July 2020

Some Results on Seymour’s Second-Neighborhood Conjecture and on Decompositions of Graphs

Farid Bouya

Follow this and additional works at: https://repository.lsu.edu/gradschool_dissertations

Part of the Discrete Mathematics and Combinatorics Commons

Recommended Citation
https://repository.lsu.edu/gradschool_dissertations/5327

This Dissertation is brought to you for free and open access by the Graduate School at LSU Scholarly Repository. It has been accepted for inclusion in LSU Doctoral Dissertations by an authorized graduate school editor of LSU Scholarly Repository. For more information, please contact gradetd@lsu.edu.
SOME RESULTS ON SEYMOUR’S SECOND-NEIGHBORHOOD CONJECTURE AND ON DECOMPOSITIONS OF GRAPHS

A Dissertation

Submitted to the Graduate Faculty of the Louisiana State University and Agricultural and Mechanical College in partial fulfillment of the requirements for the degree of Doctor of Philosophy

in

The Department of Mathematics

by

Farid Bouya
B.S. in Computer Science, Sharif University of Technology–Tehran, Iran, 2012
M.S. in Mathematics, Louisiana State University, 2014
August 2020
For Mom and Dad, who always believed in me.
Always remember that it is impossible to speak in such a way that you cannot be misunderstood: there will always be some who misunderstand you.

—Karl Popper

Unended Quest: An Intellectual Autobiography
Acknowledgments

First and foremost, praises to God, the Almighty, without Whose blessings I would have been nothing.

I would like to express my deep and sincere gratitude to my advisor Professor Bogdan Oporowski for supporting me through thick and thin. His vision, patience, sincerity, and humor kept me motivated and moving throughout my PhD years. After Bogdan, Professor James Oxley helped me the most as I tried to settle into the LSU environment. Thank you James, for the Communicating Math classes, for the detailed feedbacks on my lectures, for the occasional funny moments, and much more. I also thank Professors Guoli Ding, William Adkins, Karl Mahlburg, Mark Davidson, and Alison Grifa for helping me throughout my LSU years.

I would like to send a special thanks to Dr. Pattie Doody for helping me selflessly with my communication skills. Without her help, I would not have been successful in my talks, presentations, or my defense.

I thank all my K-12 math teachers, specially Hosein Kord, who with his passionate Discrete Math teaching made me very serious about Graph Theory. I also thank my undergraduate advisor Professor Ebadollah Mahmoodian for always being available to help, even after retirement.

I am very much thankful to my wife Saeideh for her patience, understanding, and unconditional support. I also am extremely grateful to my parents and my brother and sister for their financial and emotional support since day one.

Last but not least, I would like to thank all my friends, specially Arash “Doctor”
Ardehali, Amir “Hazrat” Vaziri, Soroush Mojtahedi, Joshua Fallon, and Tara Fife, for their nonstop motivational and spiritual support.
## Contents

Acknowledgments ......................................................... iv

Abstract ................................................................. vii

Chapter 1. Second-Neighborhood Conjecture .............................. 1
  1.1 Introduction and Basic Definitions ............................... 1
  1.2 Conjectures ..................................................... 3
  1.3 Equivalences .................................................... 5
  1.4 Counterexamples ................................................ 10

Chapter 2. Hamilton Decompositions of Hypercubes ...................... 16
  2.1 Introduction ..................................................... 16
  2.2 Notation ........................................................ 17
  2.3 The 2-Dimensional Case ....................................... 19
  2.4 The 3-Dimensional Case ....................................... 22
  2.5 Highly Symmetric Hamilton Decompositions ..................... 33
  2.6 Lemma 2.4.1 for \( n = 1 \) ..................................... 39
  2.7 Algorithms ..................................................... 40
  2.8 Correctness of Algorithm 1 .................................... 49

Chapter 3. Contraction Decomposition .................................. 51
  3.1 Introduction ..................................................... 51
  3.2 History ........................................................ 52
  3.3 First Question ................................................ 53
  3.4 Second Question ............................................... 59

Appendix. Copyright Information ..................................... 63

Bibliography ............................................................. 64

Vita .............................................................. 66
Abstract

This dissertation consists of two parts. In the first part, I examine Seymour’s Second-Neighborhood Conjecture, which states that every orientation of every simple graph has at least one vertex \( v \) such that the number of vertices of out-distance 2 from \( v \) is at least as large as the number of vertices of out-distance 1 from it. I present alternative statements of this conjecture using the language of linear algebra, the last one being completely in terms of the inverse of some matrix. In the second part of this dissertation, comprising of Chapters 2 and 3, I examine two conjectures on graph decompositions. The first one proposes that every even order hypercube \( Q_{2n} \) has a symmetric Hamilton decomposition, meaning that every cycle can be derived from every other cycle just by permuting the axes. I show that this conjecture holds when \( n \) is of the form \( 2^a3^b \). The second conjecture states that for every graph \( G \) its edge set can be partitioned into two sets \( E_1 \) and \( E_2 \) such that the contractions \( G/E_1 \) and \( G/E_2 \) are \( K_4 \)-minor free. This conjecture is currently open, but I ask and answer two slightly different questions: If I use three sets in the partition, contracting two sets at a time, I can avoid \( K_4 \) as a minor, but if I use two sets in the partition, contracting one set at a time, there are some graphs that force a \( K_{2,3} \) minor.
Chapter 1. Second-Neighborhood Conjecture

1.1. Introduction and Basic Definitions

In this chapter, all directed graphs, or digraphs for short, have underlying graphs that are simple, that is, with no loops and no multiple edges. Let $D$ be a digraph and let $u$ and $v$ be vertices of $D$. We write $d(u, v)$ to denote the length of the shortest directed path from $u$ to $v$; if no such path exists, then we put $d(u, v) = \infty$. Since we focus on vertices of out-distance one or two from a particular vertex $v$ of $D$, we set up the following notation.

$$N^+(v) = \{ u \in V(D) \mid d(v, u) = 1 \}, \quad d^+(v) = |N^+(v)|,$$

$$N^{++}(v) = \{ u \in V(D) \mid d(v, u) = 2 \}, \quad d^{++}(v) = |N^{++}(v)|,$$

$$N^-(v) = \{ u \in V(D) \mid d(u, v) = 1 \}, \quad d^-(v) = |N^-(v)|,$$

$$N^{--}(v) = \{ u \in V(D) \mid d(u, v) = 2 \}, \quad d^{--}(v) = |N^{--}(v)|.$$

Each of the symbols defined above may also have a subscript indicating to which digraph it refers. Let $\overrightarrow{D}$ be the digraph obtained from $D$ by reversing the direction on all its edges, so that $d^+_{\overrightarrow{D}}(v) = d^-_{\overrightarrow{D}}(v)$. The original form of Seymour’s Second-Neighborhood Conjecture (SNC) is therefore stated as:

**Conjecture 1.1.1** (Seymour; 1990). Every digraph has a vertex $v$ for which $d^+(v) \leq d^{++}(v)$.

One of the most important conjectures regarding digraphs is the famous Caccetta-Häggkvist Conjecture [6]:

[This chapter is adapted from: Bouya, F. and Oporowski, B., Seymour’s second-neighborhood conjecture from a different perspective, *arXiv pre-print* (2019). It is reprinted with permission from arXiv.]
Conjecture 1.1.2 (Caccetta, Häggkvist; 1978). Every digraph on $n$ vertices that satisfies $d^+(v) \geq r$ for all its vertices has a directed cycle with length at most $\left\lceil \frac{n}{r} \right\rceil$.

Conjecture 1.1.1, if true, settles a special case of Conjecture 1.1.2.

We will adopt some of the notation common in linear algebra. In particular, $\mathbf{0}$ will denote a vector or a matrix consisting of all zeros, and similarly, $\mathbf{1}$ will denote a vector or a matrix consisting of all ones. The identity matrix will be denoted by $I$. Even though the dimensions of these matrices or vectors will not be stated explicitly, they may be easily inferred from the context.

When vectors are represented in the matrix form, they will be understood as column vectors, but to save space, they will be written as transpositions of row vectors. Let $\mathbf{u} = (u_1, u_2, \ldots, u_n)^\top$ and let $\mathbf{v} = (v_1, v_2, \ldots, v_n)^\top$. When we express a numerical relation between vectors, such as $\mathbf{u} \leq \mathbf{v}$, we mean that $u_i \leq v_i$ for all $i$ in $\{1, 2, \ldots, n\}$. The relations $<, \geq, >$, and $=$ are understood in a similar way. However, the negated relations, such as $\not\leq$, $\not<$, $\not\geq$, and $\not=$ are understood in a different way. When we write, for example, $\mathbf{u} \not\leq \mathbf{v}$ we mean that $u_i > v_i$ for at least one $i$ in $\{1, 2, \ldots, n\}$, and so for vectors with more than one component, the inequality $\mathbf{u} \leq \mathbf{v}$ is not equivalent to $\mathbf{u} \not\leq \mathbf{v}$. The same idea applies to all other negated relations.

A weight function on a digraph $D$ is a function $w : V(D) \to [0, \infty)$. If the vertices of $D$ are enumerated as $v_1, v_2, \ldots, v_n$, then we can treat $w$ as a vector:

$\mathbf{w} = [w(v_1), w(v_2), \ldots, w(v_n)]^\top$. In fact, we will often blur the distinction between the values of a weight function and the components of the vector it determines, and write $w(v)$ instead of $w(v)$. We will extend this notation to sets of vertices and write $\mathbf{w}(S)$ to mean $\sum_{v \in S} w(v)$ for a subset $S$ of $V(D)$. 

2
In order to write SNC in terms of matrices, we define the *second-neighborhood matrix* of $D$ as an $n \times n$ matrix $S_D$ whose entries are denoted by $s_{ij}$ and defined as follows:

$$s_{ij} = \begin{cases} 
1 & d(v_i, v_j) = 1, \\
-1 & d(v_i, v_j) = 2, \\
0 & \text{otherwise}.
\end{cases}$$

Note that $S_D^T$ is the second-neighborhood matrix of $\overrightarrow{D}$.

**Example 1.1.3.** Below we have given a digraph $D$ together with its second-neighborhood matrix $S_D$.

![Figure 1.1](image)

Let $S_D = \begin{bmatrix} 0 & 1 & 1 & -1 \\
0 & 0 & 1 & -1 \\
-1 & 0 & 0 & 1 \\
1 & -1 & -1 & 0 \end{bmatrix}$

In this chapter, we have adopted main proof techniques from a paper of Fisher [11].

1.2. Conjectures

The main purpose of this chapter is to present several statements in the language of linear algebra, each of which is equivalent to SNC, in the hope that the tools of linear algebra may yield themselves to attacking the conjecture. These statements are the following:

**Conjecture 1.2.1.** Every digraph $D$ satisfies $S_D1 \neq 0$.

**Conjecture 1.2.2.** Every digraph $D$ and every weight vector $w$ on $D$ satisfy $S_Dw \neq 0$.  

Conjecture 1.2.3. For every digraph $D$ there is a non-zero weight vector $w$ with $S_Dw \leq 0$.

Conjecture 1.2.4. For every digraph $D$, there is a vector $v$ (not necessarily a weight vector) with at least one positive component and such that $S_Dv \leq 0$.

Conjecture 1.2.5. There is no digraph $D$ such that $S_D^{-1} \geq 0$.

We illustrate how these conjectures might be true in the following example.

Example 1.2.6. For the digraph $D$ and its second-neighborhood matrix given in Example 1.1.3 we have

\[
S_D1 = \begin{bmatrix} 0 & 1 & 1 & -1 \\ 0 & 0 & 1 & -1 \\ -1 & 0 & 0 & 1 \\ 1 & -1 & -1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \neq \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}
\]

If we pick some weight vector, say $w = [0, 3, 1, 2.5]^\top$, we get

\[
S_Dw = \begin{bmatrix} 0 & 1 & 1 & -1 \\ 0 & 0 & 1 & -1 \\ -1 & 0 & 0 & 1 \\ 1 & -1 & -1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 3 \\ 1 \\ 2.5 \end{bmatrix} = \begin{bmatrix} 1.5 \\ -1.5 \\ 2.5 \\ -4 \end{bmatrix} \neq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
\]

Putting $w = [1, 0, 1, 1]^\top$ gives

\[
S_Dw = \begin{bmatrix} 0 & 1 & 1 & -1 \\ 0 & 0 & 1 & -1 \\ -1 & 0 & 0 & 1 \\ 1 & -1 & -1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
\]

The same vector also shows that $S_D$ is not invertible, so $S_D^{-1} \neq 0$ is vacuously true.
The first major result of this chapter is the following:

**Theorem 1.2.7.** Conjectures 1.1.1, 1.2.1, 1.2.2, 1.2.3, 1.2.4, and 1.2.5 are equivalent.

Proving some of the equivalences is significantly harder than proving others, and, indeed, some of these statements, such as Conjectures 1.2.1 and 1.2.2 play only auxiliary roles in the arguments. The proof of this theorem will be presented in a series of propositions in future sections.

If one, and thus all, of these conjectures fail, the sets of counterexamples may, and, in fact, do differ between some of them. When we compare potential counterexamples and use words like “minimal” or “smaller”, we understand them in terms of the number of arcs. The fact that the sets of minimal counterexamples to Conjectures 1.2.3, 1.2.4, and 1.2.5 are the same can be easily seen from the proofs of the relevant equivalences. However, we find surprising the following:

**Theorem 1.2.8.** Every minimal counterexample to Conjecture 1.2.3 is smaller than every minimal counterexample to Conjecture 1.2.1.

1.3. Equivalences

We begin by addressing the equivalence of the first pair of the conjectures. We state it without proof, as it is evident.

**Proposition 1.3.1.** Conjectures 1.1.1 and 1.2.1 are equivalent.

We proceed now to the equivalence of the next pair of conjectures.

**Proposition 1.3.2.** Conjectures 1.2.1 and 1.2.2 are equivalent.

**Proof.** It is clear that Conjecture 1.2.2 implies Conjecture 1.2.1.
Suppose now that Conjecture 1.2.2 fails, and so there are a digraph $D$ and a weight vector $w$ on $D$ are such that $S_Dw > 0$. Since the set of positive rational numbers forms a dense subset of $[0, \infty)$, we may take a weight vector $w'$ sufficiently close to $w$ so that the components of $w'$ are rational and positive, and $S_Dw' > 0$. By multiplying $w'$ by a suitable integer, we obtain a weight vector $u$ whose components are positive integers, and such that $S_Du > 0$.

We construct a digraph $D^*$ as follows. Enumerate the vertices of $D$ as $v_1, v_2, \ldots, v_n$, and suppose that $u = [u(v_1), u(v_2), \ldots, u(v_n)]^T$. For each $i$ in $\{1, 2, \ldots, n\}$, let $V_i$ be a set of $u(v_i)$ elements, and let $V(D^*)$ be the disjoint union of all $V_i$’s. For each directed edge $(v_i, v_j)$ of $D$, put into $D^*$ directed edges from each element of $V_i$ to each element of $V_j$. Let $S_{D^*}$ be the second-neighborhood matrix of $D^*$ and note that in the vector $S_{D^*}1$, the component corresponding to a vertex $v$ of $D^*$ that lies in in some $V_i$ is equal to the component of $S_Du$ corresponding to the vertex $v_i$ of $D$. Hence $S_{D^*}1 > 0$, and so $D^*$ is a counterexample to Conjecture 1.2.1. □

Our proof of the next equivalence will make use of a classical result in linear algebra, known as Farkas’ Lemma, which is stated below.

**Theorem 1.3.3** (Farkas’ Lemma). Let $M$ be an $(m \times n)$-matrix and let $b$ be an $m$-dimensional vector. Then exactly one of the following statements holds.

1. There is an $n$-dimensional vector $x$ such that $Mx = b$ and $x \geq 0$.
2. There is an $m$-dimensional vector $y$ such that $M^Ty \geq 0$ and $b^Ty < 0$.

**Proposition 1.3.4.** Conjectures 1.2.3 and 1.2.2 are equivalent. Moreover, a digraph $D$
is a counterexample to Conjecture 1.2.3 if and only if $\overrightarrow{D}$ is a counterexample to Conjecture 1.2.2.

**Proof.** Suppose $D$ is digraph on $n$ vertices. Construct a new matrix $M$ with $n+1$ rows and $2n$ columns by assembling together smaller matrices, as follows:

$$M = \begin{bmatrix} S_D & I \\ 1^\top & 0^\top \end{bmatrix},$$

and let $b$ be the $(n+1)$-dimensional standard basis vector $[0, 0, \ldots, 0, 1]^\top$.

For the remainder of the proof, we present a list of statements (1)–(9) that are equivalent to one another. It is easy to see that consecutive statements are equivalent, and we remark that the equivalence between (5) and (6) follows from Theorem 1.3.3.

1. Digraph $D$ is a counterexample to Conjecture 1.2.3.

2. The following system fails for every $n$-dimensional vector $w$.

$$\begin{cases} S_D w \leq 0 \\
w \geq 0 \\
w \neq 0 \end{cases}$$

3. The following system fails for every $n$-dimensional vector $u$.

$$\begin{cases} S_D u \leq 0 \\
u \geq 0 \\
^\top u = 1 \end{cases}$$
4. The following system fails for every two \( n \)-dimensional vectors \( u \) and \( z \).

\[
\begin{align*}
S_D u + z &= 0 \\
z &\geq 0 \\
u &\geq 0 \\
1^\top u &= 1
\end{align*}
\]

5. The following system fails for every \( 2n \)-dimensional vector \( x \).

\[
\begin{align*}
Mx &= b \\
x &\geq 0
\end{align*}
\]

6. There is an \((n + 1)\)-dimensional vector \( y \) that satisfies the following system.

\[
\begin{align*}
M^\top y &\geq 0 \\
b^\top y &< 0
\end{align*}
\]

7. There are an \( n \)-dimensional vector \( p \) and a scalar \( r \) that satisfy the following system.

\[
\begin{align*}
S_D^\top p + r 1 &\geq 0 \\
p &\geq 0 \\
r &< 0
\end{align*}
\]

8. There is an \( n \)-dimensional vector \( p \) that satisfies the following system.

\[
\begin{align*}
S_D^\top p &> 0 \\
p &\geq 0
\end{align*}
\]

9. Digraph \( \overrightarrow{D} \) is a counterexample to Conjecture 1.2.2.
We established the equivalence of statements (1) and (9), which concludes the proof. □

Next we show that Conjectures 1.2.4 and 1.2.5 are equivalent.

**Proposition 1.3.5.** Conjectures 1.2.4 and 1.2.5 are equivalent, with the same set of counterexamples.

**Proof.** Let $D$ be a digraph, and suppose first that the matrix $S_D$ is not invertible. Then there is a non-zero vector $u$ such that $S_Du = 0$. If $u$ has a positive component, then let $v = u$; otherwise let $v = -u$. Then $v$ testifies to the fact that $D$ satisfies Conjecture 1.2.4. Also, $D$ vacuously satisfies Conjecture 1.2.5, and so both conjectures hold for digraphs with non-invertible second-neighborhood matrices.

Suppose now that $S_D$ is invertible, and let $\sigma_D$ be the map defined by $\sigma_D: w \mapsto S_Dw$.

Consider the statement:

(1) Digraph $D$ is a counterexample to Conjecture (1.2.4).

It is equivalent to the statement that no vector $w$ satisfies both $S_Dw \leq 0$ and $w \neq 0$, which, in turn, is equivalent to the statement:

(2) If $\sigma_D(w) \leq 0$, then $w \leq 0$.

Since $S_D$ is invertible, $\sigma_D$ is bijective and thus has an inverse, and so statement (2) is equivalent to the following:

(3) If $w \leq 0$, then $\sigma_D^{-1}(w) \leq 0$.

Note that $\sigma_D$ is also linear, and so (3) is equivalent to the statement:

(4) If $w \geq 0$, then $\sigma_D^{-1}(w) \geq 0$. 

Now, we observe that \( \sigma_D^{-1}(w) = S_D^{-1}w \), and so (4) may be restated as:

\[
(5) \quad S_D^{-1}w \geq 0 \quad \text{for every vector } w \geq 0.
\]

The last statement holds if and only if every entry of \( S_D^{-1} \) is non-negative. \( \square \)

The last equivalence is established in the next section.

1.4. Counterexamples

In this section, we will compare the various sets of potential counterexamples to the conjectures discussed in this chapter.

For each \( N \) in \( \{1.2.1, 1.2.3, 1.2.4\} \), let \( X_N \) denote the set of counterexamples to Conjecture \( N \), and let \( \vec{X}_N = \{ \vec{D} \mid D \in X_N \} \). Intuitively, we may think of each \( \vec{X}_N \) as the set of counterexamples to “Conjecture \( N \) stated for in-neighbors”.

The first proposition comparing the above sets of counterexamples is an immediate consequence of the statements of the conjectures, so it is stated without proof.

**Proposition 1.4.1.** \( X_{1.2.4} \subseteq X_{1.2.3} \) and \( \vec{X}_{1.2.4} \subseteq \vec{X}_{1.2.3} \).

The next proposition is almost as obvious.

**Proposition 1.4.2.** \( X_{1.2.1} \subseteq \vec{X}_{1.2.3} \).

**Proof.** Suppose \( D \in X_{1.2.1} \). It is obvious that \( D \) is also a counterexample to Conjecture 1.2.2, and Proposition 1.3.4 asserts that \( \vec{D} \) is a counterexample to Conjecture 1.2.3, as well; the conclusion follows. \( \square \)

**Lemma 1.4.3.** *Every minimal element of \( X_{1.2.3} \) is a member of \( X_{1.2.5} \).*
Proof. Let $D$ be a minimal element of $X_{1,2,3}$, and let $V(D) = \{v_1, v_2, \ldots, v_n\}$. By the minimality of $D$, for each $D - v_i$ there is a non-zero non-negative weight vector $w_i$ satisfying $S_{D-v_i}w_i \leq 0$. We can extend $w_i$ to a weight vector $w_i$ on $D$ by putting $w_i(v_i) = 0$. Note that $w_i(N^+(v_j)) - w_i(N^{++}(v_j)) \leq 0$ for $j \neq i$. If for some $i$, the weight vector $w_i$ satisfies $S_{D}w_i \leq 0$, then we reach a contradiction. Therefore, we may assume that $w_i(N^+(v_i)) - w_i(N^{++}(v_i)) > 0$ for all $i$. Let $W$ be the $(n \times n)$-matrix whose $i$th column is $w_i$, and let $C = S_{D}W$. Then the entries of $C$ may be expressed as $c_{ij} = w_j(N^+(v_i)) - w_j(N^{++}(v_i))$, which implies that $c_{ij}$ is positive if and only if $i = j$.

We use a process similar to the Gauss-Jordan elimination to turn $C$ into the identity matrix $I_n$. The only difference is that we work with columns instead of rows, so we do elementary column operations. If we are successful, the identity matrix $I_n$ may be expressed as $C$ multiplied on the right by an appropriate transformation matrix $T$, that is, $I_n = CT$. To be more precise, we do the following:

1. Start by putting $i = 1$ and $X = (x_{ij}) = C$.
2. If $i > n$, then $X$ is equal to $I_n$. Exit.
3. If $x_{ii} \leq 0$, exit. Otherwise, add suitable multiples of the $i$th column of $X$ to other columns of $X$ to make the $i$th row of $X$ zero (except for $x_{ii}$).
4. Divide the $i$th column by $x_{ii}$.
5. Add 1 to $i$. Go to (2).

If during this process we get non-positive $i$th diagonal (that is, the algorithm exits through step (3) because $x_{ii} \leq 0$), then a non-negative, non-zero linear combination of
$S_D\bar{w}_1, S_D\bar{w}_2, \ldots, S_D\bar{w}_i$ is non-positive, say,

$$a_1S_D\bar{w}_1 + a_2S_D\bar{w}_2 + \cdots + a_iS_D\bar{w}_i \leq 0.$$ 

This is equivalent to $S_D(a_1\bar{w}_1 + a_2\bar{w}_2 + \cdots + a_i\bar{w}_i) \leq 0$, which contradicts the fact that $D \in \mathcal{X}_{1,2,3}$. Therefore the procedure described above never results in the matrix $X$ having a non-positive entry on the main diagonal, so the algorithm never exits through step (3), and always exits through step (2) instead, giving us the identity matrix $I_n$. Note that in this process, we only add non-negative multiples of a column to other columns. This means that the elementary matrices associated with the matrix operations are all non-negative, therefore their product $T$ is also non-negative. Let $W' = \bar{W}T$, let $w'_i$ be the $i$th column of $W'$, and let $e_i$ be the $i$th column of $I_n$, that is, the $i$th $n$-dimensional standard basis vector. Then $W'$ is non-negative. We have

$$I_n = CT = S_D\bar{W}T = S_DW'.$$

This means that $S_D$ has non-negative inverse, so $D \in \mathcal{X}_{1,2,5}$, as required. □

Now we are ready to provide the last part of the proof of Theorem 1.2.7.

**Proposition 1.4.4.** Conjectures 1.2.3 and 1.2.4 are equivalent.

**Proof.** Clearly, Conjecture 1.2.3 implies Conjecture 1.2.4.

Suppose now that Conjecture 1.2.3 fails, and so some digraph $D$ is a minimal element of $\mathcal{X}_{1,2,3}$. Lemma 1.4.3 implies that $D \in \mathcal{X}_{1,2,5}$, so Conjecture 1.2.5 fails. Proposition 1.3.5 now implies that Conjecture 1.2.4 fails as well. □

The remainder of the chapter is devoted to proving Theorem 1.2.8. Most of the work will be contained in the following:
Lemma 1.4.5. If a digraph $D$ is a minimal member of $X_{1.2.3}$, then $D \notin X_{1.2.1}$.

Proof. Suppose, for a contradiction, that $D$ is a minimal member of $X_{1.2.3}$ that also belongs to $X_{1.2.1}$. Since Proposition 1.4.2 asserts that $X_{1.2.1} \subseteq X_{1.2.3}$, we also have

(1) $D$ is a minimal element of $X_{1.2.1}$.

The minimality of $D$ in $X_{1.2.1}$ implies that it is strongly connected, and the fact that $D$ is a counterexample to SNC implies that the minimum in-degree of $D$ is at least two; in fact it is at least seven (see [13]).

Let $y$ be an arbitrary vertex of $D$, let $xy$ be an arc of $D$, and let $D' = D \setminus xy$. For a vertex $v$ of $D$, let $a(v) = d_D(v) - d_D^{-}(v)$ and let $a'(v) = d_{D'}(v) - d_{D'}^{-}(v)$. Note that $a(v) \leq a'(v)$ whenever $v \neq y$. If $D$ has a directed path of length two from $x$ to $y$, then $a'(y) = a(y) - 2$; otherwise $a'(y) = a(y) - 1$. We show that

(2) $a'(y) = -1$ and $a'(v) \geq 1$ for $v \neq y$.

It is not hard to see that $a(v) \in \{1, 2\}$; see [5] for a justification. This means that $a'(y) \in \{-1, 0, 1\}$ and $a'(v) \geq 1$ for $v \neq y$. In the case $a'(y) = 1$, we reach a contradiction with the minimality of $D$ in $X_{1.2.1}$. We will show that $a'(y) = 0$ cannot occur either.

Suppose, for a contradiction, that $a'(y) = 0$, and let $z$ be a vertex in $N_{D'}^{-}(y)$. We define a weight vector $u$ on $D'$ as follows:

$$u(v) = \begin{cases} 1 & \text{if } v \neq z; \\
\frac{3}{2} & \text{if } v = z; 
\end{cases}$$

Now, we have $S_{D'}^{T}u > 0$, and an argument very similar to the proof of Proposition 1.3.4 im-
plies that \( D' \) fails Conjecture 1.2.3, which contradicts the minimality of \( D \) in \( \mathcal{X}_{1.2.3} \). Thus we conclude that \( a'(y) = -1 \).

Since \( D \in \mathcal{X}_{1.2.1} \), it satisfies \( a(y) > 0 \), and thus, it must be that \( a(y) = 1 \). In other words,

\[
(3) \quad d_D^-(y) = d_D^-(y) + 1.
\]

Let \( y \) be a vertex of \( D \) with the largest possible in-degree \( d \). Then (3) implies that \( d^-(y) = d - 1 \). Let \( N^-(y) = X = \{ x_1, x_2, \ldots, x_d \} \) and let \( N^+(y) = Z = \{ z_1, z_2, \ldots, z_{d-1} \} \).

Consider the digraph \( D' = D \setminus x_1y \) and note that the minimality of \( D \) implies that there is a weight vector \( w' \) such that \( S_{D'} w' \leq 0 \). The last inequality is equivalent to stating that \( w'(N^+(u)) \leq w'(N^+(u)) \) for every vertex \( u \) of \( D' \), which, in turn, implies that

\[
\sum_{u \in V(D')} w'(N_{D'}^+(u)) \leq \sum_{u \in V(D')} w'(N_{D'}^{++}(u))
\]

Note that \( w'(u) \) appears \( d_{D'}^+(u) \) times on the left side of the above inequality, while it appears \( d_{D'}^-(u) \) times on the right side. By (3), we have \( d_{D'}^-(u) = d_{D'}^+(u) + 1 \) whenever \( u \neq y \) and \( d_{D'}^-(y) = d_{D'}^+(y) - 1 \), and so \( w'(y) \geq w'(V \setminus \{ y \}) \). If \( w'(y) > w'(V \setminus \{ y \}) \), then \( w'(N_{D'}^+(x_2)) > w'(N_{D'}^{++}(x_2)) \), which is impossible. It follows that \( w'(y) = w'(V \setminus \{ y \}) \), which implies that \( w'(N_{D'}^+(u)) = w'(N_{D'}^{++}(u)) \) for every vertex \( u \) of \( D' \).

Let \( S = \{ u \in V(D') : u \neq y \) and \( w'(u) > 0 \} \), and observe that \( w'(S) = w'(y) \).

Let \( k = w'(y) \), let \( Z' = Z \cup \{ x_1 \} \), and let \( X' = X \setminus \{ x_1 \} \). By construction, \( N_{D'}^-(y) = Z' \), and so \( y \in \mathcal{N}_{D'}^+(z) \) for every \( z \in Z' \). This implies that \( w'(N_{D'}^+(z)) \geq k \), and, further, that \( w'(N_{D'}^{++}(z)) = k = w'(N^+(z)) \). This means that \( D \) has an arc \( zs \) for every \( z \in Z' \) and every \( s \in S \). Since \( y \) has the largest in-degree in \( D \), we have \( Z' = N^-(s) \) for every \( s \in S \).

Note that \( y \notin S \), and if \( X' \cap S \) had an element \( x \), then we would have \( w'(N^+(x)) < \)
\(k \leq w'(N^+(x))\), which is impossible; hence \(X' \cap S = \emptyset\). Similarly, \(Z' \cap S = \emptyset\). Therefore 
\((\{y\} \cup N^-(y) \cup N^{--}(y)) \cap S = \emptyset\). Since \(D\) is strongly connected and \(S\) is non-empty, \(D\) has a vertex \(t\) of in-distance three from \(y\), which, clearly, is in neither \(X\) nor \(Z\). Since all vertices in \(S\) have \(d\) in-neighbors in \(X \cup Z\), there is no arc in \(D\) in the form \(ts\) with \(s \in S\), and so 
\(w'(N^+(t)) = 0\). But every \(z \in N^{--}(y)\) has arcs to all members of \(S\), so \(w'(N^{++}(t)) \geq k\); a contradiction. □

Finally, we are ready to prove Theorem 1.2.8

*Proof of Theorem 1.2.8.* Suppose \(D\) is a minimal counterexample to Conjecture 1.2.1. Then \(\overrightarrow{D} \in \overrightarrow{\mathcal{X}}_{1,2,1}\). Lemma 1.4.2 implies that \(\overrightarrow{\mathcal{X}}_{1,2,1} \subseteq \mathcal{X}_{1,2,3}\), and so \(\overrightarrow{D}\) is also a counterexample to Conjecture 1.2.3. If \(\overrightarrow{D}\) were minimal, then Lemma 1.4.5 would imply that \(D \notin \mathcal{X}_{1,2,1}\), which would be a contradiction. □
Chapter 2. Hamilton Decompositions of Hypercubes

2.1. Introduction

In this section we give a brief and informal history of the problem. Formal definitions are given in the next section.

Hypercubes are widely used in computer architectures in areas like parallel computing [19], multiprocessor systems [8], processor allocation [20], 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 $n$ [21]. In 1982, Aubert and Schneider showed that every $Q_{2n}$ admits a Hamilton decomposition [2]. 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 [18] gave a direct approach for finding Hamilton decompositions for $Q_{2n}$ with $n \leq 4$. Mollard and Ramras [15] 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 chapter, stated formally as Corollary 2.5.7. In Section 2.7, we present Algo-

---

This chapter is adapted from: Bouya, F., Mahmoodian, E. S., Shokrian, M., and Tefagh, M., A Highly Symmetric Hamilton Decomposition for Hypercubes, arXiv pre-print (2020). It is reprinted with permission from arXiv.
rithms 4 and 5 that efficiently construct such decompositions. We conjecture that a similar decomposition exists for every \( n \).

### 2.2. Notation

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 of its vertices \((u,v)\) and \((u',v')\) are adjacent if and only if

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

Using this definition, it is not hard to see that

\[
Q_2 = C_4,
\]

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

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

and

\[
Q_{2n} = C_4 \square C_4 \square \cdots \square C_4.
\] (2.2.1)
As in [22], we use Equation (2.2.1) 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 \textit{dimension-}k \textit{edge} in $Q_{2n}$ is an edge that connects two vertices whose quaternary labels differ in the $k$th digit. If a dimension-\textit{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 denote it by $k$, and if it is directed in the opposite direction, we denote it by $\bar{k}$. A \textit{Hamilton cycle} in a graph is a cycle visiting all the vertices. A \textit{Hamilton decomposition} of $Q_{2n}$ is a partitioning of its edge set into $n$ disjoint Hamilton cycles. We use the notation given in [18] 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 2.1, with the origin (top left vertex) as its initial vertex, is shown by $2\bar{1}22\bar{1}11\bar{2}111$.

![Figure 2.1: The cycle in $Q_4$ matching the code $2\bar{1}22\bar{1}11\bar{2}111$.](image)
In this chapter, 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 2.2.1.** Define \( G_{n,k} \) to be the graph \( C_{4n} \square C_{4n} \square \cdots \square C_{4n} \) \( k \) times.

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 \( C_{4n} \square C_{4n} \) has coordinates \((u, v)\), where \( u \) is in the first copy of \( C_{4n} \) 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.

### 2.3. 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 \( Q_{4n} \).

#### 2.3.1. An H.D. for \( G_{n,2} \)
There is an H.D. for the graph $C_m \square C_m$ in

$$H_1 = \underbrace{11\cdots1}_m \underbrace{211\cdots1}_{m-1 \text{ times}} \underbrace{2\cdots11\cdots1}_{m-1 \text{ times}} \underbrace{2}_{m \text{ times}} \quad , \quad H_2 = \underbrace{22\cdots2}_{m \text{ times}} \underbrace{122\cdots2}_{m-1 \text{ times}} \underbrace{1\cdots22\cdots2}_{m-1 \text{ times}} \underbrace{1}_{m \text{ times}}$$

Since $G_{n,2} = C_{4^n} \square C_{4^n}$, we have the same type of H.D. for $G_{n,2}$:

$$H_1 = \underbrace{11\cdots1}_{4^n-1 \text{ times}} \underbrace{211\cdots1}_{4^n-1 \text{ times}} \underbrace{2\cdots11\cdots1}_{4^n-1 \text{ times}} \underbrace{2}_{4^n \text{ times}} \quad , \quad H_2 = \underbrace{22\cdots2}_{4^n-1 \text{ times}} \underbrace{122\cdots2}_{4^n-1 \text{ times}} \underbrace{1\cdots22\cdots2}_{4^n-1 \text{ times}} \underbrace{1}_{4^n \text{ times}} \quad (2.3.1)$$

### 2.3.2. 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 2.3.1.** 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 Hamil-
ton 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 2.3.2.** 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 2.3.3.** 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 \( 1 \leq j \leq n \).

### 2.3.3. 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 1, given in Section 2.7, takes an H.D. for \( G_{n,2} \) and an H.D. for \( Q_{2n} \) as inputs, and outputs an H.D. for \( Q_{4n} \).

### 2.4. 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 \( Q_{6n} \).

#### 2.4.1. 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 [18] and [22], we decompose the graph into three 2-factors, and then try to merge the components until we have three Hamilton cycles.

**Lemma 2.4.1.** 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 Section 2.6.

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 [22] and later in [18] to build an H.D. for \( Q_6 \). We use them to construct an H.D. for every \( G_{n,3} \).
Definition 2.4.2. 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 2.2. 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 2.2: The special cube type-I (2.2a) and the special cube type-II (2.2b).](image)

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 2.4.3. Consider $G_{n,3}$ and decompose it with the method described in Lemma 2.4.1. Then every vertex $(x, y, z)$ with $x + y + z = -1 \pmod{4^n}$ is the origin of a type-I cube.

Observation 2.4.4. Figure 2.3 shows that, a single merge operation, done on the decomposition obtained from Lemma 2.4.1, indeed merges six cycles, two in each of $X$, $Y$, and $Z$, into three cycles, one in each of $X$, $Y$, and $Z$. 
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 2.4.8 is crucial.
Definition 2.4.5. 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 2.4.6. 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 2.4.7. Let $v = (x, y, z)$ be the origin of a type-I cube $L$. Figure 2.4 shows that the $X \cap Z^n_x$-necklace-order with respect to $Z^n_x$ (before merge) is the same as $X \cap Z^n_{[x,x+1]}$-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 $Z^n_x$.

![Diagram](image)

Figure 2.4: The $X \cap Z^n_x$-necklace-order with respect to $Z^n_x$ (left) is the same as the $X \cap Z^n_{[x,x+1]}$-necklace-order with respect to $Z^n_x$ (right).

Lemma 2.4.8. Suppose that the edge set of $G_{n,3}$ is decomposed with the method given in Lemma 2.4.1, but no switches are performed. Let $v = (x, y, z)$ and $v' = (x', y', z')$ be such that $x + y + z = x' + y' + z' = -1 \pmod{4^n}$ and $z = z' + 1 \pmod{4^n}$, 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 2.4.4 that a single merge operation always succeeds. Suppose that we have merged $L$, so that $Z_{x+1}^n$ and $Z_{x+2}^n$ have merged into $Z_{[x+1,x+2]}^n$, and we are about to merge $L'$. By Observation 2.4.7, the order of vertices in $Z_{x+1}^n$ has not changed, so merging $L'$ will successfully combine $Z_{[x+1,x+2]}^n$ and $Z_x^n$ into a single cycle $Z_{[x,x+2]}^n$. 

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

Definition 2.4.9. 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 $x + y + z = -1 \pmod{4^n}$, 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 2.4.10. Consider the following procedure:

- i. Decompose the edge set of $G_{n,3}$ with the method given in Lemma 2.4.1.
- 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 \( i \), 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 \( ii \) and the type-I cubes in step \( iii \), throughout step \( iv \).
   - For a fixed \( i \), the vertices of \( Z_i^n \) remain in the same component of \( X \), the vertices of \( X_i^n \) remain in the same component of \( Y \), and the vertices of \( Y_i^n \) remain in the same component of \( Z \), and
   - Every merge operation reduces \( c_x \), \( c_y \), and \( c_z \) by 1.

In particular, after finishing step \( iv \), we have an H.D. for \( G_{n,3} \).

Proof.

1. This was shown in Lemma 2.4.1.

2. Suppose there exist \((x_1, x_2, x_3)\) and \((x_1', x_2', x_3')\) in \( S \) such that their corresponding type-I cubes have some vertex in common, so that for some \((i_1, i_2, i_3)\) and \((i_1', i_2', i_3')\) in \( \{0, 1\}^3 \) we have

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

It follows that \( x_r + i_r = x'_r + i'_r \pmod{4^n} \) for each \( 1 \leq r \leq 3 \). Adding these congruences we get \( x_1 + x_2 + x_3 + i_1 + i_2 + i_3 = x'_1 + x'_2 + x'_3 + i'_1 + i'_2 + i'_3 \pmod{4^n} \), but \( x_1 + x_2 + x_3 = x'_1 + x'_2 + x'_3 = -1 \pmod{4^n} \), so we obtain \( i_1 + i_2 + i_3 = i'_1 + i'_2 + i'_3 \pmod{4^n} \), and in particular, \( i_1 + i_2 + i_3 = i'_1 + i'_2 + i'_3 \pmod{2} \). This implies that \( i_r = i'_r \) for
some $r$, meaning that $x_r = x'_r \pmod{4^n}$ 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 $Z^n_i$ 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 2.4.4 and Lemma 2.4.8. Suppose that we have merged cubes $L_{4^n-2}$ to $L_i$. The induction hypothesis states that we have cycles $X \cap Z^n_1, X \cap Z^n_2, \ldots, X \cap Z^n_{i-1}$, and a long cycle $X \cap Z^n_{[i,4^n-1]}$. It also states that the vertices of $Z^n_i$ have been in the same component of $X$ together throughout step iv. By Observation 2.4.7, the order of the vertices in $Z^n_i$ has not changed yet, so merging $L_{i-1}$ will combine $X \cap Z^n_{[i,4^n-1]}$ and $X \cap Z^n_{i-1}$ into a single cycle $X \cap Z^n_{[i-1,4^n-1]}$. It is clear that the vertices of $Z^n_i$ have remained and will remain in the same component of $X$.

This completes the proof of the lemma. □

2.4.2. 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 2.4.11.** 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 $4n - 1$ 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 $Q_{6n} = Q_{2n} \sqcap Q_{2n} \sqcap Q_{2n}$. 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 $Q_{2n} \sqcap Q_{2n} \sqcap Q_{2n}$ as $H$ has in the subgraph $G_{n,3}$.
Lemma 2.4.12. 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 $E_1$. Similarly 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 2.4.13. 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 $1 \leq j \leq n$.

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

In Section 2.4.1 we saw how to derive a Hamilton decomposition $\{X, Y, Z\}$ from the initial partitioning given by Lemma 2.4.1. 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 2.4.1, 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 2.5 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.

![Figure 2.5: 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

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

We choose $S$ like this for two reasons:

- The origin belongs to none of the type-I cubes, so we do not need an initial case check.
- The set $S$ 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 auxiliary sets $S', D, D', M,$ and $M'$ so that we have immediate access to all the special vertices. Algorithm 2 given in Section 2.7 calculates $X$.

2.4.4. 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 Section 2.7.

2.5. Highly Symmetric Hamilton Decompositions

The theory we have developed in the previous chapters can be improved to give us highly symmetric Hamilton decompositions. Let $\sigma : [1, k] \to [1, k]$ 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 $H = e_1e_2 \ldots e_{4nk}$ is a directed Hamilton cycle in $G_{n,k}$, then $\sigma(H)$ is the Hamilton cycle

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

Note that $\sigma$ maps backward edges to backward edges: If $\sigma(i) = j$, then $\sigma(\bar{i}) = \bar{j}$. It is worth remembering that $\bar{i}$ 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 2.5.1. 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

$$H_i = \sigma_i(H) \quad \text{for all } 1 \leq i \leq k.$$

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 2.3.1 is Latin, but the one given for $G_{n,3}$ in 2.4.1 is not necessarily so. If it is not Latin, we can turn it into one with a small adjustment.

**Theorem 2.5.2.** The set $S$ mentioned in Lemma 2.4.10 step ii. can be chosen in such a way that the resulting H.D. is Latin. More specifically, if

$$S^* = \left\{ \left(0, \frac{4^t}{2} - 1, \frac{4^t}{2}\right), \left(1, \frac{4^t}{2} - 3, \frac{4^t}{2} + 1\right), \ldots, \left(\frac{4^t - 4}{6}, \frac{4^t + 2}{6}, \frac{4 \times 4^t - 4}{6}\right), \left(\frac{4 \times 4^t + 2}{6}, \frac{5 \times 4^t + 4}{6}, \frac{4^t}{2} - 2\right), \ldots, \left(\frac{5 \times 4^t - 2}{6}, -1, \frac{4^t + 2}{6}, 1\right) \right\},$$

and

$$S = \left\{(x, y, z) \mid (x, y, z) \in S^*, \text{ or } (y, z, x) \in S^*, \text{ or } (z, x, y) \in S^*\right\}, \quad (2.5.1)$$

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 $(z, x, y) \in S$. 

34
To see this, consider $G_{n,3}$ after completion of Lemma 2.4.10 step i. 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

$$\sigma_2(X) = Y \text{ and } \sigma_3(X) = Z. \quad (2.5.2)$$

We wish to show that the relations given in 2.5.2 remain valid after completion of Lemma 2.4.10 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 2.5.2 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 2.5.2 is valid after merging the three cubes. Therefore $X$ (after finishing Lemma 2.4.10 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 3 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 Section 2.7. We may also specify that Algorithm 2 takes a suitable merging set (2.5.1) 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 2.5.3.** If $\{H_1, H_2\}$ and $\{E_1, E_2, \ldots, E_n\}$ mentioned in Corollary 2.3.3 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 \( \begin{bmatrix} 1 & x \\ z & 1 \end{bmatrix} \). 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 Section 2.8 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 \( ith \) 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 9 in the algorithm and the fact that \( H_1 \) and \( H_2 \) make a Latin decomposition, for every \( 1 \leq i \leq 4^{2n} \), either all the \( F_i \)'s have a forward edge in the \( ith \) position or all the \( F_i \)'s have a backward edge in the \( ith \) 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 \( 2n \times 2n \) matrix \( Q \) via the following:

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

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 \( F_1 \) 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, 12, and 23 say that, for \( j = 0 \), \( i = s \), and \( k = 0 \), we have \( \dim = 0 \) and that for \( u = c[k][\dim] \) we have \( f[0][s][0] = e[0][u][0] \), but \( s \in S_v \), so we have \( f[0][s][0] = e[0][u][0] = v = m_{1,v} = m_{1,v} \). Therefore, \( q_{1,v} \) is well-defined and is equal to \( m_{1,v} \). Again, due to lines 10, 12, and 23, 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 2.5.4.** If \( Q_{2n} \) has a source cycle, so does \( Q_{4n} \).

**Theorem 2.5.5.** If \( \{H_1, H_2, H_3\} \) and \( \{E_1, E_2, \ldots, E_n\} \) mentioned in Corollary 2.4.13 are Latin, then the resulting Hamilton decomposition \( \{g(E_i, H_j) \mid 1 \leq i \leq n \text{ and } 1 \leq j \leq 3\} \) is also Latin.

**Proof.** The proof is very similar to that of Theorem 2.5.3, 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 \[
\begin{bmatrix}
1 & 1 \\
1 & 1 \\
1 & 1 \\
\end{bmatrix}
\] then $g(E_1, H_1)$ is a source cycle for $Q_{6n}$

with $M' = \begin{bmatrix}
M & M \\
M & M \\
M & M \\
\end{bmatrix}$ as its source matrix. □

The last theorem gives rise to another important result:

**Corollary 2.5.6.** If $Q_{2n}$ has a source cycle, so does $Q_{6n}$.

Corollaries 2.5.4 and 2.5.6 give us the main result of this chapter:

**Corollary 2.5.7.** We have a source cycle for all $Q_{2n}$ with $n = 2^a3^b$.

For future research, we conjecture the following.

**Conjecture 2.5.8.** We have a source cycle for all $Q_{2n}$. 
2.6. Lemma 2.4.1 for $n = 1$

Figure 2.6: The decomposition discussed in Lemma 2.4.1 for $n = 1$. 
2.7. Algorithms

2.7.1. 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 \(i \leftarrow 0\) to \(1\) do  
2:   for \(j \leftarrow 0\) to \(n - 1\) do  
3:     \(c[i][j] \leftarrow 0\)  
4:   end for  
5: end for
6: for \(j \leftarrow 0\) to \(n - 1\) do  
7:   for \(i \leftarrow 0\) to \(4^{2n} - 1\) do  
8:     for \(k \leftarrow 0\) to \(1\) do  
9:       dir \(\leftarrow h[k][i][1]\)  
10:      dim \(\leftarrow h[k][i][0] - 1\)  
11:     end if  
12:     if dir = 0 then  
13:       \(f[j + kn][i][0] \leftarrow e[j][c[k][\text{dim}]][0] + n(\text{dim})\)  
14:     end if  
15:   end for  
16: end for
17: if \(c[k][\text{dim}] = 4^n\) then  
18:   dir \(\leftarrow 0\)  
19: end if  
20: if \(c[k][\text{dim}] = -1\) then  
21:   c[k][\text{dim}] \(\leftarrow 4^n - 1\)  
22: end if  
23: \(f[j + kn][i][0] \leftarrow e[j][c[k][\text{dim}]][0] + n(\text{dim})\)  
24: \(f[j + kn][i][1] \leftarrow 1 - e[j][c[k][\text{dim}]][1]\)  
25: end if  
26: end for  
27: end for  
28: end for
2.7.2. 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 \( i \)th 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 \( 4^{3n} \times 2 \) array \( H \) having the edges of \( X \).

**Algorithm 2** An algorithm for finding \( X \).

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

if $s[x][0] = y$ and $s[x][1] = z$ then ▷ $(x, y, z) \in S$

if dir = 0 then ▷ we have reached $S$ from outside of cube

dir ← 1 ▷ in the next step we exit from $S'$ in negative direction

$h[i][0] ← 3$ ▷ the $i$th edge is in dimension 3

$h[i][1] ← 0$ ▷ the $i$th edge is in positive direction

c ← c + 1 ▷ adding 1 to $z$ and $c$

$z ← z + 1$

if $z = 4^n$ then ▷ mod operations

$z ← 0$

end if

else ▷ we have reached $S$ from $S'$

dir ← 1 ▷ the $i$th edge is in dimension 1

$h[i][0] ← 1$ ▷ the $i$th edge is in negative direction

$h[i][1] ← 1$ ▷ subtracting 1 from $x$ and $c$

c ← c − 1

$x ← x − 1$

if $x = −1$ then ▷ mod operations

$x ← 4^n − 1$

end if

else if $sp[x][0] = y$ and $sp[x][1] = z$ then ▷ $(x, y, z) \in S'$

if dir = 0 then ▷ we have reached $S'$ from outside of cube

dir ← 1 ▷ in the next step we exit from $S$ in negative direction

$h[i][0] ← 3$ ▷ the $i$th edge is in dimension 3

$h[i][1] ← 1$ ▷ the $i$th edge is in negative direction

c ← c − 1 ▷ subtracting 1 from $z$ and $c$

$z ← z − 1$

if $z = −1$ then ▷ mod operations

$z ← 4^n − 1$

end if

else ▷ we have reached $S'$ from $S$

dir ← 1 ▷ the $i$th edge is in dimension 2

$h[i][0] ← 2$ ▷ the $i$th edge is in negative direction

$h[i][1] ← 1$

c ← c − 1 ▷ subtracting 1 from $z$ and $c$

$y ← y − 1$

if $y = −1$ then ▷ mod operations

$y ← 4^n − 1$

end if

end if
else if $d[x][0] = y$ and $d[x][1] = z$ then  
\[ \triangleright (x, y, z) \in D \]

if $\text{dir} = 0$ then  
\[ \triangleright \text{we have reached } D \text{ from } D' \]
$$h[i][0] \leftarrow 1$$  
\[ \triangleright \text{the } ith \text{ edge is in dimension 1} \]
$$h[i][1] \leftarrow 0$$  
\[ \triangleright \text{the } ith \text{ edge is in positive direction} \]
$c \leftarrow c + 1$  
\[ \triangleright \text{adding 1 to } x \text{ and } c \]
$x \leftarrow x + 1$

if $x = 4^n$ then  
\[ \triangleright \text{mod operations} \]
$x \leftarrow 0$

end if

else  
\[ \triangleright \text{we have reached } D \text{ from } V' \]
$$h[i][0] \leftarrow 3$$  
\[ \triangleright \text{the } ith \text{ edge is in dimension 3} \]
$$h[i][1] \leftarrow 0$$  
\[ \triangleright \text{the } ith \text{ edge is in positive direction} \]
$c \leftarrow c + 1$  
\[ \triangleright \text{adding 1 to } x \text{ and } c \]
$z \leftarrow z + 1$

if $z = 4^n$ then  
\[ \triangleright \text{mod operations} \]
$z \leftarrow 0$

end if

else if $dp[x][0] = y$ and $dp[x][1] = z$ then  
\[ \triangleright (x, y, z) \in D' \]

if $\text{dir} = 0$ then  
\[ \triangleright \text{we have reached } D' \text{ from outside of cube} \]
$$h[i][0] \leftarrow 3$$  
\[ \triangleright \text{the } ith \text{ edge is in dimension 3} \]
$$h[i][1] \leftarrow 1$$  
\[ \triangleright \text{the } ith \text{ edge is in negative direction} \]
$c \leftarrow c - 1$  
\[ \triangleright \text{subtracting 1 from } z \text{ and } c \]
$z \leftarrow z - 1$

if $z = -1$ then  
\[ \triangleright \text{mod operations} \]
$z \leftarrow 4^n - 1$

end if

else  
\[ \triangleright \text{we have reached } D' \text{ from } D \]
$$h[i][0] \leftarrow 1$$  
\[ \triangleright \text{the } ith \text{ edge is in dimension 1} \]
$$h[i][1] \leftarrow 1$$  
\[ \triangleright \text{the } ith \text{ edge is in negative direction} \]
$c \leftarrow c - 1$  
\[ \triangleright \text{subtracting 1 from } z \text{ and } c \]
$x \leftarrow x - 1$

if $x = -1$ then  
\[ \triangleright \text{mod operations} \]
$x \leftarrow 4^n - 1$

end if

end if

end if
else if \(m[x][0] = y\) and \(m[x][1] = z\) then
\(\triangleright (x, y, z) \in M\)

if \(\text{dir} = 0\) then
\(\triangleright \) we have reached \(M\) from \(M'\)
\(h[i][0] \leftarrow 1\)
\(\triangleright \) the \(i\)th edge is in dimension 1
\(h[i][1] \leftarrow 0\)
\(\triangleright \) the \(i\)th edge is in positive direction
\(c \leftarrow c + 1\)
\(\triangleright \) adding 1 to \(x\) and \(c\)
\(x \leftarrow x + 1\)

if \(x = 4^n\) then
\(\triangleright \) mod operations
\(x \leftarrow 0\)

end if

else
\(\triangleright \) we have reached \(M\) from outside of cube
\(\text{dir} \leftarrow 0\)
\(\triangleright \) in the next step we exit from \(M'\) in positive direction
\(h[i][0] \leftarrow 2\)
\(\triangleright \) the \(i\)th edge is in dimension 3
\(h[i][1] \leftarrow 0\)
\(\triangleright \) the \(i\)th edge is in positive direction
\(c \leftarrow c + 1\)
\(\triangleright \) adding 1 to \(y\) and \(c\)
\(y \leftarrow y + 1\)

if \(y = 4^n\) then
\(\triangleright \) mod operations
\(y \leftarrow 0\)

end if

end if

else if \(mp[x][0] = y\) and \(mp[x][1] = z\) then
\(\triangleright (x, y, z) \in M'\)

if \(\text{dir} = 0\) then
\(\triangleright \) we have reached \(M'\) from \(M\)
\(h[i][0] \leftarrow 1\)
\(\triangleright \) the \(i\)th edge is in dimension 1
\(h[i][1] \leftarrow 0\)
\(\triangleright \) the \(i\)th edge is in positive direction
\(c \leftarrow c + 1\)
\(\triangleright \) adding 1 to \(x\) and \(c\)
\(x \leftarrow x + 1\)

if \(x = 4^n\) then
\(\triangleright \) mod operations
\(x \leftarrow 0\)

end if

else
\(\triangleright \) we have reached \(M'\) from outside of cube
\(\text{dir} \leftarrow 1\)
\(\triangleright \) in the next step we exit from \(M\) in positive direction
\(h[i][0] \leftarrow 2\)
\(\triangleright \) the \(i\)th edge is in dimension 2
\(h[i][1] \leftarrow 1\)
\(\triangleright \) the \(i\)th edge is in negative direction
\(c \leftarrow c - 1\)
\(\triangleright \) subtracting 1 from \(y\) and \(c\)
\(y \leftarrow y - 1\)

if \(y = -1\) then
\(\triangleright \) mod operations
\(y \leftarrow 4^n - 1\)

end if

end if
145: else ▷ normal vertex
146: if dir = 0 then ▷ if the current direction is positive
147: if c = -1 (mod 4^n) then ▷ if it is time to move in dimension 2
148: h[i][0] ← 2 ▷ the $i$th edge is in dimension 2
149: h[i][1] ← 0 ▷ the $i$th edge is in the positive direction
150: c ← c + 1 ▷ adding 1 to $y$ and $c$
151: y ← y + 1
152: if $y = 4^n$ then ▷ mod operations
153: y ← 0
154: end if
155: else ▷ if it is time to move in dimension 1
156: h[i][0] ← 1 ▷ the $i$th edge is in dimension 1
157: h[i][1] ← 0 ▷ the $i$th edge is in positive direction
158: c ← c + 1 ▷ adding 1 to $x$ and $c$
159: x ← x + 1
160: if $x = 4^n$ then ▷ mod operations
161: x ← 0
162: end if
163: else ▷ if the current direction is negative
164: if c = 0 (mod 4^n) then ▷ if it is time to move in dimension 2
165: h[i][0] ← 2 ▷ the $i$th edge is in dimension 2
166: h[i][1] ← 1 ▷ the $i$th edge is in negative direction
167: c ← c - 1 ▷ subtracting 1 from $y$ and $c$
168: y ← y - 1
169: if $y = -1$ then ▷ mod operations
170: y ← $4^n - 1$
171: end if
172: else ▷ if it is time to move in dimension 1
173: h[i][0] ← 1 ▷ the $i$th edge is in dimension 1
174: h[i][1] ← 1 ▷ the $i$th edge is in negative direction
175: c ← c - 1 ▷ subtracting 1 from $x$ and $c$
176: x ← x - 1
177: if $x = -1$ then ▷ mod operations
178: x ← $4^n - 1$
179: end if
180: end if
181: end if
182: end if
183: end if
184: end for
2.7.3. 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:        $c[i][j] \leftarrow 0$ ▷ initializing the $x$-, $y$-, and $z$-coordinates of the three pointers
4:    end for
5: end for
6: for $j \leftarrow 0$ to $n - 1$ do
7:    for $i \leftarrow 0$ to $4^{3n} - 1$ do
8:        for $k \leftarrow 0$ to $2$ do
9:            dir $\leftarrow h[k][i][1]$ ▷ direction of the current edge in $H_{k+1}$
10:           dim $\leftarrow h[k][i][0] - 1$ ▷ dimension of the current edge in $H_{k+1}$
11:        if dir $= 0$ then ▷ if the current edge in $H_{k+1}$ is forward
12:            $f[j + kn][i][0] \leftarrow e[j][c[k][\text{dim}]][0] + n(\text{dim})$ ▷ dimension of the current edge in $F_{j+1+kn}$
13:            $f[j + kn][i][1] \leftarrow e[j][c[k][\text{dim}]][1]$ ▷ direction of the current edge in $F_{j+1+kn}$
14:        c[k][\text{dim}] $\leftarrow c[k][\text{dim}] + 1$ ▷ moving forward in the current copy of $E_{j+1}$
15:        if c[k][\text{dim}] $= 4^n$ then ▷ mod operations
16:           c[k][\text{dim}] $\leftarrow 0$
17:        end if
18:    else ▷ if the current edge in $H_{k+1}$ is backward
19:        c[k][\text{dim}] $\leftarrow c[k][\text{dim}] - 1$ ▷ moving backward in the current copy of $E_{j+1}$
20:    if c[k][\text{dim}] $= -1$ then ▷ mod operations
21:        c[k][\text{dim}] $\leftarrow 4^n - 1$
22:    end if
23:    f[j + kn][i][0] $\leftarrow e[j][c[k][\text{dim}]][0] + n(\text{dim})$ ▷ dimension of the current edge in $F_{j+1+kn}$
24:    f[j + kn][i][1] $\leftarrow 1 - e[j][c[k][\text{dim}]][1]$ ▷ direction of the current edge in $F_{j+1+kn}$
25: end if
26: end for
27: end for
28: end for
2.7.4. 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 $4^{2n} \times 2$ array $h$ having the source cycle for $G_{n,2}$.

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

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

```plaintext
1: for $i \leftarrow 0$ to $n - 1$ do ▷ 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 ▷ initializing the $x$- and $y$-coordinates of the pointer
12:     $c[i] \leftarrow 0$
13: end for
14: for $i \leftarrow 0$ to $4^{2n} - 1$ do ▷ cycling through edges of $H$
15:     $dir \leftarrow h[i][1]$ ▷ direction of the current edge in $H$
16:     $dim \leftarrow h[i][0] - 1$ ▷ dimension of the current edge in $H$
17:     if $dir = 0$ then ▷ if the current edge in $H$ is forward
18:         $f[i][0] \leftarrow e[c[dim]][0] + n(dim)$ ▷ dimension of the current edge in $F$
19:         $f[i][1] \leftarrow e[c[dim]][1]$ ▷ direction of the current edge in $F$
20:         $c[dim] \leftarrow c[dim] + 1$ ▷ moving forward in the current copy of $E$
21:     end if
22:     if $c[dim] = 4^n$ then ▷ mod operations
23:         $c[dim] \leftarrow 0$
24:     else ▷ if the current edge in $H$ is backward
25:         $c[dim] \leftarrow c[dim] - 1$ ▷ moving backward in the current copy of $E$
26:         if $c[dim] = -1$ then ▷ mod operations
27:             $c[dim] \leftarrow 4^n - 1$
28:         end if
29:         $f[i][0] \leftarrow e[c[dim]][0] + n(dim)$ ▷ dimension of the current edge in $F$
30:         $f[i][1] \leftarrow 1 - e[c[dim]][1]$ ▷ direction of the current edge in $F$
31:     end if
32: end for
```
2.7.5. 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.

Algorithm 5 A Source Cycle for $Q_{6n}$ From Source Cycles for $Q_{2n}$ and $G_{n,3}$

1: for $i \leftarrow 0$ to $n - 1$ do
2:   for $j \leftarrow 0$ to $n - 1$ do
3:     for $k \leftarrow 0$ to $2$ do
4:       for $t \leftarrow 0$ to $2$ do
5:         $z \leftarrow (k + t) \pmod{3}$
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 $2$ do
12:   $c[i] \leftarrow 0$
13: end for
14: for $i \leftarrow 0$ to $4^{3n} - 1$ do
15:   dir $\leftarrow h[i][1]$
16:   dim $\leftarrow h[i][0] - 1$
17:   if dir = 0 then
18:     $f[i][0] \leftarrow e[c[dim]][0] + n(dim)$
19:     $f[i][1] \leftarrow e[c[dim]][1]$
20:     c[dim] $\leftarrow c[dim] + 1$
21:   if c[dim] = $4^n$ then
22:     c[dim] $\leftarrow 0$
23:   end if
24: else
25:     c[dim] $\leftarrow c[dim] - 1$
26:     if c[dim] = $-1$ then
27:       c[dim] $\leftarrow 4^n - 1$
28:     end if
29:     $f[i][0] \leftarrow e[c[dim]][0] + n(dim)$
30:     $f[i][1] \leftarrow 1 - e[c[dim]][1]$
31: end if
32: end for
2.8. 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 \( ith \) edge of \( H_1 \) is 1 or \( \bar{1} \), then the \( ith \) edge of \( F_{j+1} \) is one of \( \{1, \bar{1}, 2, \bar{2}, \ldots, n, \bar{n}\} \), and if the \( ith \) edge of \( H_1 \) is 2 or \( \bar{2} \), then the \( ith \) edge of \( F_{j+1} \) is one of \( \{n + 1, \bar{n+1}, n + 2, \bar{n+2}, \ldots, 2n, \bar{2n}\} \). 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 \( ith \) edges of \( F_{j+1} \) \( F_{j+1+n} \) are:

  - If the \( ith \) edge of \( H_1 \) is from \((a, b)\) to \((a + 1, b)\), then the \( ith \) edge of \( F_{j+1} \) has the same direction and dimension as the \((a + 1)st\) edge of \( E_{j+1} \).
  
  - If the \( ith \) edge of \( H_1 \) is from \((a, b)\) to \((a, b + 1)\), then the \( ith \) 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 \( ith \) edge of \( H_1 \) is from \((a, b)\) to \((a - 1, b)\), then the \( ith \) edge of \( F_{j+1} \) has direction opposite to that of the \( ath \) edge of \( E_{j+1} \) and dimension equal to that of the \( ath \) edge of \( E_{j+1} \).
  
  - If the \( ith \) edge of \( H_1 \) is from \((a, b)\) to \((a, b - 1)\), then the \( ith \) edge of \( F_{j+1} \) has
direction opposite to that of the \textit{bth} edge of $E_{j+1}$ and dimension equal to $n$
plus the dimension of the \textit{bth} edge of $E_{j+1}$.

- Similar statements can be made 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 $(0,0)$. 
Chapter 3. Contraction Decomposition

3.1. Introduction

In this chapter, all graphs are undirected, but may not be simple, that is, they may have loops and parallel edges. We may emphasize this fact by stating that a graph is a multigraph. Let $e$ be an edge of a graph $G$. The contraction of $e$ in $G$, denoted $G/e$, is given by identifying the endpoints of $e$ and then deleting $e$ itself. We can see that when $e$ is a loop, contracting $e$ is the same as deleting it. A graph $H$ is a minor of $G$ if it can be obtained from $G$ by a series of vertex deletions, edge deletions, and edge contractions. In this case, we write $H \preceq_m G$. Let $u$ and $v$ be two distinct vertices of a graph $G$, and let $P$ and $Q$ be two $(u, v)$-paths. The paths $P$ and $Q$ are called internally-disjoint if $V(P) \cap V(Q) = \{v, u\}$. For a graph $G$ on vertex set $V$, a partition of $V$ into $k$ subsets is a grouping of elements of $V$ into $k$ disjoint non-empty subsets. The number $k$ is the size of the partition. A partition of the edge set of $G$ is defined analogously. A graph $G$ is called $(H, k)$-positive if its edge set can be partitioned into $E(G) = E_1 \cup E_2 \cup \cdots \cup E_k$ such that $G/(E \setminus E_i)$ does not contain $H$ as a minor for $i \in \{1, 2, \ldots, k\}$. If $G$ is not $(H, k)$-positive, then it is $(H, k)$-negative. A $k$-tree is formed by starting with the complete graph $K_{k+1}$ and then repeatedly doing the following: Recognize a subgraph isomorphic to $K_k$ and add a new vertex adjacent exactly to the vertices of the $K_k$ subgraph. A graph has treewidth at most $k$ if it is a subgraph of a $k$-tree. Throughout this chapter, a graph is said to be the smallest among a set of graphs, if it has the fewest number of edges. By the intersection of two graphs, we mean the intersection of their edge sets, treated as a graph.
3.2. History

In 1971, Chartrand, Geller, and Hedetniemi asked, among many other questions, whether every planar graph is the union of two outerplanar graphs [7]. In 1996, Ding, Oporowski, Sanders, and Vertigan [10] (and, independently, Kedlaya [14]) showed that every planar graph is the union of two series-parallel graphs. In 2005, Gonçalves showed the conjecture of Chartrand, Geller, and Hedetniemi to be true [12]. Continuing in the same spirit, James Oxley asked the matroid theory question of whether the ground set of every cographic matroid may be partitioned into two sets such that the deletion of either set results in a series-parallel matroid [16]. Translating from matroids to graphs, we get the following, which is the main topic of this chapter [16]:

**Conjecture 3.2.1** (Morgan, Oporowski). *Every graph is $(K_4, 2)$-positive.*

This conjecture is currently open, but some partial results are known. A result by Demaine, Hajiaghayi, and Mohar [9] guarantees the existence of a 2 edge coloring for graphs of bounded Euler genus such that, contracting each color set, the resulting graph has bounded treewidth. The result of Gonçalves [12] settles Conjecture 3.2.1 for planar graphs, while Theorem 3.3.2, stated in the next section, proves the conjecture for 4-edge-connected graphs. We propose and deal with two slightly different questions derived from changing the parameters in Conjecture 3.2.1.
3.3. First Question

It is clear that the condition of a graph being \((K_4, 3)\)-positive is weaker than being \((K_4, 2)\)-positive. Is it true that every graph is \((K_4, 3)\)-positive? The answer to this question is affirmative.

**Theorem 3.3.1.** Every graph is \((K_4, 3)\)-positive.

In order to prove this assertion we need some build-up. For a partition \(\mathcal{P} = \{V_1, V_2, \ldots, V_k\}\) of \(V\), the loopless multigraph \(G_{\mathcal{P}}\) is defined as follows:

- \(G_{\mathcal{P}}\) has \(k\) vertices \(\{v_1, v_2, \ldots, v_k\}\),
- For every edge of \(G\) that is between \(V_i\) and \(V_j\), there is an edge in \(G_{\mathcal{P}}\) between \(v_i\) and \(v_j\).

Our proof of Theorem 3.3.1 will make use of a classical result in graph connectivity, known as the Nash-Williams Theorem, proved independently by Tutte [23] and Nash-Williams [17], which is stated below.

**Theorem 3.3.2.** A graph \(G\) has \(k\) edge-disjoint spanning trees if and only if for every partition \(\mathcal{P}\) of \(V(G)\), the multigraph \(G_{\mathcal{P}}\) has at least \(k(|\mathcal{P}| - 1)\) edges.

This is a very general result and we only need the case \(k = 3\), which we state as a corollary.

**Corollary 3.3.3.** A graph \(G\) has 3 edge-disjoint spanning trees if and only if for every partition \(\mathcal{P}\) of \(V(G)\), the multigraph \(G_{\mathcal{P}}\) has at least \(3|\mathcal{P}| - 3\) edges.

**Lemma 3.3.4.** Let \(G\) be a connected graph with \(|G| > 1\) and let \(E_1 \subseteq E(G)\). Then \(E_1\) is connected and spanning if and only if \(|G/E_1| = 1\).

**Proof.** If \(E_1\) has more than one component, contracting it results more than one vertex.
Also if \( v \in V(G) \setminus V(E_1) \) and \( u \in V(E_1) \), then \( u \) and \( v \) are two distinct vertices in \( G/E_1 \).

On the other hand, if \( E_1 \) is connected and spanning, then clearly \( G/E_1 \) is a single vertex, possibly with some loops. \( \square \)

If \( G \) is loopless, the graph \( 2G \) is obtained by adding an edge in parallel to every edge of \( G \), so that every parallel class doubles in size. Motivated by Theorem 3.3.2 and in order to prove Theorem 3.3.1, we define a graph \( G \) to be \( k \)-Nash-Williams if \( E(G) \) can be partitioned into \( E(G) = E_1 \cup E_2 \cup \ldots \cup E_k \) such that \( G/(E \setminus E_i) \) is a single vertex (possibly with some loops) for \( i \in \{1, 2, \ldots, k\} \). The following lemma enables us to put Corollary 3.3.3 into use.

Lemma 3.3.5. Let \( G \) be a connected and simple graph. Then \( G \) is \( 3 \)-Nash-Williams if and only if \( 2G \) has three edge-disjoint spanning trees.

Proof. Let \( G \) be \( 3 \)-Nash-Williams with \( E(G) = E_1 \cup E_2 \cup E_3 \). Put \( T_1 = E_1 \cup E_2 \), \( T_2 = E_1 \cup E_3 \), and \( T_3 = E_2 \cup E_3 \). Then \( |G/T_i| = 1 \) for \( 1 \leq i \leq 3 \). By Lemma 3.3.4, the graphs \( T_i \) are connected and spanning, and thus, each contain a spanning tree of \( G \). Furthermore each edge of \( G \) is in exactly two of the \( T_i \)'s, which means that \( E(2G) \) is the disjoint union of \( T_i \)'s, and thus \( 2G \) has three edge-disjoint spanning trees.

For the other direction, suppose that \( 2G \) has three edge-disjoint spanning trees \( T_1, T_2, \) and \( T_3 \). These three trees may not cover all the edges of \( 2G \). Let \( R = E(2G) \setminus (E(T_1) \cup E(T_2) \cup E(T_3)) \). For an edge \( e \) of \( 2G \), let \( f(e) \) be the unique edge that is parallel to \( e \). Partition \( R \) as follows:
\[ R_1 = \{ e \in R \mid f(e) \in T_1 \}, \]
\[ R_2 = \{ e \in R \mid f(e) \in T_2 \}, \]
\[ R_3 = \{ e \in R \mid f(e) \in T_3 \}, \]
\[ R_4 = \{ e \in R \mid f(e) \in R \}. \]

Edges in \( R_4 \) come in parallel pairs, so let \( R_4 = R_5 \cup R_6 \), where \( R_5 \) and \( R_6 \) each contain one edge from each parallel class of \( R_4 \). Finally let
\[ F_1 = T_1 \cup R_2 \cup R_5, \]
\[ F_2 = T_2 \cup R_3 \cup R_6, \]
\[ F_3 = T_3 \cup R_1. \]

Note that \( F_i \)'s may no longer be trees, but each one is connected and spanning, and they all form a partition the edge set of \( 2G \). Furthermore, parallel edges do not belong to the same \( F_i \). Put
\[ E_1 = \{ e \in E(2G) \mid e \in F_1 \text{ and } f(e) \in F_2 \}, \]
\[ E_2 = \{ e \in E(2G) \mid e \in F_1 \text{ and } f(e) \in F_3 \}, \]
\[ E_3 = \{ e \in E(2G) \mid e \in F_2 \text{ and } f(e) \in F_3 \}. \]

Then \( E(2G) = E_1 \cup E_2 \cup E_3 \). Also \( E_1 \cup E_2 = \{ e \in E(2G) \mid e \in F_1 \text{ or } f(e) \in F_1 \} \), so \( E_1 \cup E_2 \) is an isomorphic copy of \( F_1 \) in \( G \), and thus is a connected spanning subgraph of \( G \). By Lemma 3.3.4, we conclude that \( G/(E_1 \cup E_2) \) is a single vertex. A similar argument applies to the other two contractions, and thus \( G \) is 3-Nash-Williams. \( \square \)
The following proposition is a major step towards the proof of Theorem 3.3.1.

**Proposition 3.3.6.** Every simple graph \( G \) with at least \( \frac{3}{2}(n - 1) \) edges has a 3-Nash-Williams subgraph.

**Proof.** For \( n \leq 2 \), every simple graph has automatically fewer than \( \frac{3}{2}(n - 1) \) edges, and for \( n = 3 \), the only simple graph with 3 edges is \( K_3 \) which is itself 3-Nash-Williams. Now let \( G \) be a smallest counterexample. This implies that \( G \) is connected and \( G \) itself is not 3-Nash-Williams. By Lemma 3.3.5, the graph \( 2G \) does not have 3 edge-disjoint spanning trees. By Theorem 3.3.2, it has a partition \( \mathcal{P} \) of size \( p \) such that \( 2G_{\mathcal{P}} \) has fewer than \( 3(p-1) \) edges, and thus \( G_{\mathcal{P}} \) has fewer than \( \frac{3}{2}(p-1) \) edges. Note that \( p > 1 \). Let \( X_1, X_2, \ldots, X_p \) be the partition sets, and consider \( G[X_i] \) for \( 1 \leq i \leq p \).

**Lemma 3.3.7.** \( |E(G[X_i])| < \frac{3}{2}(|X_i| - 1) \)

**Proof.** Suppose the lemma fails, so that for some \( 1 \leq i \leq p \), the induced graph \( G[X_i] \) has at least \( \frac{3}{2}(|X_i| - 1) \) edges. Since \( G[X_i] \) is smaller than \( G \), it is not a counterexample to Proposition 3.3.6, and thus has some 3-Nash-Williams subgraph \( N \), but \( N \) is a subgraph of \( G \) as well; a contradiction. \( \Box \)

We can now finish the proof of Proposition 3.3.6. Considering the partition \( \mathcal{P} \), there are two types of edges in \( G \):

1. Edges between two different sets in \( \mathcal{P} \), and
2. Edges inside a set in \( \mathcal{P} \).

By the discussion before Lemma 3.3.7, there are fewer than \( \frac{3}{2}(p-1) \) edges of first type, and by Lemma 3.3.7, there are fewer than \( \sum_{i=1}^{p} \frac{3}{2}(|X_i| - 1) \) edges of second type. Thus \( G \)
has fewer than \( \frac{3}{2}(p - 1) + \sum_{i=1}^{p} \frac{3}{2}(|X_i| - 1) = \frac{3}{2}p - \frac{3}{2} + \frac{3}{2}n - \frac{3}{2}p = \frac{3}{2}(n - 1) \) edges, a contradiction.

\[ \square \]

We are now ready to prove the first major result of this chapter, Theorem 3.3.1.

**Proof of Theorem 3.3.1.** Let \( G \) be a smallest \((K_4, 3)\)-negative graph. First we will establish some basic facts about \( G \), listed as bullet points below.

- **\( G \) is loopless.** Clearly loops do not contribute to the creation of a \( K_4 \) minor.

- **\( G \) has no parallel edges.** Let \( e \) and \( f \) be two edges in parallel between some vertices \( u \) and \( v \), and let \( G' = G/e \setminus f \cong G/f \setminus e \). Since \( G' \) is smaller than \( G \), it is \((K_4, 3)\)-positive, say, with \( E(G') = E_1' \cup E_2' \cup E_3' \). Putting \( E_1 = E_1' \cup e \), letting \( E_2 = E_2' \cup f \), and setting \( E_3 = E_3' \), we see that the extra edges \( e \) and \( f \) get contracted in \( G/(E_1 \cup E_2) \) and get reduced to loops in \( G/(E_1 \cup E_3) \) and \( G/(E_2 \cup E_3) \). This shows that \( G \) is \((K_4, 3)\)-positive as well, a contradiction.

- **\( G \) is connected.** Let \( G = G' \cup G'' \) be the disjoint union of two graphs. Since \( G' \) and \( G'' \) are smaller than \( G \), they are \((K_4, 3)\)-positive, say, with \( E(G') = E_1' \cup E_2' \cup E_3' \) and \( E(G'') = E_1'' \cup E_2'' \cup E_3'' \). Putting \( E_1 = E_1' \cup E_1'' \), letting \( E_2 = E_2' \cup E_2'' \), and setting \( E_3 = E_3' \cup E_3'' \) shows that \( G \) is \((K_4, 3)\)-positive as well, a contradiction.

- **\( \delta(G) \) is greater than one.** It is not hard to see that pendant vertices do not contribute to the creation of a \( K_4 \) minor.

Showing that \( \delta(G) = 2 \) is impossible is more involved, so we present a more detailed reasoning. For a contradiction, suppose that there are vertices \( v, x, \) and \( y \) such that \( N(v) = \{x, y\} \).

If \( xy \in E(G) \), then \( vxy \) is a triangle. Let \( G' = G/\{vx, vy\} \). Since \( G' \) is smaller than
$G$, it is $(K_4, 3)$-positive, say, with $E(G') = E'_1 \cup E'_2 \cup E'_3$. Putting $E_1 = E'_1 \cup \{vx\}$, letting $E_2 = E'_2 \cup \{vy\}$, and setting $E_3 = E'_3 \cup \{xy\}$, shows that $G$ is $(K_4, 3)$-positive as well; a contradiction.

If $xy \notin E(G)$, let $G' = G/vx$. Note that $xy \in E(G')$. Since $G'$ is smaller than $G$, it is $(K_4, 3)$-positive, say, with $E(G') = E'_1 \cup E'_2 \cup E'_3$. Without loss of generality, we may suppose that $xy \in E'_1$. Put $E_1 = E'_1 \cup \{vx, vy\}$, let $E_2 = E'_2$, and set $E_3 = E'_3$. We can easily verify that $G/(E_1 \cup E_2) = G'/E'_1 \cup E'_2$ and $G/(E_1 \cup E_3) = G'/E'_1 \cup E'_3$. The third graph $G/(E_1 \cup E_2)$ has two edges $vx$ and $vy$ in place of one edge $xy$ of $G'/E'_1 \cup E'_2$, but $vx$ and $vy$ are in series, so they will not contribute to the creation of a $K_4$ minor. This proves that $\delta(G) \geq 3$.

We have shown that $G$ is simple and connected and satisfies $\delta(G) \geq 3$. By Proposition 3.3.6, the graph $G$ has a 3-Nash-Williams subgraph $H$ with $E(H) = E''_1 \cup E''_2 \cup E''_3$ such that $H/(E(H) \setminus E_i)$ is a single vertex. Let $G' = G/H$. Since $G'$ is smaller than $G$, it is $(K_4, 3)$-positive, say, with $E(G') = E'_1 \cup E'_2 \cup E'_3$. Putting $E_1 = E'_1 \cup E''_1$, letting $E_2 = E'_2 \cup E''_2$, and setting $E_3 = E'_3 \cup E''_3$, shows that $G$ is $(K_4, 3)$-positive as well; a contradiction. □

A cactus is a connected graph where every block is an edge, two parallel edges, or a cycle. Examining the proof of Theorem 3.3.1 more closely, we can be more specific about the contractions $G/(E_1 \cup E_2)$, $G/(E_1 \cup E_3)$, and $G/(E_2 \cup E_3)$. The following corollary may be proved using a similar argument.

**Corollary 3.3.8.** Every graph $G$ is $(K_4, 3)$-positive such that $G/(E_1 \cup E_2)$, $G/(E_1 \cup E_3)$, and $G/(E_2 \cup E_3)$ are cacti with loops.
3.4. Second Question

Replacing the \(K_4\) in Conjecture 3.2.1 with \(K_{2,3}\), we ask the following: Is it true that every graph is \((K_{2,3}, 2)\)-positive? The answer to this question is negative, which is the second major result of this chapter.

**Theorem 3.4.1.** There exists a graph \(G\) such that for every partition \(E(G) = E_1 \cup E_2\) we have \(G/E_1 \succeq_m K_{2,3}\) or \(G/E_2 \succeq_m K_{2,3}\).

Note that \(K_{2,3}\) is incomparable to \(K_4\) in the minor relation, so Theorem 3.4.1 does not resolve Conjecture 3.2.1.

**Proof.** Consider the graph \(H\) given in Figure 3.1. There are three pairwise internally-disjoint \((x, y)\)-paths in \(H\). Let the one containing \(u'\) be named \(P_t\), the one containing \(v'\) be named \(P_b\), and the one with length three be named \(P_m\). There are two pendant vertices: \(u\) and \(v\). We treat \(H\) like an edge between \(u\) and \(v\). We now take the complete bipartite graph \(K_{2,3}\) with \(u_1\) and \(u_2\) on one side and \(v_1\), \(v_2\), and \(v_3\) on the other side, and replace each of its six edges with a copy of \(H\). The \(H\)-copy between \(u_i\) and \(v_j\) is called \(H_{i,j}\). We name the resulting graph \(G\).

![Diagram of graphs H and G](image)

**Figure 3.1:** The structure of \(G\).
There may be, and in fact there are, many graphs that satisfy the conditions of Theorem 3.4.1. We prove that $G$ is one such graph. Suppose that $E(G) = E_1 \cup E_2$ and consider $G/E_1$. In at least one copy of $H$, say $H_{1,1}$, the edges of $E_1$ must contain an $H_{1,1}$-path from $u_1$ to $v_1$, otherwise $G/E_1$ has a $K_{2,3}$-minor, and the conclusion holds. So without loss of generality, suppose that $E_1$, restricted to $H_{1,1}$, has a path $P$ from $u'_{1,1}$ to $v'_{1,1}$. There are four cases regarding $P$:

Case 1: $P_m \subseteq P$.

In this case we have $P_m \subseteq P \subseteq E_1$, which implies that:

- $E_2 \cap P_m = \emptyset$,
- $|E_2 \cap P| \leq 3$, and
- $|E_2 \cap P_b| \leq 3$.

We can see that $G/E_2$ has three pairwise internally-disjoint $(x_{1,1}, y_{1,1})$-paths, each of length at least two, which implies that $G/E_2 \geq_m K_{2,3}$.

Case 2: $P_m \not\subseteq P$, but each of $E_2 \cap P_t$, $E_2 \cap P_m$, and $E_2 \cap P_b$ has cardinality at least 2.
In this case $G/E_1$ has three internally-disjoint $(x_{1,1}, y_{1,1})$-paths, each of length at least two, which means that $G/E_1 \geq_m K_{2,3}$.

Case 3: $P_m \notin P$, and $|E_2 \cap P_m| \leq 1$.

In this case we have:

- $|E_2 \cap P_t| \leq 3$,
- $|E_2 \cap P_b| \leq 3$, and
- $|E_2 \cap P_m| \leq 1$.

Similarly to Case 1, the graph $G/E_2$ has three pairwise internally-disjoint $(x_{1,1}, y_{1,1})$-paths, each of length at least two, which implies that $G/E_2 \geq_m K_{2,3}$.

Case 4: $P_m \notin P$, and $|E_2 \cap P_t| \leq 1$ (the case $|E_2 \cap P_b| \leq 1$ is argued similarly).

Consider $G/E_2$. Contract $P_m$ and $P_b$ to a single vertex and name it $z$. The graph $G/E_2$ has three internally-disjoint $(u'_{1,1}, z)$-paths: two in $H_{1,1}$ and one using the other copies of $H$. These paths have length at least two, which implies that $G/E_2 \geq_m K_{2,3}$. 
It follows that $G$ has a $K_{2,3}$ minor in every case, which concludes the proof. □
Appendix. Copyright Information

arXiv Publishing Agreement

“...granting rights for arXiv to distribute an article does not preclude later copyright assignment. Authors are thus free to publish submissions that already appear on arXiv.”

For more on this see

https://arxiv.org/help/license
Bibliography


Vita

Farid Bouya was born in 1988 in Tehran, Iran. He finished his undergraduate degree in computer science at Sharif University of Technology-Tehran, Iran in 2012 and earned a master of science degree in mathematics from Louisiana State University in May 2014. He is currently a candidate for the degree of Doctor of Philosophy in mathematics.