Graph Powers: Hardness Results, Good Characterizations and Efficient Algorithms

Dissertation

zur

Erlangung des akademischen Grades
Doktor-Ingenieur (Dr.-Ing.)
der Fakultät für Informatik und Elektrotechnik
der Universität Rostock

Vorgelegt von
MSc. Ngoc Tuy Nguyen

Rostock, November 2009

urn:nbn:de:gbv:28-diss2009-0206-0

Gutachter:	Prof. Dr. Van Bang Le Universität Rostock. Institut fur̈ Informatik
Gutachter:	Prof. Dr. Egon Wanke Heinrich- Heine- Universität Düsseldorf. Institut fur̈ Informatik
Gutachter:	Prof. Dr. Ekkehard Köhler Brandenburgische Technische Universität Cottbus. Institut fur̈ Mathematik

Tag der öffentlichen Verteidigung: 30. Oktober 2009

Abstract

Given a graph $H = (V_H, E_H)$ and a positive integer k, the *k-th power* of H, written H^k, is the graph obtained from H by adding new edges between any pair of vertices at distance at most k in H; formally, $H^k = (V_H, \{xy \mid 1 \leq d_H(x, y) \leq k\})$.
A graph G is the k-th power of a graph H if $G = H^k$, and in this case, H is a *k-th root* of G. For the cases of $k = 2$ and $k = 3$, we say that H^2 and H^3 is the *square*, respectively, the *cube* of H and H is a *square root* of $G = H^2$, respectively, a *cube root* of $G = H^3$.

In this thesis we study the computational complexity for recognizing k-th powers of general graphs as well as restricted graphs. This work provides new NP-completeness results, good characterizations and efficient algorithms for graph powers. The main results are the following.

- There exist reductions proving the NP-completeness for recognizing k-th powers of general graphs for fixed $k \geq 2$, recognizing k-th powers of bipartite graphs for fixed $k \geq 3$, recognizing k-th powers of chordal graphs, and finding k-th roots of chordal graphs for all fixed $k \geq 2$.

- The *girth* of G, girth(G), is the smallest length of a cycle in G,

  - For all fixed $k \geq 2$, recognizing of k-th powers of graphs with girth at most $2\lfloor \frac{k}{2} \rfloor + 2$ is NP-complete.
  - There is a polynomial time algorithm to recognize if $G = H^2$ for some graph H of girth at least 6. This algorithm also constructs a square root of girth at least 6 if one exists.
  - There exists a good characterization of squares of a graph having girth at least 7. This characterization not only leads to a simple algorithm to compute a square root of girth at least 7 but also shows such a square root, if it exists, is unique up to isomorphism.
  - There is a good characterization of cubes of a graph having girth at least 10 that gives a recognition algorithm in time $O(nm^2)$ for such graphs. Moreover, this algorithm constructs a cube root of girth at least 10 if it exists.

These results almost provide a dichotomy theorem for the complexity of the recognition problem in terms of girth of the square roots.

- There is a good characterization of squares of strongly chordal split graphs that gives a recognition algorithm in time $O(\min\{n^2, m \log n\})$ for such squares. Moreover, this algorithm also constructs a strongly chordal split graph square root if it exists.

- There exists a good characterization and a linear-time recognition algorithm for squares of block graphs. This algorithm also constructs a block graph square root if one exists. Moreover, block graph square roots in which every endblock is an edge are unique up to isomorphism.

The almost results in thesis have been published in the following papers of journal and proceedings of conferences.

- "Computing Graph Roots Without Short Cycles", *Proceedings of the* 26^{th} *International Symposium on Theoretical Aspects of Computer Science* (STACS 2009), pp. 397 - 408.
  Co-authors: Babak Farzad (Brock University, Canada), Lap Chi Lau (The Chinese University of Hong Kong), Van Bang Le (University of Rostock).

- "Hardness Results and Efficient Algorithms for Graph Powers", to appear in: *Proceedings of the* 35^{th} *International Workshop on Graph-Theoretic Concepts in Computer Science* (WG 2009).
  Co-author: Van Bang Le (University of Rostock).

- "The square of a block graph", to appear in: *Discrete Mathematics* (2009), doi:10.1016/j.disc.2009.09.004.
  Co-author: Van Bang Le (University of Rostock).

Acknowledgments

Although a few words do not do justice to their contribution I would like to thank the following people for making this work possible.

First of all I would like to express my sincere gratitude to my advisor Professor Van Bang Le, for his patience, invaluable advice and encouragement throughout. His lectures and advices have been of great value for my research. This work would not have been possible without his support, patience and encouragement.

I wish to express my sincere thanks to Professor Andreas Brandstädt for a lot of personal help during the time. I thank Professor Egon Wanke and Professor Ekkehard Köhler for their reading and commenting on my thesis.

I am grateful to my co-authors for the joint works and successful discussions, and for allowing me to include the results in this thesis.

I am also grateful to colleagues of the Institute of theoretical informatics for their help and support during my study. Special thanks go to Roswitha Fengler, Katrin Erdmann, Christian Hundt and Ragnar Nevries for a lot of help.

Furthermore, I also would like to express my gratitude to my university, HongDuc, who recommended and encouraged me to study well. Especially, I am truly deeply grateful to all my colleagues at the Department of Computer Science, HongDuc University for their encouragements.

While working on this thesis, financial support by the Ministry of Education and Training, Vietnam (322 Project) by Grant No.3766/QD-BGD&DT is gratefully acknowledged. I am also grateful to DAAD for their support for me in learning German.

Great thanks go to my buddies for their support and encouragement. I also want to thank to all friends in Rostock - "Rostockers" for everything we shared together over the years of living.

Last but not least, I am indebted to my parents, my sisters and brothers who have given me the best opportunities. All my love and thanks go to my wife Hạnh and my children Quỳnh Anh and Minh Lộc. Without their love and gentle encouragement I could not complete this work. Thank you very much!

Rostock, November 4, 2009 *Nguyễn Ngọc Túy*

Contents

List of Figures

Part I

Introduction

Chapter 1

Overview

1.1 Introduction

In graph theory, graph H is a *root* of graph $G = (V, E)$ if there exists a positive integer k such that x and y are adjacent in G if and only if their distance in H is at most k. If H is a k-th root of G, then we write $G = H^k$ and call G the k-th power of H. Graph powers and roots are fundamental graph-theoretic concepts and have been extensively studied in the literature, both in theoretic and algorithmic senses; see e.g., [17, 29, 42, 45, 46, 49] for recent results and the numerous references listed there.

The first motivation comes directly from the fact that *root* and *root computing* are concepts familiar to most branches of mathematics. Any result on graph roots and powers would be a very valuable contribution in studying graph theory. Graph powers are also very useful in designing efficient algorithms for certain combinatorial optimization problems. For instance, the papers [2, 1] used a (in linear time constructible) Hamiltonian cycle in the cube T^3 of the nontrivial tree T in their approximation algorithm for the Traveling Salesman Problem (TSP) with a parameterized triangle inequality. For the same problem, the approximation proposed in [9] used a Hamiltonian cycle in the square G^2 of a 2-connected graph G. Note that these graph power-based algorithms are the best known approximations for this variant of the TSP problem. Note also that the square of any 2-connected graph is Hamiltonian is a fundamental and deep theorem in graph theory due to Fleischner [30], and the PhD thesis [44] provided an algorithm that constructs a Hamiltonian cycle in such a square in polynomial time.

Moreover, graph roots and powers are of great interest and have a number of possible applications in different disciplines. For example, in distributed computing (cf. [55, 51]), the k-th power of a graph G represents the possible flow of information during k rounds of communication of a distributed network of processors organized according to G.

In radio frequency planning, certain situations are closely related to the coloring of the power of the associated radio network; in such applications, the associated radio network is the graph where vertices represent the transmitters and adjacencies between vertices indicate possible interferences. To avoid interference, transmitters which are close (at distance at most k in the graph) receive different frequencies. This problem is exactly the coloring problem on k-th powers of the graph. There are many studies on this problem. For example (cf. [8, 10, 21, 22, 69]), radio frequency planning without direct and hidden collisions in the radio network G (also known as $L(1,1)$-labeling problem or distance-2 coloring problem) is equivalent to the coloring of the square G^2 of G, and has been well-studied.

In computational biology, graph powers and roots, in particular tree powers and roots, are useful in the reconstruction of phylogeny (cf. [20, 50, 57, 15]).

Graph powers and roots are fundamental graph-theoretic concepts and have been extensively studied in literature, both in theoretic and algorithmic senses. These investigations considered both characterization and recognition problems, see [13, Sec. 10.6] for a survey and [46, 45] for the most recent papers.

For the characterization problem, in 1960, Ross and Harary [63], first studied the concept of a square of graphs. They characterized squares of trees and showed that tree square roots, when they exist, are unique up to isomorphism. Next, in 1967, Mukhopadhyay [56] provided a characterization of general graphs which have a square root. In 1974, Escalante *et al.* [27] characterized graphs and digraphs with a k-th root. However, these characterizations are not good in the sense that they do not lead to a polynomial time recognition algorithms for such as graphs. In fact, such a good characterization may not exist as Motwani and Sudan proved that it is NP-complete to determine if a given graph has a square root [55].

The computational complexity of recognizing of k-th powers of graphs was unresolved until 1994 when Motwani and Sudan [55] proved that recognizing of square of graphs is NP-complete. Very recently, in 2006, Lau [45] proved the NP-completeness for recognizing of cubes of graphs. He conjectured that recognizing k-th powers of some graph is NP-complete for all fixed $k \geq 2$ (cf. Conjecture 2.2.1, p. 11) and recognizing k-th powers of bipartite graphs is NP-complete for all fixed $k \geq 3$ (cf. Conjecture 2.2.2, p. 12).

By the above-mentioned negative results of Motwani and Sudan, and of Lau, a natural way (in theory and in applications) in considering k-th powers of graphs is to restrict the root graphs H.

In this thesis, we study the computational complexity for recognizing k-th powers of general graphs as well as restricted graphs. In particular, the aim of this work is as follows.

- Reductions proving the NP-completeness for recognizing k-th powers of general graphs for fixed $k \geq 2$, recognizing k-th powers of bipartite graphs for fixed

$k \geq 3$, recognizing k-th powers of chordal graphs, finding k-th roots of chordal graphs for all fixed $k \geq 2$, and recognizing k-th powers of graphs with girth at most $2\lfloor \frac{k}{2} \rfloor + 2$ for all fixed $k \geq 2$.

- Good characterizations of squares of a graph having girth at least seven and cubes of a graph having girth at least ten. Squares of graphs with girth at least six can be recognized in polynomial time.

- Good characterization of squares of strongly chordal split graphs that leads to a recognition algorithm in time $O(\min\{n^2, m \log n\})$ for such squares.

- Good characterization and a linear-time recognition algorithm for squares of block graphs.

1.2 Contributions of the thesis

There are three parts in this thesis. In Part I after a brief introduction of motivations and overview of thesis (Chapter 1), Chapter 2 provides basic notion and facts used throughout the thesis.

The main results of this thesis appear in Part II and Part III.

Part II includes the NP-completeness results for recognizing k-th powers of graphs. Chapter 3 recalls the NP-completeness results on graph power recognition problems in the literature. This chapter presents also some tools that are used in reductions in Chapter 4 and Chapter 5. Chapter 4 gives reductions proving the Conjecture 2.2.1 and Conjecture 2.2.2 are indeed true. This chapter also shows that recognizing k-th powers of chordal graphs and k-th roots of chordal graphs are NP-complete for all fixed $k \geq 2$. Chapter 5 considers the complexity for recognizing k-th powers of graphs with girth conditions. It shows that recognizing of k-th powers of graphs with girth at most $2\lfloor \frac{k}{2} \rfloor + 2$ is NP-complete.

Part III includes four chapters presenting efficient algorithms for recognizing squares and cubes of some restricted graphs. Chapter 6 provides a good characterization for graphs that are squares of some graph of girth at least seven. This characterization not only leads to a simple algorithm to compute a square root of girth at least 7 but also shows that such a square root, if it exists, is unique up to isomorphism. This chapter also shows that squares of graphs with girth at least six can be recognized in polynomial time. Chapter 7 gives a good characterization of graphs that are cubes of a graph having girth at least 10. This characterization leads to the immediate consequence that recognizing cubes of (C_4, C_6, C_8)-free bipartite graphs is polynomial time, whereas recognizing cubes of general bipartite is NP-complete. Chapter 8 shows that there exists a good characterization of squares of strongly chordal split graphs that gives a recognition algorithm in time $O(\min\{n^2, m \log n\})$

for such squares. Part III is closed by Chapter 9. This chapter provides good characterizations for squares of block graphs and a linear-time recognition algorithm for such squares. This algorithm also constructs a square block graph root if one exists. Moreover, block graph square roots in which every endblock is an edge are unique up to isomorphism.

Chapter 2

Background

Section 2.1 provides basic notions and definitions which are necessary for understanding the remaining part of this thesis. In Section 2.2 we recall two fundamental problems on graph powers, and collect several facts that are related to this work.

2.1 Definitions and notation

Notions and definitions not given here can be found in any standard textbook on graph theory or graph algorithms, e.g. [11, 13, 68, 33, 35].

In the following we always consider finite, undirected and simple graphs $G = (V, E)$ where V is the vertex set of G and E the edge set of G. To avoid ambiguities we sometimes use $V(G)$ or V_G to refer to V and $E(G)$ or E_G to refer to E. The cardinality of the vertex set V is denoted by n, and the cardinality of the edge set E is denoted by m. A set of graphs is generally called a *graph class*. The *complement* $\overline{G}$ of a graph G is the graph with vertex set V_G defined by $uv \in E_{\overline{G}}$ if and only if $uv \notin E_G$. A graph $H = (V_H, E_H)$ is a *subgraph* of $G = (V, E)$ if $V_H \subseteq V$ and $E_H \subseteq E$. A graph $H = (V_H, E_H)$ is an *induced subgraph* of G, if it is a subgraph of G and it contains all the edges uv such that $u, v \in V_H$ and $uv \in E_G$. We say that H is induced by V_H and write $G[V_H]$ for H. Given a set of vertices $X \subseteq V$, if $X = \{a, b, c, \ldots\}$, we write $G[a, b, c, \ldots]$ for $G[X]$. Also, we often identify a subset of vertices with the subgraph induced by that subset, and vice versa.

Let $\mathcal{F}$ denote a set of graphs. A graph G is $\mathcal{F} - free$ if none of its induced subgraphs is isomorphic to a graph in $\mathcal{F}$.

G is *connected* if for all $u, v \in V$, $u \neq v$, there is a path u, v-path in G connecting u and v; otherwise, G is *disconnected.* A *maximal connected subgraph* of G is a subgraph that is connected and is not contained in any other connected subgraph. The *connected components* of a graph are its maximal connected subgraphs.

Let $G = (V, E)$ be a connected graph. A subset $S \subset V$ is a *cutset (or separator)* of G if $G[V \setminus S]$ is disconnected. For a positive integer k, a *k-connected component* in a graph G is a maximal (induced) k-connected subgraph of G; the 1-connected components of G are the usual connected components, and the 2-connected components of G are also called *blocks* of G. A *k-cut* in a graph is a cutset with k vertices; a 1-cut is also called a *cut-vertex*. An *endblock* in a graph is a block that contains at most one cut-vertex of the graph.

An *isomorphism* from G to H is a bijection $f : V_G \to V_H$ such that $uv \in E_G$ if and only if $f(u)f(v) \in E_H$ for all $u, v \in V_G$. We say "G is isomorphic to H".

If G has a u, v-path, then the *distance* from u to v, written $d_G(u, v)$ is the length, i.e., number of edges, of a shortest path in G between u and v. The *diameter* of G, written $diam(G)$, is the maximum distance between two vertices (and ∞ if G is not connected).

For $k \geq 1$, let P_k denote a chordless path with k vertices and $k-1$ edges, and for $k \geq 3$, let C_k denote a chordless cycle with k vertices and k edges. The length of the cycle is the number k of its edges. An *even (odd) path (cycle)* is a path (cycle) of even (odd) length. Chordless cycles C_k, $k \geq 5$, are *holes*. Chordless cycles C_{2k+1}, $k \geq 2$, are *odd holes*. Complements $\overline{C}_k$ of chordless cycles C_k, $k \geq 5$ are *antiholes*. Complements $\overline{C}_{2k+1}$ of odd holes C_{2k+1}, are *odd antiholes*.

Let $G = (V_G, E_G)$ be a graph. We often write $xy \in E_G$ for $\{x, y\} \in E_G$. We sometimes also write $x \leftrightarrow y$ for the adjacency of x and y in the graph in question; this is particularly the case when we describe reductions in NP-completeness proofs. For disjoint sets of vertices X and Y, we write $X \leftrightarrow Y$, meaning each vertex in X is adjacent to each vertex in Y; if $X = \{x\}$, we simply write $x \leftrightarrow Y$.

The *girth* of G, girth(G), is the smallest length of a cycle in G; in case G has no cycles, we set girth$(G) = \infty$. In other words, G has girth k if and only if G contains a cycle of length k but does not contain any (induced) cycle of length $\ell = 3, \ldots, k-1$. It is quite usual to measure the sparseness of a graph in terms of its girth; a graph is "sparse" if its girth is "large enough". A girth$(G) = \infty$ if and only if G is a tree.

Note that the girth of a graph can be computed in time $O(nm)$ [41].

Definition 2.1.1 (Powers of Graphs)

Let $H = (V_H, E_H)$ be a graph. Given a positive integer k, the k-th power *of H, written H^k, is the graph obtained from H by adding new edges between any pair of vertices at distance at most k in H; formally, $H^k = (V_H, \{xy \mid 1 \leq d_H(x, y) \leq k\})$. A graph G is the k-th power of a graph H if $G = H^k$, and in this case, H is a k-th* root *of G.*

For the cases of $k = 2$ and $k = 3$, we say that H^2 and H^3 is the square, *respectively, the* cube *of H and H is a* square root *of $G = H^2$, respectively, a* cube root *of $G = H^3$.*

Definition 2.1.2 (Neighborhood)

Let $G = (V, E)$ be a graph and $v \in V$.

- $N(v) = \{u \mid u \in V, u \neq v \text{ and } uv \in E\}$ *denotes the (open) neighborhood of* v,
- $N[v] = N(v) \cup \{v\}$ *denotes the closed neighborhood of* v,
- $N^k(v) = \{u \mid u \in V \text{ and } d(u, v) = k\}$ *denotes the* k*-th neighborhood of* v,

Definition 2.1.3 (Degree of Vertex)

Let $G = (V, E)$ be a graph and $v \in V$. Set $\deg_G(v) = |N_G(v)|$, *the* degree *of v in G. The* maximum degree *is denoted* $\Delta(G)$, minimum degree *is denoted* $\delta(G)$. *A vertex with degree 0 is called an* isolated vertex. *We call vertices of degree one in G* end-vertices *of G. A* center vertex (or universal vertex) *of G is one that is adjacent to all other vertices.*

Definition 2.1.4 (Clique and Stable Set)

Let $G = (V, E)$ be a graph. A set of vertices $Q \subseteq V$ is called a clique *in G if every two distinct vertices in Q are adjacent; A* stable set or an independent set *is a set of pairwise non-adjacent vertices.*

A maximal clique (stable set) *is a clique (stable set) that is not properly contained in another clique(stable set).*

A clique (stable set) *is maximum if its cardinality is the maximum possible size of a clique (stable set) in G.*

The clique number *of graph, written $\omega(G)$, is the maximum clique size in G.*

The independence number *of graph, written $\alpha(G)$, is the maximum independent set size in G.*

The chromatic number *of graph, written $\chi(G)$, is the minimum number of colors needed to label the vertices so that adjacent veritices receive different colors.*

The clique cover number *of graph, written $\theta(G)$, is the minimum number of cliques in G needed to cover V_G.*

The next definitions give some special graph classes that we need in later chapters.

Definition 2.1.5 (Intersection Graph)

For a given set M of objects (for which intersection makes sense), the intersection graph G_M *of these objects has M as vertex set, and two objects are adjacent in G_M if the intersection of the corresponding objects is non-empty.*

Definition 2.1.6 (Some Special Graph Classes)

- *A* complete graph *is one in which every two distinct vertices are adjacent; a complete graph on k vertices is also denoted by K_k.*
- *A* star *is a graph with* at least two *vertices that has a center vertex and the other vertices are pairwise non-adjacent. Note that a star contains at least one edge and at least one center vertex; the center vertex is unique whenever the star has more than two vertices.*
- *A graph is a* block graph *if it is connected and its blocks (2-connected components) are cliques.*
- *A graph is a* parity graph *if for any two induced paths joining the same pair of vertices, the path lengths have the same parity (i.e., they are both odd or both even).*
- *A graph G is* perfect *if $\chi(H) = \omega(H)$ for every induced subgraph H of G.*
- *A graph G is* Berge graph *if it does not contain an odd hole or an odd antihole. A graph is perfect if and only if it is a Berge graph [23].*
- *A graph G is* chordal *if it contains no induced cycle of length at least four.*
- *A graph G is* weakly chordal *if G and $\overline{G}$ contain no induced cycle of length at least five.*
- *A chordal graph is* strongly chordal *if it does not contain any ℓ-sun as an induced subgraph; here a ℓ*-sun*, $\ell \geq 3$, consists of a stable set $\{u_1, u_2, \ldots, u_\ell\}$ and a clique $\{v_1, v_2, \ldots, v_\ell\}$ such that for $i \in \{1, \ldots, \ell\}$, u_i is adjacent to exactly v_i and v_{i+1} (index arithmetic modulo ℓ).*
- *A graph is a* split graph *if its vertex set can be partitioned into a clique and stable set. Clearly split graphs are chordal.*
- *A graph is an* interval graph *if it has an intersection model consisting of intervals on a straight line.*
- *A* proper interval graph *is an interval graph that has an intersection model in which no interval properly contains another.*
- *A graph is* bipartite *if there is a partition of its vertex set into two disjoint stable sets called the* bipartition *of G. It is well known that a graph is bipartite if and only if its chromatic number is at most 2.*
- *G is a* forest *if G contains no cycle. G is a* tree *if G connected and contains no cycle.*

A vertex v is *simplicial* if its neighborhood is a clique (equivalently, if it belongs to exactly one maximal clique). Let $\sigma = (v_1, v_2, \ldots, v_n)$ be an ordering of the vertices in a graph G. We say that σ is a *simplicial elimination ordering (or perfect elimination ordering)* if for all $i \in \{1, \ldots, n\}$, the vertex v_i is simplicial in $G_i = G[v_i, \ldots, v_n]$. It is proven that

Theorem 2.1.7 ([26]) *A graph is chordal if and only if it has a simplicial elimination ordering.*

Furthermore, a simplicial elimination ordering of a chordal graph can be computed in linear time [62].

2.2 Graph powers and related works

Since the power of a graph H is the union of the powers of the connected components of H, throughout this thesis, we assume that all graphs considered are connected.

Graph powers and roots are fundamental graph-theoretic concepts and have been extensively studied in the literature, both in theoretic and algorithmic senses; see e.g., [17, 29, 42, 45, 46, 49] for recent results and the numerous references listed there.

Characterizing and recognizing graph powers are two fundamental problems in the study of graph powers.

For characterization problem, one gives necessary and sufficient properties of graphs that are powers of some graph. In this way, in 1960, Ross and Harary [63] characterized squares of trees and showed that tree square roots, when they exist, are unique up to isomorphism. In 1968, Rao [59] gave a necessary and sufficient conditions for a graph to be the cube of a tree. He further showed that if G is non-complete and is the cube of a tree, then G has an unique cube root. Furthermore, he also obtained a criterion for a graph to be the fourth power of a tree [60]. In 1967, Mukhopadhyay [56] provided a characterization of graphs which have a square root. He found that a connected undirected graph G with vertices $v_1, \ldots, v_n$ has a square root if and only if G contains a collection of n cliques $G_1, \ldots, G_n$ such that for all $1 \leq i, j \leq n$:

1. $\bigcup_{1 \leq i \leq n} G_i = G$,
2. $v_i \in G_i$,
3. $v_i \in G_j$ if and only if $v_j \in G_i$.

In 1997, Flotow [31] gave sufficient conditions for graphs whose powers are chordal graphs and graphs whose powers are interval graphs.

On the other hand, the following recognition problem has attracted much attention in recent years.

k-TH POWER OF GRAPH
Instance: A graph G.
Question: Is there a graph H such that $G = H^k$?

This is motivated by the fact that all above-mentioned characterizations are not polynomial in the sense that they do not lead to a polynomial time recognition algorithms for such graphs.

The complexity of k-TH POWER OF GRAPH was unresolved until 1994 when Motwani and Sudan [55] proved that it is NP-complete to determine if a given graph has a square root. This result implies that good characterizations of graph powers may not exist in general. Moreover, by the negative results of Motwani and Sudan, a natural way (in theory and in applications) in considering graph powers is to restrict the root graphs H. Formally, given a graph class $\mathcal{C}$ and an integer $k \geq 2$, the restricted recognition problem is as follows.

k-TH POWER OF $\mathcal{C}$-GRAPH
Instance: A graph G.
Question: Is there a graph H in $\mathcal{C}$ such that $G = H^k$?

The case when $\mathcal{C}$ is the class of all trees is well-studied ([63, 34, 49, 42, 17, 45, 15]). In 1995, Lin and Skiena [49] gave an algorithm that recognizes squares of trees in linear time. In 1998, Kearney and Corneil [42] provided an algorithm to recognize k-th powers of trees in cubic time for all fixed $k \geq 2$. Recently, in 2006, Chang *et al.* [17] improved Kearney and Corneil's result by reducing the running time to linear. New and simpler linear-time algorithms for recognizing squares of trees are given in [15, 45].

Following this line of research, several important graph classes are of great interest. In 1995, Lin and Skiena [49] gave a linear-time algorithm to find square roots of planar graphs based on the characterization of Harary, Karp and Tutte [36]. In 2004, Lau and Corneil [46] also studied recognizing powers of proper interval, split, and chordal graphs. They showed that recognizing squares of chordal graphs and split graphs are NP-complete, whereas recognizing squares of proper interval graphs is polynomial time.

Recently, in 2006, Lau [45] showed that recognizing squares of bipartite graphs is polynomially solvable, while cubes of bipartite graphs is NP-complete. He conjectured that recognizing k-th powers of graphs for all fixed $k \geq 2$ and recognizing k-th powers of bipartite graphs for all fixed $k \geq 3$ are hard.

k-TH POWER OF BIPARTITE GRAPH
Instance: A graph G.
Question: Is there a *bipartite graph* H such that $G = H^k$?

Conjecture 2.2.1 ([45]) *For all fixed $k \geq 2$, k*-TH POWER OF GRAPH *is NP-complete.*

Conjecture 2.2.2 ([45]) *For all fixed $k \geq 3$, k*-TH POWER OF BIPARTITE GRAPH *is NP-complete.*

Notice that for all fixed $k \geq 3$, recognizing k-th powers of split graphs is trivial since k-th powers of split graphs, $k \geq 3$, are exactly the complete graphs. However, the computational complexity of recognizing k-th powers of chordal graphs is unknown so far.

Furthermore, we recall that the *girth* of a graph is the smallest length of a cycle in the graph. Let g be a natural number. Graphs with girth at least g form graph class $\mathcal{C}(g)$ that properly contains all trees. In this work, we deal with powers of graphs with girth conditions. In this direction, we have found several results (see Chapters 5, 6, 7).

Part II

NP-completeness

Chapter 3

Preliminaries

In this chapter, we first will recall the NP-completeness results on graph power recognition problems in the literature. Then we present some tools that will be used in our reductions in Chapters 4 and 5.

3.1 Introduction

Recently Lau and Corneil [46, 45] have shown that the following problems are NP-complete.

SQUARE OF CHORDAL GRAPH
Instance: A graph G.
Question: Is there a *chordal graph* H such that $G = H^2$?

SQUARE OF SPLIT GRAPH
Instance: A graph G.
Question: Is there a *split graph* H such that $G = H^2$?

SQUARE ROOT OF CHORDAL GRAPH
Instance: A *chordal graph* G.
Question: Is there a graph H such that $G = H^2$?

CUBE OF BIPARTITE GRAPH
Instance: A graph G.
Question: Is there a *bipartite graph* H such that $G = H^3$?

Furthermore, Lau [45] believed that k-TH POWER OF GRAPH and k-TH POWER OF BIPARTITE GRAPH are also NP-complete (cf. Conjecture 2.2.1 and Conjecture 2.2.2).

Chapter 4 deals with Conjecture 2.2.1 and Conjecture 2.2.2. First, we show that these conjectures are indeed true. Besides, we generalize the problem SQUARE ROOT OF CHORDAL GRAPH to k-TH OF CHORDAL GRAPH and k-TH ROOT OF CHORDAL

GRAPH. In particular, we prove that the following problems are NP-complete.

k-TH POWER OF CHORDAL GRAPH
Instance: A graph G.
Question: Is there a *chordal* graph H such that $G = H^k$?

k-TH ROOT OF CHORDAL GRAPH
Instance: A *chordal* graph G.
Question: Is there a (*perfect*) graph H such that $G = H^k$?

Finally, in Chapter 5 we will use similar techniques to Chapter 4 to prove the NP-completeness results of the following problems.

SQUARE OF GRAPH WITH GIRTH ≤ 4
Instance: A graph G.
Question: Is there a graph H with girth ≤ 4 such that $G = H^2$?

k-TH POWER OF GRAPH WITH GIRTH $\leq 2\lfloor \frac{k}{2} \rfloor + 2$
Instance: A graph G.
Question: Is there a graph H with girth $\leq 2\lfloor \frac{k}{2} \rfloor + 2$ such that $G = H^k$?

3.2 Preliminaries

In proving NP-completeness results we will consider the well-known NP-complete problem SET SPLITTING ([32, Problem SP4]), also known as HYPERGRAPH 2-COLORABILITY.

SET SPLITTING
Instance: Collection D of subsets of a finite set S.
Question: Is there a partition of S into disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2?

We will consider the following small instance of SET SPLITTING to illustrate our reductions in Chapter 4 and Chapter 5.

Example. $S = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7\}$ and $D = \{d_1, d_2, d_3\}$ with
$d_1 = \{u_2, u_3, u_4\}$, $d_2 = \{u_1, u_5\}$ and $d_3 = \{u_3, u_4, u_6, u_7\}$.
In this example, $S_1 = \{u_1, u_2, u_3\}$ and $S_2 = \{u_4, u_5, u_6, u_7\}$ is a possible solution.

We also apply the *tail structure* of a vertex v, first described in [55], and generalized later in [45]. The tail structure of a vertex v enables us to pin down exactly the neighborhood of v in any k-th root H of G.

Lemma 3.2.1 ([55]) *Let a, b, c be vertices of a graph G such that*

- *the only neighbors of a are b and c,*
- *the only neighbors of b are $a, c,$ and d,*

- *c and d are adjacent.*

Then the neighbors, in $V_G - \{a, b, c\}$, of d in any square root of G are the same as the neighbors, in $V_G - \{a, b, d\}$, of c in G; see Figure 3.1.

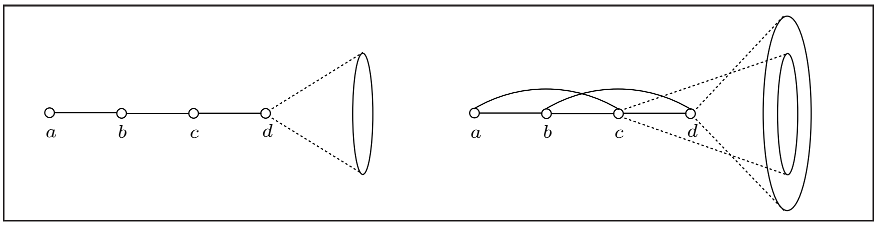

Figure 3.1: Tail in H (left) and in $G = H^2$ (right)

Lemma 3.2.2 ([45]) *Let $G = (V_G, E_G)$ be a connected graph with $\{v_1, \ldots, v_{k+1}\} \subset V_G$ where $N_G(v_1) = \{v_2, \ldots, v_{k+1}\}$ and $N_G(v_i) \subset N_G[v_{i+1}]$ for all $1 \leq i \leq k$. Then in any k-th root H of G,*

- $N_H(v_1) = \{v_2\}$,
- $N_H(v_i) = \{v_{i-1}, v_{i+1}\}$ *for all* $2 \leq i \leq k$,
- $N_H(v_{k+1}) - v_k = N_G(v_2) - \{v_1, \ldots, v_{k+1}\}$.

The vertices $v_1, \ldots, v_k$ are "tail vertices" of v_{k+1}; see Figure 3.2 for an illustration.

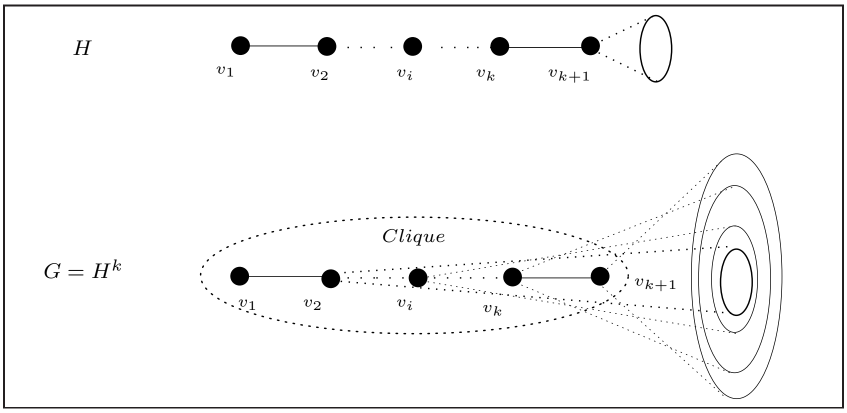

Figure 3.2: Tail in k-th root H of G and in G

We remark that k-TH POWER OF $\mathcal{C}$-GRAPH is obviously in NP whenever recognizing $\mathcal{C}$ is polynomially because guessing a k-th root H, verifying if H is in $\mathcal{C}$ and checking if $G = H^k$ can be done in polynomial time. This is the case for all graph classes considered in this thesis.

In our reductions we say that vertex u *reaches* v in k steps if u has a path in H of length at most k to v. Moreover, we say also that u reaches v in *exactly* k steps, meaning u has a path in H of length k to v.

Chapter 4

Powers of Bipartite and Chordal Graphs

In this chapter, we first prove that both Conjectures 2.2.1 and 2.2.2 are indeed true.

Next, in Section 4.2 we show that recognizing k-th powers of chordal graphs is NP-complete.

Finally, in Section 4.3 we provide a reduction to prove the NP-completeness of finding k-th roots of chordal graphs.

4.1 Powers of bipartite graphs

This section shows that the following are NP-complete.

k-TH POWER OF BIPARTITE GRAPH
Instance: A graph G.
Question: Is there a *bipartite graph* H such that $G = H^k$?

k-TH POWER OF GRAPH
Instance: A graph G.
Question: Is there a graph H such that $G = H^k$?

We prove that for fixed $k \geq 3$, k-TH POWER OF BIPARTITE GRAPH is NP-complete by reducing SET SPLITTING to it.

Our reduction generalizes the one in [45] for CUBE OF BIPARTITE GRAPH. Let $S = \{u_1, \ldots, u_n\}$, $D = \{d_1, \ldots, d_m\}$ where $d_j \subseteq S$, $1 \leq j \leq m$, be an instance of SET SPLITTING, and let $k \geq 3$ be a fixed integer. We construct an instance $G = G(D, S)$ for k-TH POWER OF BIPARTITE GRAPH as follows.

The vertex set of G consists of:

- U_i, for all $1 \leq i \leq n$. Each 'element vertex' U_i corresponds to the element u_i

in S.

- D_j, for all $1 \leq j \leq m$. Each 'subset vertex' D_j corresponds to the subset d_j in D.
- $D_j^1, \ldots, D_j^k$, for all $1 \leq j \leq m$. k 'tail vertices' $D_j^1, \ldots, D_j^k$ of the subset vertex D_j.
- $P_1^1, \ldots, P_1^{k-2}$ and $P_2^1, \ldots, P_2^{k-2}$ are $k-2$ pairs of 'partition vertices'.
- Connection vertex: X.

The edge set of G consists of:

- Edges of tail vertices:

  (E_1) $D_j, D_j^1, \ldots, D_j^k$ form a clique;

  (E_2) For all $1 \leq t \leq k-1$: $D_j^t \leftrightarrow \{U_i \mid u_i \in d_j, 1 \leq i \leq n\}$;

  (E_3) For all $1 \leq t \leq k-2$: $D_j^t \leftrightarrow X$, $D_j^t \leftrightarrow \{D_{j'} \mid d_j \cap d_{j'} \neq \varnothing\}$,
  $D_j^t \leftrightarrow \{P_1^h \mid 1 \leq h \leq k-t-1\}$ and $D_j^t \leftrightarrow \{P_2^h \mid 1 \leq h \leq k-t-1\}$;

  (E_4) For all $1 \leq t \leq k-3$: $D_j^t \leftrightarrow \{U_i \mid 1 \leq i \leq n\}$,
  $D_j^t \leftrightarrow \{D_{j'}^h \mid 1 \leq h \leq k-t-2, d_j \cap d_{j'} \neq \varnothing\}$;

  (E_5) For all $1 \leq t \leq k-4$: $D_j^t \leftrightarrow \{D_{j'} \mid 1 \leq j' \leq m\}$;

  (E_6) For all $1 \leq t \leq k-5$: $D_j^t \leftrightarrow \{D_{j'}^h \mid 1 \leq h \leq k-t-4, 1 \leq j' \leq m\}$.

- Edges of subset vertices:

  (E_7) $D_j \leftrightarrow \{X, P_1^1, \ldots, P_1^{k-2}, P_2^1, \ldots, P_2^{k-2}\}$,
  $D_j \leftrightarrow \{U_i \mid 1 \leq i \leq n\}$, $D_j \leftrightarrow \{D_{j'} \mid d_j \cap d_{j'} \neq \varnothing\}$.

  (E_8) If $k \geq 4$: $D_j \leftrightarrow \{D_{j'} \mid 1 \leq j' \leq m\}$.

- Edges of element vertices:

  (E_9) $U_1, \ldots, U_n$ form a clique, and $U_i \leftrightarrow \{X, P_1^1, \ldots, P_1^{k-2}, P_2^1, \ldots, P_2^{k-2}\}$.

- Edges of partition vertices:

  (E_{10}) $P_1^1, \ldots, P_1^{k-2}, X, U_1, \ldots, U_n$ form a clique;
  $P_2^1, \ldots, P_2^{k-2}, X, U_1, \ldots, U_n$ form a clique.

  (E_{11}) For all $1 \leq t \leq k-3$, $P_1^t \leftrightarrow \{P_2^h \mid 1 \leq h \leq k-t-2\}$,
  $P_2^t \leftrightarrow \{P_1^h \mid 1 \leq h \leq k-t-2\}$.

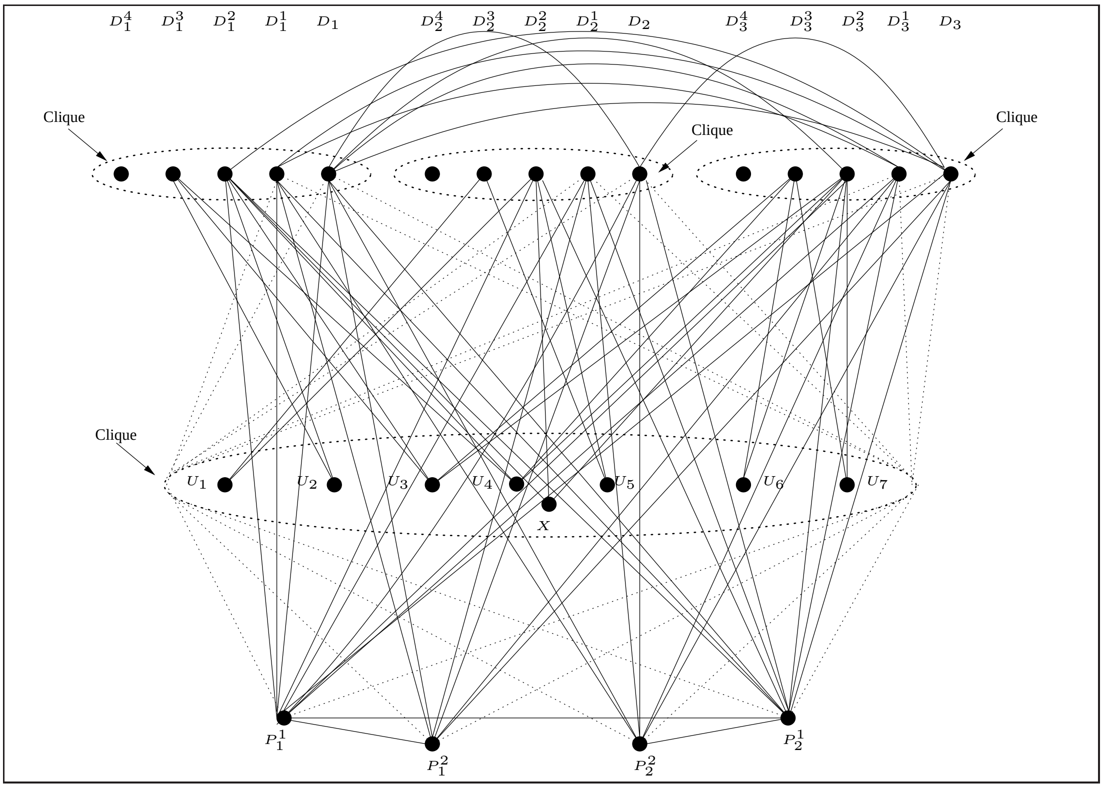

Figure 4.1: The graph G for the example instance of SET SPLITTING and $k = 4$

Clearly, G can be constructed from D, S in polynomial time. For an illustration, in case $k = 4$, the example instance yields the graph G depicted in Figure 4.1. In this and other figures in this chapter, each ellipse corresponds to a clique and we omit the clique edges to keep the figures simpler. The two dotted lines from a vertex to the cliques mean that the vertex is adjacent to all vertices in those cliques. The 4-th root graph H of G to the solution S_1, S_2 is shown in Figure 4.2.

Lemma 4.1.1 *If there exists a partition of S into two disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2, then there exists a bipartite graph H such that $G = H^k$.*

Proof. Let H have the same vertex set as G. The edges of H are as follows; see also Figure 4.3.

- Edges of subset vertices and their tail vertices: For all $2 \leq t \leq k$, $D_j^t \leftrightarrow D_j^{t-1}$ and $D_j^1 \leftrightarrow D_j$, and $D_j \leftrightarrow \{U_i \mid u_i \in d_j, 1 \leq i \leq n\}$.

- Edges of partition vertices:
  $P_1^1 \leftrightarrow \{U_i \mid u_i \in S_1, 1 \leq i \leq n\}$ and $P_2^1 \leftrightarrow \{U_i \mid u_i \in S_2, 1 \leq i \leq n\}$, and for all $2 \leq t \leq k-2$, $P_1^t \leftrightarrow P_1^{t-1}$ and $P_2^t \leftrightarrow P_2^{t-1}$.

- Edges of connection vertex: $X \leftrightarrow \{U_1, \ldots, U_n\}$.

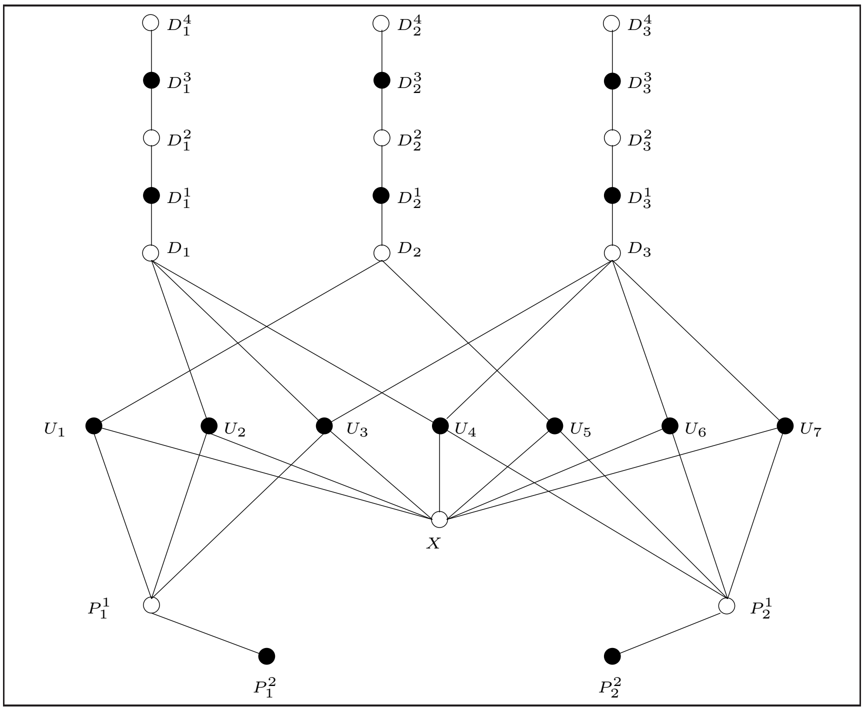

Figure 4.2: The bipartite 4-th root graph H of G to the solution S_1, S_2

Now we verify that the edge set of H^k is equal to the edge set of G. We do this by following the order of the presentation of the edge set of G; cf. (E_1) – (E_{11}).

For D_j^k, it is clear that $D_j, D_j^1, \ldots, D_j^k$ form a clique in H^k for all j, hence $N_{H^k}(D_j^k) = N_G(D_j^k)$.

For $1 \leq t \leq k-1$, the vertex D_j^t reaches D_j in t steps. By the construction of H, $D_j \leftrightarrow U_i$ whenever $u_i \in d_j$. Thus within $t+1$ steps, D_j^t reaches U_i whenever $u_i \in d_j$, therefore $D_j^t \leftrightarrow U_i$ in H^k whenever $u_i \in d_j$. By comparing with (E_2), $N_{H^k}(D_j^t) = N_G(D_j^t)$.

For $1 \leq t \leq k-2$, the vertex D_j^t reaches D_j in t steps. Since $D_j \leftrightarrow U_i$ whenever $u_i \in d_j$ and for all i, $U_i \leftrightarrow X$ in H, so within k steps, D_j^t reaches U_i whenever $u_i \in d_j$, and D_j^t reaches X. Also, in H, D_j reaches $D_{j'}$ in two steps whenever $d_j \cap d_{j'} \neq \varnothing$, thus within k steps, D_j^t reaches $D_{j'}$ whenever $d_j \cap d_{j'} \neq \varnothing$. Furthermore, since we have a solution for SET SPLITTING, every D_j has a common neighbor with P_1^1 and a common neighbor with P_2^1. So D_j^t reaches $P_1^1, \ldots, P_1^{k-t-1}$ and D_j^t reaches $P_2^1, \ldots, P_2^{k-t-1}$ within k steps. By comparing with (E_3), $N_{H^k}(D_j^t) = N_G(D_j^t)$.

For $1 \leq t \leq k-3$, the vertex D_j^t reaches D_j in t steps. Moreover, D_j reaches U_i within three steps for all i. So D_j^t reaches U_i within k steps for all i. Also, in two steps, D_j reaches $D_{j'}$ whenever $d_j \cap d_{j'} \neq \varnothing$, within k steps, D_j^t reaches

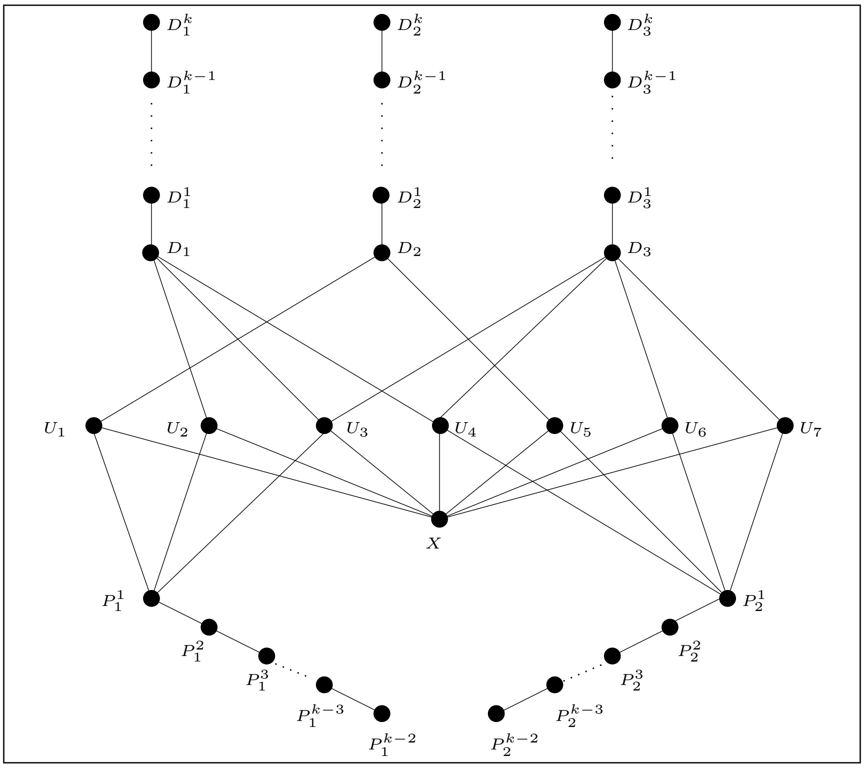

Figure 4.3: A bipartite k-th root H in Lemma 4.1.1 to the example solution S_1, S_2

$D_{j'}^1, \ldots, D_{j'}^{k-t-2}$ whenever $d_j \cap d_{j'} \neq \varnothing$. By comparing with (E_4), for all $1 \leq t \leq k-3$, $N_{H^k}(D_j^t) = N_G(D_j^t)$.

For $1 \leq t \leq k-4$, the vertex D_j^t reaches D_j in t steps. Also, D_j reaches $D_{j'}$ for all $j \neq j'$ in only four steps, thus within k steps, D_j^t reaches $D_{j'}$ for all $j \neq j'$. By comparing with (E_5), for all $1 \leq t \leq k-4$, $N_{H^k}(D_j^t) = N_G(D_j^t)$.

For $1 \leq t \leq k-5$, the vertex D_j^t reaches D_j in t steps. Also, D_j reaches $D_{j'}^1$ for all $j \neq j'$ in only five steps, thus within k steps, D_j^t reaches $D_{j'}^1, \ldots, D_{j'}^{k-t-4}$ for all $j \neq j'$. By comparing with (E_6), for all $1 \leq t \leq k-5$, $N_{H^k}(D_j^t) = N_G(D_j^t)$.

Next, the vertex D_j reaches U_i in one step whenever $u_i \in d_j$. In H, $X \leftrightarrow U_i$, so within k steps, D_j reaches X and D_j reaches U_i for all i. Moreover, since we have a solution for SET SPLITTING, every D_j has a common neighbor with P_1^1 and a common neighbor with P_2^1. In H, $\{P_1^1, \ldots, P_1^{k-2}\}$ and $\{P_2^1, \ldots, P_2^{k-2}\}$ are paths of length $k-3$. Thus, within k steps, D_j reaches $P_1^1, \ldots, P_1^{k-2}$, $P_2^1, \ldots, P_2^{k-2}$. Also, it is clear that D_j reaches $D_{j'}$ in only two steps whenever $d_j \cap d_{j'} \neq \varnothing$. Furthermore, if $k \geq 4$, D_j and $D_{j'}$ reach X in two steps, therefore D_j is adjacent to $D_{j'}$ for all $j \neq j'$. By comparing with (E_7) and (E_8), $N_{H^k}(D_j) = N_G(D_j)$.

Now consider U_i. In H, $U_i \leftrightarrow X$, hence $U_1, \ldots, U_n$ form a clique in H^k. Moreover, in H, $U_i \leftrightarrow P_1^1$ whenever $u_i \in S_1$, and $U_i \leftrightarrow P_2^1$ whenever $u_i \in S_2$, and $\{P_1^1, \ldots, P_1^{k-2}\}$ and $\{P_2^1, \ldots, P_2^{k-2}\}$ are paths of length $k-3$ in H. Thus, within k steps U_i reaches $P_1^1, \ldots, P_1^{k-2}, P_2^1, \ldots, P_2^{k-2}$. By comparing with (E_9), $N_{H^k}(U_i) = N_G(U_i)$ for all i.

Finally, for partition vertices, it is clear that $P_1^1, \ldots, P_1^{k-2}$, X, $U_1, \ldots, U_n$ form a clique in H^k and $P_2^1, \ldots, P_2^{k-2}$, X, $U_1, \ldots, U_n$ form a clique in H^k. Moreover, for all $1 \le t \le k-3$, since P_1^1 and P_2^1 have at least one neighbor U_i in H. Also, since $U_i \leftrightarrow X$ in H, P_1^t reaches $P_2^1, \ldots, P_2^{k-t-2}$ and P_2^t reaches $P_1^1, \ldots, P_1^{k-t-2}$. By comparing with (E_{10}) and (E_{11}), $N_{H^k}(P_1^t) = N_G(P_1^t)$ and $N_{H^k}(P_2^t) = N_G(P_2^t)$ $\forall t$.

We have checked that the edge set of H^k is equal to the edge set of G.

Now we will show that H is a bipartite graph. We do this by showing a 2-coloring of H. The vertices get color 1 consist of $D_j, D_j^2, \ldots, D_j^{2\lfloor \frac{k}{2} \rfloor}, X,\ P_1^1, \ldots, P_1^{2\lfloor \frac{k-3}{2} \rfloor + 1}$, $P_2^1, \ldots, P_2^{2\lfloor \frac{k-3}{2} \rfloor + 1}$. The other vertices of H get color 2. It is easy to check that vertices in the same color class are not adjacent. This completes the proof of Lemma 4.1.1. □

For the example instance, the k-th root graph H corresponds to the solution S_1, S_2 is shown in Figure 4.3.

Now we show that if G has a k-th root H (*not necessarily bipartite*), then there is a partition of S into two disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2. We first need:

Proposition 4.1.2 *If H is a k-th root of G, then, in H:*

(i) *For all i, j: D_j is only adjacent to U_i whenever $u_i \in d_j$;*

(ii) *D_j^k is only adjacent to D_j^{k-1}, D_j^1 is only adjacent to D_j and D_j^2, and D_j^t is only adjacent to D_j^{t-1} and D_j^{t+1}, $2 \le t \le k-1$;*

(iii) *If $k \ge 4$, P_ℓ^{k-2} is only adjacent to P_ℓ^{k-3}, and for all $2 \le t \le k-3$: P_ℓ^t is only adjacent to P_ℓ^{t-1} and P_ℓ^{t+1}, $\ell = 1, 2$.*

Proof. By the construction of G, we have for all j: $N_G(D_j^k) = \{D_j, D_j^1, \ldots, D_j^{k-1}\}$, $N_G(D_j^1) \subset N_G[D_j]$ and $N_G(D_j^t) \subset N_G[D_j^{t-1}]$ for all $2 \le t \le k-1$. Thus, (i) and (ii) follow immediately from Lemma 3.2.2.

For (iii), we only verify for $\ell = 1$; the case $\ell = 2$ is similar. Since $G = H^k$ and by the construction of G, and by (i), (ii), we have for all i, j:

$$d_H(D_j^{k-2}, P_1^h) > k \text{ for } 2 \le h \le k-2. \tag{4.1}$$

$$d_H(P_1^{k-2}, X) \ge k-3,\ d_H(P_1^{k-2}, U_i) \ge k-2. \tag{4.2}$$

$$d_H(P_1^{k-2}, D_j) \ge k-2,\ d_H(P_1^{k-2}, D_j^1) \ge k-2. \tag{4.3}$$

Since

$$N_G(P_1^{k-2}) = N_{H^k}(P_1^{k-2}) = \{P_1^1, \ldots, P_1^{k-3}, X, U_1, \ldots, U_n, D_1, \ldots, D_m, D_1^1, \ldots, D_m^1\}$$

and

$$N_{H^{k-3}}(P_1^{k-2}) = N_H^1(P_1^{k-2}) \cup N_H^2(P_1^{k-2}) \cup \ldots \cup N_H^{k-3}(P_1^{k-2}),$$

it follows from (4.2) and (4.3):

$$\{P_1^1, \ldots, P_1^{k-3}\} \subseteq N_{H^{k-3}}(P_1^{k-2}) \subseteq \{P_1^1, \ldots, P_1^{k-3}, X\}.$$

Case 1: $N_{H^{k-3}}(P_1^{k-2}) = \{P_1^1, \ldots, P_1^{k-3}\}$.
That is, $N_H^1(P_1^{k-2}) \cup N_H^2(P_1^{k-2}) \cup \ldots \cup N_H^{k-3}(P_1^{k-2}) = \{P_1^1, \ldots, P_1^{k-3}\}$. Note that $N_H^t(P_1^{k-2}) \neq \varnothing$ for $1 \leq t \leq k-3$: Otherwise, let u be a vertex in G such that $d_H(u, P_1^{k-2}) > k$, then there is no path between u and P_1^{k-2} in H and thus there is also no path between u and P_1^{k-2} in G which contradicts to the fact that G is connected. Therefore, $\left|N_H^t(P_1^{k-2})\right| = 1$ for $1 \leq t \leq k-3$. Thus, $P_1^1, \ldots, P_1^{k-3}$ form a path of length $k-4$ in H, say P_A. Also, we have the following claims.

Claim 1: P_1^1 must be the end-vertex of P_A, and P_1^1 must be adjacent to U_i for some $1 \leq i \leq n$.
Proof of Claim 1: Otherwise, let P_1^t be end-vertex of P_A, for $t \neq 1$. Then, by $N_{H^{k-3}}(P_1^{k-2}) = \{P_1^1, \ldots, P_1^{k-3}\}$, hence in H, $P_1^1 \not\leftrightarrow \{X, U_1, \ldots, U_n, D_j, D_j^1, \ldots, D_j^k\}$, therefore $d_H(P_1^t, D_j^{k-2}) \leq d_H(P_1^1, D_j^{k-2}) \leq k$ (as in G, $P_1^1 \leftrightarrow D_j^{k-2}$), which contradicts to (4.1).
It is clear that P_1^1 must be adjacent to U_i for some $1 \leq i \leq n$: Otherwise, by in H, $P_1^1 \not\leftrightarrow \{D_j, D_j^1, \ldots, D_j^k\}$ and $X \not\leftrightarrow \{D_j, D_j^1, \ldots, D_j^k\}$, hence $P_1^1 \not\leftrightarrow D_j^{k-2}$ in G, contradicting to the construction of G.

Claim 2: P_1^{k-3} must be the end-vertex of P_A, and P_1^{k-3} must be adjacent to P_1^{k-2}.
Proof of Claim 2: Otherwise, let $P_1^{t'}$ be end-vertex of P_A, for $t' \neq k-3$. Then, by the Claim 1, $d_H(P_1^{k-3}, D_j^3) \leq d_H(P_1^{k-3}, P_1^1) + d_H(P_1^1, D_j^3) \leq (k-5)+5$, therefore, $P_1^{k-3} \leftrightarrow D_j^3$ in G, contradicting to the construction of G.
It is clear that P_1^{k-3} must be adjacent to P_1^{k-2}: Otherwise, then $d_H(P_1^{k-2}, P_1^1) \leq k-4$, hence $d_H(P_1^{k-2}, D_j^2) \leq k$, which contradicts the fact that, in G, P_1^{k-2} is non-adjacent to D_j^2.

Claim 3: P_1^t is only adjacent to P_1^{t-1} and to P_1^{t+1} for all $2 \leq t \leq k-3$.
Proof of Claim 3: By the Claim 2, $N_H^1(P_1^{k-2}) = \{P_1^{k-3}\}$ and $N_H^{k-3}(P_1^{k-2}) = \{P_1^1\}$. Next, P_1^{k-3} must be adjacent to P_1^{k-4}: Otherwise, $d_H(P_1^{k-3}, P_1^1) \leq k-5$, hence $d_H(P_1^{k-3}, D_j^3) \leq (k-5) + d_H(P_1^1, D_j^3) \leq k$, contradicting the fact that, in G, P_1^{k-3} is non-adjacent to D_j^3.
By the same argument, we can show that P_1^{k-4} must be adjacent to P_1^{k-5}, …, P_1^2 must be adjacent to P_1^1, i.e., for all $2 \leq t \leq k-3$, P_1^t is only adjacent to P_1^{t-1} and to P_1^{t+1}.

Case 2: $N_{H^{k-3}}(P_1^{k-2}) = \{P_1^1, \ldots, P_1^{k-3}, X\}$.
Then by (4.2), $d_H(P_1^{k-2}, X) = k-3$, hence $X \in \mathcal{B} := N_H^{k-3}(P_1^{k-2})$. Set $\mathcal{A} := \bigcup_{1\le t\le k-4} N_H^t(P_1^{k-2})$, and note that $N_H^t(P_1^{k-2}) \neq \varnothing$ for $1 \le t \le k-3$, hence $|\mathcal{A}| \ge k-4$, and $1 \le |\mathcal{B}| \le 2$.

Subcase 2.1: $\mathcal{B} = \{X, P_1^{t_0}\}$ for some $1 \le t_0 \le k-3$. In this case, $\mathcal{A}$ contains the $k-4$ vertices P_1^t, for $1 \le t \le k-3, t \neq t_0$. Similar to Case 1, it can be shown that $P_1^1, \ldots, P_1^{k-2}$ form a path of length $k-3$ in H with end-vertices P_1^1 and P_1^{k-2}.

Subcase 2.2: $\mathcal{B} = \{X\}$. That is, $N_H^1(P_1^{k-2}) \cup \ldots \cup N_H^{k-4}(P_1^{k-2}) = \{P_1^1, \ldots, P_1^{k-3}\}$. Note that, $N_H^t(P_1^{k-2}) \neq \varnothing$ for $1 \le t \le k-3$, therefore there is uniquely a set $N_H^{t_0}(P_1^{k-2})$ for some $1 \le t_0 \le k-4$ such that $N_H^{t_0}(P_1^{k-2}) = \{P_1^{t_a}, P_1^{t_b}\}$ for some $1 \le t_a, t_b \le k-3$. Let $\mathcal{C} := \bigcup_{1\le t\le k-4, t\neq t_0} N_H^t(P_1^{k-2})$ containing $k-5$ vertices exactly of $\{P_1^t \mid 1 \le t \le k-3, t \notin \{t_a, t_b\}\}$. By $N_H^t(P_1^{k-2}) \neq \varnothing$ for $1 \le t \le k-3$, hence $\left|N_H^t(P_1^{k-2})\right| = 1$ for $1 \le t \le k-3$ and $t \notin \{t_a, t_b\}$. Therefore, P_1^t for all $1 \le t \le k-3$ and $t \notin \{t_a, t_b\}$, form a path of length $k-6$ in H. Since $P_1^{t_a}, P_1^{t_b} \in N_H^{t_0}(P_1^{k-2})$ for some $1 \le t_0 \le k-4$, hence $d_H(P_1^1, P_1^{k-2}) \le k-4$. Therefore, $d_H(P_1^{k-2}, D_j^2) \le d_H(P_1^{k-2}, P_1^1) + d_H(P_1^1, D_j^2) \le (k-4)+4$, this contradicts to the fact that, in G, P_1^{k-2} is non-adjacent to D_j^2.
Thus, Subcase 2.2 cannot occur. □

Now we are ready to prove the reverse direction.

Lemma 4.1.3 *If H is a k-th root of G, then there exists a partition of S into two disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2.*

Proof. From Proposition 4.1.2 and the fact that, in G, $P_1^1 \not\leftrightarrow P_2^{k-2}$, $P_2^1 \not\leftrightarrow P_1^{k-2}$, it follows that, in H, P_1^1 are not adjacent to $P_2^1, \ldots, P_2^{k-2}$ and P_2^1 are not adjacent to $P_1^1, \ldots, P_1^{k-2}$. Also, since, in G, $P_1^1 \not\leftrightarrow D_j^{k-1}$ and $P_2^1 \not\leftrightarrow D_j^{k-1}$, it follows that, in H, P_1^1 and P_2^1 are not adjacent to the tail vertices and the subset vertices. Thus, $S_1 = N_H(P_1^1) \setminus \{X, P_1^2, \ldots, P_1^{k-2}\}$ and $S_2 = N_H(P_2^1) \setminus \{X, P_2^2, \ldots, P_2^{k-2}\}$ consist of element vertices *only*. We will show that S_1 and S_2 define a desired partition of the element set S.

Claim 1: $S_1 \cap S_2 = \varnothing$.
Proof of Claim 1: In case $k = 3$, by the construction of G, P_1^1 is non-adjacent to P_2^1 in G, therefore P_1^1 and P_2^1 have no common element neighbor in H, thus $S_1 \cap S_2 = \varnothing$.
Let $k \ge 4$, by Proposition 4.1.2 (iii), $\{P_\ell^1, \ldots, P_\ell^{k-2}\}$ form an induced path in H of length $k-3$, $\ell = 1, 2$. Thus, if P_1^1 and P_2^1 have a common neighbor in H then P_2^{k-2} reaches P_1^1 in $k-3+1+1$ steps, which contradicts to the fact that, in G, P_1^1 is non-adjacent to P_2^{k-2}.

Claim 2: For all j, $S_1 \cap d_j \neq \varnothing$ and $S_2 \cap d_j \neq \varnothing$.

Proof of Claim 2: Since the partition vertices P_1^1 and P_2^1 are adjacent to all subset vertices D_j^{k-2} in G, and by Proposition 4.1.2, D^{k-2} reaches D_j in exactly $k-2$ steps, P_1^1 and P_2^1 must reach D_j in *exactly* two steps and thus each of P_1^1, P_2^1 must have a common neighbor with D_j in the element set for all j, i.e., $N_H(P_1^1) \cap N_H(D_j) \neq \varnothing$ and $N_H(P_2^1) \cap N_H(D_j) \neq \varnothing$ for all $1 \leq j \leq m$. That means, each subset in D intersects both S_1 and S_2.

By Claim 1 and Claim 2, S_1 and S_2 are actually the desired partition of S. □

Notice that in the Lemma 4.1.3, we did not use the property that H is a bipartite graph. In fact, *any* k-th root of G would tell us how to do SET SPLITTING. In particular, *any bipartite* k-th root H of G will do. Hence, by Lemmas 4.1.1 and 4.1.3, we conclude

Theorem 4.1.4 *For any fixed $k \geq 3$, k-TH POWER OF BIPARTITE GRAPH is NP-complete.*

By the same reason, k-TH POWER OF $\mathcal{C}$-GRAPH is NP-complete for all fixed $k \geq 3$ whenever $\mathcal{C}$ contains all bipartite graphs (such as triangle-free graphs, parity graphs, perfect graphs, etc.). In particular, applied for the class of all graphs, this observation and the NP-completeness of SQUARE OF GRAPH [55] together give

Theorem 4.1.5 *For all fixed $k \geq 2$, k-TH POWER OF GRAPH is NP-complete.*

4.2 Powers of chordal graphs

Lau and Corneil [46] shown that SQUARE OF CHORDAL GRAPH is NP-complete. In this section we extend that result by showing that k-TH POWER OF CHORDAL GRAPH is NP-complete for all fixed $k \geq 3$. We will reduce SET SPLITTING to it as follows.

Let $S = \{u_1, \ldots, u_n\}$, $D = \{d_1, \ldots, d_m\}$ where $d_j \subseteq S$, $1 \leq j \leq m$, be an instance of SET SPLITTING, and let $k \geq 3$ be a fixed integer. We construct an instance $G = G(D, S)$ for k-TH POWER OF CHORDAL GRAPH as follows.

The vertex set of G consists of:

- U_i, $1 \leq i \leq n$. Each 'element vertex' U_i corresponds to the element u_i in S.
- D_j, $1 \leq j \leq m$. Each 'subset vertex' D_j corresponds to the subset d_j in D.
- $D_j^1, \ldots, D_j^k$, $1 \leq j \leq m$. k 'tail vertices' $D_j^1, \ldots, D_j^k$ of the subset vertex D_j.
- $P_1^1, \ldots, P_1^{k-1}$ and $P_2^1, \ldots, P_2^{k-1}$ are $k-1$ pairs of 'partition vertices'.

The edge set of G consists of:

- Edges of tail vertices:

  (T_1) $D_j, D_j^1, \ldots, D_j^k$ form a clique.

  (T_2) For all $1 \leq t \leq k-1$, $D_j^t \leftrightarrow \{U_i \mid u_i \in d_j, 1 \leq i \leq n\}$.

  (T_3) For all $1 \leq t \leq k-2$, $D_j^t \leftrightarrow \{U_i \mid 1 \leq i \leq n\}$, $D_j^t \leftrightarrow \{D_{j'} \mid d_j \cap d_{j'} \neq \varnothing\}$, $D_j^t \leftrightarrow \{P_1^h \mid 1 \leq h \leq k-t-1\}$ and $D_j^t \leftrightarrow \{P_2^h \mid 1 \leq h \leq k-t-1\}$.

  (T_4) For all $1 \leq t \leq k-3$, $D_j^t \leftrightarrow \{D_{j'} \mid 1 \leq j' \leq m\}$; $D_j^t \leftrightarrow \{D_{j'}^h \mid 1 \leq h \leq k-t-2, d_j \cap d_{j'} \neq \varnothing\}$.

  (T_5) For all $1 \leq t \leq k-4$, $D_j^t \leftrightarrow \{D_{j'}^h \mid 1 \leq h \leq k-t-3, 1 \leq j' \leq m\}$.

- Edges of subset vertices:

  (T_6) $D_j \leftrightarrow \{P_1^1, \ldots, P_1^{k-1}, P_2^1, \ldots, P_2^{k-1}\}$, $D_j \leftrightarrow \{U_i \mid 1 \leq i \leq n\}$, and $D_j \leftrightarrow \{D_{j'} \mid 1 \leq j' \leq m\}$.

- Edges of element vertices:

  (T_7) $U_1, \ldots, U_n$ form a clique, and $U_i \leftrightarrow \{P_1^1, \ldots, P_1^{k-1}, P_2^1, \ldots, P_2^{k-1}\}$.

- Edges of partition vertices:

  (T_8) $P_1^1, \ldots, P_1^{k-1}, U_1, \ldots, U_n$ form a clique,
  $P_2^1, \ldots, P_2^{k-1}, U_1, \ldots, U_n$ form a clique.

  (T_9) For all $1 \leq t \leq k-2$, $P_1^t \leftrightarrow \{P_2^h \mid 1 \leq h \leq k-t-1\}$ and $P_2^t \leftrightarrow \{P_1^h \mid 1 \leq h \leq k-t-1\}$.

Clearly, G can be constructed from D, S in polynomial time. For an illustration, in case $k = 3$, the example instance yields the graph G is depicted in Figure 4.4. While the corresponding cube root graph H to the solution S_1, S_2 is shown in Figure 4.5.

Lemma 4.2.1 *If there exists a partition of S into two disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2, then there exists a chordal graph H such that $G = H^k$.*

Proof. Let H have the same vertex set as G. The edges of H are as follows; see also Figures 4.6.

- Edges of subset vertices and their tail vertices: For all $2 \leq t \leq k$, $D_j^t \leftrightarrow D_j^{t-1}$ and $D_j^1 \leftrightarrow D_j$, and $D_j \leftrightarrow \{U_i \mid u_i \in d_j, 1 \leq i \leq n\}$.

- Edges of element vertices: $U_1, \ldots, U_n$ form a clique.

- Edges of partition vertices:
  $P_1^1 \leftrightarrow \{U_i \mid u_i \in S_1, 1 \leq i \leq n\}$, and $P_2^1 \leftrightarrow \{U_i \mid u_i \in S_2, 1 \leq i \leq n\}$, and for all $2 \leq t \leq k-1$, $P_1^t \leftrightarrow P_1^{t-1}$ and $P_2^t \leftrightarrow P_2^{t-1}$.

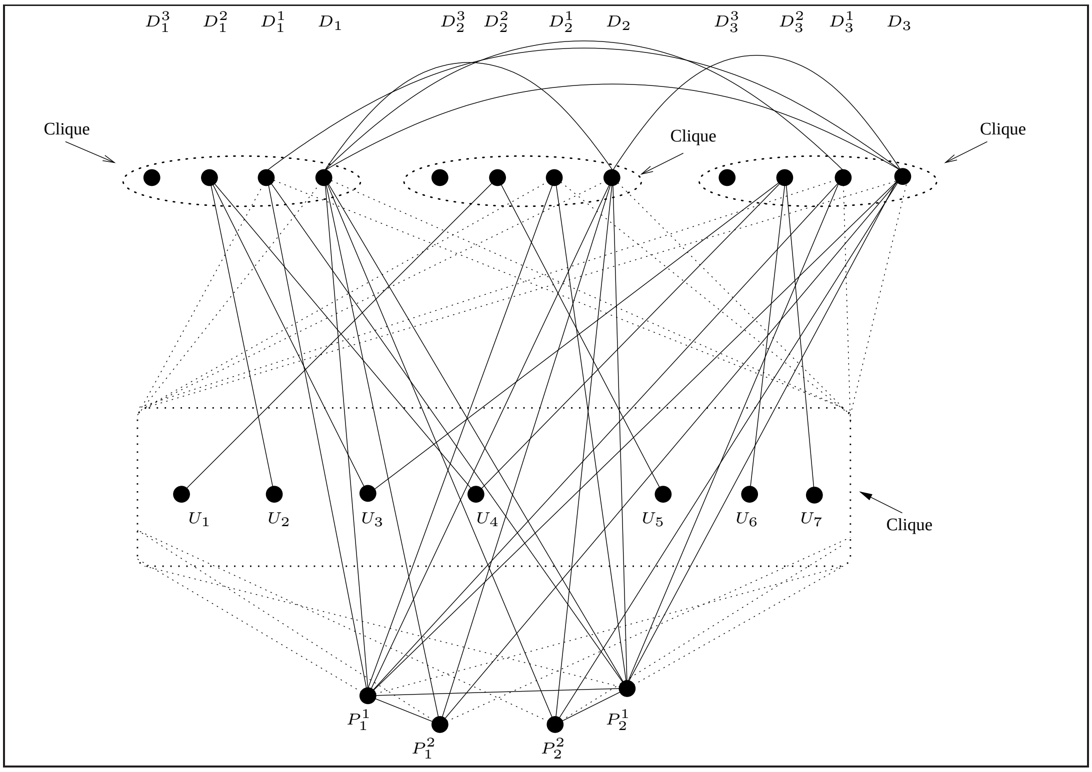

Figure 4.4: The graph G for the example instance of SET SPLITTING and $k = 3$

Similar to the proof of Lemma 4.1.1 we can show that $H^k = G$ by following the order of the presentation of the edge set of G; cf. (T_1) – (T_9), as follows.

For D_j^k, it is clear that $D_j, D_j^1, \ldots, D_j^k$ form a clique in H^k for all j, hence $N_{H^k}(D_j^k) = N_G(D_j^k)$.

For $1 \leq t \leq k-1$, the vertex D_j^t reaches D_j in H in t steps. By the construction of H, $D_j \leftrightarrow U_i$ whenever $u_i \in d_j$. Thus within $t+1$ steps, D_j^t reaches U_i whenever $u_i \in d_j$, therefore $D_j^t \leftrightarrow U_i$ in H^k whenever $u_i \in d_j$. By comparing with (T_2), $N_{H^k}(D_j^t) = N_G(D_j^t)$.

For $1 \leq t \leq k-2$, the vertex D_j^t reaches D_j in t steps. Since $D_j \leftrightarrow U_i$ whenever $u_i \in d_j$ and $U_1, \ldots, U_n$ form a clique in H. Thus within $t+2$ steps, D_j^t reaches U_i for all i, therefore $D_j^t \leftrightarrow U_i$ in H^k for all i. Also, in H, D_j reaches $D_{j'}$ in two steps whenever $d_j \cap d_{j'} \neq \varnothing$, thus within $t+2 \leq k$ steps, D_j^t reaches $D_{j'}$ whenever $d_j \cap d_{j'} \neq \varnothing$. Furthermore, since we have a solution for SET SPLITTING, every D_j has a common neighbor to P_1^1 and a common neighbor to P_2^1. Moreover, in H, $\{P_1^1, \ldots, P_1^{k-1}\}$ and $\{P_2^1, \ldots, P_2^{k-1}\}$ are induced path of length $k-2$, therefore D_j^t reaches $P_1^1, \ldots, P_1^{k-t-1}$ and D_j^t reaches $P_2^1, \ldots, P_2^{k-t-1}$ within k steps. By comparing with (T_3), $N_{H^k}(D_j^t) = N_G(D_j^t)$.

For $1 \leq t \leq k-3$, the vertex D_j^t reaches D_j, in t steps. Moreover, D_j reaches

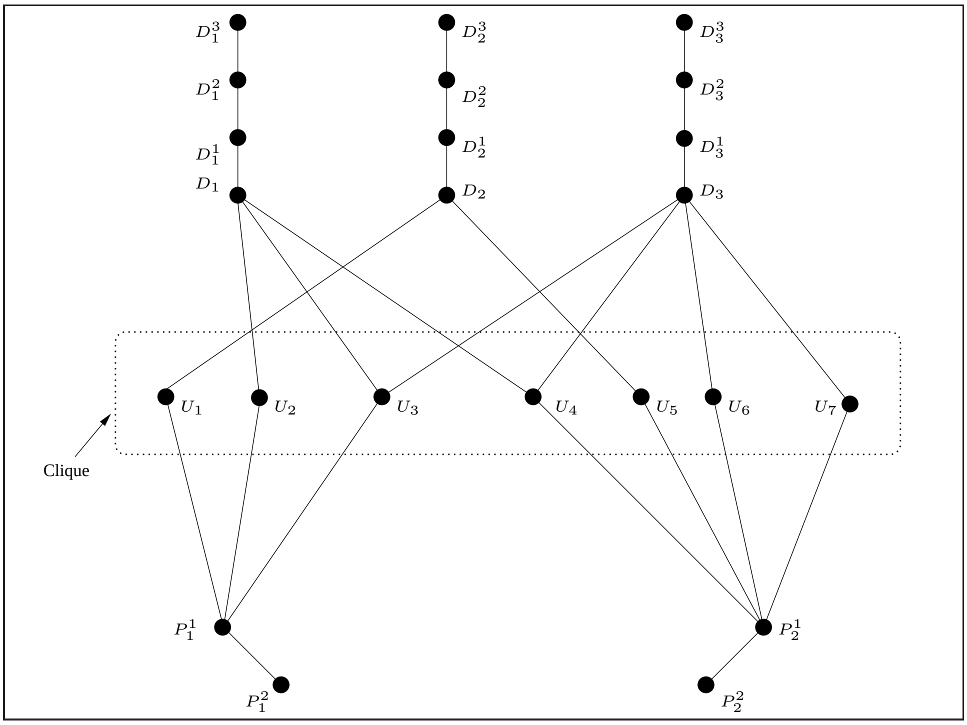

Figure 4.5: Cube root graph H to the solution S_1, S_2.

$D_{j'}$ within three steps for all j'. So within $t+3 \leq k$ steps, D_j^t reaches $D_{j'}$ for all j', therefore $D_j^t \leftrightarrow D_{j'}$ for all j'. Furthermore, as D_j reaches $D_{j'}$ in two steps whenever $d_j \cap d_{j'} \neq \varnothing$, thus within k steps, D_j^t reaches $D_{j'}^1, \ldots, D_{j'}^{k-t-2}$ whenever $d_j \cap d_{j'} \neq \varnothing$. By comparing with (T_4), $N_{H^k}(D_j^t) = N_G(D_j^t)$.

For $1 \leq t \leq k-4$, the vertex D_j^t reaches D_j in t steps. Also, D_j reaches $D_{j'}$ for all j' in three steps, therefore within k steps, D_j^t reaches $D_{j'}^1, \ldots, D_{j'}^{k-t-3}$ for all j'. By comparing with (T_5), $N_{H^k}(D_j^t) = N_G(D_j^t)$.

Next, the vertex D_j reaches U_i in one step whenever $u_i \in d_j$. In H, $U_1, \ldots, U_n$ form a clique, so within k steps, D_j reaches U_i for all i and D_j reaches $D_{j'}$ for all j'. Moreover, since we have a solution for SET SPLITTING, every D_j has a common neighbor to P_1^1 and a common neighbor to P_2^1. In H, $\{P_1^1, \ldots, P_1^{k-1}\}$ and $\{P_2^1, \ldots, P_2^{k-1}\}$ are induced path of length $k-2$. Thus, within k steps, D_j reaches $P_1^1, \ldots, P_1^{k-1}$, $P_2^1, \ldots, P_2^{k-1}$. By comparing it with (T_6), $N_{H^k}(D_j) = N_G(D_j)$.

For U_i, $U_1, \ldots, U_n$ form a clique in H, hence $U_1, \ldots, U_n$ form also a clique in H^k. In H, $U_i \leftrightarrow P_1^1$ whenever $u_i \in S_1$, and $U_i \leftrightarrow P_2^1$ whenever $u_i \in S_2$. Moreover, $P_1^1, \ldots, P_1^{k-1}$ form an induced path of length $k-2$ in H and $P_2^1, \ldots, P_2^{k-1}$ form an induced path of length $k-2$ in H. Thus, within k steps U_i reaches $P_1^1, \ldots, P_1^{k-1}, P_2^1, \ldots, P_2^{k-1}$.
By comparing with (T_7), $N_{H^k}(U_i) = N_G(U_i)$ for all i.

Finally, for partition vertices, it is clear that $P_1^1, \ldots, P_1^{k-1}, U_1, \ldots, U_n$ form a clique in H^k and $P_2^1, \ldots, P_2^{k-1}, U_1, \ldots, U_n$ form a clique in H^k. Moreover, for all $1 \leq t \leq k-2$, since P_1^1 and P_2^1 have at least one neighbor U_i in H. Also, since $U_1, \ldots, U_n$ form a clique in H, $P_1^t \leftrightarrow \{P_2^1, \ldots, P_2^{k-t-1}\}$ and $P_2^t \leftrightarrow \{P_1^1, \ldots, P_1^{k-t-1}\}$. By comparing with (T_8) and (T_9), $N_{H^k}(P_1^t) = N_G(P_1^t)$ and $N_{H^k}(P_2^t) = N_G(P_2^t)$ for all t.

We checked that the edge set of H^k is equal to the edge set of G.

Now to complete the proof we will show that H is a chordal graph. We do this by giving a simplicial elimination ordering of H.

First of all, all D_j^k are simplicial in H and we eliminate them first. Then for each vertex of $D_j^{k-1}, \ldots, D_j^1$, its neighbors in the remaining vertices of H form a clique. So we can eliminate $D_j^{k-1}, \ldots, D_j^1$ for all $1 \leq j \leq m$. By the same reason, each partition vertex of sets $\{P_1^{k-1}, \ldots, P_1^2\}$ and $\{P_2^{k-1}, \ldots, P_2^2\}$ is also simplicial and we eliminate them.

Next, since the element vertices $U_1, \ldots, U_n$ induce a complete subgraph. Moreover, D_j, P_1^1 and P_2^1 have only neighbors in the element set $\{U_1, \ldots, U_n\}$, therefore, D_j, P_1^1 and P_2^1 are also simplicial in the subgraphs induced by remaining vertices of H and we can eliminate them. Finally, the element vertices $U_1, \ldots, U_n$ are remaining vertices; they induce a complete subgraph and this completes the proof that H is chordal. □

Now we show that if G has a k-th root H (*not necessarily chordal*), then there is a partition of S into two disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2. Similar to the proof of Lemma 4.1.3, we first need:

Proposition 4.2.2 *If H is a k-th root of G, then, in H:*

(i) *For all i, j: D_j is only adjacent to U_i whenever $u_i \in d_j$;*

(ii) *D_j^k is only adjacent to D_j^{k-1}, D_j^1 is only adjacent to D_j and D_j^2, and D_j^t is only adjacent to D_j^{t-1} and D_j^{t+1} for $2 \leq t \leq k-1$;*

(iii) *P_ℓ^{k-1} is only adjacent to P_ℓ^{k-2}, and for all $2 \leq t \leq k-2$: P_ℓ^t is only adjacent to P_ℓ^{t-1} and P_ℓ^{t+1} for $\ell = 1, 2$.*

Proof. By the construction of G, we have for all j: $N_G(D_j^k) = \{D_j, D_j^1, \ldots, D_j^{k-1}\}$, $N_G(D_j^1) \subset N_G[D_j]$ and $N_G(D_j^t) \subset N_G[D_j^{t-1}]$ for all $2 \leq t \leq k-1$. Thus, (i) and (ii) follow immediately from Lemma 3.2.2.

For (iii), we only verify for $\ell = 1$; the case $\ell = 2$ is similar. Since $G = H^k$ and by the construction of G, and by (i), (ii), we have for all i, j:

$$d_H(D_j^{k-2}, P_1^t) > k \text{ for } 2 \leq t \leq k-1. \tag{4.4}$$

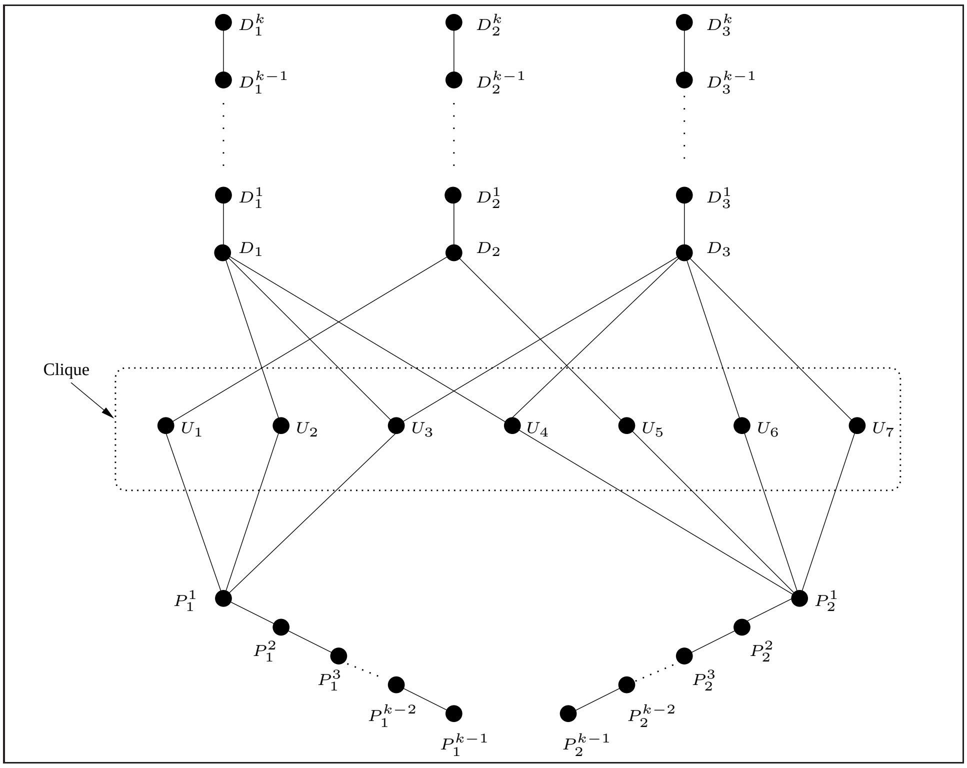

Figure 4.6: A chordal k-th root H in Lemma 4.2.1 to the example solution S_1, S_2

$$d_H(P_1^{k-1}, U_i) \geq k-1, \quad d_H(P_1^{k-1}, D_j) \geq k. \tag{4.5}$$

Since

$$N_G(P_1^{k-1}) = N_{H^k}(P_1^{k-1}) = \{P_1^1, \ldots, P_1^{k-2}, U_1, \ldots, U_n, D_1, \ldots, D_m\}$$

and

$$N_{H^{k-2}}(P_1^{k-1}) = N_H^1(P_1^{k-1}) \cup N_H^2(P_1^{k-1}) \cup \ldots \cup N_H^{k-2}(P_1^{k-1}),$$

it follows from (4.5):

$$N_H^1(P_1^{k-1}) \cup N_H^2(P_1^{k-1}) \cup \ldots \cup N_H^{k-2}(P_1^{k-1}) = \{P_1^1, \ldots, P_1^{k-2}\}.$$

Note that $N_H^t(P_1^{k-1}) \neq \varnothing$ for $1 \leq t \leq k-2$ (otherwise, let u be a vertex in G such that $d_H(u, P_1^{k-1}) > k$, then there is no path between u and P_1^{k-1} in H and thus there is also no path between u and P_1^{k-1} in G which contradicts to the fact that G is connected), hence $\left|N_H^t(P_1^{k-1})\right| = 1$ for $1 \leq t \leq k-2$. Therefore, $P_1^1, \ldots, P_1^{k-2}$ form a path of length $k-3$ in H, say P_B. Moreover, we have the following claims.

Claim 1: P_1^1 must be the end-vertex of P_B, also P_1^1 must be adjacent to U_i for some $1 \leq i \leq n$.

Proof of Claim 1: Otherwise, let P_1^t be end-vertex of P_B, for some $2 \leq t \leq k-2$. Then, by $N_{H^{k-2}}(P_1^{k-1}) = \{P_1^1, \ldots, P_1^{k-2}\}$, hence in H, $P_1^1 \not\leftrightarrow$

$\{U_1, \ldots, U_n, D_j, D_j^1, \ldots, D_j^k\}$, therefore $d_H(P_1^t, D_j^{k-2}) \leq d_H(P_1^1, D_j^{k-2}) \leq k$ (as in G, $P_1^1 \leftrightarrow D_j^{k-2}$), which contradicts (4.4).
Moreover, it is clear that P_1^1 must be adjacent to U_i for some $1 \leq i \leq n$: Otherwise, by in H, $P_1^1 \not\leftrightarrow \{D_j, D_j^1, \ldots, D_j^k\}$, hence $P_1^1 \not\leftrightarrow D_j^{k-2}$ in G, contradicting to the construction of G.

Claim 2: P_1^{k-2} must be the end-vertex of P_B, also P_1^{k-2} must be adjacent to P_1^{k-1}.
Proof of Claim 2: Otherwise, let $P_1^{t'}$ be end-vertex of P_B, for some $1 \leq t' \leq k-3$. Then, by the Claim 1, $d_H(P_1^{k-2}, D_j^2) \leq d_H(P_1^{k-2}, P_1^1) + d_H(P_1^1, D_j^2) \leq (k-4)+4$, therefore, $P_1^{k-2} \leftrightarrow D_j^2$ in G, contradicting to the construction of G; cf. (T_3).
Also, it is clear that P_1^{k-2} must be adjacent to P_1^{k-1}: Otherwise, then $d_H(P_1^{k-1}, P_1^1) \leq k-3$, hence $d_H(P_1^{k-1}, D_j^1) \leq k$, which contradicts the fact that, in G, P_1^{k-1} is non-adjacent to D_j^1.

Claim 3: P_1^t is only adjacent to P_1^{t-1} and to P_1^{t+1} for all $2 \leq t \leq k-2$.
Proof of Claim 3: By the Claim 2, $N_H^1(P_1^{k-1}) = \{P_1^{k-2}\}$ and $N_H^{k-2}(P_1^{k-1}) = \{P_1^1\}$. Next, P_1^{k-2} must be adjacent to P_1^{k-3}: Otherwise, $d_H(P_1^{k-2}, P_1^1) \leq k-4$, hence $d_H(P_1^{k-2}, D_j^2) \leq (k-4) + d_H(P_1^1, D_j^2) \leq k$, contradicting the fact that, in G, P_1^{k-2} is non-adjacent to D_j^2.
By the same argument, we can show that P_1^{k-3} must be adjacent to P_1^{k-4}, …, P_1^2 must be adjacent to P_1^1, therefore for all $2 \leq t \leq k-2$, P_1^t is only adjacent to P_1^{t-1} and to P_1^{t+1}. □

Now we are ready to prove the reverse direction.

Lemma 4.2.3 *If H is a k-th root of G, then there exists a partition of S into two disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2.*

Proof. From Proposition 4.2.2 and the fact that, in G, $P_1^1 \not\leftrightarrow P_2^{k-1}$, $P_2^1 \not\leftrightarrow P_1^{k-1}$, it follows that, in H, P_1^1 is not adjacent to $P_2^1, \ldots, P_2^{k-1}$ and P_2^1 is not adjacent to $P_1^1, \ldots, P_1^{k-1}$. Also, since, in G, $P_1^1 \not\leftrightarrow D_j^{k-1}$ and $P_2^1 \not\leftrightarrow D_j^{k-1}$, it follows that, in H, P_1^1 and P_2^1 are not adjacent to the tail vertices and the subset vertices. Thus, $S_1 = N_H(P_1^1) \setminus \{P_1^2, \ldots, P_1^{k-1}\}$ and $S_2 = N_H(P_2^1) \setminus \{P_2^2, \ldots, P_2^{k-1}\}$ consist of element vertices *only*. We will show that S_1 and S_2 define a desired partition of the element set S.

Claim 1: $S_1 \cap S_2 = \varnothing$.
Proof of Claim 1: By Proposition 4.2.2 (iii), $\{P_\ell^1, \ldots, P_\ell^{k-1}\}$ form an induced path in H of length $k-2$, $\ell = 1, 2$. Thus, if P_1^1 and P_2^1 have a common neighbor in H then P_1^1 reaches P_2^{k-1} in $k-2+1+1$ steps, this contradicts to the fact that, in G, P_1^1 is non-adjacent to P_2^{k-1}.

Claim 2: For all j, $S_1 \cap d_j \neq \varnothing$ and $S_2 \cap d_j \neq \varnothing$.

Proof of Claim 2: Since the partition vertices P_1^1 and P_2^1 are adjacent to all subset vertices D_j^{k-2} in G, and by Proposition 4.2.2 (i) and (ii), D^{k-2} reaches D_j in exactly $k-2$ steps, therefore P_1^1 and P_2^1 must reach D_j in *exactly* two steps and thus each of P_1^1, P_2^1 must have a common neighbor with D_j in the *element set* for all j, i.e., $N_H(P_1^1) \cap N_H(D_j) \neq \varnothing$ and $N_H(P_2^1) \cap N_H(D_j) \neq \varnothing$ for all $1 \leq j \leq m$. That means, the each subset in D intersects both S_1 and S_2.

By Claim 1 and Claim 2, S_1 and S_2 are actually the desired partition of S. □

Notice that, as in the case of bipartite roots, in Lemma 4.2.3, we did not use the property that H is a chordal graph. In fact, *any* k-th root of G would tell us how to do SET SPLITTING. In particular, *any chordal* k-th root H of G will do. Hence, from Lemmas 4.2.1 and 4.2.3 we conclude

Theorem 4.2.4 *For any fixed $k \geq 3$, k-TH POWER OF CHORDAL GRAPH is NP-complete.*

This result and the NP-completeness of SQUARE OF CHORDAL GRAPH [46] together give

Theorem 4.2.5 *For any fixed $k \geq 2$, k-TH POWER OF CHORDAL GRAPH is NP-complete.*

By the same reason, k-TH POWER OF $\mathcal{C}$ GRAPH is NP-complete for all fixed $k \geq 2$ whenever $\mathcal{C}$ contains all chordal graphs, such as weakly chordal graphs, HHD-free graphs (graphs which contain no induced house, hole, domino), hole-free graphs, perfect graphs, etc.

Furthermore, in case $\mathcal{C}$ is the class of all graphs, we obtain another proof for Theorem 4.1.5.

4.3 Finding k-th roots of chordal graphs

In this section, we will show that given a chordal graph G, it is NP-complete to determine if there exists a graph H such that $H^k = G$ for all fixed $k \geq 2$. In particular, we will show that the following problem is NP-complete.

k-TH ROOT OF CHORDAL GRAPH
Instance: A chordal graph G.
Question: Is there a (*perfect*) graph H such that $H^k = G$?

Notice that in case $k = 2$, it is NP-complete by the result of Lau and Corneil [46]. Thus, to complete for proving the completeness of k-TH ROOT OF CHORDAL GRAPH we only need to show that recognizing k-th roots of chordal graphs is NP-complete for all $k \geq 3$. We will reduce SET SPLITTING to it by distinguishing two cases of k as follows.

CASE 1: k **is odd**: We will show that the graph $G = G(D, S)$ constructed in the reduction proving the NP-completeness of k-TH POWER OF CHORDAL GRAPH (see Section 4.2 on page 26) is chordal, and thus, the k-TH ROOT OF CHORDAL GRAPH *is solved for all fixed odd* k.

Lemma 4.3.1 *If k is odd, then the graph $G = G(D, S)$ constructed in the reduction to prove the NP-completeness of k-*TH POWER OF CHORDAL GRAPH*(cf. Section 4.2) is chordal.*

Proof. We do this by showing a simplicial elimination ordering of G.

In case $k = 3$, by the construction of G, cf. (T_1)–(T_9), we can easily verify that

$$\{P_1^2, P_2^2, D_1^3, \ldots, D_m^3, D_1^2, \ldots, D_m^2, D_1^1, \ldots, D_m^1, D_1, \ldots, D_m, U_1, \ldots, U_n, P_1^1, P_2^1\}$$

is a possible perfect elimination ordering in G. Thus, G is chordal.

Let $k \geq 5$. By the construction of G, cf. (T_1)–(T_9), it is clear that

$$G_1 := G[D_1, \ldots, D_1^{\frac{k-3}{2}}, \ldots, D_m, \ldots, D_m^{\frac{k-3}{2}}, U_1, \ldots, U_n, P_1^1, \ldots, P_1^{\frac{k-1}{2}}, P_2^1, \ldots, P_2^{\frac{k-1}{2}}]$$

is a complete subgraph of G.

First we will determine a simplicial elimination ordering of vertices $P_1^{k-1}, \ldots, P_1^{\frac{k+1}{2}}$.

For P_1^{k-1}, its neighborhood is only $\{P_1^1, \ldots, P_1^{k-2}, U_1, \ldots, U_n, D_1, \ldots, D_m\}$ which is clique, therefore P_1^{k-1} is simplicial and we can eliminate it first.

For all P_1^t for $\frac{k+1}{2} \leq t \leq k-2$, by the construction of G, their neighborhoods are only

$$\begin{aligned} N_G(P_1^t) \ := \ & \{P_1^1, \ldots, P_1^{t-1}, U_1, \ldots, U_n, D_1, \ldots, D_m\} \\ & \cup \{D_1^1, \ldots, D_m^1, \ldots, D_1^{k-t-1}, \ldots, D_m^{k-t-1}\} \cup \{P_2^1, \ldots, P_2^{k-t-1}\} \end{aligned}$$

Note that in this case, by $\frac{k+1}{2} \leq t \leq k-2$, D_j^{k-t-1} is adjacent to $D_{j'}^{k-t-1}$, i.e., $\{D_1^1, \ldots, D_m^1, \ldots, D_1^{k-t-1}, \ldots, D_m^{k-t-1}\}$ is a clique, hence $N_G(P_1^t)$ induces a complete graph, thus P_1^t are simplicial and we can eliminate them.

Then by a similar argument $P_2^{k-1}, \ldots, P_2^{\frac{k+1}{2}}$ are simplicial and we can eliminate them.

Next, for all D_j^k, their neighbor sets are only $D_j, D_j^1 \ldots, D_j^{k-1}$ which are cliques, therefore they are simplicial and we can eliminate them.

For all D_j^{k-1}, by (T_2), their neighborhoods are only

$$\{D_j, D_j^1 \ldots, D_j^{k-2}\} \cup \{U_i \mid u_i \in d_j, 1 \leq i \leq n\}$$

which are cliques, therefore they are simplicial and we can eliminate them.

For all D_j^{k-2}, by (T_3), their neighborhoods are only

$$\{D_j, D_j^1 \ldots, D_j^{k-3}\} \cup \{U_1, \ldots, U_n\} \cup \{D_{j'} \mid d_j \cap d_{j'} \neq \varnothing\} \cup \{P_1^1, P_2^1\}$$

which are cliques, therefore they are simplicial and we can eliminate them.

For all D_j^{k-3}, by (T_3) and (T_4), their neighborhoods are only

$$\begin{aligned}&\{D_j, D_j^1 \ldots, D_j^{k-4}\} \cup \{U_1, \ldots, U_n, D_1, \ldots, D_m\}\\ &\cup\{D_{j'}^1 \mid d_j \cap d_{j'} \neq \varnothing\} \cup \{P_1^1, P_1^2, P_2^1, P_2^2\}\end{aligned}$$

which are cliques, therefore they are simplicial and we can eliminate them.

For all D_j^t for $\frac{k-1}{2} \leq t \leq k-4$, by (T_3) – (T_5), their neighborhoods are only

$$\begin{aligned}N_G(D_j^t) \;:=\; &\{D_j, D_j^1 \ldots, D_j^{t-1}\} \cup \{U_1, \ldots, U_n, D_1, \ldots, D_m\}\\ &\cup\{D_{j'}^{k-t-2} \mid d_j \cap d_{j'} \neq \varnothing\} \cup \{D_1^1, \ldots, D_m^1, \ldots, D_1^{k-t-3}, \ldots, D_m^{k-t-3}\}\\ &\cup\{P_1^1, \ldots, P_1^{k-t-1}, P_2^1, \ldots, P_2^{k-t-1}\}\end{aligned}$$

Note that in this case, by $\frac{k-1}{2} \leq t \leq k-4$, P_1^{k-t-1} is adjacent to P_2^{k-t-1}, hence $N_G(D_j^t)$ induces a complete graph, thus D_j^t are simplicial and we can eliminate them.

Next, for each vertex of $D_j^{\frac{k-1}{2}}, \ldots, D_j^1, D_j$, its neighbors in the remaining vertices of the G are only a subset of complete graph G_1. Thus, they are simplicial and we can eliminate them.

Finally, remaining vertices $U_1, \ldots, U_n, P_1^1, \ldots, P_1^{\frac{k-1}{2}}, P_2^1, \ldots, P_2^{\frac{k-1}{2}}$ induce a complete graph and this completes the proof of Lemma 4.3.1. □

Case 2: k **is even**: Let even $k \geq 4$, we will reduce SET SPLITTING to k-TH ROOT OF CHORDAL GRAPH as follows.

Let $S = \{u_1, \ldots, u_n\}$, $D = \{d_1, \ldots, d_m\}$ where $d_j \subseteq S$, $1 \leq j \leq m$, be an instance of SET SPLITTING, and let $k \geq 4$ be a fixed even integer. We construct an instance $G = G(D, S)$ for k-TH ROOT OF CHORDAL GRAPH as follows.

The vertex set of G consists of:

- U_i, $1 \leq i \leq n$. Each 'element vertex' U_i corresponds to the element u_i in S.
- D_j, $1 \leq j \leq m$. Each 'subset vertex' D_j corresponds to the subset d_j in D.
- $D_j^1, \ldots, D_j^k$, $1 \leq j \leq m$. k 'tail vertices' $D_j^1, \ldots, D_j^k$ of the subset vertex D_j.
- $P_1^1, \ldots, P_1^{k-1}$ and $P_2^1, \ldots, P_2^{k-1}$ are $k-1$ pairs of 'partition vertices'.

The edge set of G consists of:

- Edges of tail vertices:

  (R_1) $D_j, D_j^1, \ldots, D_j^k$ form a clique.

  (R_2) For all $1 \leq t \leq k-1$, $D_j^t \leftrightarrow \{U_i \mid u_i \in d_j, 1 \leq i \leq n\}$;
  $D_j^t \leftrightarrow \{D_{j'} \mid 1 \leq j' \leq m\}$.

  (R_3) For all $1 \leq t \leq k-2$, $D_j^t \leftrightarrow \{U_i \mid 1 \leq i \leq n\}$,
  $D_j^t \leftrightarrow \{D_{j'}^h \mid 1 \leq h \leq k-t-1\}$,
  $D_j^t \leftrightarrow \{P_1^h \mid 1 \leq h \leq k-t-1\}$ and $D_j^t \leftrightarrow \{P_2^h \mid 1 \leq h \leq k-t-1\}$.

- Edges of subset vertices:

  (R_4) $D_j \leftrightarrow \{P_1^1, \ldots, P_1^{k-1}, P_2^1, \ldots, P_2^{k-1}\}$, $D_j \leftrightarrow \{U_i \mid 1 \leq i \leq n\}$, and
  $D_j \leftrightarrow \{D_{j'} \mid 1 \leq j' \leq m\}$.

- Edges of element vertices:

  (R_5) $U_1, \ldots, U_n$ form a clique, and $U_i \leftrightarrow \{P_1^1, \ldots, P_1^{k-1}, P_2^1, \ldots, P_2^{k-1}\}$.

- Edges of partition vertices:

  (R_6) $P_1^1, \ldots, P_1^{k-1}, U_1, \ldots, U_n$ form a clique;
  $P_2^1, \ldots, P_2^{k-1}, U_1, \ldots, U_n$ form a clique.

  (R_7) For all $1 \leq t \leq k-2$, $P_1^t \leftrightarrow \{P_2^h \mid 1 \leq h \leq k-t-1\}$ and
  $P_2^t \leftrightarrow \{P_1^h \mid 1 \leq h \leq k-t-1\}$.

Clearly, G can be constructed from D, S in polynomial time. For an illustration, in case $k = 4$, the example instance yields the graph G is depicted in Figure 4.7. While the corresponding fourth root graph H to the solution S_1, S_2 is shown in Figure 4.8.

Lemma 4.3.2 *The graph G is chordal.*

Proof. We do this by giving a simplicial elimination ordering of G as follows.
By the construction of G; cf. (R_1) – (R_7), it is clear that

$$G_2 := G[D_1, \ldots, D_1^{\frac{k-2}{2}}, \ldots, D_m, \ldots, D_m^{\frac{k-2}{2}}, U_1, \ldots, U_n, P_1^1, \ldots, P_1^{\frac{k-2}{2}}, P_2^1, \ldots, P_2^{\frac{k-2}{2}}]$$

is a complete subgraph of G.

We first will determine a simplicial elimination ordering of vertices $P_1^{k-1}, \ldots, P_1^{\frac{k}{2}}$.

For P_1^{k-1}, its neighborhood is only $\{P_1^1, \ldots, P_1^{k-2}, U_1, \ldots, U_n, D_1, \ldots, D_m\}$ which is clique, therefore P_1^{k-1} is simplicial and we can eliminate it first.

For all P_1^t for $\frac{k}{2} \leq t \leq k-2$, by the construction of G, their neighborhoods are only

$$\begin{aligned} N_G(P_1^t) \ := \ & \{P_1^1, \ldots, P_1^{t-1}, U_1, \ldots, U_n, D_1, \ldots, D_m\} \\ & \cup \{D_1^1, \ldots, D_1^{k-t-1}, \ldots, D_m^1, \ldots, D_m^{k-t-1}\} \cup \{P_2^1, \ldots, P_2^{k-t-1}\} \end{aligned}$$

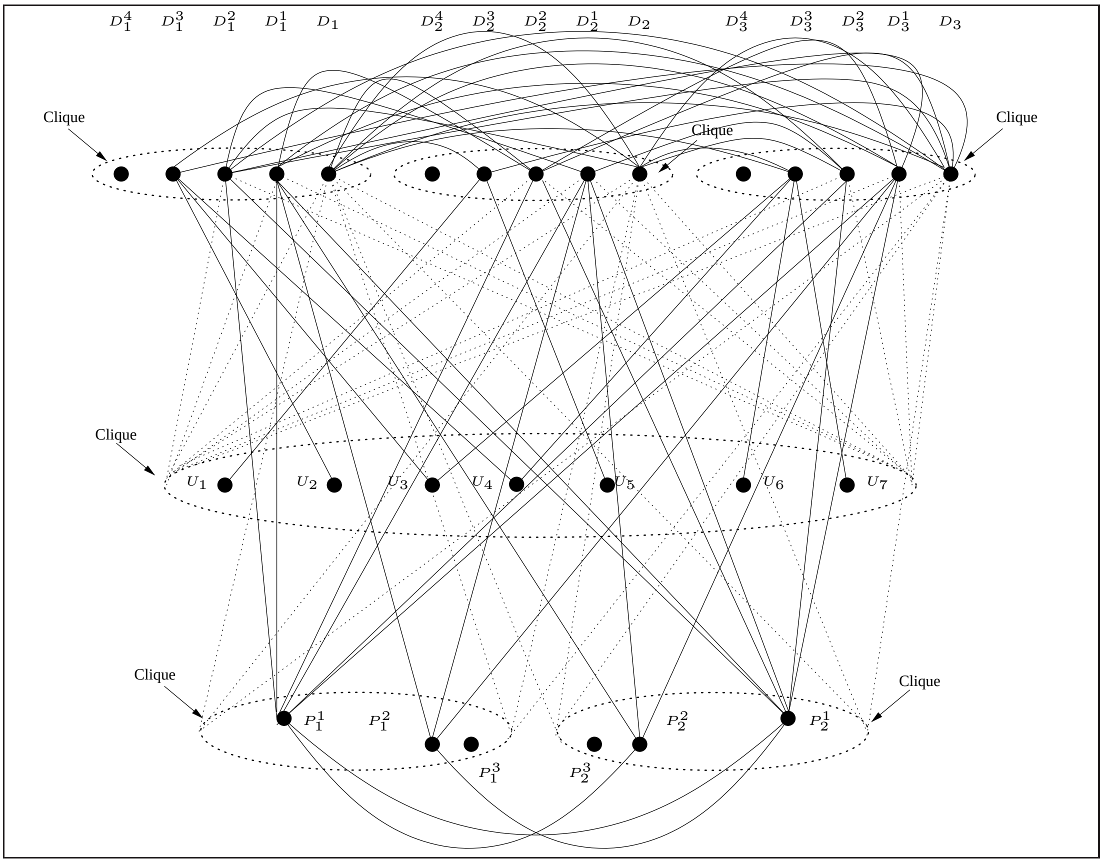

Figure 4.7: The chordal graph G for the example instance of SET SPLITTING and $k = 4$

Note that in this case, by $\frac{k}{2} \leq t \leq k-2$, D_j^{k-t-1} is adjacent to $D_{j'}^{k-t-1}$, i.e., $\{D_1^1, \ldots, D_1^{k-t-1}, \ldots, D_m^1, \ldots, D_m^{k-t-1}\}$ is a clique, hence $N_G(P_1^t)$ induces a complete graph, thus P_1^t are simplicial and we can eliminate them.

Then by similar arguments $P_2^{k-1}, \ldots, P_2^{\frac{k}{2}}$ are simplicial and we can eliminate them.

Next, for all D_j^k, their neighbor sets are only $D_j, D_j^1 \ldots, D_j^{k-1}$ which are cliques, therefore they are simplicial and we can eliminate them.

For all D_j^{k-1}, by (R_2), their neighborhoods are only

$$\{D_j, D_j^1 \ldots, D_j^{k-2}, D_1, \ldots, D_m\} \cup \{U_i \mid u_i \in d_j, 1 \leq i \leq n\}$$

which are cliques, therefore they are simplicial and we can eliminate them.

For all D_j^{k-2}, by (R_3), their neighborhoods are only

$$\{D_j, D_j^1 \ldots, D_j^{k-3}\} \cup \{U_1, \ldots, U_n, D_1, \ldots, D_m, D_1^1, \ldots, D_m^1, P_1^1, P_2^1\}$$

which are cliques, therefore they are simplicial and we can eliminate them.

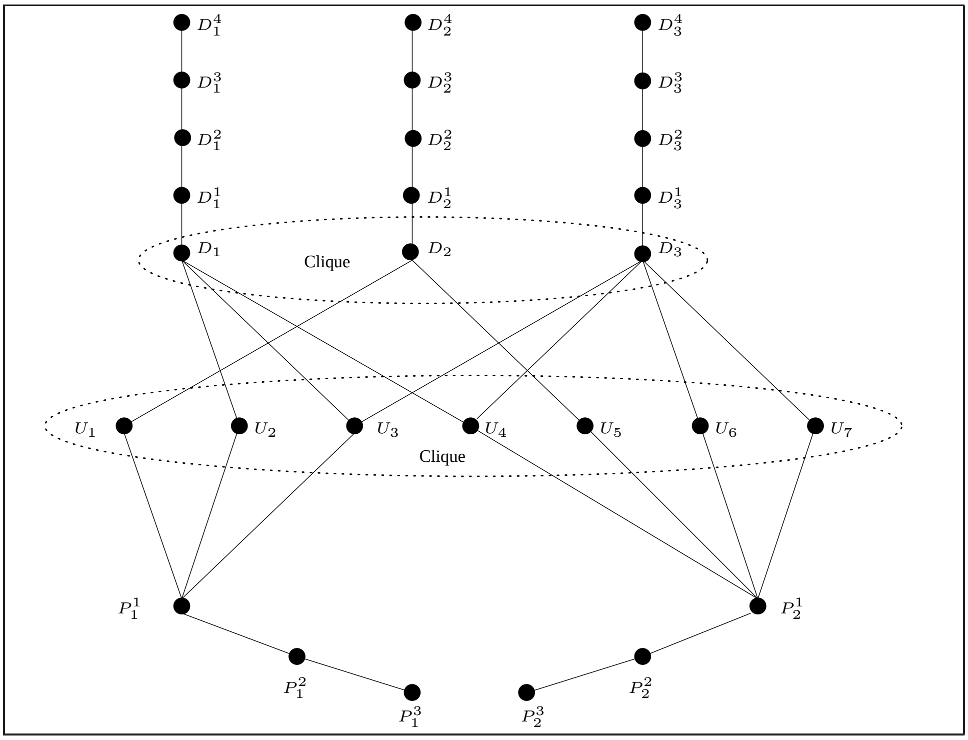

Figure 4.8: Fourth root graph H to the solution S_1, S_2

For all D_j^t for $\frac{k}{2} \leq t \leq k-3$, by (R_3), their neighborhoods are only

$$\begin{aligned} N_G(D_j^t) \;:=\; & \{D_j, D_j^1 \ldots, D_j^{t-1}\} \cup \{U_1, \ldots, U_n, D_1, \ldots, D_m\} \\ & \cup\{D_1^1, \ldots, D_m^1, \ldots, D_1^{k-t-1}, \ldots, D_m^{k-t-1}\} \\ & \cup\{P_1^1, \ldots, P_1^{k-t-1}, P_2^1, \ldots, P_2^{k-t-1}\} \end{aligned}$$

Note that in this case, by $\frac{k}{2} \leq t \leq k-3$, P_1^{k-t-1} is adjacent to P_2^{k-t-1}, hence $N_G(D_j^t)$ induces a complete graph, thus D_j^t are simplicial and we can eliminate them.

Finally, remaining vertices induce a complete graph G_2 and this completes the proof of Lemma 4.3.2. □

Lemma 4.3.3 *If there exists a partition of S into two disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2, then there exists a perfect graph H such that $G = H^k$.*

Proof. Let H have the same vertex set as G. The edges of H are as follows; see also Figures 4.9.

- Edges of subset vertices and their tail vertices: For all $2 \leq t \leq k$:
  $D_j^t \leftrightarrow D_j^{t-1}$, $D_j^1 \leftrightarrow D_j$, $D_j \leftrightarrow \{U_i \mid u_i \in d_j, 1 \leq i \leq n\}$, and
  $D_1, \ldots, D_m$ form a clique.

- Edges of element vertices: $U_1, \ldots, U_n$ form a clique.
- Edges of partition vertices:
  $P_1^1 \leftrightarrow \{U_i \mid u_i \in S_1, 1 \leq i \leq n\}$, and $P_2^1 \leftrightarrow \{U_i \mid u_i \in S_2, 1 \leq i \leq n\}$, and for all $2 \leq t \leq k-1$, $P_1^t \leftrightarrow P_1^{t-1}$ and $P_2^t \leftrightarrow P_2^{t-1}$.

Similar to the proof of Lemma 4.1.1 (see on p. 20) and Lemma 4.2.1 (see on p. 27), we can show that $H^k = G$ by following the order of the presentation of the edge set of G; cf. (R_1) – (R_7), as follows.

For D_j^k, it is clear that $D_j, D_j^1, \ldots, D_j^k$ form a clique in H^k for all j, hence $N_{H^k}(D_j^k) = N_G(D_j^k)$.

For $1 \leq t \leq k-1$, the vertex D_j^t reaches D_j in H in t steps. By the construction of H, $D_j \leftrightarrow D_{j'}$ for all $j \neq j'$ and $D_j \leftrightarrow U_i$ whenever $u_i \in d_j$. Thus within $t+1$ steps, D_j^t reaches $D_{j'}$ and D_j^t reaches U_i whenever $u_i \in d_j$, therefore $D_j^t \leftrightarrow U_i$ in H^k whenever $u_i \in d_j$ and $D_j^t \leftrightarrow D_{j'}$ in H^k for all $j \neq j'$. By comparing with (R_2), $N_{H^k}(D_j^t) = N_G(D_j^t)$.

For $1 \leq t \leq k-2$, the vertex D_j^t reaches D_j in t steps. Since $D_j \leftrightarrow U_i$ whenever $u_i \in d_j$ and $D_1, \ldots, D_m$ form a clique in H and $U_1, \ldots, U_n$ form a clique in H. Thus within $t+2$ steps, D_j^t reaches U_i for all i, therefore $D_j^t \leftrightarrow U_i$ in H^k for all i. Also, within k steps, D_j^t reaches $D_{j'}^1, \ldots, D_{j'}^{k-t-1}$, therefore $D_j^t \leftrightarrow \{D_{j'}^1, \ldots, D_{j'}^{k-t-1}\}$ in H^k for all $j \neq j'$. Furthermore, since we have a solution for SET SPLITTING, every D_j has a common neighbor to P_1^1 and a common neighbor to P_2^1. Moreover, in H, $\{P_1^1, \ldots, P_1^{k-1}\}$ and $\{P_2^1, \ldots, P_2^{k-1}\}$ are induced path of length $k-2$, therefore D_j^t reaches $P_1^1, \ldots, P_1^{k-t-1}$ and D_j^t reaches $P_2^1, \ldots, P_2^{k-t-1}$ within k steps. By comparing with (R_3), $N_{H^k}(D_j^t) = N_G(D_j^t)$.

Next, the vertex D_j reaches U_i in one step whenever $u_i \in d_j$. In H, $U_1, \ldots, U_n$ form a clique, so within k steps, D_j reaches U_i for all i and D_j reaches $D_{j'}$ for all $j' \neq j$. Moreover, since we have a solution for SET SPLITTING, every D_j has a common neighbor to P_1^1 and a common neighbor to P_2^1. In H, $\{P_1^1, \ldots, P_1^{k-1}\}$ and $\{P_2^1, \ldots, P_2^{k-1}\}$ are induced path of length $k-2$. Thus, within k steps, D_j reaches $P_1^1, \ldots, P_1^{k-1}$, $P_2^1, \ldots, P_2^{k-1}$. By comparing with (R_4), $N_{H^k}(D_j) = N_G(D_j)$.

For U_i, $U_1, \ldots, U_n$ form a clique in H, hence $U_1, \ldots, U_n$ form also a clique in H^k. In H, $U_i \leftrightarrow P_1^1$ whenever $u_i \in S_1$, and $U_i \leftrightarrow P_2^1$ whenever $u_i \in S_2$. Moreover, $P_1^1, \ldots, P_1^{k-1}$ form an induced path of length $k-2$ in H and $P_2^1, \ldots, P_2^{k-1}$ form an induced path of length $k-2$ in H. Thus, within k steps U_i reaches $P_1^1, \ldots, P_1^{k-1}, P_2^1, \ldots, P_2^{k-1}$. By comparing with (R_5), $N_{H^k}(U_i) = N_G(U_i)$ for all i.

Finally, for partition vertices, it is clear that $P_1^1, \ldots, P_1^{k-1}, U_1, \ldots, U_n$ form a clique in H^k and $P_2^1, \ldots, P_2^{k-1}, U_1, \ldots, U_n$ form a clique in H^k. Moreover, for all $1 \leq t \leq k-2$, since P_1^1 and P_2^1 have at least one neighbor U_i in H. Also, since $U_1, \ldots, U_n$ form a clique in H, therefore $P_1^t \leftrightarrow \{P_2^1, \ldots, P_2^{k-t-1}\}$ and $P_2^t \leftrightarrow \{P_1^1, \ldots, P_1^{k-t-1}\}$.

By comparing with (R_6) and (R_7), $N_{H^k}(P_1^t) = N_G(P_1^t)$ and $N_{H^k}(P_2^t) = N_G(P_2^t)$ for all t.

We checked that the edge set of H^k is equal to the edge set of G.

Now we will show that H is a *perfect* graph. We do this by using the *Strong Perfect Graph Theorem* that a graph H is perfect if and only if H and $\overline{H}$ contain no odd hole (cf. [23]).

The following two observations are helpful for the proof.

Observation 4.3.4 *If v is a simplicial vertex in H, then H has an odd hole if and only if $H - v$ has an odd hole.*

Observation 4.3.5 *If v is adjacent to all but one vertex in H, then H has an odd hole if and only if $H - v$ has an odd hole.*

Claim 1: H has no odd hole.

Proof of Claim 1. Since in H, D_j^k are simplicial, therefore by Observation 4.3.4, we can eliminate all D_j^k for all $1 \leq j \leq m$. Similarly, for each vertex of $D_j^{k-1}, \ldots, D_j^1$, its neighbors in the remaining vertices of the H form a clique, hence they are simplicial and we can eliminate $D_j^{k-1}, \ldots, D_j^1$ for all $1 \leq j \leq m$. By the same argument, each partition vertex of sets $\{P_1^{k-1}, \ldots, P_1^1\}$ and $\{P_2^{k-1}, \ldots, P_2^1\}$ is also simplicial and we eliminate them. Now only element vertices and subset vertices are remaining. For remaining vertices, since $H[D_1, \ldots, D_m]$ and $H[U_1, \ldots, U_n]$ are a complete graph, hence they cannot contain a hole of length at least five. This complete the proof that H has no odd hole.

Claim 2: $\overline{H}$ has no odd hole.

Proof of Claim 2. Since in $\overline{H}$, D_j^k is adjacent to all vertices but one vertex, therefore by Observation 4.3.5, we can eliminate all D_j^k for all $1 \leq j \leq m$.

By the same argument, each vertex of sets $D_j^{k-1}, \ldots, D_j^1$ for all $1 \leq j \leq m$ and $\{P_1^{k-1}, \ldots, P_1^2, P_2^{k-1}, \ldots, P_2^2\}$ is adjacent to all vertices but one vertex, thus we can eliminate them.

Moreover, in $\overline{H}$ both the set of element vertices and the set of subset vertices induce a stable set. Hence, $\overline{H}[D_1, \ldots, D_m, U_1, \ldots, U_n]$ is a bipartite graph. Therefore, if an odd hole exists, either P_1^1 or P_2^1 must be involved. Also, in $\overline{H}$, P_1^1 adjacent to P_2^1; P_1^1 and P_2^1 are adjacent to all $\{D_1, \ldots, D_m\}$; and P_1^1 and P_2^1 partition the vertices of $\{U_1, \ldots, U_n\}$. Thus, any induced cycle which involves both P_1^1 and P_2^1 has length three. And any induced cycle which involves only P_1^1 or P_2^1 has length four.

So no odd hole exists in $\overline{H}$. This completes the proof of Lemma 4.3.3. □

For the example instance, the k-th root H corresponds to the solution S_1, S_2 is shown in Figure 4.9.

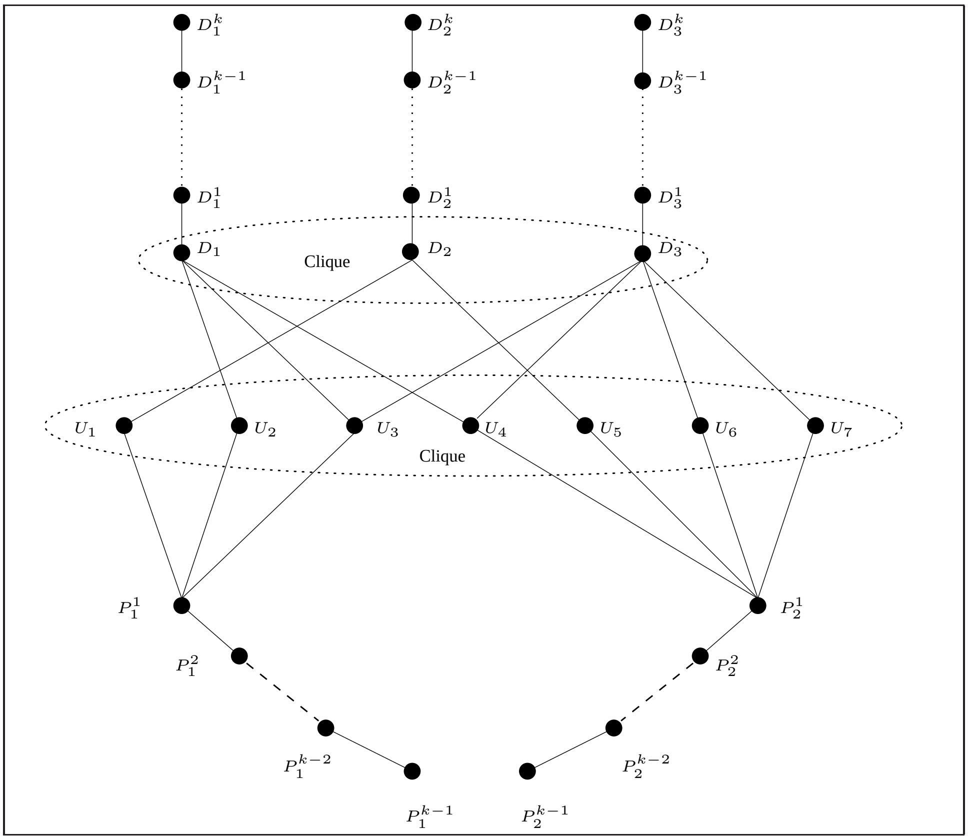

Figure 4.9: A k-th root H in Lemma 4.3.3 to the example solution S_1, S_2

Now we show that if G has a k-th root H (*not necessarily perfect*), then there is a partition of S into two disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2.

The following Proposition 4.3.6 and Lemma 4.3.7 can be proved similarly as we have done for Proposition 4.2.2 (see on p. 30) and Lemma 4.2.3 (see on p. 32), respectively.

Proposition 4.3.6 *If H is a k-th root of G, then, in H:*

(i) *For all i, j: D_j is only adjacent to U_i whenever $u_i \in d_j$, and $D_{j'}$ for all $j' \neq j$.*

(ii) *D_j^k is only adjacent to D_j^{k-1}, D_j^1 is only adjacent to D_j and D_j^2, and D_j^t is only adjacent to D_j^{t-1} and D_j^{t+1} for $2 \leq t \leq k-1$;*

(iii) *P_ℓ^{k-1} is only adjacent to P_ℓ^{k-2}, and for all $2 \leq t \leq k-2$: P_ℓ^t is only adjacent to P_ℓ^{t-1} and P_ℓ^{t+1} for $\ell = 1, 2$.*

Lemma 4.3.7 *If H is a k-th root of G, then there exists a partition of S into two disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2.*

Lemmas 4.3.3 and 4.3.7 show that k-TH ROOT OF CHORDAL GRAPH is NP-complete for all fixed even k. Hence, from CASE 1 and CASE 2 we conclude

Theorem 4.3.8 *For all fixed $k \geq 2$,* k-TH ROOT OF CHORDAL GRAPH *is NP-complete.*

Finally, the following consequences immediately follow from the reduction for proving the NP-completeness result of k-TH ROOT OF CHORDAL GRAPH:

Corollary 4.3.9 *Given chordal graph G and odd integer k, determine if there exists a chordal graph H such that $H^k = G$ is NP-complete.*

Corollary 4.3.10 *Given chordal graph G, determine if there exists a* perfect graph *H such that $H^k = G$ is NP-complete.*

4.4 Concluding remarks

Although it has been greatly expected that k-TH POWER OF GRAPH and k-TH POWER OF BIPARTITE GRAPH are NP-complete for all fixed $k \geq 2$, respectively, $k \geq 3$, our proofs here first show that these problems are indeed NP-complete. Furthermore, the reduction proving NP-completeness of k-TH POWER OF BIPARTITE GRAPH also shows that for all fixed $k \geq 3$, recognizing of k-th powers of graph class $\mathcal{C}$ containing all bipartite graphs (such as triangle-free graphs, parity graphs, perfect graphs, etc.) is NP-complete.

We also have proved that k-TH POWER OF CHORDAL GRAPH is NP-complete for all fixed $k \geq 2$. Moreover, the reduction in Section 4.2 shows that k-TH POWER OF $\mathcal{C}$-GRAPH is NP-complete for all fixed $k \geq 2$ whenever $\mathcal{C}$ contains all chordal graphs, such as weakly chordal graphs, HHD-free graphs, hole-free graphs, perfect graphs, etc.

In Section 4.3, by using similar techniques we have found proofs of NP-completeness of k-th powers of perfect graphs even if the input is restricted to chordal graphs.

Finally, we still hope that similar techniques can be found proving NP-completeness of k-th powers of strongly chordal graphs.

Chapter 5

Powers of Graphs with Girth Conditions

In this chapter we consider the complexity of recognizing k-th powers of graphs with girth conditions. The *tail structure* (cf. Lemma 3.2.1 and Lemma 3.2.2) is a useful tool to solve these problems.

In Section 5.1 we show that recognizing squares of graphs with girth at most four is NP-complete. Section 5.2 we prove that recognizing k-th powers of graphs with girth at most $2\lfloor \frac{k}{2} \rfloor + 2$ is NP-complete.

5.1 Squares of graphs with girth at most four

Note that the reductions in proving the NP-completeness results by Motwani and Sudan [55] show that recognizing squares of graphs with girth three is NP-complete. In this section we extend the idea to prove that the following problem is NP-complete.

SQUARE OF GRAPH WITH GIRTH ≤ 4

Instance: A graph G.

Question: Is there a graph H with girth ≤ 4 such that $G = H^2$?

Observe that SQUARE OF GRAPH WITH GIRTH ≤ 4 is in NP. We will reduce the NP-complete problem SET SPLITTING to it.

We also apply the tail structure of a vertex v to ensure that v has the same neighbors in any square root H of G (cf. Lemma 3.2.1, p. 15).

Let $S = \{u_1, \ldots, u_n\}$, $D = \{d_1, \ldots, d_m\}$ where $d_j \subseteq S$, $1 \leq j \leq m$, be an instance of SET SPLITTING. We construct an instance $G = G(D, S)$ for SQUARE OF GRAPH WITH GIRTH ≤ 4 as follows.

The vertex set of graph G consists of:

- U_i, $1 \leq i \leq n$. Each 'element vertex' U_i corresponds to the element u_i in S.
- D_j, $1 \leq j \leq m$. Each 'subset vertex' D_j corresponds to the subset d_j in D.
- D_j^1, D_j^2, D_j^3, $1 \leq j \leq m$. Each three 'tail vertices' D_j^1, D_j^2, D_j^3 of the subset vertex D_j correspond to the subset d_j in D.
- S_1, S_1', S_2, S_2', four 'partition vertices'.
- X, a 'connection vertex'.

The edge set of graph G consists of:

- Edges of tail vertices of subset vertices:
  For all $1 \leq j \leq m$: $D_j^3 \leftrightarrow D_j^2$, $D_j^3 \leftrightarrow D_j^1$, $D_j^2 \leftrightarrow D_j^1$, $D_j^2 \leftrightarrow D_j$, $D_j^1 \leftrightarrow D_j$, and for all i, $D_j^1 \leftrightarrow U_i$ whenever $u_i \in d_j$.
- Edges of subset vertices:
  For all $1 \leq j \leq m$: $D_j \leftrightarrow S_1$, $D_j \leftrightarrow S_1'$, $D_j \leftrightarrow S_2$, $D_j \leftrightarrow S_2'$, $D_j \leftrightarrow X$, $D_j \leftrightarrow U_i$ whenever $u_i \in d_j$, and $D_j \leftrightarrow D_k$ for all k with $d_j \cap d_k \neq \varnothing$.
- Edges of element vertices:
  For all $1 \leq i \leq n$: $U_i \leftrightarrow X$, $U_i \leftrightarrow S_1$, $U_i \leftrightarrow S_2$, $U_i \leftrightarrow S_1'$, $U_i \leftrightarrow S_2'$, and $U_1, \ldots, U_n$ form a clique.
- Edges of partition vertices:
  $S_1 \leftrightarrow X$, $S_1 \leftrightarrow S_1'$, $S_1 \leftrightarrow S_2'$, $S_2 \leftrightarrow X$, $S_2 \leftrightarrow S_1'$, $S_2 \leftrightarrow S_2'$, $S_1' \leftrightarrow X$, $S_2' \leftrightarrow X$.

Clearly, G can be constructed from D, S in polynomial time. For an illustration, given $S = \{u_1, u_2, u_3, u_4, u_5\}$ and $D = \{d_1, d_2, d_3, d_4\}$ with $d_1 = \{u_1, u_2, u_3\}$, $d_2 = \{u_2, u_5\}$, $d_3 = \{u_3, u_4\}$, and $d_4 = \{u_1, u_4\}$, the graph G is depicted in Figure 5.1. In the figure, the two dotted lines from a vertex to the clique $\{U_1, U_2, U_3, U_4, U_5, X\}$ mean that the vertex is adjacent to all vertices in that clique.

Note that, apart from the three vertices X, S_1', and S_2' (or, symmetrically, X, S_1, and S_2), our construction is the same as the one in [46, §3.1.1]. While S_1 and S_2 will represent a partition of the ground set S (Lemma 5.1.2), the vertices X, S_1', and S_2' allow us to make a square root of G being C_3-free (Lemma 5.1.1).

Lemma 5.1.1 *If there exists a partition of S into two disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2, then there exists a graph H with girth four such that $G = H^2$.*

Proof. Let H have the same vertex set as G. The edges of H are as follows; see also Figure 5.2 for an example.

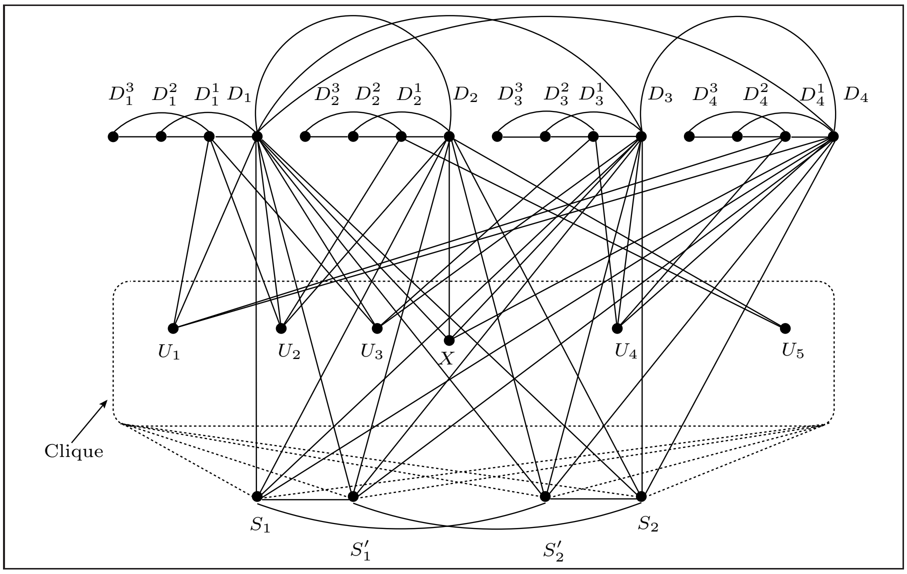

Figure 5.1: The graph G for the example instance of SET SPLITTING

- Edges of subset vertices and their tail vertices:
  For all $1 \le j \le m$: $D_j^3 \leftrightarrow D_j^2$, $D_j^2 \leftrightarrow D_j^1$, $D_j^1 \leftrightarrow D_j$, and for all i, $D_j \leftrightarrow U_i$ whenever $u_i \in d_j$.
- Edges of partition vertices:
  $S_1 \leftrightarrow S_1'$, $S_2 \leftrightarrow S_2'$, and for all i, $S_1 \leftrightarrow U_i$ and $S_2' \leftrightarrow U_i$ whenever $u_i \in S_1$, and $S_2 \leftrightarrow U_i$ and $S_1' \leftrightarrow U_i$ whenever $u_i \in S_2$.
- Edges of the connection vertex:
  $X \leftrightarrow U_i$ for all $1 \le i \le n$.

Now we verify that the edge set of H^2 is equal to the edge set of G. We do this by following the order of the presentation of the edge set of G above.

For D_j^3 and D_j^2, it is clear that $D_j^3 \leftrightarrow D_j^2$, $D_j^3 \leftrightarrow D_j^1$, $D_j^2 \leftrightarrow D_j^1$ and $D_j^2 \leftrightarrow D_j$ for all j, hence $N_{H^2}(D_j^3) = N_G(D_j^3)$ and $N_{H^2}(D_j^2) = N_G(D_j^2)$.

The vertex D_j^1 reaches D_j in one step. Since in H, $D_j \leftrightarrow U_i$ whenever $u_i \in d_j$, hence, within two steps $D_j^1 \leftrightarrow U_i$ whenever $u_i \in d_j$. Thus, $N_{H^2}(D_j^1) = N_G(D_j^1)$.

For D_j, by the construction of H, it is easily to see that within two steps, $D_j \leftrightarrow X$, $D_j \leftrightarrow U_i$ whenever $u_i \in d_j$, and $D_j \leftrightarrow D_k$ for all k with $d_j \cap d_k \neq \varnothing$. Moreover, since we have a solution for SET SPLITTING, every D_j has a common neighbor with S_1, S_2, S_1', S_2'. Therefore, within two steps D_j reaches S_1, S_2, S_1', S_2'. Thus, $N_{H^2}(D_j) = N_G(D_j)$.

Now consider U_i. In H, $U_i \leftrightarrow X$, hence $U_1, \ldots, U_n$ form a clique in H^2. Moreover, by the construction of H, it is easily to verify that in H^2, $U_i \leftrightarrow S_1$, $U_i \leftrightarrow S_2$, $U_i \leftrightarrow S_1'$, $U_i \leftrightarrow S_2'$ for all i. Thus, $N_{H^2}(U_i) = N_G(U_i)$ for all i.

Finally, for partition vertices, it is clear that in H^2, $S_1 \leftrightarrow X$, $S_1 \leftrightarrow S_1'$, $S_1 \leftrightarrow S_2'$, $S_2 \leftrightarrow X$, $S_2 \leftrightarrow S_1'$, $S_2 \leftrightarrow S_2'$, $S_1' \leftrightarrow X$ and $S_2' \leftrightarrow X$.

We have checked that the edge set of H^2 is equal to the edge set of G.

Now we will show that H has girth ≤ 4. By construction, the neighborhood in H of any vertex is a stable set, hence H has no C_3. Observe that H has girth four as it contains a C_4 consisting of X, D_i, an element vertex that corresponds to an element in $d_i \cap S_1$, and another element vertex that corresponds to an element in $d_i \cap S_2$. □

In the above example, $S_1 = \{u_1, u_3, u_5\}$ and $S_2 = \{u_2, u_4\}$ is a possible legal partition of S. The corresponding graph H constructed in the proof of Lemma 5.1.1 is depicted in Figure 5.2.

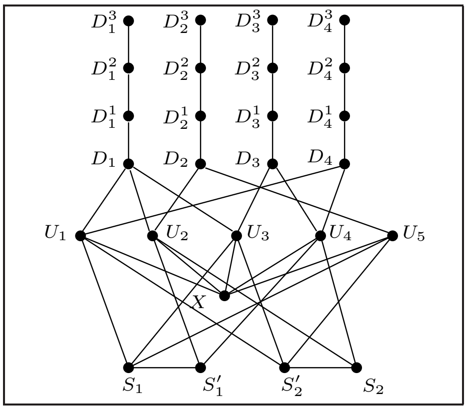

Figure 5.2: An example of root H with girth 4

Lemma 5.1.2 *If H is a square root of G, then there exists a partition of S into two disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2.*

Proof. First, observe that for each j, D_j^3, D_j^2, D_j^1, D_j satisfy the properties of Lemma 3.2.1. Hence, in H, D_j is adjacent to exactly D_j^1 and U_i for which $u_i \in d_j$. This and the fact that, in G, the partition vertices S_1, S_1', S_2, S_2' are non-adjacent to the tail vertices, show that $N_H(S_1) - \{S_1', S_2', X\}$ and $N_H(S_2) - \{S_1', S_2', X\}$ consist of element vertices only.

Now, since S_1 and S_2 are non-adjacent in G, they have no common neighbor in H. Therefore, $N_H(S_1) - \{S_1', S_2', X\}$ and $N_H(S_2) - \{S_1', S_2', X\}$ will define a partition of the element set. Since the partition vertices are adjacent to all subset vertices in G but not in H, each of S_1 and S_2 has, in H, a common neighbor with D_j in the element set for all j. Thus, $N_H(S_1) - \{S_1', S_2', X\}$ and $N_H(S_2) - \{S_1', S_2', X\}$ define a desired partition of S. □

Note that in Lemma 5.1.2 above we did not require that H has girth four. Thus, any square root of G–particularly, any square root with girth at most four–will tell us how to do SET SPLITTING. Together with Lemma 5.1.1 we conclude:

Theorem 5.1.3 SQUARE OF GRAPH WITH GIRTH ≤ 4 *is NP-complete.*

5.2 k-th powers of graphs with girth $\leq 2\lfloor\frac{k}{2}\rfloor + 2$

The results in Section 5.1 first shown that SQUARE OF GRAPH WITH GIRTH ≤ 4 is NP-complete. Moreover, the reduction for proving the NP-completeness result of cubes of bipartite graphs [45] show that recognizing cubes of graphs with girth at most four is NP-complete.

In this section we generalize those results by showing that the following problem is NP-complete.

k-TH POWER OF GRAPH WITH GIRTH $\leq 2\lfloor\frac{k}{2}\rfloor + 2$

Instance: A graph G.
Question: Is there a graph H with girth $\leq 2\lfloor\frac{k}{2}\rfloor + 2$ such that $G = H^k$ for a fixed $k \geq 4$?

We will reduce SET SPLITTING to it. The reduction for this problem is described as follows:

Let $S = \{u_1, \ldots, u_n\}$, $D = \{d_1, \ldots, d_m\}$ where $d_j \subseteq S$, $1 \leq j \leq m$, be an instance of SET SPLITTING. We construct an instance $G = G(D, S)$ for k-TH POWER OF GRAPH WITH GIRTH $\leq 2\lfloor\frac{k}{2}\rfloor + 2$ as follows.

Case 1: k is even.

The vertex set of G consists of:

- Element vertices: U_i, $1 \leq i \leq n$.
  Each 'element vertex' U_i corresponds to the element u_i in S.
- Subset vertices: D_j, $1 \leq j \leq m$.
  Each 'subset vertex' D_j corresponds to the subset d_j in D.
- Tail vertices: $D_j^1, \ldots, D_j^k$, $1 \leq j \leq m$. k 'tail vertices' of the subset vertex D_j.
- Partition vertices: $P_1^1, P_1^2, P_2^1, P_2^2$.
- Connection vertices: $X_{ji}^1, \ldots, X_{ji}^{\frac{k-2}{2}}$ for $1 \leq j \leq m$, $1 \leq i \leq n$ corresponding $u_i \in d_j$;
  $U_i^1, \ldots, U_i^{\frac{k-2}{2}}$; $Z_i^1, \ldots, Z_i^{\frac{k-2}{2}}$ for $1 \leq i \leq n$; and a vertex X.

The edge set of G consists of:

- Edges of tail vertices: For all $1 \leq j \leq m$,

(E_1) $D_j, D_j^1, \ldots, D_j^k$ form a clique;

(E_2) For all $\frac{k+2}{2} \leq t \leq k-1$: $D_j^t \leftrightarrow \{X_{ji}^h \mid 1 \leq h \leq k-t, u_i \in d_j\}$;

(E_3) For all $1 \leq t \leq \frac{k}{2}$: $D_j^t \leftrightarrow \{U_i \mid u_i \in d_j, 1 \leq i \leq n\}$,
$D_j^t \leftrightarrow \{X_{ji}^h \mid 1 \leq h \leq \frac{k-2}{2}, u_i \in d_j\}$;

(E_4) For all $1 \leq t \leq \frac{k-2}{2}$: $D_j^t \leftrightarrow X$, $D_j^t \leftrightarrow \{U_i^h, Z_i^h \mid 1 \leq h \leq \frac{k}{2} - t, u_i \in d_j\}$,
$D_j^t \leftrightarrow \{X_{j'i}^h \mid t \leq h \leq \frac{k-2}{2}, u_i \in d_j \cap d_{j'}\}$;

(E_5) For all $1 \leq t \leq \frac{k-4}{2}$: $D_j^t \leftrightarrow \{U_i \mid 1 \leq i \leq n\}$;

(E_6) For all $1 \leq t \leq \frac{k-6}{2}$: $D_j^t \leftrightarrow \{X_{j'i}^h \mid t+2 \leq h \leq \frac{k-2}{2}, u_i \in d_{j'}\}$,
$D_j^t \leftrightarrow \{U_i^h, Z_i^h \mid 1 \leq h \leq \frac{k-4}{2} - t, 1 \leq i \leq n\}$.

- Edges of subset vertices: For all $1 \leq j \leq m$,

(E_7) $D_j \leftrightarrow \{U_i \mid 1 \leq i \leq n\}$, $D_j \leftrightarrow \{X, P_1^1, P_1^2, P_2^1, P_2^2\}$,
$D_j \leftrightarrow \{X_{ji}^h, U_i^h, Z_i^h \mid 1 \leq h \leq \frac{k-2}{2}, u_i \in d_j\}$,
$D_j \leftrightarrow \{D_{j'} \mid d_j \cap d_{j'} \neq \varnothing, 1 \leq j' \leq m\}$;

(E_8) If $k \geq 6$: $D_j \leftrightarrow \{U_i^h, Z_i^h \mid 1 \leq h \leq \frac{k-4}{2}, 1 \leq i \leq n\}$, and
$D_j \leftrightarrow \{X_{j'i}^h \mid 2 \leq h \leq \frac{k-2}{2}, u_i \in d_{j'}\}$.

- Edges of connection vertices and element vertices:

(E_9)

$$\begin{aligned}\mathcal{A} \;:=\; & \{X\} \cup \{X_{ji}^h \mid 1 \leq h \leq \frac{k-2}{2}, u_i \in d_j\} \\ & \cup \{U_i^h, Z_i^h \mid 1 \leq i \leq n, 1 \leq h \leq \frac{k-2}{2}\} \\ & \cup \{U_i \mid 1 \leq i \leq n\} \text{ form a clique.}\end{aligned}$$

- Edges of partition vertices:

(E_{10}) $\{P_1^1, P_1^2, P_2^1, P_2^2\} \leftrightarrow \mathcal{A}$,
$P_1^1 \leftrightarrow \{P_1^2, P_2^2\}$, and $P_2^1 \leftrightarrow \{P_2^2, P_1^2\}$.

Case 2: k is odd.

The vertex set of G consists of:

- Element vertices: U_i, $1 \leq i \leq n$.
  Each 'element vertex' U_i corresponds to the element u_i in S.

- Subset vertices: D_j, $1 \leq j \leq m$.
  Each 'subset vertex' D_j corresponds to the subset d_j in D.

- Tail vertices: $D_j^1, \ldots, D_j^k$, $1 \leq j \leq m$. k 'tail vertices' of the subset vertex D_j.

- Partition vertices: P_1, P_2.
- Connection vertices: $X_{ji}^1, \ldots, X_{ji}^{\frac{k-3}{2}}$ for $1 \leq j \leq m$, $1 \leq i \leq n$ corresponding $u_i \in d_j$; $U_i^1, \ldots, U_i^{\frac{k-3}{2}}$ and X.

The edge set of G consists of:

- Edges of tail vertices: For all $1 \leq j \leq m$,

  (L_1) $D_j, D_j^1, \ldots, D_j^k$ form a clique;

  (L_2) For all $\frac{k+3}{2} \leq t \leq k-1$: $D_j^t \leftrightarrow \{X_{ji}^h \mid 1 \leq h \leq k-t, u_i \in d_j\}$;

  (L_3) For all $1 \leq t \leq \frac{k+1}{2}$: $D_j^t \leftrightarrow \{U_i \mid u_i \in d_j, 1 \leq i \leq n\}$,
  $D_j^t \leftrightarrow \{X_{ji}^h \mid 1 \leq h \leq \frac{k-3}{2}, u_i \in d_j\}$;

  (L_4) For all $2 \leq t \leq \frac{k-1}{2}$: $D_j^t \leftrightarrow X$, $D_j^t \leftrightarrow \{U_i^h \mid 1 \leq h \leq \frac{k+1}{2} - t, u_i \in d_j\}$,
  $D_j^t \leftrightarrow \{X_{j'i}^h \mid t-1 \leq h \leq \frac{k-3}{2}, u_i \in d_j \cap d_{j'}\}$;

  (L_5) For all $1 \leq t \leq \frac{k-3}{2}$: $D_j^t \leftrightarrow \{U_i \mid 1 \leq i \leq n\}$;

  (L_6) For all $1 \leq t \leq \frac{k-5}{2}$: $D_j^t \leftrightarrow \{X_{j'i}^h \mid t+1 \leq h \leq \frac{k-3}{2}, u_i \in d_{j'}\}$,
  $D_j^t \leftrightarrow \{U_i^h \mid 1 \leq h \leq \frac{k-3}{2} - t\}$;

  (L_7) $D_j^1 \leftrightarrow \{X, P_1, P_2\}$, $D_j^1 \leftrightarrow \{U_i^h \mid 1 \leq h \leq \frac{k-3}{2}, u_i \in d_j\}$,
  $D_j^1 \leftrightarrow \{X_{j'i}^h \mid 1 \leq h \leq \frac{k-3}{2}, u_i \in d_j \cap d_{j'}\}$,
  $D_j^1 \leftrightarrow \{D_{j'} \mid d_j \cap d_{j'} \neq \varnothing, 1 \leq j' \leq m\}$;

- Edges of subset vertices: For all $1 \leq j \leq m$,

  (L_8) $D_j \leftrightarrow \{U_i \mid 1 \leq i \leq n\}$, $D_j \leftrightarrow \{X, P_1, P_2\}$,
  $D_j \leftrightarrow \{D_{j'} \mid d_j \cap d_{j'} \neq \varnothing, 1 \leq j' \leq m\}$,
  $D_j \leftrightarrow \{U_i^h \mid 1 \leq h \leq \frac{k-3}{2}, 1 \leq i \leq n\}$,
  $D_j \leftrightarrow \{X_{j'i}^h \mid 1 \leq h \leq \frac{k-3}{2}, u_i \in d_{j'}\}$.

- Edges of connection vertices and element vertices:

  (L_9)

$$\begin{aligned}\mathcal{B} \;:=\;& \{X\} \cup \{X_{ji}^h \mid 1 \leq h \leq \frac{k-3}{2}, u_i \in d_j\} \\ & \cup \{U_i^h \mid 1 \leq i \leq n, 1 \leq h \leq \frac{k-3}{2}\} \\ & \cup \{U_i \mid 1 \leq i \leq n\} \text{ form a clique.}\end{aligned}$$

- Edges of partition vertices:

  (L_{10}) $\{P_1, P_2\} \leftrightarrow \mathcal{B}$.

Clearly, G can be constructed from D, S in polynomial time. For an illustration, in case $k = 4$, the example instance yields the graph G is depicted in Figure 5.3. And in case $k = 5$, the example instance yields the graph G is depicted in Figure 5.5. In this example (refer to Section 3.2, p. 15), $S_1 = \{u_1, u_2, u_3\}$ and $S_2 = \{u_4, u_5, u_6, u_7\}$ is a possible solution. The corresponding fourth root graph H is shown in Figure 5.4, while the corresponding fifth root graph H is shown in Figure 5.6.

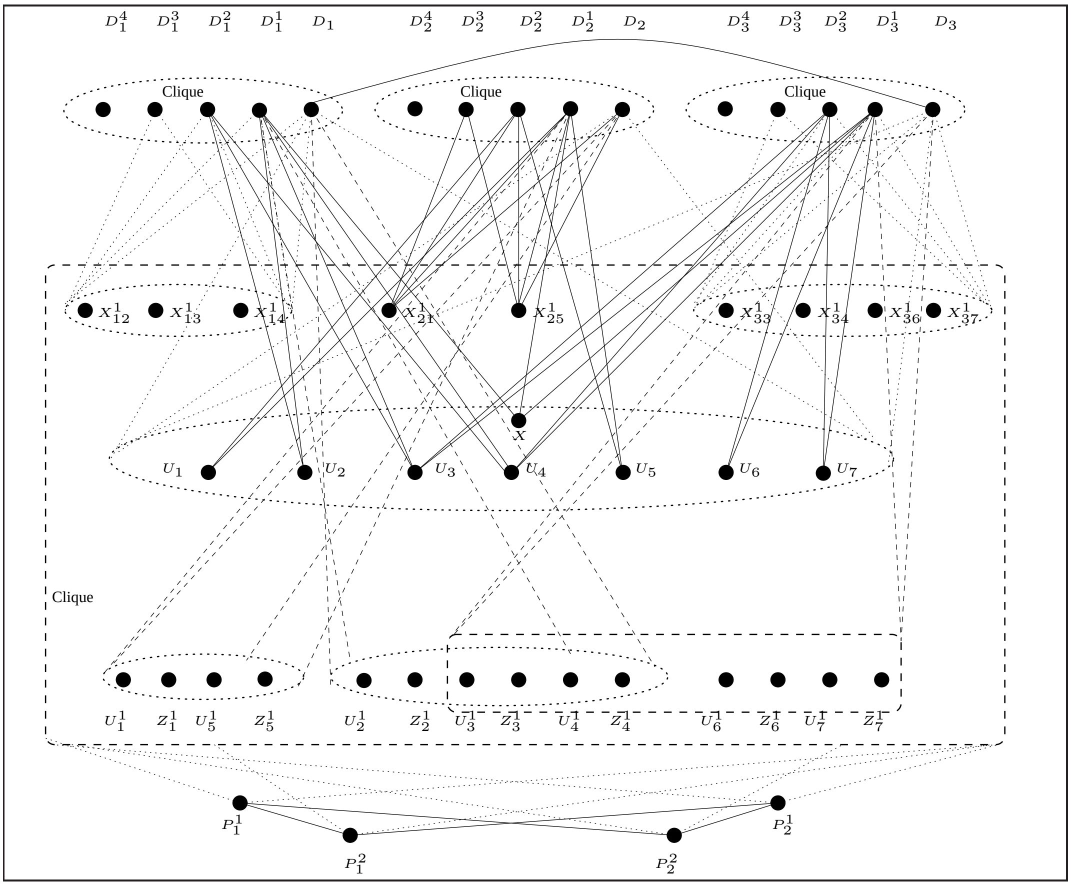

Figure 5.3: The graph G for the example instance of SET SPLITTING and $k = 4$

Lemma 5.2.1 *If there exists a partition of S into two disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2, then there exists a graph H with girth $\leq 2\lfloor \frac{k}{2} \rfloor + 2$ such that $G = H^k$.*

Proof.
CASE 1: k **is even.**

Let H have the same vertex set as G. The edges of H are as follows; see also Figure 5.7.

- Edges of subset vertices and their tail vertices:

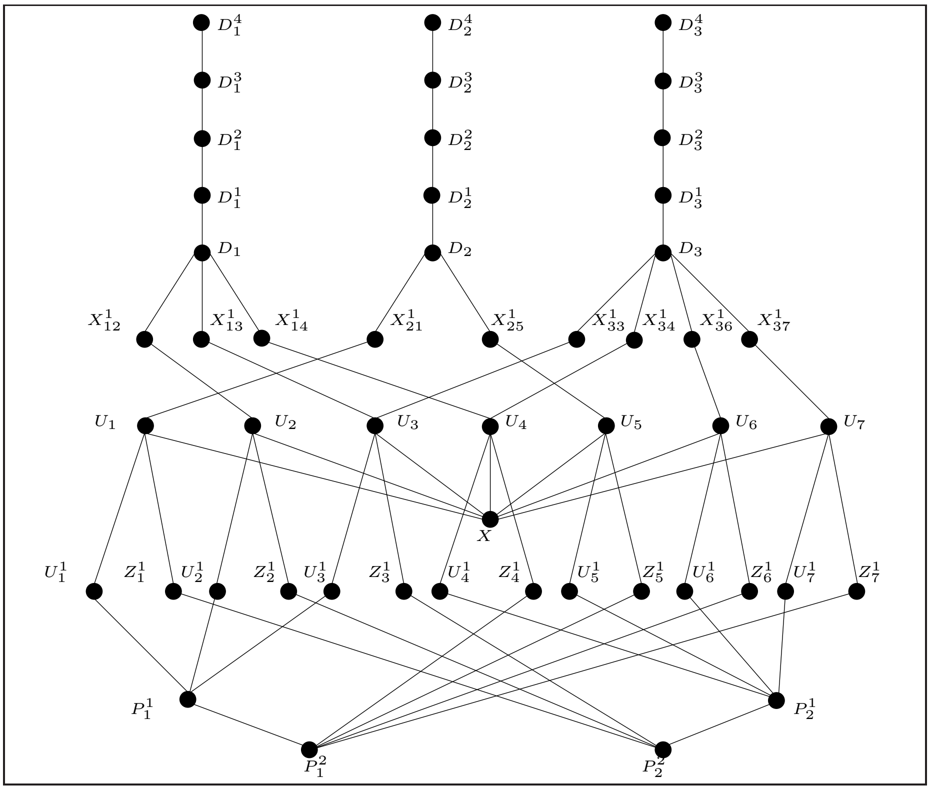

Figure 5.4: The fourth root graph H with girth six of G to the solution S_1, S_2

For all j, $2 \leq t \leq k$, $D_j^t \leftrightarrow D_j^{t-1}$ and $D_j^1 \leftrightarrow D_j$, and
$D_j \leftrightarrow \{X_{ji}^1 \mid u_i \in d_j, 1 \leq i \leq n\}$.

- Edges of partition vertices:
  $P_1^1 \leftrightarrow \{U_i^{\frac{k-2}{2}} \mid u_i \in S_1, 1 \leq i \leq n\}$, and $P_2^2 \leftrightarrow \{Z_i^{\frac{k-2}{2}} \mid u_i \in S_1, 1 \leq i \leq n\}$;
  $P_2^1 \leftrightarrow \{U_i^{\frac{k-2}{2}} \mid u_i \in S_2, 1 \leq i \leq n\}$, and $P_1^2 \leftrightarrow \{Z_i^{\frac{k-2}{2}} \mid u_i \in S_2, 1 \leq i \leq n\}$;
  $P_1^1 \leftrightarrow P_1^2$ and $P_2^1 \leftrightarrow P_2^2$.

- Edges of connection vertex and element vertices:
  $X \leftrightarrow \{U_1, \ldots, U_n\}$, $U_i \leftrightarrow U_i^1$, $U_i \leftrightarrow Z_i^1$ for all i.
  For all i, j, $1 \leq h \leq \frac{k-4}{2}$: $X_{ji}^h \leftrightarrow \{X_{ji}^{h+1} \mid u_i \in d_j\}$, and
  $U_i^h \leftrightarrow U_i^{h+1}$, $Z_i^h \leftrightarrow Z_i^{h+1}$.

Now we verify that the edge set of H^k is equal to the edge set of G. We do this by following the order of the presentation of the edge set of G; cf. $(E_1) - (E_{10})$.

For D_j^k, it is clear that $D_j, D_j^1, \ldots, D_j^k$ form a clique in H^k for all j, hence $N_{H^k}(D_j^k) = N_G(D_j^k)$.

For $\frac{k+2}{2} \leq t \leq k-1$, the vertex D_j^t reaches D_j in t steps. By the construction of H, $D_j \leftrightarrow X_{ji}^1$ whenever $u_i \in d_j$. Thus within $t+1$ steps, D_j^t reaches X_{ji}^1 whenever

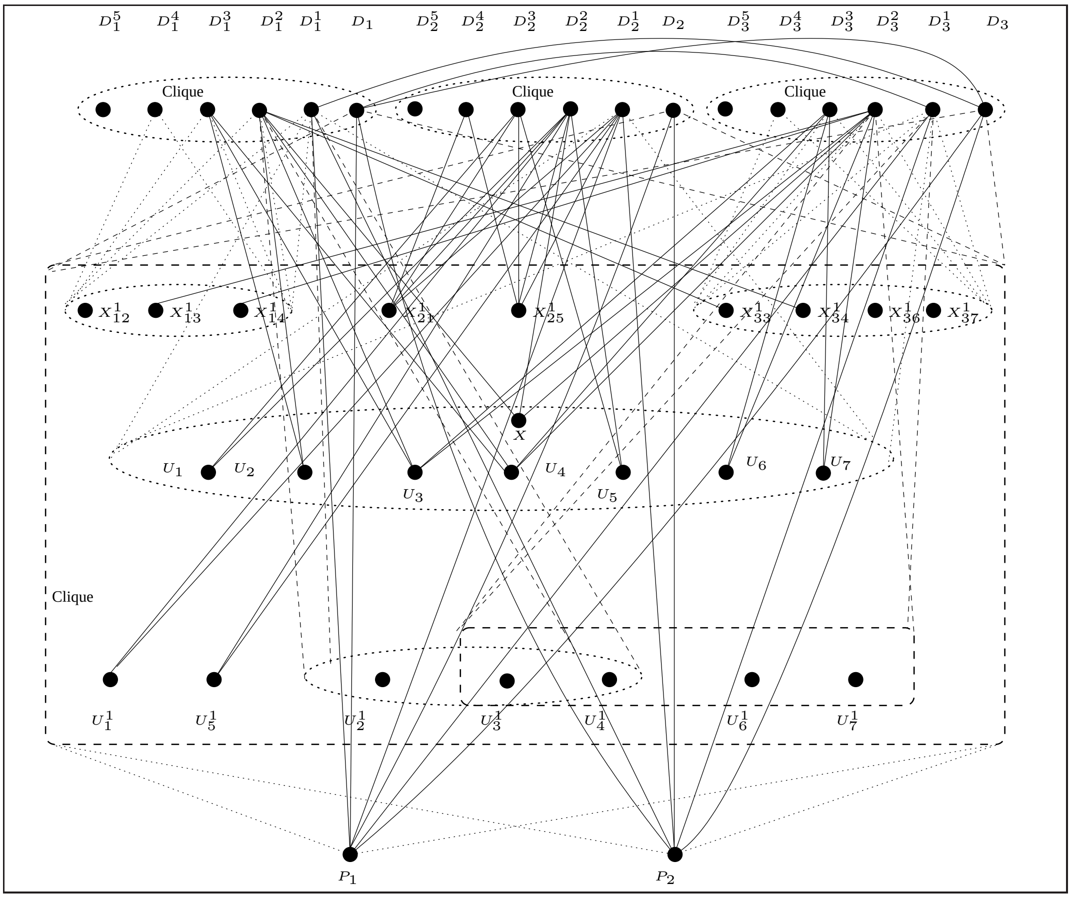

Figure 5.5: The graph G for the example instance of SET SPLITTING and $k = 5$

$u_i \in d_j$, therefore within k steps, D_j^t reaches $X_{ji}^1, \ldots, X_{ji}^{k-t}$ whenever $u_i \in d_j$. By comparing with (E_2), $N_{H^k}(D_j^t) = N_G(D_j^t)$.

For $1 \le t \le \frac{k}{2}$, the vertex D_j^t reaches D_j in t steps. Since within $\frac{k}{2}$ steps, D_j reaches U_i whenever $u_i \in d_j$. So within k steps, D_j^t reaches U_i whenever $u_i \in d_j$. Also, it is clear that in H^k, $D_j^t \leftrightarrow \{X_{ji}^h \mid 1 \le h \le \frac{k-2}{2}, u_i \in d_j\}$. By comparing with (E_3), $N_{H^k}(D_j^t) = N_G(D_j^t)$.

For $1 \le t \le \frac{k-2}{2}$, the vertex D_j^t reaches D_j in t steps. Since in H, $X \leftrightarrow \{U_1, \ldots, U_n\}$, $U_i \leftrightarrow U_i^1$, $U_i \leftrightarrow Z_i^1$ for all i, hence, within k steps, D_j^t reaches X, $U_i^1, Z_i^1, \ldots, U_i^{\frac{k}{2}-t}, Z_i^{\frac{k}{2}-t}$ whenever $u_i \in d_j$. Also, it is clear that in H^k, $D_j^t \leftrightarrow \{X_{j'i}^h \mid t \le h \le \frac{k-2}{2}, u_i \in d_j \cap d_{j'}\}$. By comparing with (E_4), $N_{H^k}(D_j^t) = N_G(D_j^t)$.

For $1 \le t \le \frac{k-4}{2}$, the vertex D_j^t reaches D_j in t steps. Since in H, D_j reaches X in $\frac{k-2}{2} + 1 + 1$ steps, and $X \leftrightarrow U_i$ for all i. Therefore, within k steps, D_j^t reaches U_i for all i. By comparing with (E_5), $N_{H^k}(D_j^t) = N_G(D_j^t)$.

For $1 \le t \le \frac{k-6}{2}$, the vertex D_j^t reaches D_j in t steps. Since in H, D_j reaches X

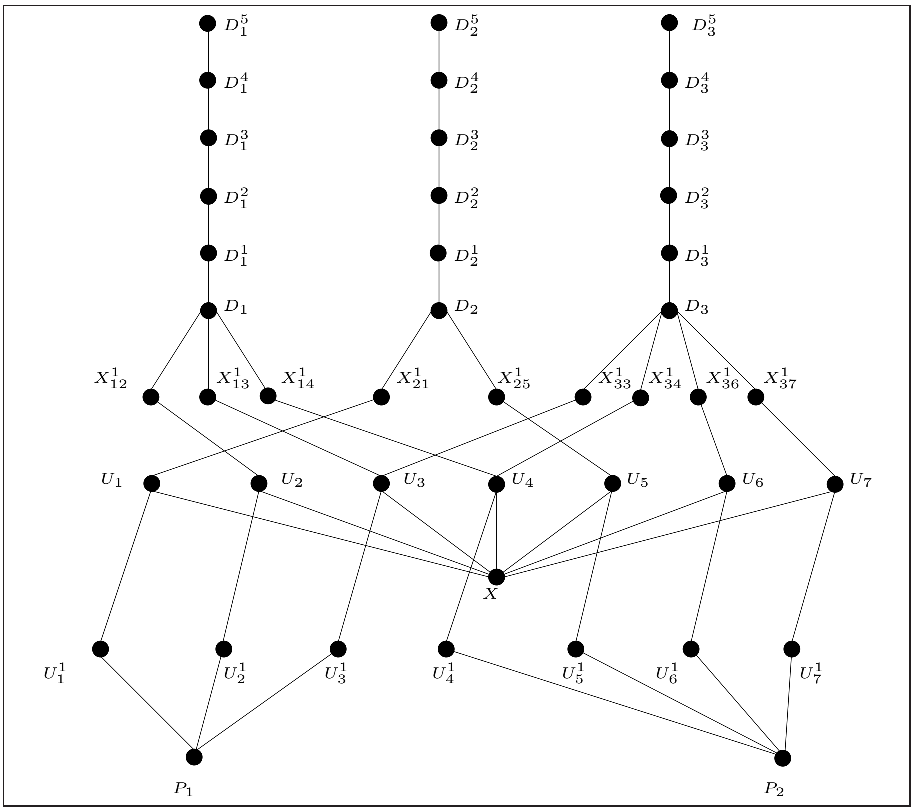

Figure 5.6: The fifth root graph H with girth six of G to the solution S_1, S_2

in $\frac{k-2}{2} + 2$ steps, and X reaches $X_{j'i}^{\frac{k-2}{2}}$ in two steps. Therefore, within k steps, D_j^t reaches $X_{j'i}^{t+2}, \ldots, X_{j'i}^{\frac{k-2}{2}}$ whenever $u_i \in d_{j'}$. Moreover, since U_i reaches $U_{i'}^1$ and $Z_{i'}^1$ within three steps. Thus, within k steps, D_j^t reaches $U_i^1, Z_i^1, \ldots, U_i^{\frac{k-4}{2}-t}, Z_i^{\frac{k-4}{2}-t}$. By comparing with (E_6), $N_{H^k}(D_j^t) = N_G(D_j^t)$.

Next, the vertex D_j, by the construction of H, it is clear that within k steps, D_j reaches X; U_i for all i; $X_{ji}^1, \ldots, X_{ji}^{\frac{k-2}{2}}$; $U_i^1, Z_i^1, \ldots, U_i^{\frac{k-2}{2}}, Z_i^{\frac{k-2}{2}}$ whenever $u_i \in d_j$; $D_{j'}$ whenever $d_j \cap d_{j'} \neq \varnothing$. Also, if $k \geq 6$, within k steps, D_j reaches $U_i^1, Z_i^1, \ldots, U_i^{\frac{k-4}{2}}, Z_i^{\frac{k-4}{2}}$ for all i; $X_{j'i}^2, \ldots, X_{j'i}^{\frac{k-2}{2}}$ for all j'.
Moreover, since we have a solution for SET SPLITTING, $d_j \cap S_1 \neq \varnothing$ and $d_j \cap S_2 \neq \varnothing$, hence, by the construction of H, every D_j must have a $\frac{k}{2}$-th common neighbor with P_1^1, P_1^2, P_2^1 and P_2^2. Thus, within $\frac{k}{2} + \frac{k}{2}$ steps, D_j reaches P_1^1, P_1^2, P_2^1 and P_2^2. By comparing with (E_7) and (E_8), $N_{H^k}(D_j) = N_G(D_j)$.

Now we consider connection vertices and element vertices.

By the construction of H, $U_i \leftrightarrow X$; and X reaches $X_{ji}^1, \ldots, X_{ji}^{\frac{k-2}{2}}$, $U_i^1, Z_i^1, \ldots, U_i^{\frac{k-2}{2}}, Z_i^{\frac{k-2}{2}}$ within $\frac{k}{2}$ steps. Therefore, it is easy to see that $\mathcal{A}$ form

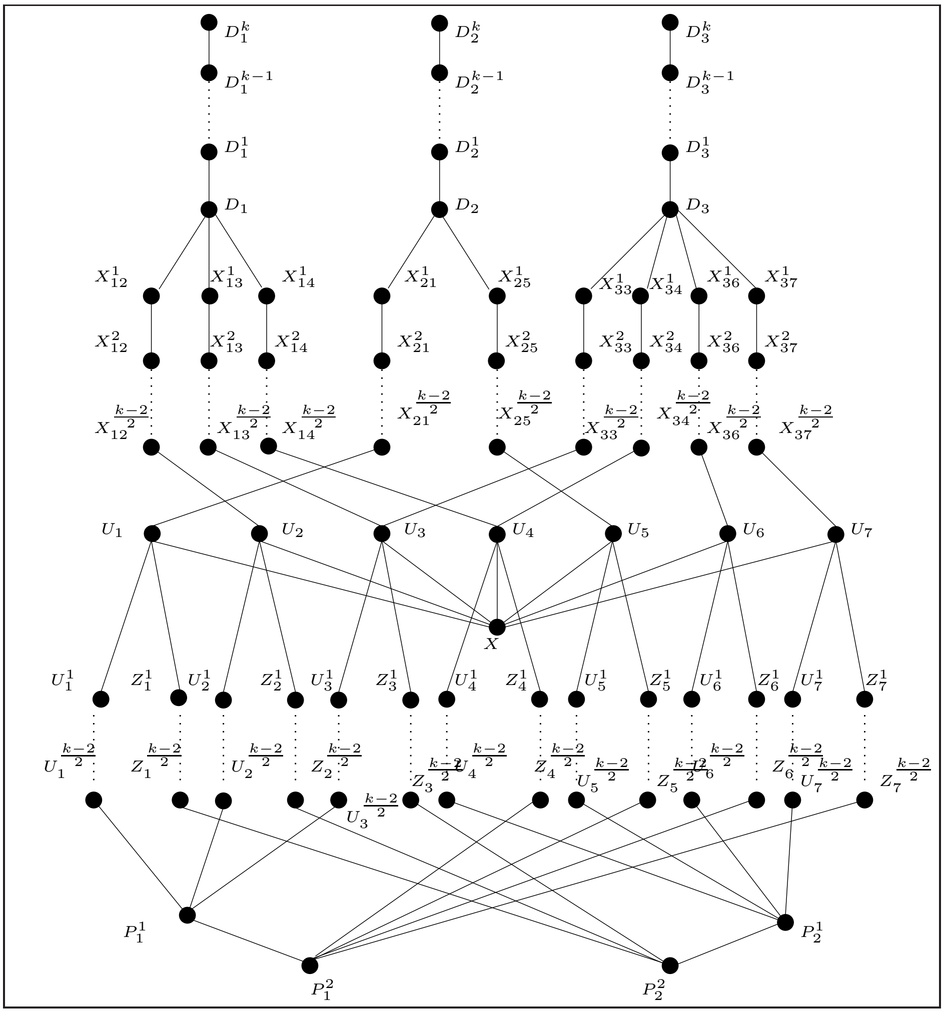

Figure 5.7: A k-th root H with girth $k+2$ for even k in Lemma 5.2.1

a clique in H^k. By comparing with (E_9), for all $v \in \mathcal{A}$, $N_{H^k}(v) = N_G(v)$.

Finally, for partition vertices, by the construction of H, it is clear that P_1^1 adjacent to P_1^2, P_2^2, and P_2^1 adjacent to P_2^2, P_1^2 in H^k.
In H, every one of partition vertices reaches X in $\frac{k}{2}+1$ steps. Hence, within k steps every partition vertices reaches
$\{U_i^h, Z_i^h \mid 1 \le i \le n, 1 \le h \le \frac{k-2}{2}\} \cup \{U_i \mid 1 \le i \le n\}$. Moreover, since we have a solution for SET SPLITTING, within k steps, D_j reaches P_1^1, P_1^2, P_2^1 and P_2^2. Therefore, consider any pair of vertices X_{ji}^h and $X_{ji'}^h$ for $u_i \in d_j$, $u_{i'} \in d_j$, $1 \le h \le \frac{k-2}{2}$, then within $k-1$ steps, every one of P_1^1, P_1^2, P_2^1 and P_2^2 must reach X_{ji}^h or $X_{ji'}^h$. Thus, within k steps every one of P_1^1, P_1^2, P_2^1 and P_2^2 reach both X_{ji}^h and $X_{ji'}^h$. By comparing with (E_{10}), $N_{H^k}(P_1^1) = N_G(P_1^1)$, $N_{H^k}(P_1^2) = N_G(P_1^2)$, $N_{H^k}(P_2^1) = N_G(P_2^1)$ and $N_{H^k}(P_2^2) = N_G(P_2^2)$.

We checked that the edge set of H^k is equal to the edge set of G in *case even k*.

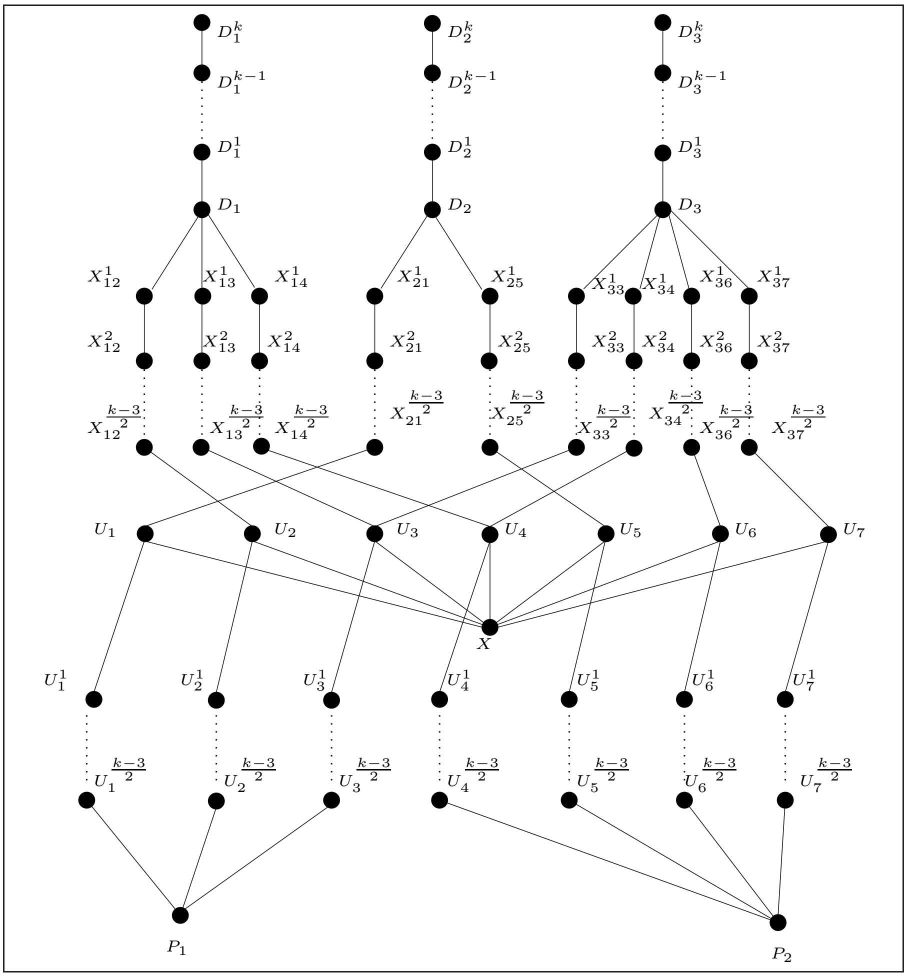

Figure 5.8: A k-th root H with girth $k+1$ for odd k in Lemma 5.2.1

Now we will show that H has girth at most $k+2$. We consider a path in H, say P. By the construction H, it is clear that if P starts froms vertices D_j^t for $1 \leq t \leq k$ then it form no cycle in H.

If the path P starts from D_j, by the construction H, D_j needs at least $\frac{k+2}{2}$ steps to reach $X, D_{j'}, P_1^1, P_1^2, P_2^1, P_2^2$. Thus, in this case, the path P can only form a cycle of length at least $\frac{k+2}{2} + \frac{k+2}{2}$ in H.

If P starts from each of vertices $X_{ji}^h, X, U_i, U_i^h, Z_i^h, P_1^1, P_1^2, P_2^1, P_2^2$, for all i, j and $1 \leq h \leq \frac{k-2}{2}$, then by a similar argument, the path P can only form a cycle of length at least $k+2$ in H. Thus, in all cases, P forms a cycle of length at least $k+2$ in H, i.e., H contains no cycle of length $\leq k+1$. Moreover, observe that H contains a cycle of length $k+2$:
$D_j X_{ji}^1 \ldots X_{ji}^{\frac{k-2}{2}} U_i X U_{i'} X_{ji'}^{\frac{k-2}{2}} \ldots X_{ji'}^1 D_j$ for $u_i \in d_j$ and $u_{i'} \in d_j$. Thus H has girth at most $k+2$.

CASE 2: k **is odd.**

Let H have the same vertex set as G. The edges of H are as follows; see also Figure 5.8.

- Edges of subset vertices and their tail vertices: For all j, $2 \leq t \leq k$, $D_j^t \leftrightarrow D_j^{t-1}$ and $D_j^1 \leftrightarrow D_j$, and $D_j \leftrightarrow \{X_{ji}^1 \mid u_i \in d_j, 1 \leq i \leq n\}$.
- Edges of partition vertices:
  $P_1 \leftrightarrow \{U_i^{\frac{k-3}{2}} \mid u_i \in S_1, 1 \leq i \leq n\}$, and $P_2 \leftrightarrow \{U_i^{\frac{k-3}{2}} \mid u_i \in S_2, 1 \leq i \leq n\}$.
- Edges of connection vertex and element vertices: $X \leftrightarrow \{U_1, \dots, U_n\}$, $U_i \leftrightarrow U_i^1$ for all i.
  For all i, j, $1 \leq h \leq \frac{k-5}{2}$: $X_{ji}^h \leftrightarrow \{X_{ji}^{h+1} \mid u_i \in d_j\}$, and $U_i^h \leftrightarrow U_i^{h+1}$.

By a similar argument to the proof of the case of even k we can show that $H^k = G$ and H has girth at most $k+1$ for odd k.

This completes the proof of Lemma 5.2.1. □

Now we show that if G has a k-th root H (*not necessarily girth* $\leq 2\lfloor \frac{k}{2} \rfloor + 2$), then there is a partition of S into two disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2.

First, by the construction of G, we have for all j:
$N_G(D_j^k) = \{D_j, D_j^1, \dots, D_j^{k-1}\}$, $N_G(D_j^1) \subset N_G[D_j]$ and $N_G(D_j^t) \subset N_G[D_j^{t-1}]$ for all $2 \leq t \leq k-1$. Thus, the following proposition follows immediately from Lemma 3.2.2.

Proposition 5.2.2 *If H is a k-th root of G, then, in H:*

(i) *For all i, j: D_j is only adjacent to X_{ji}^1 whenever $u_i \in d_j$;*

(ii) *D_j^k is only adjacent to D_j^{k-1}, D_j^1 is only adjacent to D_j and D_j^2, and D_j^t is only adjacent to D_j^{t-1} and D_j^{t+1}, $2 \leq t \leq k-1$;*

Now we are ready to prove the reverse direction.

Lemma 5.2.3 *If H is a k-th root of G, then there exists a partition of S into two disjoint subsets S_1 and S_2 such that each subset in D intersects both S_1 and S_2.*

Proof.
CASE 1: k **is even**.

First, we will show that the following claim (5.1) is true.

By Proposition 5.2.2 and the fact that, in G,
$P_1^1 \not\leftrightarrow \{D_j^t \mid 1 \leq t \leq k\}$ and $P_2^1 \not\leftrightarrow \{D_j^t \mid 1 \leq t \leq k\}$, hence, in H, P_1^1 and P_2^1 cannot

reach to vertices of
$\{X_{ji}^h \mid 1 \leq h \leq \frac{k-2}{2}\} \cup \{D_j \mid 1 \leq j \leq m\} \cup \{D_j^t \mid 1 \leq j \leq m, 1 \leq t \leq k\}$ in $\frac{k}{2}$-th step.

Next, we show that $X \notin N_H^{\frac{k}{2}}(P_1^1)$ and $X \notin N_H^{\frac{k}{2}}(P_2^1)$: Otherwise, by in G, X is adjacent to $D_j^{\frac{k-2}{2}}$ but is not adjacent to $D_j^{\frac{k}{2}}$, therefore, $d_H(X, D_j^{\frac{k-2}{2}}) = k$, hence $d_H(X, X_{ji}^1) = \frac{k}{2}$, and so $d_H(P_1^1, X_{ji}^1) = k$ and $d_H(P_2^1, X_{ji}^1) = k$. By Proposition 5.2.2(i), $d_H(P_1^1, D_j) = k+1$ and $d_H(P_2^1, D_j) = k+1$ contradicting to the construction of G.

Finally, if $N_H^{\frac{k}{2}}(P_1^1)$ and $N_H^{\frac{k}{2}}(P_2^1)$ consist of $\{U_i^h, Z_i^h \mid 1 \leq i \leq n, 1 \leq h \leq \frac{k-2}{2}\}$ *only*, by in G, U_i^1 and Z_i^1 are adjacent to $D_j^{\frac{k-2}{2}}$ but is not adjacent to $D_j^{\frac{k}{2}}$, therefore, $d_H(U_i^1, D_j^{\frac{k-2}{2}}) = k$, hence $d_H(U_i^1, X_{ji}^1) = \frac{k}{2}$, and so $d_H(P_1^1, X_{ji}^1) \geq \frac{k}{2} + \frac{k}{2}$ and $d_H(P_2^1, X_{ji}^1) \geq \frac{k}{2} + \frac{k}{2}$. By Proposition 5.2.2(i), $d_H(P_1^1, D_j) \geq k+1$ and $d_H(P_2^1, D_j) \geq k+1$ contradicting to the construction of G.

Therefore,

$$P_1^1 \text{ and } P_2^1 \text{ must have } \frac{k}{2}\text{-th neighbors in the element set } S = \{U_1, \ldots, U_n\}. \quad (5.1)$$

Let $S_1 = N_H^{\frac{k}{2}}(P_1^1) \cap S$ and $S_2 = N_H^{\frac{k}{2}}(P_2^1) \cap S$.

We will show that S_1 and S_2 define a desired partition of element set S.

Claim 1: $S_1 \cap S_2 = \varnothing$.
Proof of Claim 1: Assume contrary that $S_1 \cap S_2 \neq \varnothing$, let $U_t \in S_1 \cap S_2$, in H, the vertices P_1^1 and P_2^1 reach U_t within $\frac{k}{2}$ steps. Thus, in H, P_1^1 reaches P_2^1 within $\frac{k}{2} + \frac{k}{2}$ steps. Therefore, P_1^1 is adjacent to P_2^1 in H^k, contradicting to $P_1^1 \not\leftrightarrow P_2^1$ in $G = H^k$. Thus, S_1 and S_2 will define a partition of element set S.

Claim 2: For all j, $S_1 \cap d_j \neq \varnothing$ and $S_2 \cap d_j \neq \varnothing$.
Proof of Claim 2: Since in G, P_1^1 and P_2^1 are adjacent to all subset vertices D_j but are not adjacent to D_j^1 for all j. Hence, P_1^1 and P_2^1 must reach D_j in *exactly* k steps. Therefore,

$$N_H^{\frac{k}{2}}(P_1^1) \cap N_H^{\frac{k}{2}}(D_j) \neq \varnothing \text{ and } N_H^{\frac{k}{2}}(P_2^1) \cap N_H^{\frac{k}{2}}(D_j) \neq \varnothing. \quad (5.2)$$

Moreover,

$$\text{for all } j,\ N_H^{\frac{k}{2}}(D_j) \text{ consist of element set } \textit{only}: \quad (5.3)$$

Otherwise, if $X_{ji}^h \in N_H^{\frac{k}{2}}(D_j)$ for $1 \leq h \leq \frac{k-2}{2}$, then $d_H(D_j^{\frac{k}{2}}, X_{ji}^h) = \frac{k}{2} + \frac{k}{2}$, hence $d_H(D_j^{\frac{k+2}{2}}, X_{ji}^h) = k+1$, contradicting to the construction of G that $D_j^{\frac{k+2}{2}} \leftrightarrow \{X_{ji}^h \mid 1 \leq h \leq \frac{k-2}{2}\}$.
If $X \in N_H^{\frac{k}{2}}(D_j)$, then $d_H(D_j^{\frac{k}{2}}, X) = k$, contradicting to the construction of G that $D_j^{\frac{k}{2}} \not\leftrightarrow X$.
Finally, if U_i^h or $Z_i^h \in N_H^{\frac{k}{2}}(D_j)$ for $1 \leq h \leq \frac{k-2}{2}$, then $d_H(D_j^{\frac{k}{2}}, U_i^h) = k$ or

$d_H(D_j^{\frac{k}{2}}, Z_i^h) = k$, contradicting to the construction of G that $D_j^{\frac{k}{2}} \not\leftrightarrow \{U_i^h, Z_i^h\}$.

From (5.1), (5.2) and (5.3) we show that $N_H^{\frac{k}{2}}(P_1^1) \cap N_H^{\frac{k}{2}}(D_j) \neq \varnothing$ and $N_H^{\frac{k}{2}}(P_2^1) \cap N_H^{\frac{k}{2}}(D_j) \neq \varnothing$, therefore, $S_1 \cap d_j \neq \varnothing$ and $S_2 \cap d_j \neq \varnothing$ for all j. Thus S_1 and S_2 are actually the desired partition of S.

Case 2: k **is odd**. Similar to the proof of Case 1, we can show that P_1 and P_2 must have $\frac{k-1}{2}$-th neighbors in the element set $S = \{U_1, \ldots, U_n\}$. Moreover, let $S_1 = N_H^{\frac{k-1}{2}}(P_1) \cap S$ and $S_2 = N_H^{\frac{k-1}{2}}(P_2) \cap S$. Then S_1 and S_2 are actually the desired partition of S. □

Notice that in the Lemma 5.2.3, we did not use the property that H has girth at most $2\lfloor\frac{k}{2}\rfloor + 2$. In fact, *any k-th root* of G would tell us how to do SET SPLITTING. In particular, *any k-th root H of G with girth* $\leq 2\lfloor\frac{k}{2}\rfloor + 2$ will do. Hence, from Lemmas 5.2.1 and 5.2.3 we conclude

Theorem 5.2.4 *For any fixed $k \geq 2$, k-TH POWER OF GRAPH WITH GIRTH $\leq 2\lfloor\frac{k}{2}\rfloor + 2$ is NP-complete.*

Furthermore, it is not difficult to show that in the case of odd k, the graph H constructed in the proof of Lemma 5.2.1 is bipartite. Thus we obtain: Given a graph G and odd integer $k \geq 5$, recognizing if G is the k-th power of bipartite graph with girth at most $k + 1$ is NP-complete.

k-TH POWER OF BIPARTITE GRAPH WITH GIRTH $\leq 2\lfloor\frac{k}{2}\rfloor + 2$

Instance:	A graph G.
Question:	Is there a bipartite graph H with girth $\leq 2\lfloor\frac{k}{2}\rfloor + 2$ such that $G = H^k$ for a fixed odd integer $k \geq 5$?

This observation together with the NP-completeness result of cubes of bipartite graphs (cf. [45]), leads to the following conclusion:

Theorem 5.2.5 *For all odd fixed $k \geq 3$, k-TH POWER OF BIPARTITE GRAPH WITH GIRTH $\leq 2\lfloor\frac{k}{2}\rfloor + 2$ is NP-complete.*

5.3 Concluding remarks

We have shown that recognizing squares of graphs with girth at most four is NP-complete. More generally, we have shown that recognizing k-th powers of graphs with girth at most $2\lfloor\frac{k}{2}\rfloor + 2$ is NP-complete.

Note that the reduction for general k cannot apply to $k = 2$ since the graph $G = G(D, S)$ which is constructed in the reduction satisfied no tail structure.

Furthermore, the reductions to solve the problem k-TH POWER OF GRAPH WITH GIRTH $\leq 2\lfloor \frac{k}{2} \rfloor + 2$ also shown that for all odd fixed $k \geq 3$, recognizing k-th powers of bipartite graphs without "large girth" is NP-complete.

The NP-completeness results in this chapter together with polynomial-time results in Chapter 6 and Chapter 7 (see Theorems 6.3.5, 7.2.1) are a first step to classify polynomial and NP-complete cases for recognizing powers of graphs with girth conditions.

Part III

Good Characterizations and Efficient Algorithms

Chapter 6

Squares of Graphs with Girth At Least Six

In this chapter we show that squares of graphs with girth at least six can be recognized in polynomial time. We provide a good characterization for squares of graphs with girth at least seven. This characterization not only leads to a simple algorithm to compute a square root of girth at least 7 but also shows that such a square root, if exists, is unique up to isomorphism.

6.1 Introduction

There are many polynomial time algorithms to compute the tree square roots [49, 42, 45, 15, 17], bipartite graph square roots [45], and proper interval graph square roots [46].

The algorithms of computing tree square roots and bipartite square roots are based on the fact that the square roots have no cycles and odd cycles, respectively. Since computing the graph square uses only local information from the first and the second neighborhood, it is plausible that there are polynomial time algorithms to compute square roots that have no short cycles, and more generally to compute square roots that have no short odd cycles.

From the above idea, we are able to give a characterization and recognition algorithms of graphs that are squares of graphs without short cycles, i.e. to determine if $G = H^2$ for some graph H without short cycles. The main results of this chapter are the following.

In Section 6.2 we will provide a good characterization for graphs that are squares of some graph of girth at least 7. This characterization not only leads to a simple algorithm to compute a square root of girth at least seven but also shows that such a square root, if it exists, is unique up to isomorphism. Furthermore, using this characterization we obtain a new characterization for tree squares that allows us to derive the known results on tree square roots easily. Finally, we will close this section with some further considerations.

In Section 6.3, we will present a polynomial time algorithm to compute a square root of girth at least 6, or report that none exists.

These results together with results in Chapter 5 (Section 5.1) that squares of graphs with girth at most four is NP-complete almost provide a dichotomy theorem for the complexity of the recognition problem in terms of girth of the square roots.

The algorithmic and graph theoretical results considerably generalize previous results on tree square roots. We believe that our algorithms can be extended to compute square roots with no short odd cycles, and in fact one part of the algorithm for computing square roots of girth at least 6 uses only the assumption that the square roots have no 3-cycles or 5-cycles. Coloring properties of squares in terms of girth of the roots have been considered in the literature [5, 24, 37]; our algorithms would allow to apply those results even though a square root was not known a priori.

6.2 Squares of graphs with girth at least seven

In this section, first we give basic properties of squares of graphs with girth at least seven. Next we provide a good characterization of graphs that are squares of a graph of girth at least seven. Our characterization leads to a simple polynomial-time recognition algorithm for such graphs.

6.2.1 Basic facts

Lemma 6.2.1 *Let G be a connected, non-complete graph such that $G = H^2$ for some graph H.*

(i) *If girth$(H) \geq 6$ and v is vertex with* $\deg_H(v) \geq 2$ *then $N_H[v]$ is a maximal clique in G;*

(ii) *If girth$(H) \geq 7$ and Q is a maximal clique in G then $Q = N_H[v]$ for some vertex v where* $\deg_H(v) \geq 2$.

Proof. (i) Let v be a vertex with $\deg_H(v) \geq 2$. Clearly, $Q = N_H[v]$ is a clique in G. Consider an arbitrary vertex w outside Q; in particular, w is non-adjacent in H to v. If w is non-adjacent in H to all vertices in Q, then $d_H(w, v) > 2$. If w is adjacent in H to a vertex $x \in Q - v$, let $y \in Q \setminus \{v, x\}$. Then $N_H[w] \cap N_H[y] = \varnothing$ (otherwise H would contain a cycle of length at most five), hence $d_H(w, y) > 2$. Thus, in any case, w cannot be adjacent, in G, to all vertices in Q. Therefore, Q is a maximal clique in G.

(ii) Let Q be a maximal clique in G. Let $v \in Q$ be a vertex with maximum $|Q \cap N_H[v]|$. We will see that $Q = N_H[v]$ and $\deg_H(v) \geq 2$.

First, we show that $\deg_H(v) \geq 2$. Assume not and let u be the (only) neighbor of v in H. Then, as G is not complete, $\deg_H(u) \geq 2$. Moreover, for every vertex x, if $d_H(x,v) \leq 2$, then $d_H(x,u) \leq 1$. Hence $Q \subseteq N_H[u]$, and by the maximality of Q, $Q = N_H[u]$. But then $|Q \cap N_H[u]| = |N_H[u]| \geq 3 > |Q \cap N_H[v]| = 2$, contradicting to the choice of v.

Now, we show that if $w \in Q \setminus N_H[v]$ and $x \in Q \cap N_H[v]$, then $wx \notin E_H$: As $w \notin N_H[v]$, this is clear in case $x = v$. So, let $x \neq v$ and assume to the contrary that $wx \in E_H$. Then, by the choice of v, there exists a vertex $w' \in Q \setminus N_H[x]$, $w' \in N_H[v]$. Note that $w'x, w'w \notin E_H$ because H has no C_3, C_4. As $ww' \in E_G \setminus E_H$, there exists a vertex $u \notin \{w, w', x, v\}$ with $uw, uw' \in E_H$. But then $H[w, w', x, v, u]$ contains a C_4 or C_5. Contradiction.

Finally, we show that $Q \subseteq N_H[v]$, and by the maximality of Q, $Q = N_H[v]$: Assume not and $w \in Q \setminus N_H[v]$. As $wv \in E_G \setminus E_H$, there exists a vertex x such that $xw, xv \in E_H$, and so, $x \in N_H[v] \setminus Q$. By the maximality of Q, x must be non-adjacent (in G) to a vertex $w' \in Q \setminus N_H[v]$. Since $w'v \in E_G \setminus E_H$, there exists a vertex a such that $aw', av \in E_H$; note that $a \notin \{x, w\}$. Now, if $ww' \in E_H$ then $H[w, w', a, v, x]$ contains a cycle of length at most five. If $ww' \notin E_H$, let b be a vertex such that $bw, bw' \in E_H$; possibly $b = a$. Then $H[w, w', a, b, v, x]$ contains a cycle of length at most six. In any case we have a contradiction, hence $Q \setminus N_H[v] = \varnothing$. □

The 5-cycle C_5 and the 6-cycle C_6 show that (i), respectively, (ii) in Lemma 6.2.1 is best possible with respect to the girth condition of the root. More generally, the maximal cliques in the square of the subdivision of any complete graph on $n \geq 3$ vertices do not satisfy the Condition (ii).
In order to characterize squares of graphs of girth at least seven, we need to define *forced edge* as follows.

Definition 6.2.2 *Let G be an arbitrary graph. An edge of G is called* forced edge *if it is contained in (at least) two distinct maximal cliques in G.*

Proposition 6.2.3 *Let G be a connected, non-complete graph such that $G = H^2$ for some graph H with girth at least seven, and let F be the subgraph of G consisting of all forced edges of G. Then*

- *F is obtained from H by deleting all end-vertices in H;*
- *for every maximal clique Q in G, $F[Q \cap V_F]$ is a star; and*
- *every vertex in $V_G - V_F$ belongs to exactly one maximal clique in G.*

Proof. We first make the following two observations.

1) Consider a forced edge xy in G. Let $Q_1 \neq Q_2$ be two maximal cliques in G containing xy. By Proposition 6.2.1, there exist vertices v_i, $i = 1, 2$, with $\deg_H(v_i) \geq 2$ and $Q_i = N_H[v_i]$. As $Q_1 \neq Q_2$, $v_1 \neq v_2$. As $x, y \in N_H[v_1] \cap N_H[v_2]$ and H has no C_3, C_4, $\{x, y\} = \{v_1, v_2\}$ and $xy = v_1v_2 \in E_H$. Thus, every forced edge xy in G is an edge in H with $\deg_H(x) \geq 2$ and $\deg_H(y) \geq 2$.

2) Let xy be an edge in H. If x or y is an end-vertex in H, then clearly xy belongs to exactly one maximal clique in G, hence xy is not a forced edge in G. If $\deg_H(x) \geq 2$ and $\deg_H(y) \geq 2$, then by Lemma 6.2.1, $N_H[x]$ and $N_H[y]$ are two (distinct) maximal cliques in G containing xy, hence xy is a forced edge in G.

Now, (i) follows directly from the above observations. For (ii), consider a maximal clique Q in G. By Lemma 6.2.1, $Q = N_H[v]$ for some vertex v with $\deg_H(v) \geq 2$. Let X be the set of all neighbors of v in H that are end-vertices in H and $Y = N_H(v) \setminus X$. Since G is not complete, $Y \neq \varnothing$. By (i), $X \cap V_F = \varnothing$, hence $F[Q \cap V_F] = F[\{v\} \cup Y]$ which implies (ii). For (iii), consider a vertex $u \in V_G - V_F$ and a maximal clique Q containing u. Then, u cannot belong to Y and therefore Q is the only maximal clique containing u. □

6.2.2 Good characterizations of squares of graphs with girth at least seven

We now are able to characterize squares of graphs with girth at least seven as follows.

Theorem 6.2.4 *Let G be a connected, non-complete graph. Let F be the subgraph of G consisting of all forced edges in G. Then G is the square of a graph with girth at least seven if and only if the following conditions hold.*

(i) *Every vertex in $V_G - V_F$ belongs to exactly one maximal clique in G.*

(ii) *Every edge in F belongs to exactly two distinct maximal cliques in G.*

(iii) *Every two non-disjoint edges in F belong to a common maximal clique in G.*

(iv) *For each maximal clique Q of G, $F[Q \cap V_F]$ is a star.*

(v) *F is connected and has girth at least seven.*

Proof. For the only if-part, (ii) and (iii) follow easily from Lemma 6.2.1, and (i), (iv) and (v) follow directly from Proposition 6.2.3.

For the if-part, let G be a connected graph satisfying (i) – (v). We will construct a spanning subgraph H of G with girth at least seven such that $G = H^2$ as follows (see also in Figure 6.1).

For each edge xy in F let, by (ii) and (iv), $Q \neq Q'$ be *the* two maximal cliques in G with $Q \cap Q' = \{x, y\}$. Let, without loss of generality, $|Q \cap V_F| \geq |Q' \cap V_F|$.

Assuming x is a center vertex of the star $F[Q \cap V_F]$, then y is a center vertex of the star $F[Q' \cap V_F]$: Otherwise, by (iv), x is the center vertex of the star $F[Q' \cap V_F]$ and there exists some $y' \in Q' \cap V_F$ such that $yy' \notin F$; note that $xy' \in F$ (by (iv)). As $|Q \cap V_F| \geq |Q' \cap V_F|$, there is an edge $xz \in F - xy$ in $Q - Q'$. By (iii), $zy' \in E_G$. Now, as Q' is maximal, the maximal clique Q'' containing x, y, z, y' is different from Q'. But then $\{y, y'\} \subseteq Q' \cap Q''$, i.e., $yy' \in F$, hence F contains a triangle xyy', contradicting (v).

Thus, assuming x is a center vertex of the star $F[Q \cap V_F]$, y is a center vertex of the star $F[Q' \cap V_F]$. Then put the edges xq, $q \in Q - x$, and yq', $q' \in Q' - y$, into H.

By construction, $F \subseteq H \subseteq G$ and by (i),

$$\text{for all vertices } u \in V_H \setminus V_F,\ \deg_H(u) = 1, \tag{6.1}$$

$$\forall v \in V_F, \forall a, b \in V_H \text{ with } va, vb \in E_H:\ a \text{ and } b \text{ belong to the same clique in } G. \tag{6.2}$$

Furthermore, as every maximal clique in G contains a forced edge (by (iv)), H is a spanning subgraph of G. Moreover, F is an induced subgraph of H: Consider an edge $xy \in E_H$ with $x, y \in V_F$. By construction of H, x or y is a center vertex of the star $F[Q \cap V_F]$ for some maximal clique Q in G. Since $x, y \in V_F$, xy must be an edge of this star, i.e., $xy \in E_F$. Thus, F is an induced subgraph of H. In particular, by (6.1) and (v), H is connected and $\text{girth}(H) = \text{girth}(F) \geq 7$.

Now, we complete the proof of Theorem 6.2.4 by showing that $G = H^2$.

Claim 1 $E_G \subseteq E_{H^2}$.
Proof of Claim 1: Let $uv \in E_G \setminus E_H$ and let Q be a maximal clique in G containing uv. By (iv), Q contains a forced edge xy and x or y is a center vertex of the star $F[Q \cap V_F]$. By construction of H, xu and xv, or else yu and yv are edges of H, hence $uv \in E_{H^2}$.

Claim 2 $E_{H^2} \subseteq E_G$.
Proof of Claim 2: Let $ab \in E_{H^2} \setminus E_H$. Then there exists a vertex x such that $xa, xb \in E_H$. By (6.1), $x \in V_F$, and by (6.2), $ab \in E_G$.

It follows by Claims 1 and 2 that $G = H^2$, and Theorem 6.2.4 is proved. □

Corollary 6.2.5 *Given a graph $G = (V_G, E_G)$, it can be recognized in $O(|V_G|^2 \cdot |E_G|)$ time if G is the square of a graph H with girth at least seven. Moreover, such a square root, if any, can be computed in the same time.*

Proof. Note that by Lemma 6.2.1, any square of an n-vertex graph with girth at least seven has at most n maximal cliques. Now, to avoid triviality, assume G is connected and non-complete. We first use the algorithm in [67] to list the maximal cliques in G in time $O(n^2 m)$. If there are more than n maximal cliques, G is not the square of any graph with girth at least seven. Otherwise, compute the forced edges of G to form the subgraph F of G. This can be done in time $O(n^2)$ in an obvious

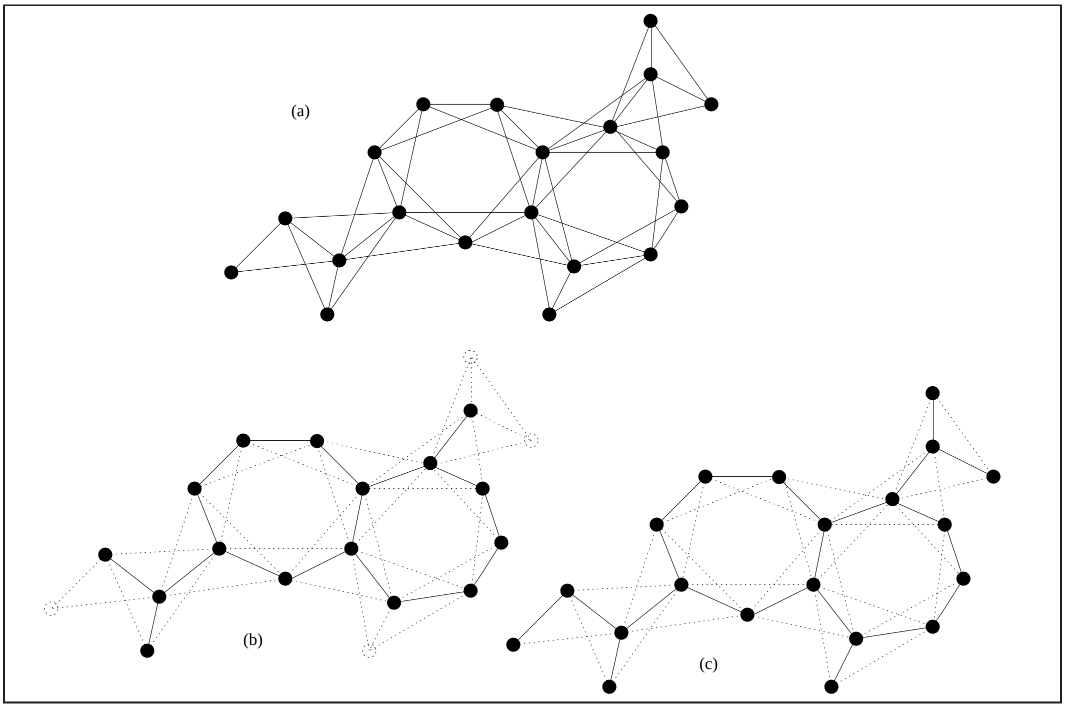

Figure 6.1: An input graph G (a), the subgraph F of G (b) and a square root H (c) constructed according to the proof of Theorem 6.2.4

way (but see also explanation in Corollary 6.2.9).
The conditions (i) – (v) in Theorem 6.2.4 then can be tested within the same time bound, as well as the square root H, in case all conditions are satisfied, according to the proof of Theorem 6.2.4. $\square$

Corollary 6.2.6 *The square roots with girth at least seven of squares of graphs with girth at least seven are unique, up to isomorphism.*

Proof. Let G be the square of some graph H with girth ≥ 7. If G is complete, clearly, every square root with girth ≥ 6 of G must be isomorphic to the star $K_{1,n-1}$ where n is the vertex number of G. Thus, let G be non-complete, and let F be the subgraph of G formed by the forced edges. If F has only one edge, G clearly consists of exactly two maximal cliques, Q_1, Q_2, say, and $Q_1 \cap Q_2$ is the only forced edge of G. Then, it is easily seen that every square root with girth ≥ 6 of G must be isomorphic to the double star T having center edge v_1v_2 and $\deg_T(v_i) = |Q_i|$.

So, assume F has at least two edges. Then for each two maximal cliques Q, Q' in G with $Q \cap Q' = \{x, y\}$, x or y is the unique center vertex of the star $F[V_F \cap Q]$ or $F[V_F \cap Q']$. Hence, for any end-vertex u of H, i.e., $u \in V_G - V_F$, the neighbor of u in F is unique. Since F is the graph resulting from H by deleting all end-vertices, H is therefore unique. $\square$

6.2.3 Squares of trees revisited

Squares of trees have been widely discussed in the literature. Using the results in the Subsection 6.2.1 and Subsection 6.2.2, we obtain a new characterization for tree squares that allow us to derive the known results on tree square roots easily.

Observe that the proof of Theorem 6.2.4 shows that if F is a tree, then also the square root H is a tree. This fact and Lemma 6.2.1 and 6.2.3 immediately imply the following good characterization for squares of trees in terms of forced edges. Recall Definition 6.2.2 in Subsection 6.2.1 for the notion of forced edges in a graph.

Theorem 6.2.7 *Let G be a connected, non-complete graph. Let F be the subgraph of G consisting of all forced edges in G. Then G is the square of a tree if and only if the following conditions hold.*

(i) *Every vertex in $V_G - V_F$ belongs to exactly one maximal clique in G;*

(ii) *Every edge in F belongs to exactly two distinct maximal cliques in G;*

(iii) *Every two non-disjoint edges in F belong to a common maximal clique in G;*

(iv) *For each maximal clique Q of G, $F[Q \cap V_F]$ is a star;*

(v) *F is a tree.*

The class of strongly chordal graphs (refer to Lemma 8.1.1) is closed under powers. It is clear that trees are strongly chordal. In [25, 52, 61] it was shown that the square of a tree is strongly chordal; later, not knowing this fact, [49, 3] proved that the square of a tree is chordal. Our characterization of tree squares, Theorem 6.2.7, give a new and short proof for this fact:

Corollary 6.2.8 ([25, 52, 61]) *Squares of trees are strongly chordal.*

Proof. Let G be a non-complete graph that is the square of a tree, and let F be the forced subgraph of G. Then F satisfies (i) – (v) in Theorem 6.2.7. In particular, G cannot contain an induced sun otherwise F would contain a cycle, contradicting (v). Now, assume $v_1 v_2 \dots v_\ell v_1$ is an induced cycle in G with $\ell \geq 4$. Consider the maximal cliques Q_i in G containing the edge $v_i v_{i+1}$, $1 \leq i \leq \ell$ (modulo ℓ). Note that the Q_is are pairwise distinct, hence by (i), $v_i \in V_F$. Thus, with (iv), $F[Q_i \cap V_F]$ is a star containing v_i and v_{i+1}, $1 \leq i \leq \ell$, implying F contains a cycle; a contradiction to (v). □

Corollary 6.2.9 ([15, 17, 42, 45, 49]) *Given a graph $G = (V_G, E_G)$, it can be recognized in $O(|V_G| + |E_G|)$ time if G is the square of a tree. Moreover, a tree root of a square of a tree can be computed in the same time.*

Proof. In order to obtain linear time, we use Corollary 6.2.8 saying that squares of trees are chordal, and that all maximal cliques of a chordal graph can be computed in linear time (see, for example, [33]).

Thus, given $G = (V_G, E_G)$, we may assume that G is chordal and all maximal cliques of G are available. To detect all forced edges in G, create for each edge e of G a linked list $L(e)$ consisting of all maximal cliques in G that contain e: Scan each maximal clique Q_i and for each edge e_j in Q_i add Q_i to $L(e_j)$; this can be done in time $O(n + m)$. If $|L(e)| \geq 3$ for some edge e, then (i) fails, and G is not the square of a tree. So, let $|L(e)| \leq 2$ for all edges e, and F consists of all edges e with $|L(e)| = 2$. Clearly, F can be obtained in $O(m)$ time, and (ii) – (iv) can be tested in $O(n + m)$ time. □

Corollary 6.2.10 ([15, 45, 63]) *The tree roots of squares of trees are unique, up to isomorphism.*

Proof. By Corollary 6.2.6 □

Finally, we note that the following new and good characterization for squares of trees has been shown in [15], which also easily follows from our Theorem 6.2.7.

Theorem 6.2.11 ([15]) *Let G be a connected, non-complete graph. Then G is the square of a tree if and only if G is chordal, 2-connected, and has the following properties:*

(i) *Every two distinct maximal cliques have at most two vertices in common;*

(ii) *Every 2-cut belongs to exactly two maximal cliques of G;*

(iii) *Every pair of non-disjoint 2-cuts belongs to the same maximal clique; and*

(iv) *All 2-cuts contained in the same maximal clique of G have a common vertex.*

6.2.4 Further considerations

Squares of bipartite graphs can be recognized in $O(\Delta \cdot M(n))$ time in [45], where $\Delta = \Delta(G)$ is the maximum degree of the n-vertex input graph G and $M(n)$ is the time needed to perform the multiplication of two $n \times n$-matrices. However, no good characterization is known so far. As bipartite graphs with girth at least seven are exactly the (C_4, C_6)-free bipartite graphs, we immediately have:

Corollary 6.2.12 *Let G be a connected, non-complete graph. Let F be the subgraph of G consisting of all forced edges in G. Then G is the square of a (C_4, C_6)-free bipartite if and only if the following conditions hold.*

(i) *Every vertex in $V_G - V_F$ belongs to exactly one maximal clique in G;*

(ii) *Every edge in F belongs to exactly two distinct maximal cliques in G;*

(iii) *Every two non-disjoint edges in F belong to a common maximal clique in G;*

(iv) *For each maximal clique Q of G, $F[Q \cap V_F]$ is a star;*

(v) *F is a connected (C_4, C_6)-free bipartite.*

Moreover, squares of (C_4, C_6)-free bipartite graphs can be recognized in $O(n^2m)$ time, and the (C_4, C_6)-free square bipartite roots of such squares are unique, up to isomorphism.

Furthermore, results in the Subsection 6.2.1 and Subsection 6.2.2 also allow us to consider the computational complexity of some optimization problems on the class of squares of graphs with girth at least 7.

The following problems belong to the most basic NP-complete problems ([32, Problems GT19, GT20]):

CLIQUE

Instance: A graph $G = (V, E)$ and an integer k.
Question: Is there a clique in G with at least k vertices ?

STABLE SET

Instance: A graph G and an integer k.
Question: Is there a stable set in G with at least k vertices ?

It was shown in [49] that CLIQUE and STABLE SET remain NP-complete on the graph-class squares of graphs (of girth three). Another consequence of our results is:

Corollary 6.2.13 *The weighted version of* CLIQUE *can be solved in $O(n^2m)$ time on squares of graphs with girth at least seven, where n and m are the number of vertices, respectively, edges of the input graph.*

Proof. Let $G = (V_G, E_G)$ be the square of some graph with girth at least seven. By Lemma 6.2.1, G has $O(|V_G|)$ maximal cliques. By using the algorithm in [67], all maximal cliques in G can be listed in time $O(|V_G| \cdot |E_G| \cdot |V_G|)$. □

A graph H is a *subdivision* of G if it is obtained from G by replacing each edge of G by a path of length 2.

In [38], it was shown that STABLE SET is even NP-complete on squares of the subdivision of some graph. As the subdivision of a graph has girth at least six, STABLE SET is NP-complete on squares of graphs with girth at least six.

6.3 Squares of graphs with girth at least six

The main result of this section is Theorem 6.3.5, which shows that squares of graphs with girth at least six can be recognized efficiently. Formally, we will show that the following problem

SQUARE OF GRAPH WITH GIRTH AT LEAST SIX

Instance: A graph G.
Question: Is there a graph H with girth at least 6 such that $G = H^2$?

is polynomially solvable (Theorem 6.3.5).

Similar to the algorithm in [45], our recognition algorithm consists of two steps. The first step (Subsection 6.3.1) is to show that if we fix a vertex $v \in V$ and a subset $U \subseteq N_G(v)$, then there is at most one $\{C_3, C_5\}$-free (locally bipartite) square root graph H of G with $N_H(v) = U$. Then, in the second step (Subsection 6.3.2), we show that if we fix an edge $e = uv \in E_G$, then there are at most two possibilities of $N_H(v)$ for a square root H with girth at least 6. Furthermore, both steps can be implemented efficiently, and thus it will imply that SQUARE OF GRAPH WITH GIRTH AT LEAST SIX is polynomially solvable.

6.3.1 Square root with a specified neighborhood

This subsection deals with the first auxiliary problem.

$\{C_3, C_5\}$-FREE SQUARE ROOT WITH A SPECIFIED NEIGHBORHOOD

Instance: A graph G, $v \in V_G$ and $U \subseteq N_G(v)$.
Question: Is there a $\{C_3, C_5\}$-free graph H such that $H^2 = G$ and $N_H(v) = U$?

An efficient recognition algorithm for $\{C_3, C_5\}$-FREE SQUARE ROOT WITH A SPECIFIED NEIGHBORHOOD relies on the following fact.

Lemma 6.3.1 *Let $G = H^2$ for some $\{C_3, C_5\}$-free graph H. Then, for all vertices $x \in V$ and all vertices $y \in N_H(x)$, $N_H(y) = N_G(y) \cap \big(N_G[x] \setminus N_H(x)\big)$.*

Proof. First, consider an arbitrary vertex $w \in N_H(y) - x$. Clearly, $w \in N_G(y)$, as well as $w \in N_G(x)$. Also, since H is C_3-free, $wx \notin E_H$. Thus $w \in N_G(y) \cap \big(N_G(x) \setminus N_H(x)\big)$.

Conversely, let w be an arbitrary vertex in $N_G(y) \cap \big(N_G[x] \setminus N_H(y)\big)$. Assuming $wy \notin E(H)$, then $w \neq x$ and there exist vertices z and z' such that $zx, zw \in E(H)$ and $z'y, z'w \in E_H$. As H is C_3-free, $zy \notin E_H$, $z'x \notin E_H$, and $zz' \notin E_H$. But then x, y, w, z and z' induce a C_5 in H, a contradiction. Thus $w \in N_H(y)$. □

Recall that $M(n)$ stands for the time needed to perform a matrix multiplication of two $n \times n$ matrices; currently, $M(n) = O(n^{2.376})$.

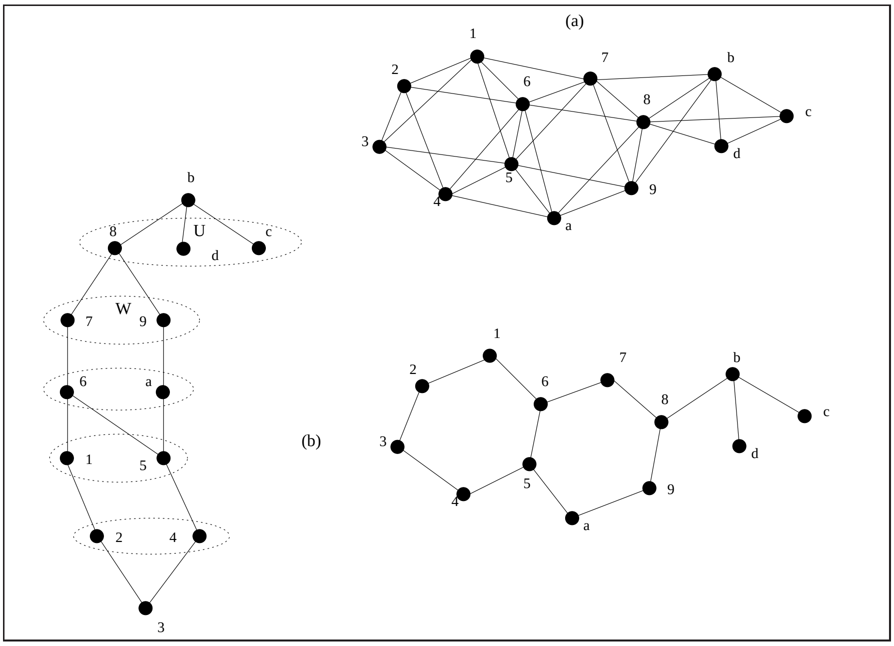

Figure 6.2: An input graph G (a) with the specified neighborhood $U = \{8, c, d\} \subseteq N_G(b)$; a (C_3, C_5)-free square root H (b) constructed by Algorithm 6.3.1: `c3c5-free`

Theorem 6.3.2 $\{C_3, C_5\}$-FREE SQUARE ROOT WITH A SPECIFIED NEIGHBORHOOD *has at most one solution. The unique solution, if any, can be constructed in time $O(M(n))$.*

Proof. Given G, $v \in V_G$ and $U \subseteq N_G(v)$, and assume H is a $\{C_3, C_5\}$-free square root of G such that $N_H(v) = U$. Then, by Lemma 6.3.1, the neighborhood in H of each vertex $u \in U$ is uniquely determined by $N_H(u) = N_G(u) \cap \big(N_G[v] \setminus U\big)$. By repeatedly applying Lemma 6.3.1 for each $v' \in U$ and $U' = N_H(v')$ and noting that all considered graphs are connected, we can conclude that H is unique.

Lemma 6.3.1 also suggests the following BFS-like procedure, Algorithm 6.3.1: `c3c5-free` below, for constructing the $\{C_3, C_5\}$-free square root H of G with $U = N_H(v)$, if it exists (see also Figure 6.2).

It can be seen, by construction, that H is $\{C_3, C_5\}$-free, and thus the correctness of Algorithm 6.3.1: `c3c5-free` follows from Lemma 6.3.1. Moreover, since every vertex is enqueued at most once, lines 1–13 take $O(m)$ steps for $m = |E_G|$. Checking if $G = H^2$ (line 14) takes $O(M(n))$ steps for $n = |V_G|$. $\square$

Algorithm 6.3.1: `c3c5-free`

Input: A graph G, a vertex $v \in V_G$ and a subset $U \subseteq N_G(v)$.
Output: A $\{C_3, C_5\}$-free graph H with $H^2 = G$ and $N_H(v) = U$, or else 'NO' if such a square root H of G does not exist.

Add all edges vu, $u \in U$, to E_H
$Q \leftarrow \varnothing$
for each $u \in U$ **do**
  enqueue(Q, u)
  parent$(u) \leftarrow v$
while $Q \neq \varnothing$ **do**
  $u \leftarrow$ dequeue(Q)
  set $W := N_G(u) \cap \big(N_G(\text{parent}(u)) \setminus N_H(\text{parent}(u))\big)$
  for each $w \in W$ **do**
    add uw to E_H
    if parent$(w) = \varnothing$
    then parent$(w) \leftarrow u$
      enqueue(Q, w)
if $G = H^2$ **then return** H
  else return 'NO'

6.3.2 Square root with a specified edge

This subsection discusses the second auxiliary problem.

GIRTH ≥ 6 ROOT GRAPH WITH ONE SPECIFIED EDGE

Instance: A graph G and an edge $xy \in E_G$.
Question: Is there a graph H with girth at least six such that $H^2 = G$ and $xy \in E_H$?

The question is easy if $\mid G \mid \leq 2$. So, for the rest of this section, assume that $\mid G \mid > 2$. Then, we will reduce this problem to $\{C_3, C_5\}$-FREE SQUARE ROOT WITH A SPECIFIED NEIGHBORHOOD. Given a graph G and an edge xy of G, write $C_{xy} = N_G(x) \cap N_G(y)$, i.e., C_{xy} is the set of common neighbors of x and y in G.

Lemma 6.3.3 *Suppose H is of girth at least 6, $xy \in E_H$ and $H^2 = G$. Then $G[C_{xy}]$ has at most two connected components. Moreover, if A and B are the connected components of $G[C_{xy}]$ (one of them maybe empty) then*

(i) *$A = N_H(x) - y$ and $B = N_H(y) - x$, or*

(ii) *$B = N_H(x) - y$ and $A = N_H(y) - x$.*

Proof. Set $X = N_H(x) - y$ and $Y = N_H(y) - x$. Notice that X or Y (but not both) maybe empty. First we show that $X \cup Y = C_{xy}$. Consider an arbitrary vertex $v \in C_{xy}$; we claim that v is either in X or Y: Otherwise, there is a length 2 path from v to x and a length 2 path from v to y, which implies that there is either a 3-cycle or a 5-cycle, a contradiction. So we have $C_{xy} \subseteq X \cup Y$.

On the other hand, consider an arbitrary vertex $u \in X$, it is obvious that $u \in N_{H^2}(x)$. Also, since $xy \in E(H)$, $u \in N_{H^2}(y)$. A similar argument applies if $u \in Y$. Therefore, $u \in N_{H^2}(x) \cap N_{H^2}(y)$. Since $H^2 = G$, $u \in C_{xy}$. Hence $X \cup Y = C_{xy}$.

Next, observe that X and Y induce cliques in H^2 and thus in G. Moreover, $X \cap Y = \varnothing$ (as H has no 3-cycle) and no vertex in X is adjacent in H to a vertex in Y (as H has no 4-cycle). Now, no vertex $u \in X$ is adjacent in G to a vertex $w \in Y$: Otherwise, there is a vertex $v \notin X \cup Y$ adjacent in H to u and to w, implying that x, y, u, w, v induce a 5-cycle in H, a contradiction.

Thus, the cliques $G[X]$ and $G[Y]$ are exactly the connected components of $G[C_{xy}]$ and the lemma follows. □

By Lemma 6.3.3, we can solve GIRTH ≥ 6 ROOT GRAPH WITH ONE SPECIFIED EDGE as follows:

Compute C_{xy}. If $G[C_{xy}]$ has more than two connected components, there is no solution. If $G[C_{xy}]$ is connected, solve $\{C_3, C_5\}$-FREE SQUARE ROOT WITH A SPECIFIED NEIGHBORHOOD for inputs
$I_1 = (G, v = x, U = C_{xy} + y)$ and $I_2 = (G, v = y, U = C_{xy} + x)$.
If, for I_1 or I_2, Algorithm 6.3.1: `c3c5-free` outputs H and H is C_4-free, then H is a solution. In the other cases there is no solution.
If $G[G_{xy}]$ has two connected components, A and B, solve $\{C_3, C_5\}$-FREE SQUARE ROOT WITH A SPECIFIED NEIGHBORHOOD for inputs
$I_1 = (G, v = x, U = A + y)$, $I_2 = (G, v = x, U = B + y)$, $I_3 = (G, v = y, U = A + x)$, $I_4 = (G, v = y, U = B + x)$, and make a decision similar as before. In this way, checking if a graph H is C_4-free is the most expensive step, and we obtain

Theorem 6.3.4 GIRTH ≥ 6 ROOT GRAPH WITH ONE SPECIFIED EDGE *can be solved in time $O(n^4)$.* □

Let $\delta = \delta(G)$ denote the minimum vertex degree in G. Now we can state the main result of this section as follows.

Theorem 6.3.5 SQUARE OF GRAPH WITH GIRTH AT LEAST SIX *can be solved in time $O(\delta \cdot n^4)$.*

Proof. Given G, let x be a vertex of minimum degree in G. For each vertex $y \in N_G(x)$ check if the instance $(G, xy \in E_G)$ for GIRTH ≥ 6 ROOT GRAPH WITH ONE SPECIFIED EDGE has a solution. □

6.4 Concluding remarks

We have shown that squares of graphs with girth at least six can be recognized in polynomial time (Theorem 6.3.5).

We have found a good characterization for squares of graphs with girth at least seven that gives a faster recognition algorithm in this case (Theorem 6.2.4).

These results together with results in Chapter 5 (Theorem 5.1.3) that squares of graphs with girth at most four is NP-complete, almost provide a dichotomy theorem for the complexity of the recognition problem in terms of girth of the square roots.

However, the following interesting question is still open.

Problem 6.4.1 *What is the complexity status of computing square root with girth (exactly) five ?*

However, we believe that this problem should be efficiently solvable. Also, we believe that the algorithm to compute a square root of girth 6 can be extended to compute a square root with no C_3 or C_5.

More generally, let k be a positive integer and consider the following problem.

k-TH POWER OF GRAPH WITH GIRTH $\geq 3k - 1$

Instance: A graph G.
Question: Is there a graph H with girth $\geq 3k - 1$ such that $G = H^k$?

Then the following question remains open.

Problem 6.4.2 *What is the complexity status of* k-TH POWER OF GRAPH WITH GIRTH $\geq 3k - 1$ *?*

Although we did not solve this problem, we hope that the same approach would work. So, we believe that k-TH POWER OF GRAPH WITH GIRTH $\geq 3k - 1$ is polynomial time solvable.

Conjecture 6.4.3 k-TH POWER OF GRAPH WITH GIRTH $\geq 3k - 1$ *is polynomially solvable.*

Moreover, in the case that the k-th root has girth at least $3k + 1$, we strongly believe that the methods used to characterize for squares of graphs with girth at least seven (cf. Section 6.2) can apply to give a good characterization for k-th powers of graphs with girth at least $3k+1$. In fact, we have done this for $k = 3$ (cf. Chapter 7, Section 7.2).

The truth of Conjecture 6.4.3 together with the results in this chapter would imply a complete dichotomy theorem: SQUARES OF GRAPHS OF GIRTH g is polynomial if $g \geq 5$ and NP-complete otherwise.

Chapter 7

Cubes of Graphs with Girth At Least Ten

In this chapter we give a good characterization of graphs that are cubes of a graph having girth at least 10. Our characterization leads to an $O(nm^2)$-time recognition algorithm for such graphs. Moreover, this algorithm constructs a cube root of girth at least 10 if it exists.

7.1 Basic facts

In this section, we give basic properties of cubes of graphs with girth at least ten. The following fact is the key observation for further discussions.

Lemma 7.1.1 *Let $G = (V, E_G)$ be a connected, non-complete graph such that $G = H^3$ for some graph $H = (V, E_H)$ with girth at least 10. Then $Q \subseteq V$ is a maximal clique in G if and only if $Q = N_H[u, v]$ for some edge $uv \in E_H$ with $\deg_H(u) \geq 2$ and $\deg_H(v) \geq 2$.*

Proof. First, consider an edge uv in H with $\deg_H(u) \geq 2$ and $\deg_H(v) \geq 2$. Obviously, $N_H[u, v]$ is a clique in G. This clique is indeed maximal: Otherwise there exists a vertex $w \in V \setminus N_H[u, v]$ adjacent in G to all vertices in $N_H[u, v]$. Let P be a shortest path in H connecting w and u. If $v \in P$, let x be an arbitrary vertex in $N_H(u) - v$; otherwise let x be any vertex in $N_H(v) - u$. Let P' be a shortest path in H connecting w and x. Note that both P and P' have length at most 3. But then the subgraph of H formed by u, v, P and P' contains a cycle of length at most 8, a contradiction.

Next, let Q be a maximal clique in G. Observe that

for all $u, v \in Q$, there exists exactly one shortest u, v-path in H.

(As $u, v \in Q$, any shortest u, v-path in H has length at most 3. Thus, by the girth condition on H, such u, v-path must be unique.) Moreover,

> for every two vertices $u, v \in Q$, all vertices on the shortest u, v-path P in H belong to Q.

(This is clear if any vertex of Q is on P. So, let w be an arbitrary vertex of Q outside P, and consider shortest paths P', P'' in H connecting u and w, respectively, v and w. Then $H[P \cup P' \cup P'']$ is a tree, otherwise it contains a cycle of length at most 9. Hence every vertex $x \in P$ must belong to P' or to P'', implying $d_H(x, w) \leq 3$. Thus, by the maximality of Q, $x \in Q$.) Furthermore,

> for all $u, v \in Q$, no u, v-path in H of length at least 4 belongs to $H[Q]$.

(If $x_0 x_1 \ldots x_r$ is a path in $H[Q]$ connecting $x_0 = u$ and $v = x_r$, $r \geq 4$, then the subgraph of H formed by x_0, x_1, x_2, x_3, x_4 and the shortest x_0, x_4-path in H would contain a cycle of length at most 7.)

It follows from the facts above that $H[Q]$ is connected and any two vertices in Q are connected by exactly one path in $H[Q]$ and thus $H[Q]$ is a tree. The tree $H[Q]$ must have diameter 3 because $G \neq Q$ is connected and Q is a maximal clique. It follows, by the maximality of Q, that $Q = N_H[u, v]$ for an edge uv in H with $\deg_H(u) \geq 2$ and $\deg_H(v) \geq 2$. □

The example of the cube of the 9-cycle C_9 shows that Lemma 7.1.1 is best possible with respect to the girth condition of the root. Lemma 7.1.1 immediately implies:

Corollary 7.1.2 *If $G = (V_G, E_G)$ is the cube of some graph with girth at least* 10*, then G has at most $|E_G|$ maximal cliques.*

We now introduce the main notion in this chapter.

Definition 7.1.3 *Let G be an arbitrary graph. An edge e of G is called* forced *if e is the intersection of two distinct maximal cliques in G.*

The meaning of forced edges is that if the graph considered is the cube of some graph with girth at least 10, then its forced edges must belong to the edge set of any cube root with such girth condition.

Observation 7.1.4 *Let $G = H^3$ for some graph H with girth at least* 10*. Then, an edge of G is forced if and only if it is the mid-edge of a P_6 in H.*

Proof. Let $xy \in E_G$ be a forced edge, and let Q and Q' be two maximal cliques of G such that $Q \cap Q' = \{x, y\}$. By Lemma 7.1.1, $H[Q] = N_H[u, v]$ and $H[Q'] = N_H[u', v']$ for some edges $uv \neq u'v' \in E_H$. If x and y are nonadjacent in H then two x, y-paths, one in $N_H[u, v]$ and one in $N_H[u', v']$, form a cycle of length at most six, a contradiction. Thus, $xy \in E_H$, and it follows that xy is the mid-edge of a P_6 in $N_H[u,v] \cup N_H[u',v']$.

Conversely, if $abxycd$ is a P_6 in H, then $N_H[b,x] \cap N_H[c,y] = \{x,y\}$, hence, by Lemma 7.1.1, xy is a forced edge of G. $\square$

So, cubes of graphs with large girth but without long induced paths do not contain forced edges. Such graphs are trivial in the following sense.

Definition 7.1.5 *A connected graph G is said to be* trivial *if it contains a non-empty clique C such that $G \setminus C$ is the disjoint union of at most $|C|-1$ cliques and every vertex in C is adjacent to every vertex in $G \setminus C$.*

Observation 7.1.6

(i) *A graph is trivial if and only if it is the cube of some tree of diameter at most 4*

(ii) *Trivial graphs can be recognized in linear time.*

Proof. (i): The case of complete graphs, respectively, of trees of diameter at most 3 is clear. Let G be a trivial non-complete graph and C be a clique of G with the properties given in Definition 7.1.5. Let $S_1, \ldots, S_p$ be the connected components of $G \setminus C$, $1 \le p \le |C|-1$. For each S_i choose a vertex $v_i \in C$ such that $v_i \ne v_j$ for $i \ne j$ and fix a vertex $v \in C \setminus \{v_1, \ldots, v_p\}$. Consider the tree T with the same vertex set V_G and edge set

$$\big\{vw \mid w \in C \setminus \{v, v_1, \ldots, v_p\}\big\} \cup \{vv_i \mid 1 \le i \le p\} \cup \{v_i w \mid w \in S_i, 1 \le i \le p\}.$$

Then T has diameter 4 and $G = T^3$. Conversely, let T be a tree of diameter 4. Then the center of T consists of exactly one vertex, say v. Then, clearly, $G = T^3$ is a trivial graph with clique $C = N_T(v)$.

(ii): By standard arguments, the set of universal vertices C of a given graph G can be computed in linear time, as well as computing the connected components of $G \setminus C$ and testing if a vertex set forms a clique. $\square$

By Observation 7.1.6, we need to consider non-trivial graphs only. A *star* is a tree with at least two vertex and diameter at most two. Let $\mathcal{C}(G)$ denote the set of all maximal cliques of G and for an edge e, let $\mathcal{C}_e$ denote the set of all maximal cliques containing e.

Proposition 7.1.7 *Let G be a connected, non-trivial graph such that $G = H^3$ for some graph H with girth at least* 10*, and let F be the subgraph of G consisting of all forced edges of G. Then*

(i) *F is a connected induced subgraph of H;*

(ii) *For each* $e \in F$*, there exists a unique maximal clique* $Q_e \in \mathcal{C}_e$ *such that*

(a) *for every two distinct non-disjoint forced edges* e *and* e', $e \cup e' \subseteq Q_e \cap Q_{e'}$,

(b) *for every* $Q \in \mathcal{C}(G) \setminus \{Q_e \mid e \in F\}$*, and for all forced edges* e_1, e_2 *in* Q, $Q_{e_1} \cap Q = Q_{e_2} \cap Q$;

(iii) *For each* $e \in F$, $\mathcal{C}_e \setminus \{Q_e\}$ *can be partitioned into non-empty disjoint sets* $\mathcal{A}_e$ *and* $\mathcal{B}_e$ *with*

(a) $Q \cap Q' = e$ *if and only if* $Q \in \mathcal{A}_e$ *and* $Q' \in \mathcal{B}_e$ *or vice versa,*

(b) *setting* $A_e = \bigcap_{Q \in \mathcal{A}_e} Q$, $B_e = \bigcap_{Q \in \mathcal{B}_e} Q$*, all pairs of maximal cliques in* $\mathcal{A}_e$ *have the same intersection* A_e*, all pairs of maximal cliques in* $\mathcal{B}_e$ *have the same intersection* B_e,

(c) $Q_e = A_e \cup B_e$*, and* $|A_e| \geq |\mathcal{A}_e| + 2$, $|B_e| \geq |\mathcal{B}_e| + 2$,

(d) $F[A_e \cap V_F]$ *and* $F[B_e \cap V_F]$ *are stars with distinct universal vertices in* e;

(iv) $\mathcal{C}(G) = \bigcup_{e \in F} \mathcal{C}_e$;

(v) $V_G \setminus \bigcup_{e \in F} Q_e$ *consists of exactly the simplicial vertices of* G.

Proof.
(i): We claim that, for all $u, v \in V_F$, every u,v-path P in H belongs to F: By definition of F, there are $u', v' \in V_F$ (possibly $u' = v'$) such that $uu', vv' \in E_F$. By Observation 7.1.4, uu' and vv' are P_6 mid-edges in H. From this fact and the girth condition on H it is easily seen that every edge of P is a mid-edge of some P_6 in H, hence $P \subseteq F$.

For (ii) – (v), consider an arbitrary forced edge $e = xy$. By Observation 7.1.4, $xy \in E_H$ and $\deg_H(x) \geq 2$, $\deg_H(y) \geq 2$. Set

$$
\begin{aligned}
Q_e &:= N_H[x, y],\\
\mathcal{A}_e &:= \{Q \subseteq V_G \mid Q = N_H[x, x'], x' \in N_H(x) - y, \deg_H(x') \geq 2\},\\
\mathcal{B}_e &:= \{Q \subseteq V_G \mid Q = N_H[y, y'], y' \in N_H(y) - x, \deg_H(y') \geq 2\}.
\end{aligned}
$$

Note that $\mathcal{A}_e$ and $\mathcal{B}_e$ are nonempty because xy is mid-edge of a P_6 in H, and by Lemma 7.1.1, $\mathcal{C}_e = \mathcal{A}_e \cup \{Q_e\} \cup \mathcal{B}_e$.

Now, (ii) (a) follows directly from the definition of Q_e. Furthermore, $Q_1 \cap Q_2 = N_H[x]$ for all $Q_1 \neq Q_2 \in \mathcal{A}_e$, $Q_1' \cap Q_2' = N_H[y]$ for all $Q_1' \neq Q_2' \in \mathcal{B}_e$, $Q_e = N_H[x] \cup N_H[y]$, and $\big|N_H[x]\big| \geq |\mathcal{A}_e| + 2$, $\big|N_H[y]\big| \geq |\mathcal{B}_e| + 2$. Hence (iii).

For the rest we will make use of the following fact: If $Q = N_H[x, y] \notin \{Q_e \mid e \in F\}$ is a maximal clique in G, then by Observation 7.1.4, all vertices in $N_H(x) - y$ or all vertices in $N_H(y) - x$ are end-vertices in H.

(ii) (b): Let $Q = N_H[x, y] \notin \{Q_e \mid e \in F\}$ and assume that all vertices in $N_H(x) - y$ are end-vertices in H. Then all P_6 mid-edges e of H contained in Q must contain y, hence $Q \cap Q_e = N_H[y]$ for all forced edges e in Q.

(iv): Consider a maximal clique $Q = N_H[x, y]$ of G. If $e = xy \in F$, $Q = Q_e$. Otherwise, we may assume that all vertices in $N_H(x) - y$ are end-vertices in H. Then there exists $y' \in N_H(y) - x$ such that yy' is the mid-edge of a P_6 in H (otherwise, H would be a tree with diameter at most four, and G would be trivial by Observation 7.1.6 (i)). Hence $e = yy' \in F$ and $Q \in \mathcal{C}_e$.

(v): If $v \in Q_e$ for some $e \in F$, then v also belongs to another maximal clique in $\mathcal{A}_e \cup \mathcal{B}_e$, hence v is not simplicial in G. Let $v \in Q = N_H[x, y] \notin \{Q_e \mid e \in F\}$, and assume that all vertices in $N_H(x) - y$ are end-vertices. Then $N_{H^3}[v] = N_H[x, y]$, i.e., v belongs to exactly the maximal clique Q. □

7.2 Good characterization

We now are able to characterize cubes of graphs with girth at least 10 as follows.

Theorem 7.2.1 *Let G be a connected, non-trivial graph. Let F be the subgraph of G consisting of all forced edges in G. Then, G is the cube of a graph with girth at least* 10 *if and only if the following conditions hold.*

(i) *For each $e \in F$, there exists a unique maximal clique $Q_e \in \mathcal{C}_e$ such that*

  (a) *for every two distinct non-disjoint forced edges e and e', $e \cup e' \subseteq Q_e \cap Q_{e'}$,*

  (b) *for every $Q \in \mathcal{C}(G) \setminus \{Q_e \mid e \in F\}$, and for all forced edges e_1, e_2 in Q, $Q_{e_1} \cap Q = Q_{e_2} \cap Q$;*

(ii) *For each $e \in F$, $\mathcal{C}_e \setminus \{Q_e\}$ can be partitioned into non-empty disjoint sets $\mathcal{A}_e$ and $\mathcal{B}_e$ with*

  (a) *$Q \cap Q' = e$ if and only if $Q \in \mathcal{A}_e$ and $Q' \in \mathcal{B}_e$ or vice versa,*

  (b) *setting $A_e = \bigcap_{Q \in \mathcal{A}_e} Q$, $B_e = \bigcap_{Q \in \mathcal{B}_e} Q$, all pairs of maximal cliques in $\mathcal{A}_e$ have the same intersection A_e, all pairs of maximal cliques in $\mathcal{B}_e$ have the same intersection B_e,*

  (c) *$Q_e = A_e \cup B_e$, and $|A_e| \geq |\mathcal{A}_e| + 2$, $|B_e| \geq |\mathcal{B}_e| + 2$,*

  (d) *$F[A_e \cap V_F]$ and $F[B_e \cap V_F]$ are stars with distinct universal vertices in e;*

(iii) *$\mathcal{C}(G) = \bigcup_{e \in F} \mathcal{C}_e$;*

(iv) *$V_G \setminus \bigcup_{e \in F} Q_e$ consists of exactly the simplicial vertices of G.*

(v) *F is connected and has girth at least* 10.

Proof. The only if-part is shown by Proposition 7.1.7. For the if-part, let G satisfy the conditions (i) – (v). Write $\mathcal{K} = \{Q_e \mid e \in F\}$. We construct a cube root H for G by Algorithm 7.2: `CubeRootGirthTen` below.

Algorithm 7.2: `CubeRootGirthTen`

Input: Connected graph $G = (V_G, E_G)$ with $n = |V_G|$ and $m = |E_G|$.
Output: A cube root H with girth at least 10 if such H exists or 'NO' otherwise.

1. $H := F$
2. **for** each $Q_e \in \mathcal{K}$ **do**
3. let $e = xy$ where x is universal in $F[A_e \cap V_F]$ and
4. y is universal in $F[B_e \cap V_F]$ // cf. Theorem 7.2.1 (ii)(d)
5. put all edges ux, vy into H, $u \in A_e \setminus V_F, v \in B_e \setminus V_F$
6. // $H[Q_e] = N_H[x, y]$ for all $e = xy \in F$
7. **for** each $Q \notin \mathcal{K}$ **do**
8. let $Q \in \mathcal{C}_e$ for some forced edge e
9. choose a vertex $c_Q \in (Q \cap Q_e) \setminus V_F$; $c_Q \neq c_{Q'}$ for $Q \neq Q' \notin \mathcal{K}$
10. // Note that $Q \cap Q_e = A_e$ or $Q \cap Q_e = B_e$, hence the choices
11. // of c_Q's are possible by (ii)(c); c_Q is independent of e by (i)(b)
13. put all edges vc_Q, $v \in Q \setminus V_H$, into H
14. **return** H

Note that by (iii), all maximal cliques of G are considered by the algorithm, hence H is a spanning subgraph of G. Write $H_0 = F$, and let H_1 denote the graph H after the first for-loop (at line 2), and H_2 be the output graph H of Algorithm 7.2: `CubeRootGirthTen`. Then the following facts hold by construction ($i = 1, 2$; for $i = 2$ note that $V_{H_2} \setminus V_{H_1}$ consists of simplicial vertices of G):

$$\text{For all } u \in V_{H_i} \setminus V_{H_{i-1}}\text{: } deg_{H_i}(u) = 1, \text{ and if } N_{H_i}(u) = \{w\}, \text{ then } w \in V_{H_{i-1}}; \tag{7.1}$$

$$\text{For all } v, v' \in V_{H_i} \setminus V_{H_{i-1}}\text{: } vv' \in E_G \text{ if and only if } N_{H_i}(v) = N_{H_i}(v'); \tag{7.2}$$

$$\text{For all } uw \in E_H \setminus F\text{: } w \in V_F \Rightarrow u \in Q_e \text{ for all } e \in F \text{ containing } w; \tag{7.3}$$

For all $vu, uw \in E_H \setminus F$: $w \in V_F \Rightarrow v \in V_{H_2} \setminus V_{H_1}$ and $u = c_Q \in Q \cap Q_e$ for all $e \in F$ containing w, where $Q \in \mathcal{C}_e \setminus \{Q_e\}$ is the unique maximal clique containing v. (7.4)

It follows from (7.1) that H_{i-1} is an induced subgraph of H_i, and H_1 and H_2 have the same girth as $H_0 = F$. Hence by (v) H is connected and has girth at least 10.

We now show that $G = H^3$. By construction it is clear that each edge uv of G is an edge in H^3 (consider a maximal clique of G containing uv). Conversely, let $uv \in E_{H^3} \setminus E_H$. Then there is a path in H of length two or three.

Case 1: uwv is a path in H.
If $uw, vw \in F$, then by (i) (a), $u, v \in Q_{uw} \cap Q_{vw}$, hence $uv \in E_G$. If $uw \notin F, vw \in F$ (or vice versa), then by (7.3), $u \in Q_{vw}$, hence $uv \in E_G$. Finally, if $uw, vw \notin F$, then, by the construction of H and by (7.2), $uv \in E_G$.

Case 2: uw_1w_2v is a path in H.
If $w_1w_2 \in F$, then, as in Case 1, by (i) (a) and by (7.3), $u, v \in Q_{w_1w_2}$, hence $uv \in E_G$. So, we may assume that $w_1w_2 \in E_H \setminus F$. Since F is an induced subgraph in H, it follows that $uw_1 \notin F$ or $vw_2 \notin F$, say $uw_1 \in E_H \setminus F$. Then $vw_2 \in F$, otherwise w_1 or $w_2 \notin V_F$ because F is induced in H, contradicting (7.1). Now, by (7.4), $u \in Q_{vw_2}$, hence $uv \in E_G$. □

Theorem 7.2.2 *Given an n-vertex m-edge graph G, recognizing if G is the cube of some graph H with girth at least 10 can be done in time $O(nm^2)$, and if so, such a cube root H for G can be constructed within the same time bound.*

Proof. Note that by Corollary 7.1.2, any cube of an m-edge graph with girth at least ten has at most m maximal cliques. Then, use the algorithm in [67] to list the maximal cliques of G in time $O(nm^2)$. If there are more than m maximal cliques, G is not the cube of any graph with girth at least ten. Otherwise, the at most m maximal cliques of G are available. Then computing the forced edges of G to form the subgraph F of G, as well as the lists $\mathcal{C}_e$ for each $e \in F$ can be done in time $O(m^2)$ in an obvious way.

Assuming the partitions $\mathcal{C}_e = \mathcal{A}_e \cup \{Q_e\} \cup \mathcal{B}_e$ for all forced edges e are given, conditions (i) – (v) in Theorem 7.2.1 then can be tested within the same time bound, as well as the cube root H, in case all conditions are satisfied, can be constructed according to the constructive proof (Algorithm 7.2: `CubeRootGirthTen`) of Theorem 7.2.1.

We now point out how to find the partition $\mathcal{C}_e = \mathcal{A}_e \cup \{Q_e\} \cup \mathcal{B}_e$ satisfying (ii) (if any) for each e (initially, $\mathcal{A}_e = \varnothing = \mathcal{B}_e$):

- Find two $Q, Q' \in \mathcal{C}_e$ with $Q \cap Q' = e$ (if such two cliques do not exist we just return 'NO'). Put Q into $\mathcal{A}_e$ and Q' into $\mathcal{B}_e$. Set $\mathcal{D} := \mathcal{C}_e \setminus \{Q, Q'\}$;
- While there exists $Q^* \in \mathcal{D}$ s.t. $Q^* \cap Q = e$ for all $Q \in \mathcal{A}_e$, put Q^* into $\mathcal{B}_e$ and remove Q^* from $\mathcal{D}$;
- While there exists $Q^* \in \mathcal{D}$ s.t. $Q^* \cap Q = e$ for all $Q \in \mathcal{B}_e$, put Q^* into $\mathcal{A}_e$ and remove Q^* from $\mathcal{D}$;
- If $|\mathcal{D}| = 1$, let Q_e be the clique in $\mathcal{D}$;
- If $|\mathcal{D}| \neq 1$ or $\mathcal{C}_e = \mathcal{A}_e \cup \{Q_e\} \cup \mathcal{B}_e$ does not satisfy (ii), return 'NO'.

The correctness is obvious. Since $|\mathcal{C}_e| < n$ this will take $O(n^2)$ time. Since there are at most m forced edges, it will take $m \cdot O(n^2) = O(nm^2)$ time in total (note that $m \geq n$). □

7.3 Further considerations

Observe that the proof of Theorem 7.2.1 shows that if F is a tree, then also the cube root H is a tree. This fact and Lemma 7.1.1 and Proposition 7.1.7 immediately imply the following good characterization for cubes of trees in terms of forced edges. Recall Definition 7.1.3 in Section 7.1 for the notion of forced edges in this case.

Theorem 7.3.1 *Let G be a connected, non-complete graph. Let F be the subgraph of G consisting of all forced edges in G. Then G is the cube of a tree if and only if the following conditions hold.*

(i) *For each $e \in F$, there exists a unique maximal clique $Q_e \in \mathcal{C}_e$ such that*

  (a) *for every two distinct non-disjoint forced edges e and e', $e \cup e' \subseteq Q_e \cap Q_{e'}$,*

  (b) *for every $Q \in \mathcal{C}(G) \setminus \{Q_e \mid e \in F\}$, and for all forced edges e_1, e_2 in Q, $Q_{e_1} \cap Q = Q_{e_2} \cap Q$;*

(ii) *For each $e \in F$, $\mathcal{C}_e \setminus \{Q_e\}$ can be partitioned into non-empty disjoint sets $\mathcal{A}_e$ and $\mathcal{B}_e$ with*

  (a) *$Q \cap Q' = e$ if and only if $Q \in \mathcal{A}_e$ and $Q' \in \mathcal{B}_e$ or vice versa,*

  (b) *setting $A_e = \bigcap_{Q \in \mathcal{A}_e} Q$, $B_e = \bigcap_{Q \in \mathcal{B}_e} Q$, and all pairs of maximal cliques in $\mathcal{A}_e$ have the same intersection A_e, all pairs of maximal cliques in $\mathcal{B}_e$ have the same intersection B_e,*

  (c) *$Q_e = A_e \cup B_e$, and $|A_e| \geq |\mathcal{A}_e| + 2$, $|B_e| \geq |\mathcal{B}_e| + 2$,*

  (d) *$F[A_e \cap V_F]$ and $F[B_e \cap V_F]$ are stars with distinct universal vertices in e;*

(iii) *$\mathcal{C}(G) = \bigcup_{e \in F} \mathcal{C}_e$;*

(iv) *$V_G \setminus \bigcup_{e \in F} Q_e$ consists of exactly the simplicial vertices of G.*

(v) *F is tree.*

Similarly, as in the proof of Theorem 7.2.1, if F a (C_4, C_6, C_8)-free bipartite graph, then the cube root H for G is also a (C_4, C_6, C_8)-free bipartite graph. Thus, if we replace the condition on F in Theorem 7.2.1 by 'F is a (C_4, C_6, C_8)-free bipartite graph', we obtain a good characterization and an $O(nm^2)$-time recognition for cubes of bipartite roots of this kind, while CUBE OF BIPARTITE GRAPH is NP-complete in general [45]. Thus we have the following result.

Corollary 7.3.2 *There is a good characterization and an $O(nm^2)$-time recognition algorithm for cubes of (C_4, C_6, C_8)-free bipartite graphs.*

Finally, similar to Subsection 6.2.4 (see in page 68) we consider the computational complexity of CLIQUE on the class of cubes of graphs with girth at least 10.

It was shown in [49] that CLIQUE remains NP-complete on the graph class of cubes of graphs (of girth three). By Corollary 7.1.2, G has $O(|E_G|)$ maximal cliques and CLIQUE efficiently solvable in the class of cubes of graphs with girth at least 10.

Corollary 7.3.3 *The weighted version of* CLIQUE *can be solved in* $O(nm^2)$ *time on cubes of graphs with girth at least ten, where* n *and* m *are the number of vertices, respectively, edges of the input graph.*

Proof. Let $G = (V_G, E_G)$ be the cube of some graph with girth at least ten. By Corollary 7.1.2, G has at most $|E_G|$ maximal cliques. We first apply the algorithm in [67] to find all maximal cliques in G in time $O(|V_G| \cdot |E_G| \cdot |E_G|)$.
Then, to solve the problem CLIQUE we iterate over the list of all maximal cliques, selecting the largest one. □

7.4 Concluding remarks

We gave good characterizations of graphs that are cubes of a graph having girth at least 10. Our characterization leads to an $O(nm^2)$-time recognition algorithm for such graphs. It is interesting that while recognizing cubes of bipartite graphs is NP-complete, our results in this section gave a good characterization of cubes of (C_4, C_6, C_8)-free bipartite graphs (bipartite without "short cycles").

These results are related to Conjecture 6.4.3, which appears in Chapter 6 that k-POWER OF GRAPH WITH GIRTH $\geq 3k-1$ is polynomially solvable. We strongly believe that this technique can be used to give a good characterization for k-th powers of graphs with girth at least $3k+1$.

Furthermore, we have shown that CLIQUE can be solved in $O(nm^2)$ time on cubes of graphs with girth at least ten.

Chapter 8

Squares of Strongly Chordal Split Graphs

Lau and Corneil [46] first showed that SQUARE OF SPLIT GRAPH and SQUARE OF CHORDAL GRAPH are NP-complete. In contrast, we will show in this chapter that there exists a good characterization of squares of strongly chordal split graphs that gives a recognition algorithm in time $O(\min\{n^2, m \log n\})$ for such squares.

8.1 Squares of strongly chordal split graphs

A graph G is a *strongly chordal split graph* if it is strongly chordal and a split graph. We will make use of the following well-known fact:

Lemma 8.1.1 ([25, 52, 61]) *Powers of strongly chordal graphs are strongly chordal.*

In a graph, a vertex is *maximal* if its closed neighborhood is maximal with respect to set-inclusion. As an example, the vertices u, v in Figure 8.1 are maximal vertices, while the other vertices are not.

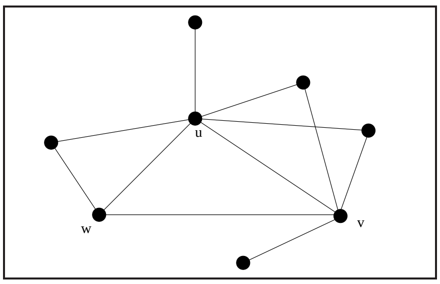

Figure 8.1: The vertices u, v are maximal vertices in the given graph

For split graphs $H = (V_H, E_H)$ we write $H = (C \cup S, E_H)$, meaning $V_H = C \cup S$ is a partition of the vertex set of H into a clique C and a stable set S.

Lemma 8.1.2 *Let $H = (C \cup S, E_H)$ be a connected split graph without 3-sun. Then Q is a maximal clique in H^2 if and only if $Q = N_H[v]$ for some maximal vertex $v \in C$ of H.*

Proof. The assertion is trivially true if $S = \varnothing$. So let us assume that $S \neq \varnothing$.

First, let $v \in C$ be a maximal vertex of H. Then $N_H(v) \cap S \neq \varnothing$, and, obviously, $N_H[v]$ is a clique in H^2. Write $T = N_H(v) \cap S$. Suppose to the contrary that some vertex $w \in S \setminus T$ satisfies $d_H(w, x) = 2$ for all $x \in T$. Then

$$\text{every vertex in } T \text{ is adjacent in } H \text{ to a vertex in } N_H(w) \cap C. \tag{8.1}$$

Moreover, as v is a maximal vertex in H,

$$\text{every vertex in } N_H(w) \cap C \text{ is non-adjacent in } H \text{ to a vertex in } T. \tag{8.2}$$

Now let $u \in N_H(w) \cap C$ be a vertex with maximal number of H-neighbors in T. By (8.1), there exists a vertex $x \in T \setminus N_H(u)$. By (8.2), x is adjacent to a vertex $y \in N_H(w) \cap C$. Furthermore, by the choice of u, y is non-adjacent to a vertex $z \in T \cap N_H(u)$. But then u, v, y, z, x and w induce a 3-sun in H. This contradiction shows that every vertex $w \in S \setminus T$ is at distance 3 in H to some vertex in T. Hence, in H^2, the clique $N_H[v]$ is maximal.

Second, let Q be a maximal clique in H^2. Note that every vertex in C is a universal vertex in H^2, hence Q properly contains C. Let $v \in C$ be a vertex with maximal $T = N_H(v) \cap Q \cap S$. We claim that $T = Q \cap S$. If not, let x be a vertex in $(Q \cap S) \setminus T$, and let $u \in N_H(x) \cap C$ be a vertex with maximal number of H-neighbors in T. Note that every pair of a vertex in T and a vertex in $(Q \cap S) \setminus T$ must have a common H-neighbor because they must have distance two in H. Now, by the choice of v, u has a non-neighbor $y \in T$. Consider a common H-neighbor w of x and y. By the choice of u, there exists a vertex $z \in T$ adjacent, in H, to u but non-adjacent to w. But then u, w, v, x, y and z induce a 3-sun in H. This contradiction shows that $Q \cap S \subseteq T$, hence $Q \cap S = T$. Therefore, $Q \subseteq N_H[v]$, and by the maximality of the clique Q, $Q = N_H[v]$ and v is a maximal vertex of H. $\square$

We now are ready to characterize squares of strongly chordal split graphs. Let $\mathcal{C}(G)$ denote the set of all maximal cliques in G.

Theorem 8.1.3 *G is the square of a strongly chordal split graph if and only if G is strongly chordal and $\left|\bigcap_{Q \in \mathcal{C}(G)} Q\right| \geq |\mathcal{C}(G)|$.*

Proof. For the only if-part, let $G = H^2$ for some strongly chordal split graph $H = (C \cup S, E_H)$. By Lemma 8.1.1, G is strongly chordal. By Lemma 8.1.2, $N_H[v]$, $v \in C$ maximal vertices of H, are exactly the maximal cliques in G, hence $C \subseteq \bigcap_{Q \in \mathcal{C}(G)} Q$.

Obviously, $N_H[v] \neq N_H[v']$ for distinct maximal vertices $v, v' \in C$, hence $|C| \geq |\mathcal{C}(G)|$.

For the if-part, let $G = (V_G, E_G)$ be a strongly chordal graph satisfying $\big|\bigcap_{Q \in \mathcal{C}(G)} Q\big| \geq |\mathcal{C}(G)|$. Write $C = \bigcap_{Q \in \mathcal{C}(G)} Q$. Since $|C| \geq |\mathcal{C}(G)|$, we can choose a unique vertex $v_Q \in C$ for each maximal clique $Q \in \mathcal{C}(G)$ such that $v_Q \neq v_{Q'}$ whenever $Q \neq Q'$. Now Lemma 8.1.2 indicates the way how to construct a square root H for G (see also in Figure 8.2).

Put the clique C into H, and then, for each maximal clique Q of G, put the edges vv_Q, $v \in Q \setminus C$, into H. Clearly, $S = V_G \setminus C$ is a stable set in H, hence H is a split graph with $V_H = C \cup S$. Moreover, by construction, H satisfies the following properties:

$$\text{For all } v \in C, N_H(v) \text{ belongs to a maximal clique in } G, \tag{8.3}$$

and

$$\text{for all } u \in S, N_H(u) \subseteq C. \tag{8.4}$$

We now show that H is a square root of G and that H is strongly chordal.

Claim 1: $E_G = E_{H^2}$.
Proof of Claim 1: If $xy \in E_G$, then x, y belong to a maximal clique Q of G. By construction of H, if $v_Q \in \{x, y\}$, then $xy \in E_H$. Otherwise, $xv_Q, yv_Q \in E_H$. In both cases, $xy \in E_{H^2}$. Conversely, let $xy \in E_{H^2} \setminus E_H$. Then there exists a vertex z such that $xz, yz \in E_H$, and by (8.3) (if $z \in C$), respectively, by (8.4) (if $z \in S$), $xy \in E_G$. Claim 1 follows.

Claim 2: H is S_3-free.
Proof of Claim 2: Assume that $u_1, u_2, u_3, v_1, v_2, v_3$ induce a 3-sun in H where $\{u_1, u_2, u_3\}$ is a stable set and $\{v_1, v_2, v_3\}$ is a clique and for $i \in \{1, 2, 3\}$, u_i is adjacent to exactly v_i and v_{i+1} (index arithmetic modulo 3). Then, by construction of H, in G, $u_i \in Q_i \cap Q_{i+1}$ and $u_i \notin Q_{i+2}$ for some maximal cliques Q_1, Q_2, Q_3 of G. Because of the maximality of the cliques, u_i is non-adjacent to a vertex $w_{i+2} \in Q_{i+2}$. Note that $w_{i+2} \notin Q_i \cup Q_{i+1}$. But then $u_1, u_2, u_3, w_1, w_2, w_3$ form a 3-sun in G (if w_1, w_2, w_3 are pairwise non-adjacent in G) or G contains an induced 4-cycle (otherwise). Claim 2 follows.

Claim 3: H is S_ℓ-free for all $\ell \geq 4$.
Proof of Claim 3: Assume to the contrary that H contains an induced ℓ-sun, $\ell \geq 4$. Consider an induced S_ℓ with smallest ℓ, consisting of a stable set $\{u_1, u_2, \ldots, u_\ell\}$ and a clique $\{v_1, v_2, \ldots, v_\ell\}$ such that for $i \in \{1, \ldots, \ell\}$, u_i is adjacent to exactly v_i and v_{i+1} (index arithmetic modulo ℓ). Since the u_i's form a ℓ-cycle in G and G is chordal, this cycle must have a short chord, say $u_1u_3 \in E_G$. Hence in H, u_1 and u_3 have a common neighbor v; note that $v \in C$ and so v is adjacent in H to all v_i. Then, by the minimality of ℓ and by Claim 2, v is adjacent in H to all u_i. Therefore,

$$u_1, \ldots, u_\ell \text{ are pairwise adjacent in } G. \tag{8.5}$$

Furthermore, by construction of H, there exist some maximal cliques $Q_1, \dots, Q_\ell$ of G such that for all i

$$u_i \in Q_{i+1} \text{ and, for all } j \in \{1, \dots, \ell\} \setminus \{i, i+1\},\ u_j \notin Q_{i+1}. \tag{8.6}$$

Next, fixing an index i, all u_j, $j \in \{1, \dots, \ell\} \setminus \{i, i+1\}$, are non-adjacent in G to a common vertex in Q_{i+1}: Because of the maximality of Q_{i+1} and by (8.6), each u_j is non-adjacent to vertex in Q_{i+1}. Let $x \in Q_{i+1}$ be a vertex with maximal number of non-neighbors in $\{u_1, \dots, u_\ell\} \setminus \{u_i, u_{i+1}\}$. If for some $j \neq i, i+1$, u_j is adjacent to x, let $y \in Q_{i+1}$ be non-adjacent to u_j. By the choice of x, y must be adjacent to a non-neighbor $u_{j'}$ of x. But then, by (8.5) and because $xy \in E_G$, $x, y, u_j, u_{j'}$ induce a 4-cycle in G, a contradiction. Hence x is non-adjacent to all $\{u_1, \dots, u_\ell\} \setminus \{u_i, u_{i+1}\}$.

Now, for each i, let $w_{i+1} \in Q_{i+1}$ be a vertex non-adjacent in G to all $\{u_1, \dots, u_\ell\} \setminus \{u_i, u_{i+1}\}$. Then, by (8.5) and (8.6), $\{u_1, \dots, u_\ell, w_1, \dots, w_\ell\}$ induce a ℓ-sun in G (if $w_1, \dots, w_\ell$ are pairwise non-adjacent in G) or G contains an induced 4-cycle (otherwise). This contradiction finally proves Claim 3. □

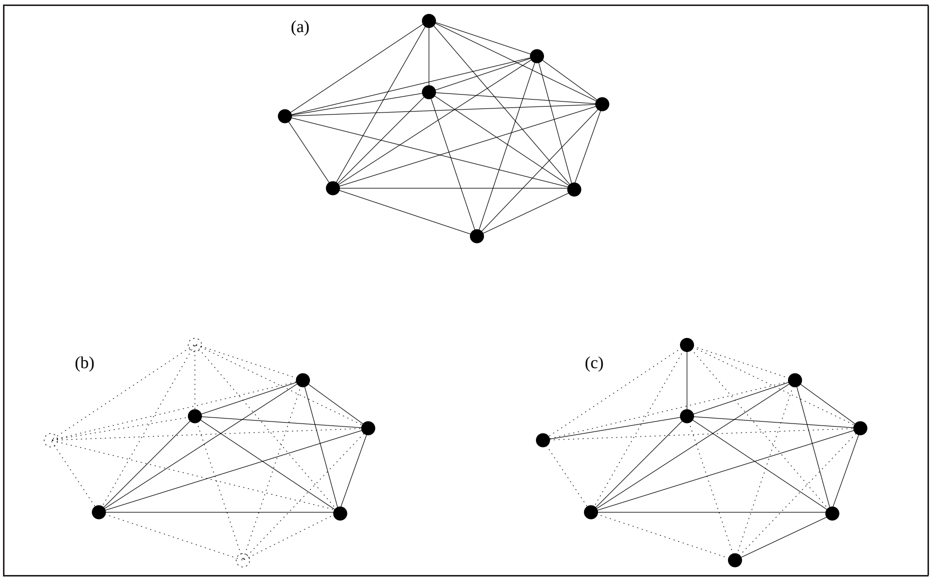

Figure 8.2: An input graph G (a), a clique C in G (b) and a square root H (c) constructed by Algorithm 8.1

Theorem 8.1.4 *Given an n-vertex and m-edge graph G, recognizing if G is the square of some strongly chordal split graph H can be done in time $O(\min\{n^2, m \log n\})$, and if so, such a square root H for G can be constructed in the same time.*

Proof. By the constructive proof of Theorem 8.1.3, the following Algorithm 8.1: `StronglyChordalSplitRoot` correctly computes a strongly chordal

Algorithm 8.1: `StronglyChordalSplitRoot`

Input: Connected graph $G = (V_G, E_G)$ with $n = |V_G|$ and $m = |E_G|$.
Output: A strongly chordal split graph H with $G = H^2$ if such H exists or 'NO' otherwise.

1. **if** G is strongly chordal **then**
2.   compute all maximal cliques $Q_1, \ldots, Q_q$ of G
3.   compute $C = \bigcap_{1 \leq i \leq q} Q_i$
4.   **if** $|C| \geq q$ **then**
5. $V_H := V_G$; $E_H := \{xy \mid x, y \in C\}$
6.   **for** $i := 1$ **to** q **do**
7.   choose a vertex $v_i \in C$ with $v_i \neq v_j$ for $i \neq j$
8.   **for** $i := 1$ **to** q **do**
9. $E_H := E_H \cup \{vv_i \mid v \in Q_i \setminus C\}$
10.   **return** H
11.   **else return** 'NO'
12. **else return** 'NO'

split graph H that is a square root for G, if any. The time complexity of Algorithm 8.1: `StronglyChordalSplitRoot` is dominated by the time consumed at lines 1 and 2. Testing if G is strongly chordal can be done in time $O(\min\{n^2, m \log n\})$ ([28, 53, 58, 64]). Assuming G is strongly chordal, all maximal cliques $Q_1, \ldots Q_q$ of G can be listed in linear time (cf. [33, 64]); note that $q \leq n$. So, the total time of the algorithm is bounded by $O(\min\{n^2, m \log n\})$. □

8.2 Concluding remarks

SQUARE OF SPLIT GRAPH and SQUARE OF CHORDAL GRAPH are NP-complete. On the positive side, we have found efficient algorithms for recognizing squares of strongly chordal split graphs.

Some interesting open questions are: What is the computational complexity of recognizing powers of strongly chordal graphs and of powers of chordal bipartite graphs (bipartite graphs without cycles of length at least six)?

We note that strongly chordal split graphs and chordal bipartite graphs are closely related. Hence, our given result on squares of strongly chordal split graphs, cubes of chordal bipartite graphs could be polynomially solvable.

Conjecture 8.2.1 CUBE OF CHORDAL BIPARTITE GRAPH *is polynomially solvable.*

Chapter 9

Squares of Block Graphs

In this chapter, we give a good characterization for squares of block graphs and a linear-time recognition algorithm for such squares. This algorithm also constructs a square block graph root if one exists. Moreover, block graph square roots in which every endblock is an edge are unique up to isomorphism.

9.1 Introduction

Note that bipartite graphs, as well as graphs having girth at least six generalize trees in such a way that these do not have cliques of size larger than two. It should be remarked that known polynomial time recognitions for squares of trees, of bipartite graphs, and of graphs having girth at least six depend partly on this fact; chordal graphs also generalize trees but deciding if a graph is the square of a chordal graph is NP-complete; see [46].

Another natural generalization of trees are *block graphs*; these are exactly the connected graphs in which every block (i.e., every maximal 2-connected subgraph) is a clique. Powers of block graphs have been considered in [18] in the context of interval number, and in [12] in the context of leaf powers and simplicial powers. To the best of our knowledge, the complexity of recognizing powers of block graphs, as well as the characterization problem are not yet discussed in the literature.

In this chapter we consider the characterization and recognition problems of graphs that are squares of block graphs, i.e., for a given graph G, to determine if $G = H^2$ for some block graph H. We first give relevant properties of squares of block graphs in Section 9.2. Then, based on these properties, we will provide in Section 9.3 good characterizations for graphs that are squares of block graphs and in Section 9.4 a simple linear-time algorithm to compute a square root that is a block graph (if any). In Section 9.5 we will derive known results for squares of trees from our discussions.

We recall that a connected graph is a *block graph* if its blocks are cliques. The following theorem collects several known characterizations of block graphs.

Theorem 9.1.1 *For all graphs G, the following statements are equivalent*

(i) *G is a block graph;*

(ii) *G is the intersection graph of blocks of some connected graph;*

(iii) *G is a connected diamond-free chordal graph;*

(iv) *Between every two vertices in G there is exactly one chordless path.*

The *diamond* is a K_4 minus an edge. The equivalence (i) $\Leftrightarrow$ (ii) is Theorem 3.5 in [35], and the equivalence (i) $\Leftrightarrow$ (iii) can be easily seen, e.g., by [15, Observation 3]. The equivalence (i) $\Leftrightarrow$ (iv) can easily be seen as follows: The direction (i) $\Rightarrow$ (iv) is obvious; (iv) implies that every 2-connected component of G must be a clique, hence (i).

Finally, we remark that block graphs can be recognized in linear-time: By an algorithm in [66], the blocks of a given graph $G = (V_G, E_G)$ can be detected in linear time. Then, testing if all blocks of G are cliques can be done in an obvious way in $O(|V_G| + |E_G|)$ time.

9.2 Basic facts

In this section we give basic properties of squares of block graphs which form a starting point for our characterizations of such graphs in Section 9.3.

Let x, y be two non-adjacent vertices in a graph $G = (V_G, E_G)$. A subset $S \subseteq V_G$ is an *x, y-separator* if x and y belong to different connected components of $G - S$. A *separator* is an x, y-separator for some non-adjacent vertices x, y. A *minimal* separator is an x, y-separator that is not properly contained in an other x, y-separator.

Observation 9.2.1 *Let $G = H^2$ for some block graph H. Let B be a non-endblock of H and let $u \neq v$ be two cut-vertices of H in B. Let X and Y be two connected components of $H - B$ such that $N_H(u) \cap X \neq \varnothing$ and $N_H(v) \cap Y \neq \varnothing$. Then B is a minimal x, y-separator in G for any pair of vertices $x \in X$, $y \in Y$.*

Proof. Clearly, B is a separator in G disconnecting any pair of vertices $x \in X, y \in Y$. Moreover, in G, every vertex $w \in B$ is adjacent to a vertex in X and to a vertex in Y, implying $B - w$ does not separate X and Y. Thus, B is a minimal x, y-separator in G for any pair of vertices $x \in X, y \in Y$. $\square$

The following fact is the key observation for further discussions.

Proposition 9.2.2 *Let G be a connected, non-complete graph such that $G = H^2$ for some block graph H. Then the maximal cliques in G are exactly the closed neighborhoods $N_H[v]$ for cut-vertices v in H.*

Proof. (i) Let v be a cut-vertex in H. Clearly, $Q = N_H[v]$ is a clique in G. Consider an arbitrary vertex $x \in V_H \setminus Q$ (note that such a vertex exists as G is not complete), and let B be a block of H containing v such that x does not belong to the connected component of $H - v$ containing $B - v$. Then $d_H(x, y) \geq 3$ for all vertices $y \in B - v$, hence x cannot be adjacent, in G, to all vertices in Q. Therefore, Q is a maximal clique in G.

(ii) Let Q be a maximal clique in G. Among all vertices in Q, let $v \in Q$ be a vertex with inclusion-maximal $Q \cap N_H[v]$. We will see that v is a cut-vertex of H and $Q = N_H[v]$.

Assume first, by way of contradiction, that v is a simplicial vertex in H and let B be the unique block of H containing v. Then, as G is not complete, B contains at least one cut-vertex. Clearly, for all cut-vertices u of H in B and for all vertices $x \in V_H$, if $d_H(v, x) \leq 2$, then $d_H(u, x) \leq 2$. In particular $d_H(u, q) \leq 2$ for all $q \in Q$. Hence, by the maximality of Q, $u \in Q$. Moreover, $Q \cap N_H[v] = Q \cap B \subseteq Q \cap N_H[u]$, and therefore, by the choice of v, $Q \cap B = Q \cap N_H[u] \subseteq B$ for all cut-vertices u of H in B. This implies that $Q \subseteq B$, contradicting the maximality of Q.

Hence v must be a cut-vertex in H. Next, we claim that $Q \setminus N_H[v] = \varnothing$. If not, consider a vertex $w \in Q \setminus N_H[v]$. As $d_H(w, v) = 2$, there exists a cut-vertex u such that $vu, uw \in E_H$. Note that, in H, u separates v and w. Hence $Q \subseteq N_H[u]$ because $d_H(q, v) \leq 2$ and $d_H(q, w) \leq 2$ for all $q \in Q$. By the maximality of Q, $Q = N_H[u]$. But then $Q \cap N_H[v]$ is the block in H containing vu which is properly contained in $N_H[u] = Q \cap N_H[u]$, contradicting the choice of v. Hence $Q \setminus N_H[v] = \varnothing$, as claimed.

Thus $Q \subseteq N_H[v]$, and by the maximality of Q, $Q = N_H[v]$. □

Minimal separators in squares of block graphs can be characterized as follows.

Proposition 9.2.3 *Let G be a connected, non-complete graph such that $G = H^2$ for some block graph H. Then the following conditions are equivalent:*

(i) *S is a minimal separator of G;*

(ii) *S is a non-endblock of H;*

(iii) *$|S| \geq 2$ and S is the intersection of two maximal cliques of G.*

Proof.
(i) $\Rightarrow$ (ii) Let S be a minimal x, y-separator of G. Let $xv_1 \ldots v_\ell y$ be the shortest path in H connecting x and y. Since $d_H(x, y) \geq 3$, $\ell \geq 2$. Note that all v_i are cut-vertices of H. For each $1 \leq i < \ell$, let B_i be the block of H containing $v_i v_{i+1}$. By Observation 9.2.1, each B_i is a minimal x, y-separator of G. If $S \neq B_i$ for all i,

then, by the minimality of the x, y-separators S and B_i, $B_i - S \neq \varnothing$ for all i. Let $b_i \in B_i - S$, $1 \leq i < \ell$ (possibly $b_i = b_j$ for some $i \neq j$). Now by noting that x and b_1, $b_{\ell-1}$ and y are adjacent in G, as well as $G[\{b_i \mid 1 \leq i < \ell\}]$ contains a path with endpoints $b_1, b_{\ell-1}$, we get the contradiction that S does not separate x and y. Thus, we conclude that $S = B_i$ for some i, hence (ii).

(ii) $\Rightarrow$ (iii) Let S be a non-endblock of H. Then $|S| \geq 2$, and S contains at least two cut-vertices $u \neq v$ of H. By Proposition 9.2.2, $Q = N_H[u]$ and $Q' = N_H[v]$ are maximal cliques in G. Clearly, $S = Q \cap Q'$. The second part follows by Observation 9.2.1.

(iii) $\Rightarrow$ (i) Let Q, Q' be two maximal cliques in G such that $S = Q \cap Q'$ has at least two vertices. By Proposition 9.2.2, $Q = N_H[u]$ and $Q' = N_H[v]$ for some cut-vertices $u \neq v$ in H. Since $\left|N_H[u] \cap N_H[v]\right| = |S| \geq 2$, u and v must be adjacent in H. Hence S is the non-endblock in H containing uv, and (i) follows from Observation 9.2.1.□

As a corollary of Proposition 9.2.3, all minimal separators of the square of a block graph are cliques with at least two vertices, hence squares of block graphs are chordal (indeed, it is well-known that a graph is chordal if and only if each of its minimal separators is a clique; see, e.g., [33]) and 2-connected.

Recall that a chordal graph is *strongly chordal* if it does not contain any ℓ-sun as an induced subgraph; here a *ℓ-sun*, $\ell \geq 3$, consists of a clique $\{u_1, u_2, \ldots, u_\ell\}$ and a stable set $\{v_1, v_2, \ldots, v_\ell\}$ such that for $i \in \{1, \ldots, \ell\}$, v_i is adjacent to exactly u_i and u_{i+1} (index arithmetic modulo ℓ). Clearly, block graphs are strongly chordal.

It was known that powers of strongly chordal graphs are strongly chordal (refer to Lemma 8.1.1). In particular, squares of block graphs (hence of trees) are strongly chordal; later, not knowing this fact, [49, 3] proved that the square of a tree is chordal. As another consequence of Proposition 9.2.3, we give a new and short proof for this fact:

Corollary 9.2.4 ([25, 52, 61]) *Squares of block graphs are strongly chordal.*

Proof. Let G be a non-complete graph that is the square of block graph H. As pointed out, G is chordal. Suppose G contains an induced ℓ-sun with clique $\{u_1, u_2, \ldots, u_\ell\}$ and stable set $\{v_1, v_2, \ldots, v_\ell\}$. Let Q be a maximal clique in G containing $\{u_1, u_2, \ldots, u_\ell\}$, and for each $i \in \{1, \ldots, \ell\}$, let Q_i be a maximal clique of G containing v_i, u_i and u_{i+1}. Now, $Q \cap Q_i$, $1 \leq i \leq \ell$, contains u_i and u_{i+1} but none of $\{u_1, u_2, \ldots, u_\ell\} \setminus \{u_i, u_{i+1}\}$, hence they are pairwise distinct blocks in H. But then the cycle in H with edges $u_1u_2, u_2u_3, \ldots, u_{\ell-1}u_\ell, u_\ell u_1$ belongs to distinct blocks of H, a contradiction. Thus, G is a strongly chordal graph. □

The structure of the minimal separators in squares of block graphs is now described in the following proposition. Given a block graph H, a simplicial vertex of H belonging to an endblock of H is called a *leaf*.

Proposition 9.2.5 *Let G be a connected, non-complete graph such that $G = H^2$ for* *some block graph H. Let F be the subgraph of G formed by all minimal separators of G. Then*

(i) *F is obtained from H by deleting all leaves of H. In particular, F is a block graph whose blocks are exactly the minimal separators of G;*

(ii) *For all maximal cliques Q and Q' of G with $|Q \cap Q'| \geq 2$, $Q \cap Q'$ is a block of F;*

(iii) *Every block S of F belongs to at least two and at most $|S|$ maximal cliques of G;*

(iv) *Every two non-disjoint blocks of F belong to a common maximal clique of G;*

(v) *For all maximal cliques Q of G, $Q \cap V_F = N_F[v]$ for some vertex v of F;*

(vi) *$V_G \setminus V_F$ is the set of all simplicial vertices of G. Moreover, for every vertex $v \in V_G \setminus V_F$, $|N_H(v) \cap V_F| = 1$, and if $N_H(v) \cap V_F = \{u\}$, then $N_G(u) \setminus V_F = N_H(u) \setminus V_F$.*

Proof.
(i) and (ii) follow from Proposition 9.2.3.

(iii) follows from Propositions 9.2.3 and 9.2.2, by noting that any block S in H clearly contains at most $|S|$ cut-vertices, and if S is a non-endblock, S contains at least two cut-vertices.

(iv): Two non-disjoint minimal separators S and S' of G are two non-endblocks of H (by Proposition 9.2.3) having a common cut-vertex, say v. Hence $S \cup S' \subseteq N_H[v]$, and (iv) follows from Proposition 9.2.2.

(v): Let Q be a maximal clique in G. By Proposition 9.2.2, $Q = N_H[v]$ for some cut-vertex v of H. Since G is not complete, some block of H in $N_H[v]$ must be a non-endblock, hence $v \in V_F$, and by (i), $N_F[v] = Q \cap V_F$.

(vi): If v has two non-adjacent neighbors $x \neq y$, then any minimal x, y-separator in G must contain v, hence $v \in V_F$. Thus, every vertex in $V_G \setminus V_F$ must be simplicial in G. On the other side, by (iii), every vertex in F belongs to at least two maximal cliques. Thus, $V_G \setminus V_F$ consists of exactly the simplicial vertices of G. The second part follows directly from the following observation: By (i), any vertex $v \in V_G \setminus V_F$ is a leaf of H and belongs to an endblock B_v of H. As G is not complete, the cut-vertex u of H in B_v must belong to a non-endblock of H, hence $N_H(v) \cap V_F = (B_v \setminus \{v\}) \cap V_F = \{u\}$, and $N_G(u) \setminus V_F$ consists of exactly the leaves of H that belong to an endblock containing u. □

Unlike tree roots, block graph roots are not unique in general. Indeed, if H is a block graph and H' is the block graph obtained from H by deleting all edges joining two simplicial vertices in an endblock of H (thus, every endblock in H' is an edge), then clearly H^2 and $(H')^2$ coincide; see also Figure 9.1.

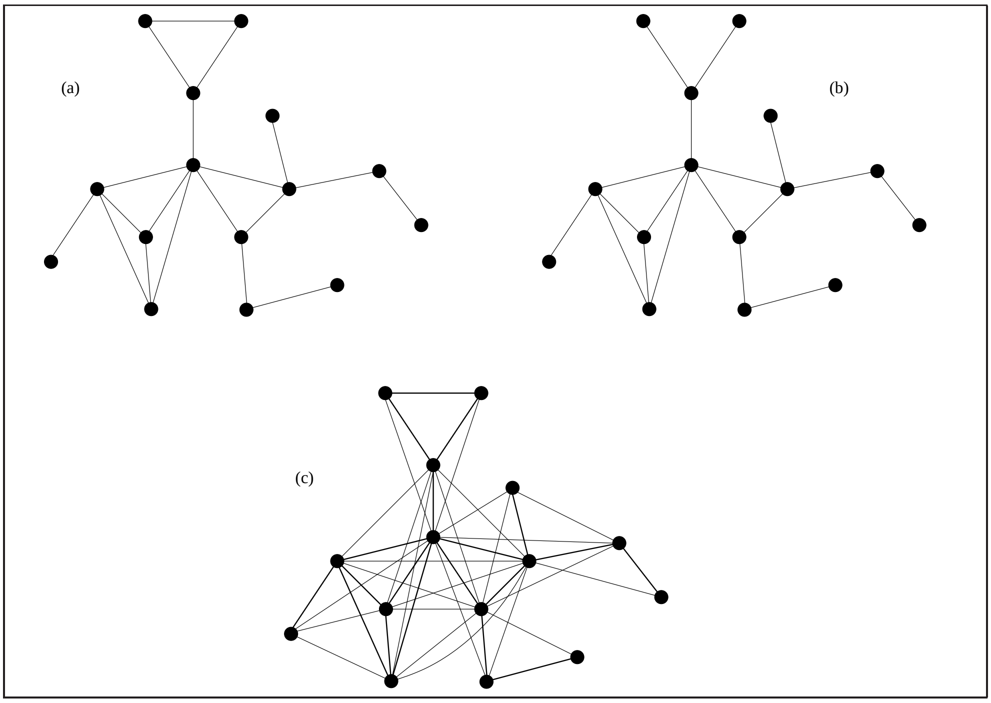

Figure 9.1: Two block graphs (a) and (b) with the same square (c)

Theorem 9.2.6 *Block graph roots in which every endblock is an edge are unique up to isomorphism.*

Proof. Let H_1, H_2 be two block graphs in which every endblock is an edge, and assume that $f : H_1^2 \to H_2^2$ is an isomorphism. We will show that H_1 and H_2 are isomorphic by pointing out that the restriction $f : H_1 \to H_2$ of f is an isomorphism.

Write $G_i = H_i^2$, $i = 1, 2$. If G_1 or G_2 is a clique, then H_i must be stars (as every endblock in H_i is an edge) with the same vertex number, hence they are isomorphic. So, assume that G_i are non-complete, and let F_i be the subgraph of G_i formed by the minimal separators of G_i. By Proposition 9.2.5 (i), F_i is a block graph and each block of F_i is a non-endblock of H_i.

Claim 1: The restrictions $f : V_{F_1} \to V_{F_2}$ and $f : V_{H_1} \setminus V_{F_1} \to V_{H_2} \setminus V_{F_2}$ of f are bijections, and $V_{H_i} \setminus V_{F_i}$ is a stable set in H_i, $i = 1, 2$.

Proof of Claim 1: The first part follows from Proposition 9.2.5 (vi), the second part follows from our assumption on the block graphs H_i.

Claim 2: For all $v, v' \in V_{F_1}$: $vv' \in E_{F_1}$ if and only if $f(v)f(v') \in E_{F_2}$.

Proof of Claim 2: Note that, by Claim 1, $f(v), f(v') \in V_{F_2}$. Let $vv' \in E_{F_1}$. Then $f(v)f(v')$ is an edge in G_2. If $f(v)f(v') \notin E_{F_2}$, then $f(v)$ and $f(v')$ must belong to different blocks $B_2 \neq B_2'$ in F_2 with $B_2 \cap B_2' \neq \varnothing$. Consider the block B in F_1 containing vv'. As B is a non-endblock of H_1, there are different blocks B_1, B_1' of H_1 with $\varnothing \neq B \cap B_1 \neq B \cap B_1' \neq \varnothing$. Let $x \in B_1 \setminus B, x' \in B_1' \setminus B$. Then $xx' \notin E_{G_1}$

but $f(x)f(x') \in E_{G_2}$ because $f(x)$ and $f(x')$ are adjacent to both $f(v)$ and $f(v')$, hence $f(x)$ and $f(x')$ must belong to some blocks in H_2 containing the cut vertex $B_2 \cap B_2'$. This contradiction shows that $f(v)f(v') \in E_{F_2}$. Along the same line, it can be seen that $f(v)f(v') \in E_{F_2}$ implies $vv' \in E_{F_1}$.

Claim 3: For all $v \in V_{F_1}$, $v' \in V_{H_1} \setminus V_{F_1}$: $vv' \in E_{H_1}$ if and only if $f(v)f(v') \in E_{H_2}$. *Proof of Claim 3:* Note that, by Claim 1, $f(v) \in V_{F_2}$, $f(v') \in V_{H_2} \setminus V_{F_2}$. Let $vv' \in E_{H_1}$. Then $f(v)f(v') \in E_{G_2}$. Assume that $f(v)f(v') \notin E_{H_2}$. Then there exists vertex $u \in V_{H_1}$ such that $f(v)f(u)$ and $f(v')f(u)$ are edges of H_2. As $f(v)$ is a cut-vertex of H_2, there exists $w \in V_{H_1}$ such that $f(w)f(v) \in E_{H_2}$ and $f(w), f(u)$ belong to different blocks of H_2. Hence $f(w)f(v') \notin E_{G_2}$, and by Proposition 9.2.5 (vi) (second part), $f(w) \notin V_{F_2}$. Therefore, $w \in V_{F_1}$ (by Claim 1) and $wv \in E_{H_1}$ (by Claim 2), implying $wv' \in E_{G_1}$. This contradicts $f(w)f(v') \notin E_{G_2}$. Thus, $f(v)f(v') \in E_{H_2}$, as claimed. Similarly, it can be seen that $f(v)f(v') \in E_{H_2}$ implies $vv' \in E_{H_1}$.

It follows from Claims 1 – 3 that the restriction $f : H_1 \to H_2$ of f is an isomorphism. □

9.3 Good characterizations of squares of block graphs

We now are ready to characterize graphs that are squares of a block graph. Our characterizations are good in the sense that they lead to polynomial time recognition algorithms for such graphs.

Theorem 9.3.1 *Let G be a connected non-complete graph and let F be the subgraph of G formed by all minimal separators of G. Then G is the square of a block graph if and only if the following conditions hold.*

(i) *F is a block graph whose blocks are exactly the minimal separators of G;*

(ii) *For all maximal cliques Q and Q' of G with $|Q \cap Q'| \geq 2$, $Q \cap Q'$ is a block of F;*

(iii) *Every block S of F belongs to at least two and at most $|S|$ maximal cliques of G;*

(iv) *Every two non-disjoint blocks of F belong to a common maximal clique of G;*

(v) *For all maximal clique Q of G, $Q \cap V_F = N_F[s]$ for some vertex s of F.*

Proof. The only if-part follows directly from Proposition 9.2.5.

For the if-part, let G be a connected, non-complete graph satisfying (i) – (v). Then note that $V_G \setminus V_F$ is the set of all simplicial vertices of G (cf. also the proof of Proposition 9.2.5 (vi)): If v has two non-adjacent neighbors $x \neq y$, then any minimal x, y-separator in G must contain v, hence $v \in V_F$. On the other side, by (iii), every vertex in F belongs to at least two maximal cliques.

Now, we will construct a spanning subgraph H of G such that H is a block graph and $G = H^2$ by attaching the simplicial vertices in $V_G \setminus V_F$ to F in a suitable way (see also Fig. 9.2): For each $v \in V_G \setminus V_F$, $Q = N_G[v]$ is a maximal clique of G (as v is simplicial in G). By (v) we have two cases. If $Q \cap V_F = N_F[s]$ for some cut-vertex s of F, then $Q \cap V_F$ consists of all blocks of F at s. Since H should be a square root of G, $d_H(v, s) \leq 2$ for all $v \in Q$. Hence we must attach all $v \in Q \setminus V_F$ to F at the vertex s. In the other case, $S = Q \cap V_F$ is a block of F. Then we take a simplicial vertex $s \in S$ of F and attach all $v \in Q \setminus V_F$ to F at the vertex s. A simplicial vertex of F in S always exists: If $s \in S$ is a cut-vertex of F, i.e., there is another block S' of F at s, then by (iv), $N_F[s] = Q' \cap V_F$ for another maximal clique $Q' \neq Q$ (hence we cannot join $v \in Q \setminus V_F$ to s). Thus, letting q_1 be number of the maximal cliques C of G with $C \cap V_F = S$ and q_2 be the number of cut-vertices of F in S, we have $q_1 \leq |S| - q_2$ because of (iii) at most $|S|$ maximal cliques may contain S. To sum

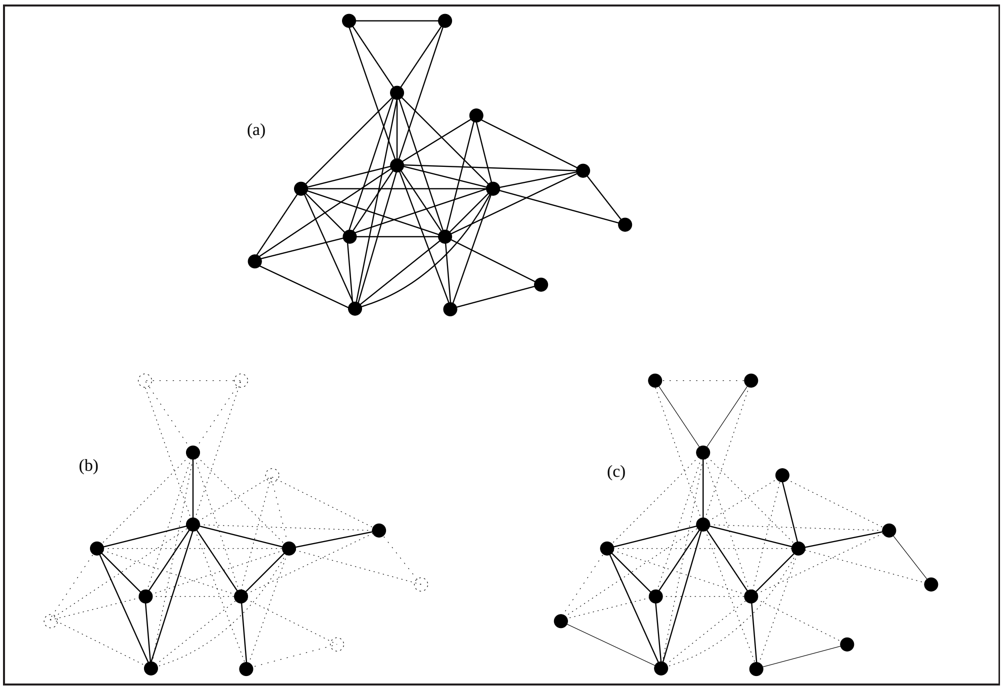

Figure 9.2: An input graph G (a), the subgraph F of G (b) and a square root H (c) constructed by Algorithm 9.3: `BlockGraphRoot`

up, a block graph H that will be a square root of G is constructed by the following Algorithm 9.3: `BlockGraphRoot`:

Algorithm 9.3: `BlockGraphRoot`

1. $H := F$
2. let X be the set of all cut-vertices of F
3. **for** each $v \in V_G \setminus V_H$ **do**
4. $\quad Q := N_G[v]$ //note: Q is a maximal clique in G
5. $\quad$ **if** $Q \cap V_F$ is a block of F **then**
6. $\qquad$ choose an arbitrary vertex $s_Q \in (Q \cap V_F) \setminus X$
7. $\qquad X := X \cup \{s_Q\}$
8. $\qquad$ **else** let s_Q be the universal vertex of $F[Q \cap V_F]$
9. $\quad$ //note: $Q \cap V_F = N_F[s_Q]$ by (v)
10. $\quad V_H := V_H \cup Q$
11. $\quad E_H := E_H \cup \{vs_Q \mid v \in Q \setminus V_H\}$

The output graph H of Algorithm 9.3: `BlockGraphRoot` has the following properties:

Claim 1: The following facts hold:

(a) H is a spanning subgraph of G and contains F as an induced subgraph;

(b) Every vertex $v \in V_G \setminus V_F$ has exactly one neighbor in H, and if $N_H(v) = \{u\}$, then $u \in V_F$;

(c) If $v \in V_G \setminus V_F$ with $N_H(v) = \{u\}$, then $vw \in E_G$ for each $w \in N_F(u)$, and for all $v \neq v'$ in $V_G \setminus V_F$: $vv' \in E_G$ if and only if $N_H(v) = N_H(v')$;

(d) H is a block graph.

Proof of Claim 1: (a): As discussed before, by Conditions (iii) – (v), the vertex s_Q chosen in the for-loop at line 6, respectively, line 8 always exists, hence H is a spanning subgraph of G. Since the algorithm only attaches vertices outside F to F to obtain H, F is an induced subgraph of H.

(b) is obvious by construction; c.f. lines 6, 8 and 11 of the algorithm: Every $v \in V_G \setminus V_H$ is contained in a unique maximal clique Q of G and, by construction, $N_H(v) = \{s_Q\}$ and $s_Q \in V_F$.

(c): First, let $v \in V_G \setminus V_F$ with $N_H(v) = \{u\}$. Then, by (b), $u \in F$, and by construction, $Q \cap V_F = N_F[u]$ where Q is the unique maximal clique of G containing v. Hence the first assertion holds.

Next, let $v \neq v'$ in $V_G \setminus V_F$. If $vv' \in E_G$, then v and v' belong to a unique maximal clique Q of G, hence by construction $N_H(v) = N_H(v') = \{s_Q\}$. Conversely, if $vv' \notin E_G$, then the maximal cliques Q, Q' of G containing v, respectively v', are distinct. By construction, $s_Q \neq s_{Q'}$ whenever s_Q or $s_{Q'}$ is a simplicial vertex of F. So, let us consider the case $Q \cap V_F = N_F[s_Q]$, $Q' \cap V_F = N_F[s_{Q'}]$ with cut-vertices $s_Q, s_{Q'}$

in F. If $s_Q = s_{Q'}$, then $Q \cap Q' \cap V_F = N_F[s_Q] \subseteq Q \cap Q'$, contradicting (ii) because $N_F[s_Q]$ contains at least two blocks of F. Thus, $s_Q \neq s_{Q'}$, i.e., $N_H(v) \neq N_H(v')$.

(d): Since F is a block graph (by (i)), (c) directly follows from (a) and (b). (It should be remarked that every endblock in H is an edge)

Claim 2: $E_{H^2} \subseteq E_G$.
Proof of Claim 2: Let $vv' \in E_{H^2} \setminus E_H$. Then there exists a vertex u such that $vu, v'u \in E_H$. We distinguish three cases. First, if $v, v' \in V_F$, then also, by Claim 1 (b), $u \in V_F$, and hence by Claim 1 (a), $vu, v'u \in E_F$. Now, as $vv' \notin E_H$, uv and uv' belong to different blocks of F, and by (iii), v and v' are contained in a common maximal clique of G, hence $vv' \in E_G$. Second, if $v, v' \in V_G \setminus V_F$, then by Claim 1 (c), $vv' \in E_G$. Third, without loss of generality, we may assume that $v \in V_G \setminus V_F$ and $v' \in V_F$. Then by Claim 1 (b), $u \in V_F$ is the unique neighbor of v in H, and by Claim 1 (a), $v'u \in E_F$. Now, again by Claim 1 (c), $vv' \in E_G$.

Claim 3: $E_G \subseteq E_{H^2}$.
Proof of Claim 3: Let $vv' \in E_G \setminus E_H$ and let Q be a maximal clique in G containing vv'. First assume that $Q \cap V_F = N_F[s]$ for some cut-vertex s of F. Then, as $vv' \notin E_H$, $s \notin \{v, v'\}$. Hence $sv, sv' \in E_F$ (if $v, v' \in V_F$), or by construction $sv, sv' \in E_H$ (if $v, v' \notin V_F$) or one of sv, sv' is in E_F and the other is in E_H (otherwise). Thus $vv' \in E_{H^2}$. Next, if $Q \cap V_F$ is a block of F, then $Q \setminus V_F \neq \varnothing$ (by (iii)), and hence by construction $sv, sv' \in E_H$ for some $s \in Q \cap V_F$, $s \neq v, v'$. Thus $vv' \in E_{H^2}$.

It follows by Claims 2 and 3 that $G = H^2$, and Theorem 9.3.1 is proved. □

Another formulation of Theorem 9.3.1 is:

Theorem 9.3.2 *Let G be a connected graph. Then G is the square of a block graph if and only if G is 2-connected, chordal, and satisfies the following conditions:*

(i) *Every two distinct minimal separators of G have at most one vertex in common;*

(ii) *For all maximal cliques Q and Q' of G with $|Q \cap Q'| \geq 2$, $Q \cap Q'$ is a minimal separator of G;*

(iii) *Every minimal separator S belongs to at least two and at most $|S|$ maximal cliques of G;*

(iv) *Every two non-disjoint minimal separators belong to a common maximal clique of G;*

(v) *All minimal separators belonging to the same maximal clique have exactly one vertex in common.*

Proof. For complete graphs the theorem is trivially true. So, let us assume that G is non-complete. The if-part then follows from Observation 9.2.1, Corollary 9.2.4, and Theorem 9.3.1.

For the only if-part, let G be a 2-connected, non-complete, chordal graph satisfying (i) – (v). Let F be the subgraph of G formed by all minimal separators. We will show that F is a block graph in which each block is a minimal separator, and thus G satisfies the conditions in Theorem 9.3.1 and we are done.

To this end, we first note that for every maximal clique Q of G there exists another maximal clique Q' with $|Q \cap Q'| \geq 2$ (this is because G is non-complete, chordal and 2-connected). This together with (ii) and (v) imply that F is connected.

Next we show that F is chordal. Suppose not. Then there exists an induced cycle $v_1v_2 \dots v_\ell v_1$ in F, $\ell \geq 4$. Since every minimal separator is a clique (by (iii)), all edges v_iv_{i+1} (indices taken modulo ℓ) belong to pairwise distinct minimal separators. Hence, by (iv),

$$v_iv_{i+2} \in E_G \setminus E_F \quad \text{for all } i. \tag{9.1}$$

Therefore, by (v),

$$v_iv_{i+3} \notin E_G \quad \text{for all } i. \tag{9.2}$$

In particular, $\ell \geq 6$. Consider the cycle C in G, $C = v_1v_3v_5 \dots v_\ell v_1$ if ℓ is odd and $C = v_1v_3v_5 \dots v_{\ell-1}v_1$ otherwise.

If $\ell = 6$, let Q be a maximal clique of G containing $v_1v_3v_5$, Q' be a maximal clique containing $v_1v_2v_3$. By (9.2), $Q \neq Q'$. Now $v_1v_3 \in Q \cap Q'$, hence by (ii), $v_1v_3 \in F$, contradicting (9.1).

If $\ell \geq 7$, C has length $\lceil \frac{\ell}{2} \rceil \geq 4$. Since G is chordal, C has a short chord, say $v_1v_5 \in E_G$. As before we conclude $v_1v_3 \in F$, a contradiction to (9.1).

Thus, F is chordal. Furthermore, (i) and (v) imply that F cannot contain an induced diamond, hence F is a block graph. Finally, (i) implies that the blocks of F are exactly the minimal separators of G. $\square$

Note that all conditions in Theorem 9.3.2, respectively, Theorem 9.3.1, can be tested in polynomial time (in fact, it is straightforward to do this in $O(nm)$ steps). Hence these characterizations give polynomial-time recognition algorithms for squares of block graphs.

9.4 A linear time recognition for squares of block graphs

In this section we will describe how to recognize squares of block graphs in linear time. Instead of testing the Conditions (i) – (v) given in the characterizations explicitly, we will need the following fact:

Lemma 9.4.1 *Given a graph G and a block graph H on the same vertex set, testing if $G = H^2$ can be done in $O(m)$ time.*

Proof. The argument is similar to the case of tree squares given in [45, Lemma 6.1]. For the sake of completeness, we give the proof here.

Recall that leaves in a block graph H are simplicial vertices of H belonging to an endblock of H. Pick an arbitrary leaf v of H, and let B be the endblock of H containing v. Let u be the cut-vertex of H in B if $H \neq B$. Otherwise let u be an arbitrary vertex in $B - v$. Obviously, $N_{H^2}[v] = N_H[u]$. Therefore, if $G = H^2$, then $N_G[v] = N_H[u]$. Thus, if $N_G[v] \neq N_H[u]$, we return 'NO', meaning $G \neq H^2$. Otherwise, we replace G and H by $G - v$ and $H - v$, respectively, and repeat the process. If only one vertex has remained in H and G, it implies that $N_{H^2}[w] = N_G[w]$ for all vertices w. In this case $H^2 = G$ and we return 'YES'. The total time complexity is bounded by $\sum_{v \in V_G} O(\deg_G(v)) = O(\sum_{v \in V_G} \deg_G(v)) = O(m)$. □

Theorem 9.4.2 *Given a graph G, it can be recognized in $O(n+m)$ time if G is the square of a block graph, and if so, such a block graph square root can be computed in the same time.*

Proof. It is well-known that 2-connectedness can be tested in linear time $O(n + m)$ (see [66]). It is also well-known that testing chordality and listing all maximal cliques, as well as all minimal separators of a given chordal graph can be done in linear time (see, for example, [16, 33, 43, 64]).

So, given $G = (V_G, E_G)$, we assume that G is chordal and 2-connected, otherwise we just return 'NO', meaning that G is not the square of a block graph. We may also assume that all maximal cliques and all minimal separators of G are available, and that there are at most $n = |V_G|$ maximal cliques (cf. Proposition 9.2.2) and at most $m = |E_G|$ minimal separators (cf. Proposition 9.2.3). In particular, we may assume further that the subgraph F of G formed by all minimal separators is a block graph, otherwise we return 'NO' (cf. Proposition 9.2.5 (i)).

Next, construct the block graph H from F according to Algorithm 9.3: `BlockGraphRoot` in the proof of Theorem 9.3.1; with small modifications: In line 5 instead of testing if $Q \cap V_F$ is a block of F we just test if $Q \cap V_F$ is a clique. Then, in line 6, respectively, line 8, if the vertex s_Q does not exists, we return 'NO' and stop. This takes $\sum_{v \in V_G \setminus V_F} O(\deg(v)) = O(m)$ steps.

Note that if G is indeed the square of a block graph, then all Conditions (i) – (v) are satisfied, hence the so constructed block graph H is indeed exactly the block graph obtained from Algorithm 9.3: `BlockGraphRoot`, and therefore, H is a square root of G (cf. proof of Theorem 9.3.1). Thus, we have to check if H is really a square root of G. If not, we correctly return 'NO' . This last check takes $O(m)$ steps as pointed out by lemma 9.4.1, and Theorem 9.4.2 follows. □

9.5 Squares of trees revisited

Given the fact that the squares of trees have been widely discussed in the literature, we will derive from our results some previously known results for tree squares.

First, tree squares are strongly chordal by Corollary 9.2.4. Second, as every endblock in a tree is an edge, Theorem 9.2.6 implies directly:

Theorem 9.5.1 ([15, 45, 63]) *The tree roots of squares of trees are unique, up to isomorphism.*

Third, observe that Proposition 9.2.3 shows that each minimal separator in a tree square consists of exactly two vertices, and therefore, in tree squares minimal separators and 2-cuts coincide. Hence, in Theorem 9.3.2, applied for tree squares, (i) is trivially satisfied, (ii) means that every two maximal cliques have at most two vertices in common (this plus chordality and 2-connectedness implies that if $|Q \cap Q'| = 2$ then $Q \cap Q'$ is a 2-cut), and (iii) means that every 2-cut belongs to exactly two maximal cliques. Thus, we also obtain the good characterization of squares of trees (refer to Theorem 6.2.11).

Furthermore, in the proof of Theorem 9.4.2, if F is a tree, then H is also a tree. Hence we obtain:

Theorem 9.5.2 ([15, 17, 42, 45, 49]) *Given a graph $G = (V_G, E_G)$, it can be recognized in $O(|V_G| + |E_G|)$ time if G is the square of a tree. Moreover, a tree root of a square of a tree can be computed in the same time.*

9.6 Concluding remarks

Block graphs generalize trees in a very natural way, and in a sense, they are not too far from trees. Discussing powers of block graphs is motivated by a number of results on tree powers in the literature. In this chapter we have found good characterizations for squares of block graphs and a linear-time recognition algorithm for such squares. Our algorithm also constructs a square block graph root if one exists. Moreover, block graph square roots in which every endblock is an edge are unique up to isomorphism. Finally, our discussion on squares of block graphs generalizes some previously known results on squares of trees.

For all fixed $k \geq 3$, the complexity status of recognizing k-th powers of block graphs is not yet determined. However, we strongly believe that k-TH POWER OF BLOCK GRAPH should be efficiently solvable for all fixed k:

k-TH POWER OF BLOCK GRAPH
Instance: A graph G.
Question: Is there a block graph H such that $G = H^k$?

Also, it would be interesting to see if there exists a good graph-theoretic characterization for k-th powers of block graphs for all k.

Conjecture 9.6.1 *For all fixed $k \geq 2$, k-*TH POWER OF BLOCK GRAPH *is polynomially solvable.*

Bibliography

[1] T. Andreae, On the traveling salesman problem restricted to inputs satisfying a relaxed triangle inequality, *Networks* 38 (2001) 59-67.

[2] T. Andreae, H.-J. Bandelt, Performance guarantees for approximation algorithms depending on parametrized triangle inequalities, *SIAM J. Discrete Math.* 8 (1995) 1-16.

[3] Geir Agnarsson, Raymond Greenlaw, Magnús M. Halldórsson, On powers of chordal graphs and their colorings, *Congressus Numer.* 144 (2000) 41–65.

[4] G. Agnarsson and M.M. Halldórsson, Coloring powers of a planar graph, *SIAM J. Discrete Math*, 16(4) (2003): 651–662.

[5] Noga Alon, Bojan Mohar, The chromatic number of graph powers, *Combinatorics, Probability and Computing* 11 (2002) 1–10.

[6] H.-J. Bandelt, A. Henkmann, F. Nicolai, Powers of distance-hereditary graphs, *Discrete Mathematics* 145 (1995) 37-60.

[7] H. J. Bandelt, H.M. Mulder, Distance-hereditary graphs, *J. Combin. Theory* B 41 (1986) 182-208.

[8] R. Battiti, A.A. Bertossi, M.A. Bonuccelli, Assigning codes in wireless networks: Bounds and scaling properties, *Wireless Networks* 5 (1999) 195-209.

[9] M.A. Bender, C. Chekuri, Performance guarantees for the TSP with a parametrized triangle inequality, *Information Processing Letters* 73 (2000) 17-21.

[10] A.A. Bertossi, M.A. Bonuccelli, Code assignment for hidden terminal interference avoidance in multihop packet radio networks, *IEEE-ACM Trans. on Networking* 3 (1995) 441-449.

[11] J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications*, American Elsevier Publishing Co., Inc., New York, 1976.

[12] Andreas Brandstädt, Van Bang Le, Simplicial powers of graphs, In: Proceedings of the 2nd Annual International Conference on Combinatorial Optimization and Applications (COCOA'08), *Lecture Notes in Computer Science* 5165 (2008) 160–170.

[13] Andreas Brandstädt, Van Bang Le, Jeremy P. Spinrad, *Graph Classes: A Survey*, SIAM Monographs on Discrete Math. Appl., Vol. 3, SIAM, Philadelphia (1999).

[14] Andreas Brandstädt, Van Bang Le, Structure and linear time recognition of 3-leaf powers., *Information Processing Letters* 98 (2006) 133–138.

[15] Andreas Brandstädt, Van Bang Le, and R. Sritharan, Structure and linear time recognition of 4-leaf powers, *ACM Transactions on Algorithms* 5 (2008) 11:1–11:22

[16] L. Sunil Chandran, Fabrizio Grandoni, A linear time algorithm to list the minimal separators of chordal graphs, *Discrete Math.* 306 (2006) 351–358.

[17] Maw-Shang Chang, Ming-Tat Ko, and Hsueh-I Lu, Linear time algorithms for tree root problems, *Lecture Notes in Computer Science*, 4059 (2006) 411–422.

[18] Mingjang Chen, Gerard J. Chang, Douglas B. West, Interval numbers of powers of block graphs, *Discrete Math.* 275 (2004) 87–96.

[19] Mingjang Chen, Gerard J. Chang, Families of Graphs Closed Under Taking Powers, *Graphs and Combinatorics* (2001) 17, pp.207–212.

[20] Z.-Z.Chen, T. Jiang, G. Lin, Computing phylogenetic roots with bounded degrees and errors, *SIAM J. Computing* 32 (2003) 864-879.

[21] I. Chlamtac, S. Kutten, On broadcasting in radio networks – Problem analysis and protocol design, *IEEE Trans. on Comm.* 33 (1985) 1240-1246.

[22] I. Chlamtac, S. Pinter, Distributed nodes organization algorithm in a multihop dynamic radio network, *IEEE Trans. on Comm.* 36 (1987) 728-737.

[23] M. Chudnovsky, N. Robertson, P. Seymour, R. Thomas, The Strong Perfect Graph Theorem, *Annals of Math*, Vol 164(2006) 51–229.

[24] Daniel W. Cranston, Seog-Jin Kim, List-coloring the square of a subcubic graph, *J. Graph Theory* 57 (2007) 65–87.

[25] Elias Dahlhaus, P. Duchet, On strongly chordal graphs, *Ars Combin.* 24 B (1987) 23–30.

[26] G.A. Dirac, On rigid circuit graphs, *Abh. Math. Sem. Univ. Hamburg.* 25, pp. 71–76, 1961.

[27] F. Escalante, L. Montejano, and T. Rojano, Characterization of n-path graphs and of graphs having nth root, *J. Combin. Theory* B 16 (1974) 282–289.

[28] M. Farber, Characterizations of strongly chordal graphs, *Discrete Math.* 43 (1983) 173-189.

[29] Babak Farzad, Lap Chi Lau, Van Bang Le, Ngoc Tuy Nguyen, Computing Graph Roots Without Short Cycles, *Proceedings of Symposium on Theoretical Aspects of Computer Science (STACS 2009)*, pp. 397 - 408.

[30] H. Fleischner, The square of every two-connected graph is Hamiltonian, *J. Combin. Theory* B 16 (1974) 29–34.

[31] C. Flotow, Graphs whose powers are chordal and graphs whose powers are interval garphs, *J. Graph Theory* 24 (1997) 323-330.

[32] Michael R. Garey, David S. Johnson, *Computers and Intractability–A Guide to the Theory of NP-Completeness*, Freeman, New York (1979), twenty-third printing 2002.

[33] Martin C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, New York (1980).

[34] Takashi Hamada, Graphs which are n-th powers of trees, *TRU Math.* 15 (1979) 1-17.

[35] F. Harary, *Graph Theory*, Addison-Wesley, Massachusetts (1972).

[36] Frank Harary, R.M. Karp, and W.T. Tutte, A criterion for planarity of the square of a graph, *J. Combin. Theory* 2 (1967) 395–405.

[37] Frédéric Havet, Choosability of the square of planar subcubic graphs with large girth, *Discrete Math.*, to appear.

[38] J.D. Horton, K. Kilakos, Minimum edge dominating sets, *SIAM J. Discrete Math* 6 (1993) 375–387.

[39] E. Howorka, A characterization of distance-hereditary graphs, *Quart. J. Math. Oxford Ser.* 2 (28) (1977) 417-420.

[40] E. Howorka, A characterization of ptolemaic graphs, *J. Graph Theory* 5 (1981) 323-331.

[41] Alon Itai, Michael Rodeh, Finding a minimum circuit in a graph, *SIAM J. Computing* 7 (1978) 413–423.

[42] Paul E. Kearney, Derek G. Corneil, Tree powers, *J. Algorithms* 29 (1998) 111–131.

[43] P. Sreenivasa Kumar, C.E. Veni Madhawan, Minimal vertex separators of chordal graphs, *Discrete Appl. Math.* 89 (1998) 155–168.

[44] H.T. Lau, *Finding a Hamiltonian Cycle in the Square of a Block*, PhD Thesis, School of Computer Science, McGill University, Montreal, QC, Canada (1980).

[45] Lap Chi Lau, Bipartite roots of graphs, *ACM Transactions on Algorithms* 2 (2006) 178–208.

[46] Lap Chi Lau, Derek G. Corneil, Recognizing powers of proper interval, split and chordal graphs, *SIAM J. Discrete Math.* 18 (2004) 83–102.

[47] Van Bang Le and Ngoc Tuy Nguyen, Hardness Results and Efficient Algorithms for Graph Powers, to appear in: *Proceedings of the* 35^{th} *International Workshop on Graph-Theoretic Concepts in Computer Science* (WG 2009).

[48] Van Bang Le and Ngoc Tuy Nguyen, The square of a block graph, to appear in: *Discrete Mathematics* (2009), doi:10.1016/j.disc.2009.09.004.

[49] Yaw.-Ling Lin, Steven S. Skiena, Algorithms for square roots of graphs, *SIAM J. Discrete Math.* 8 (1995) 99–118.

[50] G.-H. Lin, P.E. Kearney, T. Jiang, Phylogenetic k-root and Steiner k-root, *Proceedings of ISAAC 2000, Lecture Notes in Computer Science* 1969 (2000) 539–551.

[51] N. Linial, Locality in distributed graph algorithms, *SIAM J. Computing* 21 (1992) 193–201.

[52] Anna Lubiw, Γ-*free matrices*, Master Thesis, Dept. of Combinatorics and Optimization, University of Waterloo, Canada, 1982.

[53] Anna Lubiw, Doubly lexical orderings of matrices *SIAM J. Computing* 16 (1987) 854–879.

[54] Kazuhisa Makino, Takeaki Uno, New Algorithms for Enumerating all Maximal Cliques, *SWAT* 2004, pp.260-272.

[55] Rajeev Motwani, Madhu Sudan, Computing roots of graphs is hard, *Discrete Appl. Math.* 54 (1994) 81-88.

[56] A. Mukhopadhyay, The square root of a graph, *J. Combin. Theory* 2 (1967) 290-295.

[57] N. Nishimura, P. Ragde, D.M. Thilikos, On graph powers for leaf-labeled trees, *J. Algorithms* 42 (2002) 69-108.

[58] Robert Paige, Robert E. Tarjan, Three partition refinement algorithms, *SIAM J. Computing* 16 (1987) 973–989.

[59] S.B. Rao, The cube of a tree, *I.S.I. Technical Report*. No 8(1968).

[60] S.B. Rao, The fourth power of tree, *I.S.I. Technical Report*. No 18(1968).

[61] A. Raychaudhuri, On powers of strongly chordal and circular arc graphs, *Ars Combin.* 34 (1992) 147–160.

[62] Donald J. Rose, R. Endre Tarjan, and George S. Lueker, Algorithmic aspects of vertex elimination on graphs, *SIAM J. Computing* 5 (1976) 266–283.

[63] I.C. Ross, Frank Harary, The square of a tree, *Bell System Tech. J.* 39 (1960) 641–647.

[64] J.P. Spinrad, *Efficient Graph Representations*, Fields Institute Monographs, Toronto (2003).

[65] Robert E. Tarjan, Mihalis Yannakakis, Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs, *SIAM J. Computing* 13 (1984) 566–579 (Addendum: *SIAM J. Comput.* 14 (1985) 254–255).

[66] Robert E. Tarjan, Depth first search and linear time algorithms, *SIAM J. Computing* 1 (1972) 146–160.

[67] Shuji Tsukiyama, Mikio Ide, Hiromu Ariyoshi, and Isao Shirakawa, A new algorithm for generating all the maximal independent sets, *SIAM J. Computing* 6 (1977) 505–517.

[68] D. B. West, *Introduction to Graph Theory*, Prentice Hall Inc., Upper Saddle River, NJ, 1996.

[69] R.K. Yeh, *Labeling graphs with a condition at distance two*, PhD Thesis, University of South Carolina, USA (1990).

Index

Thesen zur

"Graph Powers: Hardness Results, Good Characterizations and Efficient Algorithms"

Given a graph $H = (V_H, E_H)$ and a positive integer k, the *k-th power* of H, written H^k, is the graph obtained from H by adding new edges between any pair of vertices at distance at most k in H; formally, $H^k = (V_H, \{xy \mid 1 \leq d_H(x, y) \leq k\})$. A graph G is the k-th power of a graph H if $G = H^k$, and in this case, H is a *k-th root* of G. For the cases of $k = 2$ and $k = 3$, we say that H^2 and H^3 the *square*, respectively, the *cube* of H and H a *square root* of $G = H^2$, respectively, a *cube root* of $G = H^3$.

Graph powers and roots are fundamental graph-theoretic concepts and have a number of applications in different fields. They have been extensively studied in the literature, both in the theoretic and the algorithmic sense.

In this thesis, our investigations deal with the computational complexity for recognizing k-th powers of general graphs as well as restricted graphs. We organize the following summary of results according to the considered graph classes.

1. General graphs: We proved that recognizing k-th powers of graphs is NP-complete for all fixed $k \geq 2$, and thus proved a conjecture in [Lap Chi Lau, Bipartite roots of graphs, *ACM Transactions on Algorithms* 2 (2006) 178–208]. By this we have completed the computational complexity status for recognizing k-th powers of general graphs for all fixed $k \geq 2$.

2. Bipartite graphs: We gave reductions proving the NP-completeness for recognizing k-th powers of bipartite graphs for all fixed $k \geq 3$, and thus proved another conjecture in [Lap Chi Lau, Bipartite roots of graphs, *ACM Transactions on Algorithms* 2 (2006) 178–208]. On the other hand, we found a good characterization of cubes of (C_4, C_6, C_8)-free bipartite graphs that gives a recognition algorithm in polynomial-time for such cubes.

3. Chordal graphs: We generalized the results of Lau and Corneil [Lap Chi Lau, Derek G. Corneil, Recognizing powers of proper interval, split and chordal graphs, *SIAM J. Discrete Math.* 18 (2004) 83–102] by showing that recognizing k-th powers of chordal graphs and k-th roots of chordal graphs are NP-complete for all fixed $k \geq 2$. On the positive side, we provided a good characterization of squares of strongly chordal split graphs that gives a recognition algorithm in time $O(\min\{n^2, m \log n\})$ for such squares. Moreover, this algorithm also constructs a strongly chordal split graph square root if it exists.

4. Graphs in terms of their girth: The *girth* of G is the smallest length of a cycle in G,

- For all fixed $k \geq 2$, recognizing k-th powers of graphs with girth at most $2\lfloor\frac{k}{2}\rfloor + 2$ is NP-complete.
- There is a polynomial time algorithm to recognize if $G = H^2$ for some graph H of girth at least 6. This algorithm also constructs a square root of girth at least 6 if one exists.
- There exists a good characterization of squares of a graph having girth at least 7. This characterization not only leads to a simple algorithm to compute a square root of girth at least 7 but also shows such a square root, if it exits, is unique up to isomorphism.
- There is a good characterization of cubes of a graph having girth at least 10 that gives a recognition algorithm in time $O(nm^2)$ for such graphs. Moreover, this algorithm constructs a cube root of girth at least 10 if it exists.

These results almost provide a dichotomy theorem for the complexity of the recognition problem in terms of girth of the square roots.

5. Block graphs: We have found good characterizations for squares of block graphs and a linear-time recognition algorithm for such squares. This algorithm also constructs a block graph square root if one exists. Moreover, block graph square roots in which every endblock is an edge are unique up to isomorphism.

6. Tree squares: Given the fact that the squares of trees have been widely discussed in the literature, powers of $\mathcal{C}$-graph where $\mathcal{C}$ contains all trees are particular interest. Block graphs and graphs of girth at least seven properly contain trees. By using the characterizations for squares of block graphs and squares of graphs with girth at least seven, we obtain new characterizations for tree squares that allow us to derive known results on tree square roots easily.

Zusammenfassung

Sei $H = (V_H, E_H)$ ein Graph. Für eine gegebene natürliche Zahl k sei H^k die *k-te Graphpotenz* von H, wenn H^k genau die Knotenpaare $u, v \in V_H$ durch eine Kante verbindet, deren Distanz in H höchstens k ist. Formal wird H^k als der Graph $H^k = (V_H, \{xy \mid 1 \leq d_H(x, y) \leq k\})$ beschrieben.

Ein gegebener Graph G heißt k-te Graphpotenz von H, wenn G isomorph zu H^k ist. In diesem Fall sei H auch *k-te Wurzel* von G. Die Spezialfälle H^2 und H^3 bezeichnen das *Quadrat* bzw. den *Kubus* von H. Gleichermaßen soll H in diesen Fällen *Quadratwurzel* bzw. *kubische Wurzel* heißen.

Diese Arbeit untersucht die Berechnungskomplexität des Erkennungsproblems für k-te Potenzen von allgemeinen Graphen und einigen speziellen Graphenklassen. Die Hauptresultate der Arbeit sind:

- Die Erkennung von Graphen G, die eine k-te Wurzel H besitzen, ist NP-vollständig für alle $k \geq 2$. Das Problem bleibt NP-vollständig, wenn G chordal ist. Auch unter der Einschränkung von H auf bipartite oder chordale Graphen bleibt das Problem NP-vollständig.
- Die *Weite* eines Graphen ist die Länge eines kürzesten Kreises des Graphen.
  - Für alle $k \geq 2$ ist es NP-vollständig zu entscheiden, ob $G = H^k$ für einen Graphen H mit Weite höchstens $2\lfloor \frac{k}{2} \rfloor + 2$.
  - Es kann in Polynomialzeit entschieden werden, ob G das Quadrat eines Graphen H mit Weite höchstens 6 ist.
  - Es gibt algorihmisch gute Charakterisierungen für Quadrate von Graphen mit Weite mindestens 7 und für Kuben von Graphen mit Weite mindestens 10.

  Diese Resultate stellen beinahe eine Dichotomie für die Komplexität des Erkennungsproblems von Quadraten bezüglich der Weite der Quadratwürzel dar.
- Die Quadrate streng chordaler Splitgraphen können gut charakterisiert werden. Diese Eigenschaft führt zu einem Erkennungsalgorithmus für diese Graphen, der in $O(\min\{n^2, m \log n\})$ Zeit arbeitet.

- Die Quadrate G von Blockgraphen besitzen eine Charakterisierung, die zu einem Erkennungsalgorithmus mit linearer Laufzeit führt. Der Algorithmus ist gegebenenfalls sogar in der Lage, die Quadratwurzel von G zu konstruieren.

Summary

Given a graph $H = (V_H, E_H)$ and a positive integer k, the *k-th power* of H, written H^k, is the graph obtained from H by adding new edges between any pair of vertices at distance at most k in H; formally, $H^k = (V_H, \{xy \mid 1 \leq d_H(x, y) \leq k\})$.
A graph G is the k-th power of a graph H if $G = H^k$, and in this case, H is a *k-th root* of G. For the cases of $k = 2$ and $k = 3$, we say that H^2 and H^3 the *square*, respectively, the *cube* of H and H a *square root* of $G = H^2$, respectively, a *cube root* of $G = H^3$.

In this thesis we study the computational complexity for recognizing k-th powers of general graphs as well as restricted graphs. This work provides new NP-completeness results, good characterizations and efficient algorithms for graph powers. The main results are the following.

- There exist reductions proving the NP-completeness for recognizing k-th powers of general graphs for fixed $k \geq 2$, recognizing k-th powers of bipartite graphs for fixed $k \geq 3$, recognizing k-th powers of chordal graphs, and finding k-th roots of chordal graphs for all fixed $k \geq 2$.
- The *girth* of G, girth(G), is the smallest length of a cycle in G,
  - For all fixed $k \geq 2$, recognizing of k-th powers of graphs with girth at most $2\lfloor \frac{k}{2} \rfloor + 2$ is NP-complete.
  - There is a polynomial time algorithm to recognize if $G = H^2$ for some graph H of girth at least 6. This algorithm also constructs a square root of girth at least 6 if one exists.
  - There exists a good characterization of squares of a graph having girth at least 7. This characterization not only leads to a simple algorithm to compute a square root of girth at least 7 but also shows such a square root, if it exits, is unique up to isomorphism.
  - There is a good characterization of cubes of a graph having girth at least 10 that gives a recognition algorithm in time $O(nm^2)$ for such graphs. Moreover, this algorithm constructs a cube root of girth at least 10 if it exists.

These results almost provide a dichotomy theorem for the complexity of the recognition problem in terms of girth of the square roots.

- There is a good characterization of squares of strongly chordal split graphs that gives a recognition algorithm in time $O(\min\{n^2, m \log n\})$ for such squares. Moreover, this algorithm also constructs a strongly chordal split graph square root if it exists.

- There exists a good characterization and a linear-time recognition algorithm for squares of block graphs. This algorithm also constructs a block graph square root if one exists. Moreover, block graph square roots in which every endblock is an edge are unique up to isomorphism.

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbständig und nur unter Vorlage der angegebenen Literatur und Hilfsmittel angefertigt habe.

Rostock, 4. November 2009

Ngoc Tuy Nguyen

Curriculum Vitae

Name:	Ngoc Tuy Nguyen
Date of birth:	December 10, 1968
Place of birth:	Thanhhoa, Vietnam
Marital status:	Married, two children
Nationality:	Vietnam
Address in VN:	13/41 Cuu Huu Street, Tanson Quarter, Thanhhoa City, Vietnam.
	Email: nntuy@yahoo.com

EDUCATION

1982 - 1985:	Lamson Thanhhoa Secondary School, Abitur (May 1985).
1985 - 1989:	Faculty of Mathematics and Informatics of Vinh University, Bachelor of Mathematics (July 1989).
1990 - 1992:	Faculty of Mathematics and Informatics of Vinh University, MSc of Mathematics (December 1993).
1996 - 1998:	Hanoi University of Education, MSc of Education (April 1998).
1998 - 2002:	Hanoi University of Technology, Engineer of Information Technology (May 2002).
2005 - present:	Institut of Informatics, University of Rostock, Germany. PhD. Candidate of Computer Science (Scholarship of Vietnam Government).

TEACHING EXPERIENCE

1989 - 1997:	Lecturer of Mathematics and Informatics Department of Mathematics, Thanhhoa College, Vietnam.
1998 - present:	Lecturer of Informatics at Department of Computer Science, Hongduc University, Thanhhoa, Vietnam.

Rostock, June 11, 2009