A HIGHLY SYMMETRIC HAMILTON DECOMPOSITION FOR HYPERCUBES

FARID BOUYA, EBADOLLAH S. MAHMOODIAN, MODJTABA SHOKRIAN ZINI, AND MOJTABA TEFAGH

Abstract. A Hamilton decomposition of a graph is a partitioning of its edge set into disjoint spanning cycles. The existence of such decompositions is known for all hypercubes of even dimension $2n$. We give a decomposition for the case $n = 2^a3^b$ that is highly symmetric in the sense that every cycle can be derived from every other cycle just by permuting the axes. We conjecture that a similar decomposition exists for every $n$.

Key words: Hamilton decomposition, Hypercubes

1. Introduction

Hypercubes are widely used in computer architectures in areas like parallel computing [5], multiprocessor systems [2], processor allocation [6], and fault-tolerant computing [1]. Hamilton decomposition (H.D.) of hypercubes is of central importance in the aforementioned areas.

In 1954, Ringel showed that the hypercube $Q_n$ is Hamilton decomposable whenever $n$ is a power of two and posed the problem of whether a similar decomposition exists for all even $n$ [3]. In 1982, Aubert and Schneider showed that every $Q_{2n}$ admits a Hamilton decomposition [3]. Many different algorithms and methods have been used to find explicit Hamilton decompositions for $Q_{2n}$. Our work is inspired by two such methods. Okuda and Song [4] gave a direct approach for finding Hamilton decompositions for $Q_{2n}$ with $n \leq 4$. Mollard and Ramras [3] gave a fast and efficient method of generating and storing Hamilton decompositions when $n$ is a power of two by constructing one special cycle and permuting the axes to obtain the other cycles. We use Okuda and Song’s method to continue the work of Mollard and Ramras and extend it to all $n$ of the form $2^a3^b$, which is the main result of this paper, stated formally as Corollary 5.7. In Appendix B, we present Algorithms 4 and 5 that efficiently construct such decompositions. We conjecture that a similar decomposition exists for every $n$.

2. Notations

The hypercube of dimension $n$, denoted by $Q_n$, is the graph whose vertices are the $2^n$ binary strings of length $n$ and two vertices are adjacent if and only if their corresponding strings differ in exactly one bit. The Cartesian product of two graphs $G$ and $H$, denoted by $G \square H$, has vertex set

$$ V(G \square H) = \{(u, v) | u \in V(G) \text{ and } v \in V(H)\} $$

and two vertices $(u, v)$ and $(u', v')$ are connected if and only if

- $u = u'$ and $vv' \in E(H)$, or
- $v = v'$ and $uu' \in E(G)$.

Date: April 7, 2020.
Using this definition, it is not hard to see that

\[ Q_{m+n} = Q_m \boxtimes Q_n, \]

\[ Q_n = K_2 \boxtimes K_2 \boxtimes \cdots \boxtimes K_2, \]

and

\[ Q_{2n} = C_4 \boxtimes C_4 \boxtimes \cdots \boxtimes C_4, \]

(2.0.1)

As in [7], we use Equation (2.0.1) to make another coordinate system for the vertices of \(Q_{2n}\):

Each vertex is assigned a quaternary string \(q_1q_2 \ldots q_n\) of length \(n\), where \(q_i \in \{0, 1, 2, 3\}\). There is an edge between two vertices if and only if their labels differ in exactly one position, and in that position, their difference is either 1 or \(-1\) modulo 4. We wish to consider directed cycles, so we assign directions to edges of \(Q_{2n}\) as follows: A dimension-\(k\) edge in \(Q_{2n}\) is an edge that connects two vertices whose quaternary labels differ in the \(k\)th digit. If a dimension-\(k\) edge is directed in the positive direction, that is, it is directed from \((x_1, x_2, \ldots, x_{k-1}, x_k, x_{k+1}, \ldots, x_n)\) towards \((x_1, x_2, \ldots, x_{k-1}, x_k + 1 \pmod{4}, x_{k+1}, \ldots, x_n)\), we show it by \(k\), and if it is directed in the opposite direction, we show it by \(k\). A Hamilton cycle in a graph is a cycle visiting all the vertices. A Hamilton decomposition of \(Q_{2n}\) is a partitioning of its edge set into \(n\) disjoint Hamilton cycles. We use the notation given in [4] to show directed cycles: We start from the initial vertex, and simply move in the positive direction of \(C\), writing down the dimension and the direction of the edges we pass. For example, the cycle given in Figure 1, with the origin (top left vertex) as its initial vertex, is shown by 21122112211.

In this paper, we only deal with Hamilton cycles, so the initial vertex is always taken to be the origin, that is, the vertex \(0 = (0, 0, \ldots, 0)\).

**Definition 2.1.** Define \(G_{n,k}\) to be the graph \(C_4 \boxtimes C_4 \boxtimes \cdots \boxtimes C_4\).

Note that \(Q_{2k} \cong G_{1,k}\). We show directed edges and cycles in \(G_{n,k}\) the same way we show them in \(Q_{2n}\). The only difference is that coordinates in \(G_{n,k}\) are calculated modulo \(4^n\), whereas they are calculated modulo 4 in \(Q_{2n}\). We are especially interested in the cases \(k = 2\) and \(k = 3\), so we recognize that these cases require special treatment. Since \(G_{n,2}\) is the Cartesian product of two cycles, we think of \(G_{n,2}\) as a 2-dimensional cyclic grid. Every vertex of \(C_4 \boxtimes C_4\) has coordinates \((u, v)\), where \(u\) is in the first copy of \(C_4\) and \(v\) is in the second copy. We think of this coordinate \((u, v)\) in two ways:
(1) The vertices \( u \) and \( v \) are elements of \( Q_{2n} \), and thus quaternary strings of length \( n \). Therefore, \((u, v)\) is a quaternary string of length \( 2n \).

(2) Fixing some order in \( Q_{2n} \), we assign the integers 0 to \( 4^n - 1 \) to its vertices. Thus, every vertex in \( Q_{4n} \) has integral coordinates \((u, v)\) where \( 0 \leq u, v \leq 4^n - 1 \).

In order to derive Hamilton decompositions for larger hypercubes from smaller hypercubes, we study the graphs \( G_{n,2} \) and \( G_{n,3} \) in more detail.

3. The 2-Dimensional Case

We start by finding an H.D. for \( G_{n,2} \). We will see how an H.D. for \( G_{n,2} \) and an H.D. for \( Q_{2n} \) can be combined to give an H.D. for \( Q_{4n} \).

3.1. An H.D. for \( G_{n,2} \)

There is an H.D. for the graph \( C_m \square C_m \) in

\[
H_1 = \begin{array}{cccccccc}
1 & \cdots & 1 & 2 & 1 & \cdots & 1 & 2 \\
\text{m-1 times} & \text{m-1 times} & \text{m-1 times} & \text{m-1 times} & \text{m-1 times} & \text{m times}
\end{array}
\]

\[
H_2 = \begin{array}{cccccccc}
1 & \cdots & 2 & 2 & 1 & \cdots & 2 & 2 \\
\text{m-1 times} & \text{m-1 times} & \text{m-1 times} & \text{m-1 times} & \text{m-1 times} & \text{m times}
\end{array}
\]

Since \( G_{n,2} = C_4 \square C_4 \), we have the same type of H.D. for \( G_{n,2} \):

\[
H_1 = \begin{array}{cccccccc}
1 & \cdots & 1 & 2 & 1 & \cdots & 1 & 2 \\
4^n-1 times & 4^n-1 times & 4^n-1 times & 4^n-1 times & 4^n times
\end{array}
\]

\[
H_2 = \begin{array}{cccccccc}
1 & \cdots & 2 & 2 & 1 & \cdots & 2 & 2 \\
4^n-1 times & 4^n-1 times & 4^n-1 times & 4^n-1 times & 4^n times
\end{array}
\]

3.2. Deriving an H.D. for \( Q_{4n} \) From an H.D. for \( Q_{2n} \)

Noting that \( Q_{2n} \) has order \( 4^n \) and \( Q_{4n} = Q_{2n} \square Q_{2n} \), we propose the following:

**Definition 3.1.** Let \( E \) be a directed Hamilton cycle in \( Q_{2n} \). A 2-dimensional seating of \( Q_{4n} \) onto \( G_{n,2} \) via \( E \), is a representation of the vertices of \( Q_{4n} \) by assigning them integral coordinates as follows:

1. Consider \( E \) and its positive direction. Take \( 0 \) as the initial vertex. Assign 0 to \( 0 \), assign 1 to the next vertex in \( E \), and continue until \( 4^n - 1 \) is assigned to the last vertex of \( E \).
2. Induce the order of \( E \) onto \( Q_{2n} \), so that each vertex has the same order in either graph.
3. Using the coordinates in [2], assign coordinates to every member of \( Q_{4n} = Q_{2n} \square Q_{2n} \). Put the vertices on the 2-dimensional grid using their coordinates.

Using the natural order of \( E \), we have mapped the vertices of \( Q_{4n} \) onto \( G_{n,2} \) and recognized \( Q_{4n} \) as a supergraph of \( G_{n,2} \). Any subgraph of \( G_{n,2} \), therefore, is also a subgraph of \( Q_{4n} \). In particular, if \( H \) is a directed Hamilton cycle in \( G_{n,2} \), the 2-dimensional directed Hamilton cycle derived from \( E \) and \( H \), denoted by \( f(E, H) \), is a Hamilton cycle in \( Q_{4n} \) and is defined in the natural way:

1. 2-dimensionally seat \( Q_{4n} \) onto \( G_{n,2} \) via \( E \).
2. \( Q_{4n} \) has \( 2n \) axes \( 1, 2, \ldots, 2n \), while \( G_{n,2} \) has an \( x \)-axis and a \( y \)-axis. The axes \( 1, 2, \ldots, n \) are in direction \( x \) and the axes \( n + 1, n + 2, \ldots, 2n \) are in direction \( y \).
3. \( f(E, H) \) has the same edges in the supergraph \( Q_{2n} \square Q_{2n} \) as \( H \) has in the subgraph \( G_{n,2} \).

The following lemma is very useful.
Lemma 3.2. Let $H_1$ and $H_2$ be two disjoint Hamilton cycles in $G_{n,2}$ (which form an H.D.) and $E_1$ and $E_2$ be two disjoint Hamilton cycles in $Q_{2n}$. Then the four Hamilton cycles $F_1 = f(E_1, H_1)$, $F_2 = f(E_1, H_2)$, $F_3 = f(E_2, H_1)$, and $F_4 = f(E_2, H_2)$ in $Q_{4n}$ are pairwise disjoint.

Proof. It suffices to show that $F_1 = f(E_1, H_1)$ is disjoint from the other three cycles $F_2$, $F_3$, and $F_4$. To achieve this, we 2-dimensionally seat $Q_{4n}$ onto $G_{n,2}$ via $E_1$. This enables us to see that $F_1$ and $F_2$ have all their edges on the grid, whereas $F_3$ and $F_4$ have all their edges off the grid. This means that $F_1$ is disjoint from $F_3$ and from $F_4$. Furthermore, $F_1$ and $F_2$ represent $H_1$ and $H_2$, respectively, and $H_1$ and $H_2$ are disjoint, so $F_1$ and $F_2$ must be disjoint as well. $\square$

This provides us with a recursive tool to construct Hamilton decompositions.

Corollary 3.3. If $\{H_1, H_2\}$ is an H.D. for $G_{n,2}$ and $\{E_1, E_2, \ldots, E_n\}$ is an H.D. for $Q_{2n}$, then the family $\{f(E_i, H_j) \mid 1 \leq i \leq n, 1 \leq j \leq 2\}$ is an H.D. for $Q_{4n}$. The new Hamilton cycles are named $F_1, F_2, \ldots, F_{2n}$ via $F_j = f(E_j, H_1)$ and $F_{j+n} = f(E_j, H_2)$ for $1 \leq j \leq n$.

3.3. 2-Dimensional Algorithm.

Using the definition of $f(E, H)$, it is not difficult to devise an algorithm for computing an H.D. for $Q_{4n}$. Algorithm III given in Appendix III takes an H.D. for $G_{n,2}$ and an H.D. for $Q_{2n}$ as inputs, and outputs an H.D. for $Q_{4n}$.

4. The 3-Dimensional Case

Just like in the 2-dimensional case, finding an H.D. for the graph $G_{n,3}$ is essential for transitioning from $Q_{2n}$ to $Q_{6n}$. An H.D. for $Q_{2n}$ can be combined with an H.D. for $G_{n,3}$ to give an H.D. for $Q_{6n}$.

4.1. An H.D. for $G_{n,3}$.

Compared to the 2-dimensional case, finding an H.D. for $G_{n,3}$ is not easy. Motivated by [3] and [7], we decompose the graph into three 2-factors, and then try to merge the components until we have three Hamilton cycles.

Lemma 4.1. The graph $G_{n,3}$ with the partitioning given below decomposes into $3 \times 4^n$ copies of the directed cycle with $4^{2n}$ edges:

If $e$ is in direction 1 and is between $(x, y, z)$ and $(x+1, y, z)$, we direct $e$ from $(x, y, z)$ to $(x+1, y, z)$ and

$$e \in Z \quad \text{if } x + y + z = -1 \pmod{4^n},$$
$$e \in X \quad \text{otherwise}.$$

If $e$ is in direction 2 and is between $(x, y, z)$ and $(x, y+1, z)$, we direct $e$ from $(x, y, z)$ to $(x, y+1, z)$ and

$$e \in X \quad \text{if } x + y + z = -1 \pmod{4^n},$$
$$e \in Y \quad \text{otherwise}.$$

If $e$ is in direction 3 and is between $(x, y, z)$ and $(x, y, z+1)$, we direct $e$ from $(x, y, z)$ to $(x, y, z+1)$ and

$$e \in Y \quad \text{if } x + y + z = -1 \pmod{4^n},$$
$$e \in Z \quad \text{otherwise}.$$

We have demonstrated the case $n = 1$ in Appendix A.
Proof. Choosing an arbitrary vertex \( v \) and moving along the edges of \( X \), we can see that \( v \) belongs to a unique cycle of length \( 4^{2n} \) that is in \( X \). Similarly, it belongs to a unique cycle of length \( 4^{2n} \) in \( Y \) and another one in \( Z \). There are \( 4^{3n} \) vertices in total, so there are \( 4^n \) cycles in each of \( X, Y, \) and \( Z \), for a total of \( 3 \times 4^n \) cycles. \( \square \)

We wish to merge these cycles together and end up with just three, so that we have an H.D. for \( G_{n,3} \). To this end, we introduce two cubes and a merge operation. These cubes and the merge operation were first introduced in [7] and later in [4] to build an H.D. for \( Q_6 \). We use them to construct an H.D. for every \( G_{n,3} \).

**Definition 4.2.** Let \( c_X, c_Y, \) and \( c_Z \), denote the number of (current) connected components of \( X, Y, \) and \( Z \), respectively. The type-I cube and the type-II cube are given in Figure 2. The top left vertex is the origin of the cube, that is, the vertex \((x, y, z)\) such that any other vertex \((x', y', z')\) of the cube satisfies \(0 \leq x - x', y - y', z - z' \leq 1 \pmod{4^n}\).

![Figure 2](image)

**Figure 2.** The special cube type-I (2a) and the special cube type-II (2b).

By merging a type-I cube we replace it with a type-II cube. Note that the vertices maintain their \( X \)-, \( Y \)-, and \( Z \)-degrees during the merge operation.

The aim of the merge operation is to reduce each of \( c_X, c_Y, \) and \( c_Z \) by 1. Before starting to merge, we need to make sure that we have enough type-I cubes and that this three-way switch in colors indeed merges six cycles into three. We make a couple of observations.

**Observation 4.3.** Consider \( G_{n,3} \) and decompose it with the method described in Lemma 4.1. Then every vertex \((x, y, z)\) with \(x + y + z = -1 \pmod{4^n}\) is the origin of a type-I cube.

**Observation 4.4.** Figure 3 shows that, a single merge operation, done on the decomposition obtained from Lemma 4.1, indeed merges six cycles, two in each of \( X, Y, \) and \( Z \), into three cycles, one in each of \( X, Y, \) and \( Z \).
Figure 3. The cycles merged during the merge operation

Of course, we need another $4^n - 2$ of these merge operations, and as we progress, the structures of the cycles change, which could possibly cause a merge operation to “fail” to combine six cycles into three. Hence, Lemma H.S is crucial.

**Definition 4.5.** For $i \leq j$ let $[i, j]$ be the set $\{i, i + 1, \ldots, j\}$. For $0 \leq i < 4^n$, define $Z^n_i$ to be the set of vertices of $G_{n,3}$ that have their 3rd coordinate equal to $i$. Finally, let

$$Z^n_{[i,j]} = \bigcup_{i \leq k \leq j} Z^n_k$$

**Definition 4.6.** Let $C$ be a cycle and $S$ be a subset of $V(C)$. The $C$-necklace-order with respect to $S$ is the order in which the vertices of $S$ appear in $C$. As its name suggests, shifting or reversing the direction of $C$ does not change its order (with respect to any vertex set).
Observation 4.7. Let \( v = (x, y, z) \) be the origin of a type-I cube \( L \). Figure 4 shows that the \( X \cap Z^n_x \)-necklace-order with respect to \( Z^n_x \) (before merge) is the same as \( X \cap Z^n_{x+1} \)-necklace-order with respect to \( Z^n_x \) (after merge). Indeed, the only change to \( X \cap Z^n_x \) is the removal of the edge \( uv \), which is replaced by a detour through \( Z^n_{x+1} \). This augmentation does not change the order of vertices of \( Z^n_x \).

![Figure 4](image_url)

**Figure 4.** The \( X \cap Z^n_x \)-necklace-order with respect to \( Z^n_x \) (left) is the same as the \( X \cap Z^n_{x+1} \)-necklace-order with respect to \( Z^n_x \) (right).

Lemma 4.8. Suppose that we just decomposed the edge set of \( G_{n,3} \) with the method given in Lemma 4.1. Let \( v = (x, y, z) \) and \( v' = (x', y', z') \) be such that \( x + y + z = x' + y' + z' = -1 \) (mod \( 4^n \)) and \( z = z' + 1 \) (mod \( 4^n \)), and let \( L \) and \( L' \) be the type-I cubes with origins at \( v \) and \( v' \), respectively. If we merge \( L \) first and then \( L' \), we reduce \( c_X \) by 2.

**Proof.** We saw in Observation 4.7 that a single merge operation always succeeds. Suppose that we have merged \( L \), so that \( Z^n_{x+1} \) and \( Z^n_{x+2} \) have merged into \( Z^n_{x+1,x+2} \), and we are about to merge \( L' \). By Observation 4.7, the order of vertices in \( Z^n_{x+1} \) has not changed, so merging \( L' \) will successfully combine \( Z^n_{x+1,x+2} \) and \( Z^n_{x+2} \) into a single cycle \( Z^n_{x+2} \). \( \square \)

We now specify a condition under which all the merge operations are guaranteed to succeed.

**Definition 4.9.** Let \( S \subseteq [0, 4^n - 1]^3 \). We say that \( S \) is a **merging set** if it satisfies the following:

- \( |S| = 4^n - 1 \),
- Members \((x, y, z)\) of \( S \) satisfy \( x + y + z = -1 \) (mod \( 4^n \)), and
- Distinct members \((x, y, z)\) and \((x', y', z')\) of \( S \) satisfy \( x \neq x' \), \( y \neq y' \), and \( z \neq z' \).

We need \( 4^n - 1 \) merge operations, each merging six cycles into three. In order for all these operations to successfully take place, it suffices for the type-I cubes to have their origins in a merging set. This we show next.

**Lemma 4.10.** Consider the following procedure:

i Decompose the edge set of \( G_{n,3} \) with the method given in Lemma 4.1.

ii Select a merging set \( S \).

iii Recognize the \( 4^n - 1 \) type-I cubes that have their origins in \( S \).

iv Replace each type-I cube with a type-II cube.
The following statements hold:

(1) After completing step \[3\] we have \( c_X = c_Y = c_Z = 4^n \), with different components of \( X \) being \( Z_i^n \)'s, different components of \( Y \) being \( X_i^n \)'s, and different components of \( Z \) being \( Y_i^n \)'s.

(2) The type-I cubes are pairwise disjoint.

(3) After fixing \( S \) in step \[3\] and the type-I cubes in step \[3\] throughout step \[3\]

\[ x_1 + x_2 + x_3 + i_1 + i_2 + i_3 = x_1' + x_2' + x_3' + i_1' + i_2' + i_3' \quad (\text{mod } 4^n), \]

so we obtain \( i_1 + i_2 + i_3 = i_1' + i_2' + i_3' \) (mod 2). This implies that \( i_r = i_r' \) for some \( r \), meaning that \( x_r = x_r' \) (mod 4) for the same \( r \). This gives \( x_r = x_r' \), which contradicts the assumption that \( S \) is a merging set.

(3) Due to the symmetry involved in \[3\], it suffices to prove the assertions in just one direction, that is, to prove

- The vertices of \( Z_i^n \) remain in the same component of \( X \), and
- Every merge operation reduces \( c_X \) by 1.

Without loss of generality, suppose that \( S = \{v_0, v_1, \ldots, v_{4^n-2}\} \), where \( v_i = (x_i, y_i, i) \), and let \( L_i \) be the type-I cube with origin at \( v_i \). Because of (2), the order in which we merge the cubes does not matter, so for the sake of simplicity, assume that \( L_{4^n-2} \) is merged first, \( L_{4^n-3} \) is merged next, and so on.

We proceed by induction. The base case is satisfied due to Observation [4.4] and Lemma [4.8]. Suppose that we have merged cubes \( L_{4^n-2} \) to \( L_i \). The induction hypothesis states that we have cycles \( X \cap Z_i^n, X \cap Z_{i+1}^n, \ldots, X \cap Z_{4^n-1}^n \), and a long cycle \( X \cap Z_{[i,4^n-1]}^n \). It also states that the vertices of \( Z_i^n \) have been in the same component of \( X \) together throughout step \[3\].

By Observation [4.7], the order of the vertices in \( Z_i^n \) has not changed yet, so merging \( L_{i+1} \) will combine \( X \cap Z_{[i,4^n-1]}^n \) and \( X \cap Z_{[i+1,4^n-1]}^n \) into a single cycle \( X \cap Z_{[i-1,4^n-1]}^n \). It is clear that the vertices of \( Z_i^n \) have remained and will remain in the same component of \( X \).

\[ \square \]
4.2. Deriving an H.D. for $Q_{6n}$ from an H.D. for $Q_{2n}$.

Combining the Hamilton decompositions for $Q_{2n}$ and $G_{n,3}$ is very similar to the 2-dimensional case. We think of $G_{n,3}$ as a 3-dimensional cyclic grid, and assign coordinates like $(x, y, z)$ to its vertices.

**Definition 4.11.** Let $E$ be a directed Hamilton cycle in $Q_{2n}$. A *3-dimensional seating* of $Q_{6n}$ onto $G_{n,3}$ via $E$, is a representation of the vertices of $Q_{6n}$ by assigning them integral coordinates as follows:

1. Consider $E$ and its positive direction. Assign 0 to 0, assign 1 to the next vertex in $E$, and continue until $4^n - 1$ is assigned to the last vertex of $E$.

2. Induce the order of $E$ onto $Q_{2n}$, so that each vertex has the same order in either graph.

3. Using the coordinates in $E$, assign coordinates to every member of $Q_{6n} = Q_{2n} \square Q_{2n} \square Q_{2n}$.

Put the vertices on the 3-dimensional grid using their coordinates.

Using the natural order of $E$, we have mapped the vertices of $Q_{6n}$ onto $G_{n,3}$ and recognized $Q_{6n}$ as a supergraph of $G_{n,3}$. If $H$ is a directed Hamilton cycle in $G_{n,3}$, the *3-dimensional directed Hamilton cycle derived from* $E$ and $H$, denoted by $g(E, H)$, is a Hamilton cycle in $Q_{6n}$ and is defined in the natural way:

1. 3-dimensionally seat $Q_{6n}$ onto $G_{n,3}$ via $E$.

2. $Q_{6n}$ has $3n$ axes $1, 2, \ldots , 3n$, while $G_{n,3}$ has an $x$-axis, a $y$-axis, and a $z$-axis. The axes $1, 2, \ldots , n$ are in direction $x$, the axes $n + 1, n + 2, \ldots , 2n$ are in direction $y$, and the axes $2n + 1, 2n + 2, \ldots , 3n$ are in direction $z$.

3. $g(E, H)$ has the same edges in the supergraph $Q_{2n} \square Q_{2n} \square Q_{2n}$ as $H$ has in the subgraph $G_{n,3}$.

**Lemma 4.12.** Let $H_1$ and $H_2$ be two disjoint Hamilton cycles in $G_{n,3}$ and $E_1$ and $E_2$ be two disjoint Hamilton cycles in $Q_{2n}$. Then the four Hamilton cycles $G_1 = g(E_1, H_1)$, $G_2 = g(E_1, H_2)$, $G_3 = g(E_2, H_1)$, and $G_4 = g(E_2, H_2)$ in $Q_{6n}$ are pairwise disjoint.

**Proof.** We show that $G_1 = g(E_1, H_1)$ is disjoint from the other three cycles $G_2$, $G_3$, and $G_4$. To achieve this, we 3-dimensionally seat $Q_{6n}$ onto $G_{n,3}$ via $E_1$. Similar to the 2-dimensional case, $G_1$ and $G_2$ have all their edges on the grid, while $G_3$ and $G_4$ have all their edges off the grid. Thus $G_1$ is disjoint from $G_3$ and $G_4$. Furthermore, $G_1$ and $G_2$ represent $H_1$ and $H_2$, respectively, and $H_1$ and $H_2$ are disjoint, so $G_1$ and $G_2$ must be disjoint as well. 

**Corollary 4.13.** If $\{H_1, H_2, H_3\}$ is an H.D. for $G_{n,3}$ and $\{E_1, E_2, \ldots , E_n\}$ is an H.D. for $Q_{2n}$, then the family $\{g(E_i, H_j) \mid 1 \leq i \leq n, 1 \leq j \leq 3\}$ is an H.D. for $Q_{6n}$. The new Hamilton cycles are named $F_1, F_2, \ldots , F_{3n}$ via $F_j = f(E_j, H_1)$, $F_{j+n} = f(E_j, H_2)$, and $F_{j+2n} = f(E_j, H_3)$ for $1 \leq j \leq n$.

4.3. An algorithm for computing an H.D. for $G_{n,3}$.

In section 4.1, we saw how to derive a Hamilton decomposition $\{X, Y, Z\}$ from the initial partitioning given by Lemma 4.1. We now give an algorithm to compute $X$. Algorithms for $Y$ and $Z$ are similar.

The idea is to apply the edge decomposition given in Lemma 4.1 and then proceed from the origin, initially moving in the positive direction of $X$, until we reach a chosen type-I cube (one whose origin belongs to the merging set). We then recognize the special vertex, take the necessary actions mandated by the merge operation, and continue to walk in $X$. Figure 6 shows all the special vertices and the reasoning behind our actions. For example, if we reach $m'$ and the current direction
is negative, it means that we came from outside of the cube (and not from \(m\)), so we should go to \(m\) and change the direction to positive, so that we move outside in the next step. If we reach \(m'\) and the current direction is positive, however, it means that we came from \(m\) (and not from outside), so we should move outside and leave the direction unchanged.

![Image of a merge operation together with the attached X-edges. The above vertex labelling conforms to that of Algorithm 2.](image)

We choose the merging set to be

\[
S = \left\{ (0, 0, 4^t - 1), (1, 1, 4^t - 3), \ldots, \left(\frac{4^t}{2} - 1, \frac{4^t}{2} - 1, 1\right), \left(\frac{4^t}{2} + 1, \frac{4^t}{2} + 1, 4^t - 2\right), \left(\frac{4^t}{2} + 1, \frac{4^t}{2} + 2, 4^t - 4\right), \ldots, (4^t - 2, 4^t - 1, 2) \right\}.
\]

We choose \(S\) like this for two reasons:

- The origin does not belong to any of the type-I cubes, so we do not need an initial case check.
- \(S\) has all the \(x\)-coordinates from 0 to \(4^n - 2\), so it is easy to check if a coordinate belongs to it.

We define five helping sets \(S', D, D', M,\) and \(M'\) so that we have instant access to all the special vertices. Algorithm 2 given in Appendix B calculates \(X\).

4.4. 3-Dimensional Algorithm.

Just like in the 2-dimensional case, we use the definition of \(g(E, H)\) to devise an algorithm for computing an H.D. for \(Q_{6n}\). Algorithm 3 is very similar to Algorithm 1 and is given in Appendix B.

5. Highly Symmetric Hamilton Decompositions

The theory we have developed in the previous chapters can be improved to give us highly symmetric Hamilton decompositions. Let \(\sigma : [1, k] \rightarrow [1, k]\) be a permutation. Then \(\sigma\) induces a homomorphism of \(G_{n,k}\) by relabelling the axes: The axis previously referred to as \(i\) is now called \(\sigma(i)\). More specifically, the vertex \(v = (x_1, x_2, \ldots, v_k)\) is mapped to \(\sigma(v) = (x_{\sigma^{-1}(1)}, x_{\sigma^{-1}(2)}, \ldots, x_{\sigma^{-1}(k)})\). As \(\sigma\)
is a homomorphism, it maps Hamilton cycles to Hamilton cycles. If $H = e_1 e_2 \ldots e_{4n} k$ is a directed Hamilton cycle in $G_{n,k}$, then $\sigma(H)$ is the Hamilton cycle

$$\sigma(e_1)\sigma(e_2)\ldots\sigma(e_{4n})$$

Note that $\sigma$ maps backward edges to backward edges: If $\sigma(i) = j$, then $\sigma(\overline{i}) = \overline{j}$. It is worth remembering that $\overline{i}$ stands for an edge from $(x_1, x_2, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_n)$ to $(x_1, x_2, \ldots, x_{i-1}, x_i - 1 \pmod{4^n}, x_{i+1}, \ldots, x_n)$.

**Definition 5.1.** A family $S = \{\sigma_1, \sigma_2, \ldots, \sigma_k\}$ of $k$ permutations on $[1, k]$ is called a Latin family if the matrix $m_{ij} = \sigma_i(j)$ is a Latin square. We do not differentiate between $\sigma_i$ and the $i$th row of the matrix. For the sake of simplicity, we require that $\sigma_1$, the first row of the matrix, is the identity.

Let $T = \{H_1, H_2, \ldots, H_k\}$ be an H.D. for $G_{n,k}$. We say that $T$ is a *Latin Hamilton decomposition* if there exists a Hamilton cycle $H$ in $G_{n,k}$ and a Latin family $S = \{\sigma_1, \sigma_2, \ldots, \sigma_k\}$ of permutations on $[1, k]$ such that

$$H_i = \sigma_i(H) \quad 1 \leq i \leq k$$

The Hamilton cycle $H (= H_1)$ is then called a *source cycle* for $G_{n,k}$ and the matrix $m_{ij} = \sigma_i(j)$ is called a source matrix for $G_{n,k}$. The pair $(H, M)$ is called a source pair for $G_{n,k}$.

The H.D. given for $G_{n,2}$ in (3.1) is Latin, but the one given for $G_{n,3}$ in (4.1) is not necessarily so. If it is not Latin, we can turn it into one with a small adjustment.

**Theorem 5.2.** The set $S$ mentioned in Lemma 4.10 step iv can be chosen in such a way that the resulting H.D. is Latin. More specifically, if

$$S^* = \left\{ \left(0, 4^t \frac{4^t - 1}{2}, 4^t \frac{4^t - 1}{2}\right), \left(1, 4^t \frac{4^t - 2}{6}, 4^t \frac{4^t - 2}{6}\right), \ldots, \left(4^t - 4, 4^t + 2, 4 \times 4^t - 4\right), \ldots, \left(\frac{4 \times 4^t + 2}{6}, \frac{5 \times 4^t + 4}{6}, \frac{4^t - 4}{6}\right), \ldots, \right\}$$

and

(5.2.1) \[ S = \{(x, y, z) \mid (x, y, z) \in S^*, \text{ or } (y, z, x) \in S^*, \text{ or } (z, x, y) \in S^* \}, \]

then the resulting H.D. is Latin.

**Proof.** We show that it suffices for $S$ to have the following property:

If $(x, y, z) \in S$, then $(y, z, x) \in S$ and $(z, x, y) \in S$.

To see this, consider $G_{n,3}$ after completion of Lemma 4.10 step iv. Let $\sigma_i : [1, 3] \to [1, 3]$ be defined via $\sigma_i(j) = i + j - 1 \pmod{3}$ for $i$ and $j$ in $[1, 3]$. It is not hard to see that

(5.2.2) \[ \sigma_2(X) = Y \text{ and } \sigma_3(X) = Z. \]

We wish to show that the relations given in (5.2.2) remain valid after completion of Lemma 4.10 step iv. To achieve this, we merge the cubes three at a time and use induction.

Suppose that $u_1 = (x, y, z)$, $u_2 = \sigma_2(u_1) = (z, x, y)$, and $u_3 = \sigma_3(u_1) = (y, z, x)$ belong to $S$, and let $L_1$, $L_2$, and $L_3$ be type-I cubes with their origins at $u_1$, $u_2$, and $u_3$, respectively. By the induction hypothesis, we know that (5.2.2) is valid before merging $L_1$, $L_2$, and $L_3$. 


Since \( u_2 = \sigma_2(u_1) \), we have \( L_2 = \sigma_2(L_1) \), and because \( u_3 = \sigma_3(u_1) \), we get \( L_3 = \sigma_3(L_1) \). Furthermore, analyzing the merge operator gives \( \sigma_2(L_1 \cap X) = L_2 \cap Y \) and \( \sigma_3(L_1 \cap X) = L_3 \cap Z \). This means that \( 5.2.2 \) is valid after merging the three cubes. Therefore \( X \) (after finishing Lemma \( 4.10 \)) step \( iv \) is a source cycle for \( G_{n,3} \) in the H.D. \( \{X, Y, Z\} \), and its source matrix is

\[
\begin{bmatrix}
1 & 2 & 3 \\
2 & 3 & 1 \\
3 & 1 & 2
\end{bmatrix}
\]

We may modify Algorithm \( 1 \) to take source pairs for \( Q_{2n} \) and \( G_{n,2} \) and produce a source pair for \( Q_{4n} \). We may also modify Algorithm \( 2 \) to take source pairs for \( Q_{2n} \) and \( G_{n,3} \) and produce a source pair for \( Q_{6n} \). Algorithms \( 4 \) and \( 5 \) are the Latin counterparts to Algorithms \( 1 \) and \( 3 \), respectively, and are given in Appendix \( B \). We may also specify that Algorithm \( 2 \) takes a suitable merging set \( 5.2.1 \) so that it produces a source cycle for \( G_{n,3} \). Hence, it is not necessary to give a Latin counterpart to Algorithm \( 2 \).

**Theorem 5.3.** If \( \{H_1, H_2\} \) and \( \{E_1, E_2, \ldots, E_n\} \) mentioned in Corollary \( 3.3 \) are Latin, then the resulting Hamilton decomposition \( \{f(E_i, H_j) \mid 1 \leq i \leq n \text{ and } 1 \leq j \leq 2\} \) is also Latin.

**Proof.** Let \( E_1 \), our source cycle for \( Q_{2n} \), have source matrix \( M \). For \( G_{n,2} \), the cycle \( H_1 \) is a source cycle and has source matrix \( \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{bmatrix} \). We show that \( F_1 = f(E_1, H_1) \) is a source cycle for \( Q_{4n} \) with

\[
M' = \begin{bmatrix}
M & M + n \\
M + n & M
\end{bmatrix}
\]

as its source matrix, where \( M + n \) is obtained from \( M \) by adding \( n \) to every entry.

Our proof is based on Algorithm \( 1 \). In Appendix \( C \) it is shown that Algorithm \( 1 \) computes \( f(E, H) \) correctly. We know that, for \( 1 \leq j \leq n \), this algorithm stores \( f(E_j, H_1) \) and \( f(E_j, H_2) \) as \( F_j \) and \( F_{j+n} \), respectively. The dimension of the \( j \)th edge of \( F_j \) is stored in \( f[j - 1][i - 1][0] \) and its direction is stored in \( f[j - 1][i - 1][1] \). Due to line \( 4 \) in the algorithm and the fact that \( H_1 \) and \( H_2 \) make a Latin decomposition, for every \( 1 \leq i \leq 4n \), either all the \( F_i \)'s have a forward edge in the \( j \)th position or all the \( F_i \)'s have a backward edge in the \( j \)th position. So the directions of the edges are as required and we only need to focus on their dimensions.

To show that the edge dimensions are as we want, we define a \( 2n \) by \( 2n \) matrix \( Q \) via

\[
q_{i,j} = t \text{ if there is some } 0 \leq s < 4^{2n} \text{ such that } f[0][s][0] = j \text{ and } f[i + 1][s][0] = t.
\]

We show that

- \( Q \) is well-defined, and
- \( Q = M' \).

This would complete the proof of the theorem.

For \( 1 \leq i \leq 2n \), let \( S_i \) be the set of edge numbers in \( F_i \) with dimension \( i \). More precisely

\[
S_i = \{j \mid 0 \leq j < 4^{2n} \text{ and } f[0][j][0] = i\}
\]

Suppose that \( 1 \leq v \leq n \) and let \( s \in S_v \). Lines \( 10 \) and \( 12 \) say that, for \( j = 0 \), \( i = s \), and \( k = 0 \), we have \( \text{dim} = 0 \) and that for \( u = c[k]\text{dim} \) we have \( f[0][s][0] = e[0][u][0] \), but \( s \in S_v \), so we have \( f[0][s][0] = e[0][u][0] = v = m_{i,v} = m_{1,v} \). Therefore, \( q_{1,v} \) is well-defined and is equal to \( m_{1,v} \). Again, due to lines \( 10 \) and \( 12 \) for \( 0 \leq w < n \), putting \( j = w \) but keeping the same \( i \) and \( k \), we have the same \( u \), and thus \( f[w][s][0] = e[w][u][0] = m_{w+1,v} \). This means that \( q_{w+1,v} \) is well-defined as is equal to \( m_{w+1,v} \). Since \( w \) and \( v \) were arbitrary in \([0, n - 1]\) and \([1, n]\), respectively, we get \( q_{w+1,v} = m_{w+1,v} \) for \( 1 \leq v \leq n \) and \( 0 \leq w < n \).

A similar argument for the other cases shows that
• for $n + 1 \leq v \leq 2n$ and $1 \leq w \leq n$ we have $q_{w,v} = m_{w,v-n} + n$,
• for $1 \leq v \leq n$ and $n + 1 \leq w \leq 2n$ we have $q_{w,v} = m_{w-n,v} + n$, and
• for $n + 1 \leq v \leq 2n$ and $n + 1 \leq w \leq 2n$ we have $q_{w,v} = m_{w-n,v-n}$.

This shows that $Q$ is well-defined and $Q = M'$.

□

As a corollary, we have the following important result.

Corollary 5.4. If $Q_{2n}$ has a source cycle, so does $Q_{4n}$.

Theorem 5.5. If $\{H_1, H_2, H_3\}$ and $\{E_1, E_2, \ldots, E_n\}$ mentioned in Corollary 4.13 are Latin, then the resulting Hamilton decomposition $\{g(E_i, H_j) \mid 1 \leq i \leq n$ and $1 \leq j \leq 3\}$ is also Latin.

Proof. The proof is very similar to that of Theorem 5.3, therefore we only sketch it here. Based on Algorithm 3 if $E_1$ is a source cycle for $Q_{2n}$ with source matrix $M$, and if $H_1$ is a source cycle for $G_{n,3}$ with source matrix $\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}$, then $g(E_1, H_1)$ is a source cycle for $Q_{6n}$ with source matrix $M' = \begin{pmatrix} M & M+n & M+2n \\ M+n & M+2n & M \\ M+2n & M & M+n \end{pmatrix}$ as its source matrix. □

The last theorem gives rise to another important result:

Corollary 5.6. If $Q_{2n}$ has a source cycle, so does $Q_{6n}$.

Corollaries 5.4 and 5.6 give us the main result of this paper:

Corollary 5.7. We have a source cycle for all $Q_{2n}$ with $n = 2^a 3^b$.

For future research, we conjecture the following.

Conjecture 5.8. We have a source cycle for all $Q_{2n}$.

ACKNOWLEDGEMENTS

We thank Negin Karisani for her useful comments during this project.

REFERENCES

[1] M. Abd-El-Barr and F. Gebali, Reliability analysis and fault tolerance for hypercube multi-computer networks, Information Sciences, 276 (2014), pp. 295–318.
[2] S. R. Das, N. H. Vaidya, and L. M. Patnaik, Design and implementation of a hypercube multiprocessor, Microprocessors and Microsystems, 14 (1990), pp. 101–106.
[3] M. Mollard and M. Ramras, Edge decompositions of hypercubes by paths and by cycles, Pre-print, (2013).
[4] K. Okuda and S. W. Song, Revisiting hamiltonian decomposition of the hypercube, in Proceedings of the 13th symposium on Integrated circuits and systems design, SBCCI’00, Washington, DC, USA, 2000, IEEE Computer Society, pp. 55–60.
[5] G. Ostrouchov, Parallel computing on a hypercube: an overview of the architecture and some applications, 03 1987.
[6] S. Rai, J. L. Trahan, and T. Smailus, Processor allocation in hypercube multiprocessors, IEEE Transactions on Parallel and Distributed Systems, 6 (1995), pp. 606–616.
[7] S. W. Song, Towards a simple construction method for hamiltonian decomposition of the hypercube, Discrete Math., 21 (1995), pp. 297–306.
A. Lemma 4.1 for $n = 1$

Figure 6. The decomposition discussed in Lemma 4.1 for $n = 1$. 
Appendix B. Algorithms

B.1. An H.D. for $Q_{4n}$.

Input:

- An $n \times 4^n$ array $e$ with its $i$th row showing the $i$th Hamilton cycle of $Q_{2n}$.
- A $2 \times 4^{2n}$ array $h$ with its $i$th row showing the $i$th Hamilton cycle of $G_{n,2}$.

Output:

- A $2n \times 4^{2n}$ array $f$ with its $i$th row showing the $i$th Hamilton cycle of $Q_{4n}$.

Algorithm 1 An H.D. for $Q_{4n}$ from an H.D. for $Q_{2n}$ and an H.D. for $G_{n,2}$

1: for $i$ ← 0 to 1 do
2:     for $j$ ← 0 to 1 do
3:         $c[i][j] ← 0$ \hspace{1cm} \triangleright initializing the $x$- and $y$-coordinates of the two pointers
4:     end for
5: end for
6: for $j$ ← 0 to $n - 1$ do \hspace{1cm} \triangleright cycling through $E_1$ to $E_n$
7:     for $i$ ← 0 to $4^{2n} - 1$ do \hspace{1cm} \triangleright cycling through $H_1$ and $H_2$
8:         for $k$ ← 0 to 1 do \hspace{1cm} \triangleright direction through edges of $H_{k+1}$
9:             dir ← $h[k][i][1]$ \hspace{1cm} \triangleright direction of the current edge in $H_{k+1}$
10:             dim ← $h[k][i][0] - 1$ \hspace{1cm} \triangleright dimension of the current edge in $H_{k+1}$
11:             if dir = 0 then \hspace{1cm} \triangleright if the current edge in $H_{k+1}$ is forward
12:                 $f[j + kn][i][0] ← e[j][c[k][dim]][0] + n(dim)$ \hspace{1cm} \triangleright dimension of the current edge in $F_{j+1+kn}$
13:                 $f[j + kn][i][1] ← e[j][c[k][dim]][1]$ \hspace{1cm} \triangleright direction of the current edge in $F_{j+1+kn}$
14:             end if
15:             $c[k][dim] ← c[k][dim] + 1$ \hspace{1cm} \triangleright moving forward in the current copy of $E_{j+1}$
16:             if $c[k][dim] = 4^n$ then \hspace{1cm} \triangleright mod operations
17:                 $c[k][dim] ← 0$
18:             end if
19:         else \hspace{1cm} \triangleright if the current edge in $H_{k+1}$ is backward
20:             $c[k][dim] ← c[k][dim] - 1$ \hspace{1cm} \triangleright moving backward in the current copy of $E_{j+1}$
21:             if $c[k][dim] = -1$ then \hspace{1cm} \triangleright mod operations
22:                 $c[k][dim] ← 4^n - 1$
23:         end if
24:         $f[j + kn][i][0] ← e[j][c[k][dim]][0] + n(dim)$ \hspace{1cm} \triangleright dimension of the current edge in $F_{j+1+kn}$
25:         $f[j + kn][i][1] ← 1 - e[j][c[k][dim]][1]$ \hspace{1cm} \triangleright direction of the current edge in $F_{j+1+kn}$
26:     end for
27: end for
28: end for
B.2. An H.D. for $G_{n,3}$.

Input:
- A $4^n \times 2$ array $S$ having the merging set $S$ in its first $4^n - 1$ rows.
  The elements of $S$ are sorted by their $x$-coordinates, with the $ith$ row of $S$ having the element with $x$-coordinate $i$. The first entry gives the $y$-coordinate and the second gives the $z$-coordinate.

Output:
- A $4^n \times 2$ array $H$ having the edges of $X$.

Algorithm 2 An algorithm for finding $X$.

1: $x \leftarrow 0$ \hspace{1cm} \triangleright \text{initializing the pointer’s } x \text{-coordinate}
2: $y \leftarrow 0$ \hspace{1cm} \triangleright \text{initializing the pointer’s } y \text{-coordinate}
3: $z \leftarrow 0$ \hspace{1cm} \triangleright \text{initializing the pointer’s } z \text{-coordinate}
4: $c \leftarrow 0$ \hspace{1cm} \triangleright c = x + y + z
5: dir $\leftarrow 0$
6: $s[4^n - 1][0] \leftarrow -2$ \hspace{1cm} \triangleright \text{no element of $S$ has } x \text{-coordinate equal to } 4^n - 1$
7: $s[4^n - 1][1] \leftarrow -2$
8: for $i \leftarrow 0$ to $4^n - 1$ do
9:   $sp[i][0] \leftarrow s[i][0]$ \hspace{1cm} \triangleright \text{creating the } ith \text{ member of } S'$
10:  $sp[i][1] \leftarrow s[i][1] + 1$
11:  if $sp[i][1] = 4^n$ then \hspace{1cm} \triangleright \text{mod operations}
12:     $sp[i][1] \leftarrow 0$
13:  end if
14:  $d[i][0] \leftarrow s[i][0] + 1$ \hspace{1cm} \triangleright \text{creating the } ith \text{ member of } D$
15:  $d[i][1] \leftarrow s[i][1]$
16:  if $d[i][0] = 4^n$ then \hspace{1cm} \triangleright \text{mod operations}
17:     $d[i][0] \leftarrow 0$
18:  end if
19:  $dp[i][0] \leftarrow d[i][0]$ \hspace{1cm} \triangleright \text{creating the } ith \text{ member of } D'$
20:  $dp[i][1] \leftarrow sp[i][1]$
21:  $m[i + 1][0] \leftarrow s[i][0] + 1$ \hspace{1cm} \triangleright \text{creating the } ith \text{ member of } M$
22:  $m[i + 1][1] \leftarrow sp[i][1]$\hspace{1cm} 
23:  $mp[i + 1][0] \leftarrow d[i][0]$ \hspace{1cm} \triangleright \text{creating the } ith \text{ member of } M'$
24:  $mp[i + 1][1] \leftarrow sp[i][1]$
25: end for
26: $m[0][0] \leftarrow -1$ \hspace{1cm} \triangleright \text{no element of } M \text{ has } x \text{-coordinate equal to } 0$
27: $m[0][1] \leftarrow -1$
28: $mp[0][0] \leftarrow -1$ \hspace{1cm} \triangleright \text{no element of } M' \text{ has } x \text{-coordinate equal to } 0
29: $mp[0][1] \leftarrow -1$
for $i \leftarrow 0$ to $4^{3n} - 1$ do

\[ \text{\Comment{main loop for building the $i$th edge}} \]

if $s[x][0] = y$ and $s[x][1] = z$ then

\[ \text{\Comment{(x, y, z) \in S}} \]

if dir = 0 then

\[ \text{\Comment{we have reached S from outside of cube}} \]

\[ \text{\Comment{in the next step we exit from $S'$ in negative direction}} \]

\[ \text{\Comment{the $i$th edge is in dimension 3}} \]

\[ \text{\Comment{adding 1 to $z$ and $c$}} \]

\[ h[i][0] \leftarrow 3 \]

\[ h[i][1] \leftarrow 0 \]

\[ c \leftarrow c + 1 \]

\[ z \leftarrow z + 1 \]

\[ \text{if } z = 4^n \text{ then} \]

\[ \text{\Comment{mod operations}} \]

\[ z \leftarrow 0 \]

end if

else

\[ \text{\Comment{we have reached $S$ from $S'$}} \]

\[ \text{\Comment{the $i$th edge is in dimension 1}} \]

\[ \text{\Comment{subtracting 1 from $x$ and $c$}} \]

\[ h[i][0] \leftarrow 1 \]

\[ h[i][1] \leftarrow 1 \]

\[ c \leftarrow c - 1 \]

\[ x \leftarrow x - 1 \]

\[ \text{if } x = -1 \text{ then} \]

\[ \text{\Comment{mod operations}} \]

\[ x \leftarrow 4^n - 1 \]

end if

else if $sp[x][0] = y$ and $sp[x][1] = z$ then

\[ \text{\Comment{(x, y, z) \in S'}} \]

if dir = 0 then

\[ \text{\Comment{we have reached $S'$ from outside of cube}} \]

\[ \text{\Comment{in the next step we exit from $S$ in negative direction}} \]

\[ \text{\Comment{the $i$th edge is in dimension 3}} \]

\[ \text{\Comment{subtracting 1 from $z$ and $c$}} \]

\[ h[i][0] \leftarrow 3 \]

\[ h[i][1] \leftarrow 1 \]

\[ c \leftarrow c - 1 \]

\[ z \leftarrow z - 1 \]

\[ \text{if } z = -1 \text{ then} \]

\[ \text{\Comment{mod operations}} \]

\[ z \leftarrow 4^n - 1 \]

end if

else

\[ \text{\Comment{we have reached $S'$ from $S$}} \]

\[ \text{\Comment{the $i$th edge is in dimension 2}} \]

\[ \text{\Comment{subtracting 1 from $z$ and $c$}} \]

\[ h[i][0] \leftarrow 2 \]

\[ h[i][1] \leftarrow 1 \]

\[ c \leftarrow c - 1 \]

\[ y \leftarrow y - 1 \]

\[ \text{if } y = -1 \text{ then} \]

\[ \text{\Comment{mod operations}} \]

\[ y \leftarrow 4^n - 1 \]

end if

end if
else if $d[x][0] = y$ and $d[x][1] = z$ then
  \[ (x, y, z) \in D \]
  if dir = 0 then
    \[ h[i][0] \leftarrow 1 \]
    \[ h[i][1] \leftarrow 0 \]
    \[ c \leftarrow c + 1 \]
    \[ x \leftarrow x + 1 \]
  \[ (x, y, z) \in D' \]
  if $x = 4^n$ then
    \[ x \leftarrow 0 \]
  end if
else
  \[ h[i][0] \leftarrow 3 \]
  \[ h[i][1] \leftarrow 0 \]
  \[ c \leftarrow c + 1 \]
  \[ z \leftarrow z + 1 \]
  if $z = 4^n$ then
    \[ z \leftarrow 0 \]
  end if
else if $dp[x][0] = y$ and $dp[x][1] = z$ then
  \[ (x, y, z) \in D' \]
  if dir = 0 then
    \[ h[i][0] \leftarrow 3 \]
    \[ h[i][1] \leftarrow 1 \]
    \[ c \leftarrow c - 1 \]
    \[ z \leftarrow z - 1 \]
  \[ (x, y, z) \in D' \]
  if $z = -1$ then
    \[ z \leftarrow 4^n - 1 \]
  end if
else
  \[ h[i][0] \leftarrow 1 \]
  \[ h[i][1] \leftarrow 1 \]
  \[ c \leftarrow c - 1 \]
  \[ x \leftarrow x - 1 \]
  if $x = -1$ then
    \[ x \leftarrow 4^n - 1 \]
  end if
end if
else if $m[x][0] = y$ and $m[x][1] = z$ then
   \begin{align*}
   \text{\triangleright} \quad & (x, y, z) \in M \\
   \text{\triangleright} \quad & \text{we have reached } M \text{ from } M' \\
   \text{\triangleright} \quad & \text{the } i \text{th edge is in dimension } 1 \\
   \text{\triangleright} \quad & \text{the } i \text{th edge is in positive direction} \\
   \text{\triangleright} \quad & \text{adding } 1 \text{ to } x \text{ and } c \\
   \text{end if}
   \end{align*}

if $\text{dir} = 0$ then
   \begin{align*}
   \text{\triangleright} \quad & \text{we have reached } M \text{ from } M' \\
   \text{\triangleright} \quad & \text{the } i \text{th edge is in dimension } 1 \\
   \text{\triangleright} \quad & \text{the } i \text{th edge is in positive direction} \\
   \text{\triangleright} \quad & \text{adding } 1 \text{ to } x \text{ and } c \\
   \text{\triangleright} \quad & \text{mod operations}
   \end{align*}

\begin{align*}
   h[i][0] & \leftarrow 1 \\
   h[i][1] & \leftarrow 0 \\
   c & \leftarrow c + 1 \\
   x & \leftarrow x + 1 \\
   \text{if } x = 4^n \text{ then} \\
   \quad x & \leftarrow 0
   \end{align*}

else
   \begin{align*}
   \text{\triangleright} \quad & \text{we have reached } M \text{ from outside of cube} \\
   \text{\triangleright} \quad & \text{in the next step we exit from } M' \text{ in positive direction} \\
   \text{\triangleright} \quad & \text{the } i \text{th edge is in dimension } 3 \\
   \text{\triangleright} \quad & \text{the } i \text{th edge is in positive direction} \\
   \text{\triangleright} \quad & \text{adding } 1 \text{ to } y \text{ and } c \\
   \text{\triangleright} \quad & \text{mod operations}
   \end{align*}

\begin{align*}
   y & \leftarrow y + 1 \\
   \text{if } y = 4^n \text{ then} \\
   \quad y & \leftarrow 0
   \end{align*}

end if

else if $mp[x][0] = y$ and $mp[x][1] = z$ then
   \begin{align*}
   \text{\triangleright} \quad & (x, y, z) \in M' \\
   \text{\triangleright} \quad & \text{we have reached } M' \text{ from } M \\
   \text{\triangleright} \quad & \text{the } i \text{th edge is in dimension } 1 \\
   \text{\triangleright} \quad & \text{the } i \text{th edge is in positive direction} \\
   \text{\triangleright} \quad & \text{adding } 1 \text{ to } x \text{ and } c \\
   \text{\triangleright} \quad & \text{mod operations}
   \end{align*}

if $\text{dir} = 0$ then
   \begin{align*}
   \text{\triangleright} \quad & \text{we have reached } M' \text{ from } M \\
   \text{\triangleright} \quad & \text{the } i \text{th edge is in dimension } 1 \\
   \text{\triangleright} \quad & \text{the } i \text{th edge is in positive direction} \\
   \text{\triangleright} \quad & \text{adding } 1 \text{ to } x \text{ and } c \\
   \text{\triangleright} \quad & \text{mod operations}
   \end{align*}

\begin{align*}
   h[i][0] & \leftarrow 1 \\
   h[i][1] & \leftarrow 0 \\
   c & \leftarrow c + 1 \\
   x & \leftarrow x + 1 \\
   \text{if } x = 4^n \text{ then} \\
   \quad x & \leftarrow 0
   \end{align*}

else
   \begin{align*}
   \text{\triangleright} \quad & \text{we have reached } M' \text{ from outside of cube} \\
   \text{\triangleright} \quad & \text{in the next step we exit from } M \text{ in positive direction} \\
   \text{\triangleright} \quad & \text{the } i \text{th edge is in dimension } 2 \\
   \text{\triangleright} \quad & \text{the } i \text{th edge is in negative direction} \\
   \text{\triangleright} \quad & \text{subtracting } 1 \text{ from } y \text{ and } c \\
   \text{\triangleright} \quad & \text{mod operations}
   \end{align*}

\begin{align*}
   h[i][0] & \leftarrow 2 \\
   h[i][1] & \leftarrow 1 \\
   c & \leftarrow c - 1 \\
   y & \leftarrow y - 1 \\
   \text{if } y = -1 \text{ then} \\
   \quad y & \leftarrow 4^n - 1
   \end{align*}

end if

end if
else

if dir = 0 then

if c = −1 (mod 4^n) then

h[i][0] ← 2
h[i][1] ← 0

c ← c + 1

y ← y + 1

if y = 4^n then

y ← 0

end if

else

h[i][0] ← 1
h[i][1] ← 0

c ← c + 1

x ← x + 1

if x = 4^n then

x ← 0

end if

end if

else

if c = 0 (mod 4^n) then

h[i][0] ← 2
h[i][1] ← 1

c ← c − 1

y ← y − 1

if y = −1 then

y ← 4^n − 1

end if

else

h[i][0] ← 1
h[i][1] ← 1

c ← c − 1

x ← x − 1

if x = −1 then

x ← 4^n − 1

end if

end if

end if

end for
B.3. An H.D. for $Q_{6n}$.

Input:

- An $n \times 4^n$ array $e$ with its $i$th row showing the $i$th Hamilton cycle for $Q_{2n}$.
- A $3 \times 4^{3n}$ array $h$ with its $i$th row showing the $i$th Hamilton cycle for $G_{n,3}$.

Output:

- A $3n \times 4^{3n}$ array $g$ with its $i$th row showing the $i$th Hamilton cycle for $Q_{6n}$.

Algorithm 3 An H.D. for $Q_{6n}$ from an H.D. for $Q_{2n}$ and an H.D. for $G_{n,3}$

1: for $i \leftarrow 0$ to 2 do
2:   for $j \leftarrow 0$ to 2 do
3:     $c[i][j] \leftarrow 0$ \hspace{1cm} \triangleright \text{initializing the } x-, y-, \text{ and } z\text{-coordinates of the three pointers}
4:   end for
5: end for
6: for $j \leftarrow 0$ to $n - 1$ do \hspace{1cm} \triangleright \text{cycling through } E_1 \text{ to } E_n
7:   for $i \leftarrow 0$ to $4^{3n} - 1$ do \hspace{1cm} \triangleright \text{cycling through edges of } H_1, H_2, \text{ and } H_3
8:     for $k \leftarrow 0$ to 2 do \hspace{1cm} \triangleright \text{cycling through } H_1, H_2, \text{ and } H_3
9:       dir $\leftarrow h[k][i][1]$ \hspace{1cm} \triangleright \text{direction of the current edge in } H_{k+1}
10:      dim $\leftarrow h[k][i][0] - 1$ \hspace{1cm} \triangleright \text{dimension of the current edge in } H_{k+1}
11:     if dir $= 0$ then \hspace{1cm} \triangleright \text{if the current edge in } H_{k+1} \text{ is forward}
12:         $f[j + kn][i][0] \leftarrow e[j][c[k][\text{dim}]][0] + n(\text{dim})$ \hspace{1cm} \triangleright \text{dimension of the current edge in } F_{j+1+kn}
13:         $f[j + kn][i][1] \leftarrow e[j][c[k][\text{dim}]][1]$ \hspace{1cm} \triangleright \text{direction of the current edge in } F_{j+1+kn}
14:     if $c[k][\text{dim}] = 4^n$ then \hspace{1cm} \triangleright \text{mod operations}
15:       $c[k][\text{dim}] \leftarrow c[k][\text{dim}] + 1$ \hspace{1cm} \triangleright \text{moving forward in the current copy of } E_{j+1}
16:   else \hspace{1cm} \triangleright \text{if the current edge in } H_{k+1} \text{ is backward}
17:     if $c[k][\text{dim}] = 4^n - 1$ then \hspace{1cm} \triangleright \text{mod operations}
18:       $c[k][\text{dim}] \leftarrow 4^n - 1$ \hspace{1cm} \triangleright \text{moving backward in the current copy of } E_{j+1}
19:   end if
20: end if
21: end for
22: end for

B.4. A source cycle for $Q_{4n}$.

**Input:**
- A $4^n \times 2$ array $e$ having the source cycle for $Q_{2n}$.
- An $n \times n$ source matrix $A$ for the cycle $E$.
- A $2^{2n} \times 2$ array $h$ having the source cycle for $G_{n,2}$.

**Output:**
- A $2^n \times 2$ matrix $P$ as the accompanying source matrix.
- A $2^n \times 2$ array $f$ having the source cycle for $Q_{4n}$.

**Algorithm 4** A Source Cycle for $Q_{4n}$ From Source Cycles for $Q_{2n}$ and $G_{n,2}$

1: for $i \leftarrow 0$ to $n - 1$ do \Comment{building $P$}
2:     for $j \leftarrow 0$ to $n - 1$ do
3:         for $k \leftarrow 0$ to 1 do
4:             for $t \leftarrow 0$ to 1 do
5:                 $z \leftarrow (k + t) \mod 2$
6:                 $p[i + k][j + tn] \leftarrow a[i][j] + zn$
7:             end for
8:         end for
9:     end for
10: end for
11: for $i \leftarrow 0$ to 1 do \Comment{initializing the $x$- and $y$-coordinates of the pointer}
12:     $c[i] \leftarrow 0$
13: end for
14: for $i \leftarrow 0$ to $2^{2n} - 1$ do \Comment{cycling through edges of $H$}
15:     dir $\leftarrow h[i][1]$ \Comment{dimension of the current edge in $H$}
16:     dim $\leftarrow h[i][0] - 1$ \Comment{direction of the current edge in $H$}
17:     if dir = 0 then \Comment{if the current edge in $H$ is forward}
18:         $f[i][0] \leftarrow e[c[dim]][0] + n(dim)$ \Comment{dimension of the current edge in $F$}
19:         $f[i][1] \leftarrow e[c[dim]][1]$ \Comment{direction of the current edge in $F$}
20:         $c[dim] \leftarrow c[dim] + 1$ \Comment{mod operations}
21:     end if
22:     if $c[dim] = 4^n$ then \Comment{if the current edge in $H$ is backward}
23:         $c[dim] \leftarrow 0$ \Comment{mod operations}
24:     end if
25:     if $c[dim] = 4^n - 1$ then \Comment{moving backward in the current copy of $E$}
26:         $c[dim] \leftarrow 4^n - 1$ \Comment{mod operations}
27:     end if
28:     if $c[dim] = -1$ then \Comment{mod operations}
29:         $f[i][0] \leftarrow e[c[dim]][0] + n(dim)$ \Comment{dimension of the current edge in $F$}
30:         $f[i][1] \leftarrow 1 - e[c[dim]][1]$ \Comment{direction of the current edge in $F$}
31:     end if
32: end for
B.5. A source cycle for $Q_{6n}$.

Input:

- A $4^n \times 2$ array $e$ having the source cycle for $Q_{2n}$.
- An $n \times n$ source matrix $A$ for the cycle $E$.
- A $4^{2n} \times 2$ array $h$ having the source cycle for $G_{n,2}$.

Output:

- A $4^{3n} \times 2$ array $f$ having the source cycle for $Q_{6n}$.
- A $3n \times 3n$ matrix $P$ as the accompanying source matrix.

\begin{algorithm}
\caption{A Source Cycle for $Q_{6n}$ From Source Cycles for $Q_{2n}$ and $G_{n,3}$}
\begin{algorithmic}[1]
\For {$i \leftarrow 0$ to $n - 1$} \Comment building $P$
  \For {$j \leftarrow 0$ to $n - 1$}
    \For {$k \leftarrow 0$ to $2$}
      \For {$t \leftarrow 0$ to $2$}
        \State $z \leftarrow (k + t) \pmod{3}$
        \State $p[i + k][j + tn] \leftarrow a[i][j] + zn$
      \EndFor
    \EndFor
  \EndFor
\EndFor
\For {$i \leftarrow 0$ to $2$} \Comment initializing the $x$-, $y$-, and $z$-coordinates of the pointer
  \State $c[i] \leftarrow 0$
\EndFor
\For {$i \leftarrow 0$ to $4^{3n} - 1$} \Comment cycling through edges of $H$
  \State $dir \leftarrow h[i][1]$ \Comment direction of the current edge in $H$
  \State $dim \leftarrow h[i][0] - 1$ \Comment dimension of the current edge in $H$
  \If {$dir = 0$} \Comment if the current edge in $H$ is forward
    \State $f[i][0] \leftarrow e[c[dim]][0] + n(\text{dim})$ \Comment dimension of the current edge in $F$
    \State $f[i][1] \leftarrow e[c[dim]][1]$ \Comment direction of the current edge in $F$
    \State $c[dim] \leftarrow c[dim] + 1$ \Comment moving forward in the current copy of $E$
  \Else \Comment if the current edge in $H$ is backward
    \If {$c[dim] = 4^n$} \Comment mod operations
      \State $c[dim] \leftarrow 0$
    \Else
      \State $c[dim] \leftarrow c[dim] - 1$ \Comment moving backward in the current copy of $E$
      \State $c[dim] \leftarrow 4^n - 1$ \Comment mod operations
    \EndIf
  \EndIf
  \State $f[i][0] \leftarrow e[c[dim]][0] + n(\text{dim})$ \Comment dimension of the current edge in $F$
  \State $f[i][1] \leftarrow 1 - e[c[dim]][1]$ \Comment direction of the current edge in $F$
\EndFor
\end{algorithmic}
\end{algorithm}
Appendix C. Correctness of Algorithm 1

For a fixed \( j \), in the \( i \)-loop, Algorithm 1 outputs two cycles \( F_{j+1} = f(E_{j+1}, H_1) \) and \( F_{j+1+n} = f(E_{j+1}, H_2) \). It does so by traversing the edges of \( H_1 (k = 0) \) and \( H_2 (k = 1) \) and mimicking them:

- If the \( i \)th edge of \( H_1 \) is 1 or 2, then the \( i \)th edge of \( F_{j+1} \) is one of \( \{1, 2, \ldots, n, \overline{1}, \overline{2}, \ldots, n\} \), and if the \( i \)th edge of \( H_1 \) is 2 or \( \overline{2} \), then the \( i \)th edge of \( F_{j+1} \) is one of \( \{n + 1, n + 1, \ldots, 2n, 2n\} \). Same thing is true for \( H_2 \) and \( F_{j+1+n} \).

- A pointer, with its \( x \)- and \( y \)-coordinates being \( c[0][0] \) and \( c[0][1] \), tracks movement through \( H_1 \) on the 2-dimensional grid. Another pointer, with its \( x \)- and \( y \)-coordinates being \( c[1][0] \) and \( c[1][1] \), tracks movement through \( H_2 \) on the 2-dimensional grid. These pointers together with \( E_{j+1} \) determine in what dimension and direction the \( i \)th edges of \( F_{j+1} \) and \( F_{j+1+n} \) are:
  - If the \( i \)th edge of \( H_1 \) is from \((a, b)\) to \((a + 1, b)\), then the \( i \)th edge of \( F_{j+1} \) has the same direction and dimension as the \((a + 1)\)st edge of \( E_{j+1} \).
  - If the \( i \)th edge of \( H_1 \) is from \((a, b)\) to \((a, b + 1)\), then the \( i \)th edge of \( F_{j+1} \) has direction equal to that of the \((b + 1)\)st edge of \( E_{j+1} \) and dimension equal to \( n \) plus the dimension of the \((b + 1)\)st edge of \( E_{j+1} \).
  - If the \( i \)th edge of \( H_1 \) is from \((a, b)\) to \((a - 1, b)\), then the \( i \)th edge of \( F_{j+1} \) has direction opposite to that of the \( a \)th edge of \( E_{j+1} \) and dimension equal to that of the \( a \)th edge of \( E_{j+1} \).
  - If the \( i \)th edge of \( H_1 \) is from \((a, b)\) to \((a, b - 1)\), then the \( i \)th edge of \( F_{j+1} \) has direction opposite to that of the \( b \)th edge of \( E_{j+1} \) and dimension equal to \( n \) plus the dimension of the \( b \)th edge of \( E_{j+1} \).
  - Same things can be said about \( H_2 \) and \( E_{j+1} \).

- The pointers are initially set to \((0, 0)\). After each iteration of \( j \), the pointers become \((0, 0)\) because \( H_1 \) and \( H_2 \) start and end at \((0, 0)\).