4 \) using the above method (Again the rest of the edges form the other Hamiltonian cycle).

Lemma 1 \[ Method \ 4 \ proposed \ above \ produces \ a \ Lee \ distance \ Gray \ code \ and \ hence \ an \ Hamiltonian \ cycle \ in \ T_{k_{n-1}k_{n-2} \cdots k_0}, \ where \ k_i \geq 3, \ for \ i = 0, 1, 2, \cdots, n - 1, \ and \ all \ k_i \'s \ are \ odd \ (or \ all \ k_i \'s \ are \ even). \]
Proof: The proof is given when all $k_i$'s, $i = n - 1, n - 2, \cdots, 0$ are odd. A similar proof is obtained when they are all even.

Case 1: 
\[
f_i(0000 \cdots 0) = 0000 \cdots 0
\]
\[
f_i(k_{n-1} - 1, k_{n-2} - 1, \cdots, k_0 - 1)
= (k_{n-1} - 1, k_{n-2} - 1, k_{n-3} - 1, \cdots, k_0 - 1)
\]
But for $i = n - 2, n - 3, \cdots, 0, \frac{k_i}{k_i - 1} = 0$ because $(k_{i+1} - 1)$'s are all even and $(k_i - 1) < k_{i+1} - 1$. Thus the first and the last words are at a distance of 1.

Case 2: Let $X$ and $Y$ be two consecutive numbers in the mixed radix numbers and let $m$ be the first index from left in which $X$ and $Y$ differ; i.e.,
\[
X = \{x_{n-1}x_{n-2}\cdots x_{m+1}\}^*x_m
\]
\[
\{(k_{m-1} - 1)(k_{m-2} - 1)\cdots (k_0 - 1)\}^*
\]
\[
Y = \{x_{n-1}x_{n-2}\cdots x_{m+1}\}^*(x_m + 1)(00\cdots 0)^*
\]
where the segment marked by a * may or may not exist depending on the value of $m$. Let $f_i(X) = a_{n-1}a_{n-2}\cdots a_0$ and $f_i(Y) = b_{n-1}b_{n-2}\cdots b_0$. $f_i(X)$ and $f_i(Y)$ are considered in three segments: (a) between dimension $n - 1$ and $m + 1$, (b) dimension $m$, and (c) between dimension $m - 1$ and 0. It is shown below that $a_i = b_i$ for $i \neq m$ and that $D_L(a_m, b_m) = 1$. This shows $D_L(f_i(X), f_i(Y)) = 1$.

(a) $[n - 1 \geq i \geq m + 1]$: For this range, $x_i = y_i$ and so $a_i = b_i$.

(b) $[i = m]$: Also because of (a), either $(a_m = x_m$ and $b_m = x_m + 1)$, or $(a_m = x_m$ and $b_m = x_m + 1)$. Further
\[
\frac{x_m}{x_m + 1} = \frac{k_m - 1 - x_m}{k_m - 1 - (x_m + 1)} = \frac{(k_m - 1) - (x_m + 1) - 1}{x_m - 1}.
\]
Note that $D_L(x_m, x_{m+1}) = x_m + 1 - x_m = 1$ and that $D_L(x_m, x_{m+1}) = x_m - (x_m - 1) = 1$. Therefore in either case $D_L(a_m, b_m) = 1$.

(c) $[m - 1 \geq i \geq 0]$: Since $k_i - 1$ are all even and $k_i - 1 \geq k_{i+1} - 1, a_{m-2} = a_{m-3} = \cdots = a_0 = 0$. Further $b_{m-2} = b_{m-3} = \cdots = b_0 = 0$. We show that $a_{m-1} = b_{m-1}$ by considering the following three cases.

Case i $[x_m + 1 < k_{m-1}]$: Then $a_{m-1} = k_{m-1} - 1 - x_m$ and $b_{m-1} = 0 - (x_m + 1) = k_{m-1} - x_m - 1$. So $a_{m-1} = b_{m-1}$.

Case ii $[x_m + 1 = k_{m-1} and so x_m = k_{m-1} - 1]$:
\[
a_{m-1} = k_{m-1} - 1 - x_m = k_{m-1} - 1 - k_{m-1} - 1 = 0; b_{m-1} = 0 since k_{m-1} is odd. Thus, in this case also $a_{m-1} = b_{m-1}$.
\]

Case iii $[x_m + 1 > k_{m-1}]$: If $x_m + 1$ is even (and so $x_m$ is odd) then $a_{m-1} = k_{m-1} - 1$ and $b_{m-1} = 0 = k_{m-1} - 1$; On the other hand, if $x_{m+1}$ is odd (and so $x_m$ is even) then $a_{m-1} = k_{m-1} - 1 = 0$ and $b_{m-1} = 0$. Thus in both cases $a_{m-1} = b_{m-1}$.

4. Edge Disjoint Hamiltonian Cycles

When edge disjoint Hamiltonian cycles are used in a communication algorithm, their effectiveness is improved if more than one cycle exists. As mentioned earlier, the existence of disjoint Hamiltonian cycles in the cross product of various graphs has been discussed in the literature [13, 7, 8, 9, 12, 10, 11]; however a straightforward way of generating such disjoint cycles is not clear. This section contains the functions that generate these disjoint cycles for the k-ary n-cubes, and the hypercubes. Because of space limitation, results are given for $n$, a power of 2. Results for other cases are described in [17] and will be presented in future.

Two Gray codes, $G_1$ and $G_2$, over $Z^n_k$ are said to be independent if two words, $a$ and $b$, are adjacent in $G_1$ (or $G_2$), then they are not adjacent in $G_2$ (or $G_1$). If $k \geq 3$, we can have at most $n$ sets of independent Gray codes; for $k = 2$, this number is $\lfloor \frac{n}{2} \rfloor$.

Theorem 2 The independent set of Gray codes over $Z^n_k$ is equivalent to the set of edge disjoint Hamiltonian cycles in the graph of k-ary n-cube, $C^n_k$.

In the following, $h_i(X) = h_i((x_{n-1}, x_{n-2}, \cdots, x_0); Z^n_k) = (g_{n-1}, g_{n-2}, \cdots, g_0)$ represents the mapping of the radix number form to the corresponding Gray code form, for $i = 0, 1, 2, \cdots, n - 1$ (assuming $k \geq 3$), and these mappings give the independent Gray codes. Similarly, $H_0(C^n_k), H_1(C^n_k), \cdots, H_{n-1}(C^n_k)$ denote the corresponding Hamiltonian cycles derived by the functions $h_i(X), i = 0, 1, \cdots, n - 1$.

For $n = 1$, only one Gray code exists, and so in the rest of the chapter it is assumed that $n \geq 2$ and $k \geq 3$. Further, for $X = (x_{n-1}, x_{n-2}, \cdots, x_0) \in Z^n_k$, let $(X, Z^n_{k+1})$ represent the $n_1$ digit long radix-k representation of the radix-k number X. For example, if $X = (2, 4, 3) \in Z^3_8$, then $(X, Z^n_{k+1}) = (14, 23)$. If there is no confusion, we also assume $X = (X, Z^n_{k+1})$, i.e., $X$ can be treated an integer in the range $0, 1, \cdots, k^n - 1$. 
The following sections show the functions that generate the edge disjoint Hamiltonian cycles for various \(k\)-ary \(n\)-cubes.

### 4.1. \(k\)-ary 2-cube (i.e., \(n = 2\))

**Theorem 3** There are two independent Gray codes in \(C_k^2\) and these are generated by the functions \(h_0\) and \(h_1\), where

\[
h_0(x; Z_k^1) = h_0((x_1, x_0); Z_k^1) = (x_1, (x_0 - x_1) \mod k)
\]

\[
h_1(x; Z_k^1) = h_1((x_1, x_0); Z_k^1) = ((x_0 - x_1) \mod k, x_1)
\]

**Proof:** \(h_0\) is the same as the function \(f_1\) (Method 1) described in Section 3 of this paper (and also see [5]). Thus this gives a Gray code. \(h_1\) gives two digit sequence which is a permutation of the two digit sequence obtained from \(h_0\). Thus \(h_1\) also gives a Gray code and so an Hamiltonian cycle in \(k\)-ary \(n\)-cube. We need to prove that the edges are disjoint. In the \(i\)-th row, \(i = 0, 1, \cdots, k - 1\), \(h_0\) uses all edges except the edge \(((i, k - 1 - i), (i, k - i))\) which is the only \(i\)-th row edge used by \(h_1\). Similarly, along the \(j\)-th column, \(j = 0, 1, \cdots, k - 1\), \(h_1\) uses all edges except the edge \(((i, k - 1 - j), (j, k - j))\), which is the only \(j\)-th column edge used by \(h_1\). Thus the two Hamiltonian cycles are disjoint.

The inverse functions of \(h_0\) and \(h_1\) are as follows:

\[
h_0^{-1}((g_1, g_0); Z_k^1) = (g_1, (g_0 + g_1) \mod k)
\]

\[
h_1^{-1}((g_1, g_0); Z_k^1) = ((g_0 + g_1) \mod k, g_1)
\]

**Example 1** The two Gray code sequences are shown in Figure 1 for \((k = 3, n = 2)\). The solid lines represent the first Gray code and the dotted line the second.

### 4.2. Two Dimensional Torus, \(T_{k^r,k}\)

The results given in Section 4.1 can be generalized to two dimensional torus of size \(k^r \times k\), for \(k \geq 3\) and \(r \geq 1\), as shown in the following theorem.

**Theorem 4** There are two independent Gray codes in \(T_{k^r,k}\) for \(k \geq 3\) and \(r \geq 1\) and are generated by the functions \(h_0\) and \(h_1\), where

\[
h_0(x_1, x_0) = (a_1, a_0) = (x_1, (x_0 - x_1) \mod k)
\]

\[
h_1(x_1, x_0) = (b_1, b_0) = ((x_1(k - 1) + x_0) \mod k^r, x_1 \mod k)
\]

The inverse functions are given by

\[
h_0^{-1}(a_1, a_0) = (x_1, x_0) = (a_1, (a_1 + a_0) \mod k)
\]

\[
h_1^{-1}(b_1, b_0) = (x_1, x_0) = (b_1 - b_0(k - 1)\mod k^r, (b_1 + b_0) \mod k)
\]

The results given in Section 4.1 can be generalized to two dimensional torus of size \(k^r \times k\), for \(k \geq 3\) and \(r \geq 1\), as shown in the following theorem.

### 4.3. \(k\)-ary \(n\)-cube, where \(n = 2^r\), and \(k \geq 3\)

The disjoint Hamiltonian cycles can be recursively constructed as described in the following theorem.

**Theorem 5** The \(n\) independent sets of Gray codes are defined by the functions \(h_i, i = 0, 1, 2, \cdots, n - 1\) as described below. Let \(i_1 = \frac{n}{2}\) and \(h_i(X; Z_k^n) = h_i, ((X_1, X_0); Z_{kn/2}^2) = (Y_1, Y_0), \text{ i.e.}

\[
(Y_1, Y_0) = h_i, ((X_1, X_0); Z_{kn/2}^2) = \begin{cases} (X_1, (X_0 - X_1) \mod k^{n/2}) & \text{for } i_1 = 0, \\ ((X_0 - X_1) \mod k^{n/2}, X_1) & \text{for } i_1 = 1 \\ \end{cases}
\]

Then, the \(i\)-th function \(h_i\) is recursively defined as

\[
h_i(X; Z_k^n) = (y_{n-1}, y_{n-2}, \cdots, y_{n/2}, y_{n/2-1}, \cdots, y_0)
\]

where,

\[
(y_{n-1}, y_{n-2}, \cdots, y_{n/2}) = h(i mod n/2) (Y_1; Z_{k^{n/2}}^2), \\
(y_{n/2-1}, y_{n/2-2}, \cdots, y_0) = h(i mod n/2) (Y_0; Z_{k^{n/2}}^2).
\]

**Proof:** By induction on \(n\).

**Base:** when \(n = 2\), this theorem is reduced to Theorem 3.

**Induction Hypothesis:** Assume that there are \(n\) independent sets of Gray codes in \(C_k^n, H_0^n, H_1^n, \cdots, H_{n-1}^n\) for \(n = 2^r\), i.e., \(C_k^n = H_0^n + H_1^n + \cdots + H_{n-1}^n\).

**Induction Step:** Now consider the case for \(n' = 2n = 2^{r+1}\). A \(C_k^n\) can be decomposed as

\[
C_k^n = C_k^n \times C_k^n = (H_0^n + H_1^n + \cdots + H_{n-1}^n) \times (H_0^n + H_1^n + \cdots + H_{n-1}^n) = (H_0^n \times H_0^n + \cdots + H_{n-1}^n \times H_{n-1}^n)
\]
Here $G_1 + G_2$ indicates the union of two edge disjoint graphs, $G_1$ and $G_2$. Now $H'_i \times H''_i$ is a two dimensional torus of size $k^2 \times k^2$ and this is edge-disjoint from $H'_j \times H''_j$ for $i \neq j$. From each $H'_i \times H''_i$, two disjoint Hamiltonian cycles can be constructed using Theorem 3. Thus 2n disjoint Hamiltonian cycles can be constructed from $C^n_{2n}$, and the $2^{th}$ Gray code is the $i^{th}$ cycle of the component $(H'_i \times H''_i)$, where $i_0 = i \mod \lfloor n/2 \rfloor$ and $i_1 = \lfloor n/4 \rfloor$.

**Example 3** In $Z^4_2$, there are 8 independent sets of Gray codes, which are generated by the functions $h_i, i = 0, 1, 2, \ldots, 7$. Let $X = (2,1,1,3,2,3,0,1)$ be a given vector over $Z^4_2$. We now describe how this vector is mapped under $h_3$. Since $i = 3, i_1 = 1 = 1$. Let $h_0((2,1,1,3,2,3,0,1); Z^{16}_2) = (151, 177); Z^{16}_2 = (151, 176) = (Y_1, Y_0)$.

$$h_3(X; Z^4_2) = (h_{3 \mod 4}(151; Z^4_1), h_{3 \mod 8}(177; Z^4_1)) = (h_3((2,1,1,3); Z^4_1), h_3((0,1,2,2); Z^4_1)).$$

To find $h_3((2,1,1,3); Z^4_1)$, note that $((2,1,1,3); Z^4_1) = ((9,7), Z^{16}_2).$. In this recursion, $i_1 = 1 = 1$ and so $h_1((9,7); Z^{16}_2) = (14, 9)$. Thus $h_3((2,1,1,3); Z^4_1) = (14, 9), h_3((0,1,2,2); Z^4_1) = (14, 9)$. Similarly, it can be shown that $h_3((0,1,2,2); Z^4_1) = (3,2,1,0)$ and thus $h_3((2,1,1,3,2,3,0,1); Z^{16}_2) = (3,3,3,2,3,2,1,0)$. Further, it can be verified that $h_0(X) = (2,3,3,0,1,2,3), h_1(X) = (3,2,3,3,1,0,3,2), e t c.$

**Note:** In Theorem 5, the functions $h_i(x_{n-1}, x_{n-2}, \ldots, x_0)$, for $i = 1, 2, \ldots, n-1$ can be obtained from $h_0(x_{n-1}, x_{n-2}, \ldots, x_0) = (a_{n-1}, a_{n-2}, \ldots, a_0)$ by a simple permutation of $a_i$’s. Let $i = (i_{n-1}, i_{n-2}, \ldots, i_0)$ in binary. To get $h_i(x_{n-1}, x_{n-2}, \ldots, x_0)$ if $i_j = 1$ for $j = 0, 1, 2, \ldots, n-1$, then permute the least 0-th $2^i$ bits of $h_0$ with the next 1-st $2^i$ bits, the 2-nd $2^i$ bits with the 3-rd $2^i$ bits, etc. For example, suppose $h_0(X) = h_0(x_7, x_6, \ldots, x_0) = (a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0)$

Then $h_1(X) = h_0(x_7, x_6, \ldots, x_0) = (a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0)$

$$h_2(X) = h_1(x_7, x_6, \ldots, x_0) = (a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0)$$

$$h_3(X) = h_2(x_7, x_6, \ldots, x_0) = (a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0)$$

$$(3,3,3,2,3,2,1,0) = (3,3,3,2,3,2,1,0).$$

Thus in the above example, $h_0(X) = (2,1,1,3,2,3,0,1)$

$$h_1(X) = (3,2,3,3,1,0,3,2)$$

$$h_2(X) = (3,2,3,3,1,0,3,2)$$

$$h_3(X) = (3,3,3,2,3,2,1,0)$$

$$h_4(X) = (0,1,2,3,2,3,0,3,2)$$

$$h_5(X) = (1,0,3,2,3,2,3,0,3,2)$$

$$h_6(X) = (2,3,0,1,3,1,2,3)$$

$$h_7(X) = (3,2,1,0,3,3,3,0,3,2)$$

**Example 4** Based on Theorem 5, Figure 2 shows the decomposition of $C_3 \times C_3 \times C_3 \times C_3$ into two edge disjoint Hamiltonian cycles (solid and dotted lines).

**5. Hypercubes**

The $n$-dimensional hypercube, $Q_n$, can be defined as

$$Q_n = Q_{n-1} \times Q_1$$

where $Q_1$ is a line joining two nodes (0,1). A two dimensional hypercube, $Q_1 \times Q_1$, is isomorphic to $C_4$. This can be seen by the mapping 00 $\leftrightarrow$ 01 $\leftrightarrow$ 11 $\leftrightarrow$ 10 $\leftrightarrow$ 2. Thus the method described in Section 4.3 can be used to generate the edge disjoint Hamiltonian cycles.

**Example 5** Fig 5 shows two edge disjoint Hamiltonian cycles in $Q_4$.

**6. Conclusion**

The Lee distance Gray codes presented in this paper are shown to be useful in edge disjoint decompositions of higher dimensional torus to lower dimensional tori - in particular edge disjoint Hamiltonian cycles. The mapping functions, which map a node in radix number system to the corresponding Gray codewords are simple. In fact, for $k$-ary $n$-cube, it is required to calculate only one mapping function,
which gives the first Gray code and so the first Hamiltonian cycle. The other $n - 1$ mappings are some simple permutation of this function. Because of the space limitations, the results are given for $n$, the dimension of torus, to be a power of 2. For other cases, similar results will be given in the future.

References