Technische Universität Berlin
Electrical Engineering and Computer Science
Institute of Software Engineering
and Theoretical Computer Science
Algorithmics and Computational Complexity (AKT)

Proximity and Intractibility

Revisiting Classic Graph Problems

Master thesis

Pascal Kunz

zur Erlangung des Grades „Master of Science“ (M. Sc.)
im Studiengang Computer Science (Informatik)

Erstgutachter:	Prof. Dr. Rolf Niedermeier
Zweitgutachter:	Prof. Dr. Stefan Felsner
Betreuer:	Till Fluschnik, Malte Renken, Prof. Dr. Rolf Niedermeier

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und eigenhändig sowie ohne unerlaubte fremde Hilfe und ausschließlich unter Verwendung der aufgeführten Quellen und Hilfsmittel angefertigt habe.

Die selbständige und eigenhändige Ausfertigung versichert an Eides statt

Berlin, den ____________________ ____________________________

Datum Unterschrift

Zusammenfassung

Proximity-Graphen werden von einer Menge von Punkten in der Ebene oder in einer höherdimensionalen Struktur induziert. Diese Graphen beschreiben, welche Paare von Punkten einander nah sind und welche nicht. Es gibt etliche verschiedene Proximity-Graphen, die unterschiedlichen Definitionen von Nähe entsprechen. Wir untersuchen drei Klassen von Proximity-Graphen: Relative-Neighborhood-Graphen (RNG), Relatively-Closest-Graphen (RCG) und Gabriel-Graphen. Wir untersuchen klassische NP-vollständige Probleme darauf, ob sie auch auf diesen Graphklassen NP-vollständig bleiben. Wir kommen zu dem Ergebnis, dass sie dies mit wenigen Ausnahmen tun. Wir betrachten dabei die algorithmischen Probleme Independent Set, Vertex Cover, 3-Colorability, Dominating Set, Feedback Vertex Set und Hamiltonian Cycle.

Die Reduktionen, mithilfe derer wir beweisen, dass diese Probleme NP-schwer sind, basieren auf *zweiseitigen Book Embeddings*, die eine sehr strukturierte planare Einbettung bestimmter Graphen vorgeben. Daher untersuchen wir auch kurz die Berechnung von Book Embeddings der genannten Proximity-Graphen. Zudem betrachten wir Proximity-Graphen, die sich mit der Zeit verändern und so Punktmengen modellieren, die sich bewegen. Wir zeigen, dass das Finden kleiner Separatoren in solchen temporalen Graphen NP-schwer ist. Zum Schluss besprechen wir noch eine Reihe verwandter Fragen, die offen bleiben.

Abstract

Proximity graphs are induced by sets of points in the plane or higher-dimensional structures. Such graphs describe which sets of points are close to one another and which are not. There are many different kinds of proximity graphs, corresponding to different definitions of when points are close. We will study three classes of proximity graphs: relative neighborhood graphs (RNG), relatively closest graphs (RCG), and Gabriel graphs. We investigate which classic NP-complete problems remain NP-complete on these graph classes. We show that with a few exceptions they do. We consider the algorithmic problems Independent Set, Vertex Cover, 3-Colorability, Dominating Set, Feedback Vertex Set, and Hamiltonian Cycle.

The reductions we use to prove these problems' NP-hardness are based on *two-page book embeddings*. Such book embeddings give very structured planar embeddings of certain graphs. We, therefore, also briefly consider the computation of book embeddings of the aforementioned proximity graphs. We also consider proximity graphs that change over time, modeling sets of points that are in motion. We show that finding small separators in such temporal graphs is NP-hard. Finally, we discuss several related questions that remain open.

Table of Contents

1 Introduction **6**
1.1 Related work . . . 9
1.2 Our contributions . . . 9

2 Preliminaries **12**
2.1 Graphs . . . 12
2.2 Planar graphs . . . 13
2.3 Proximity graphs . . . 14
2.4 Book embeddings . . . 18
2.5 Complexity . . . 20

3 Overview **21**

4 Dominating Set **25**
4.1 Definitions and intermediate results . . . 25
4.2 Reduction . . . 34
4.3 Conclusions . . . 37

5 Independent Set and Vertex Cover **39**
5.1 Definitions and intermediate results . . . 39
5.2 Reduction . . . 41
5.3 Conclusions . . . 46

6 3-Colorability **48**
6.1 Definitions and intermediate results . . . 48
6.2 Reduction . . . 50
6.3 Conclusions . . . 54

7 Feedback Vertex Set **56**
7.1 Definitions and intermediate results . . . 56
7.2 Reduction . . . 60
7.3 Conclusions . . . 68

8 Hamiltonian Cycle **69**
8.1 Definitions and intermediate results . . . 69
8.2 Reduction . . . 77
8.3 Conclusions . . . 84

9 Temporal separators **86**
9.1 A 3-SAT variant 86
9.2 Temporal graphs, paths, and separators 88
9.3 Reduction 89
9.4 Conclusions 91

10 Recognition **93**

11 Conclusion **95**

Literature **97**

1 Introduction

Given a set of points in the plane, which pairs of points are close to one another and which are far apart? Of course, the answer to this question depends on how one defines “close”. Frequently, it is convenient to describe proximity relationships using graphs. Such graphs are known as proximity graphs. In these graphs, points that are “close” are connected by an edge. An example of a set of points along with a proximity graph based on a purely intuitive notion of proximity is pictured in Figure 1.1.

Proximity graphs are mostly studied in computational geometry, but have a wide array of applications. Describing the proximity relationships between points in the plane or higher-dimensional structures is a problem that arises in several fields of science and engineering, most obviously in geography, less obviously in data mining [GS15; Vri+16], computer vision [Xu+19], the design of mobile ad-hoc networks [BCJ10; PKV16], the design of crowd-movement sensors [Chi+15], analyzing road traffic [Wat10], describing the spread of a species of mold [Ada09]. and the analysis of electroencephalograms (EEGs) for the extraction of information on the functionality of the human brain [Dim+18].

In order to illustrate proximity graphs' usefulness more concretely we will give an intuitive example. Such graphs could arise in the context of railway networks. If two cities are close, then it makes sense to build a railway directly from one to the other, but if they are too far apart, then one can expect passengers to take a train to some city “in between” and transfer there. When defining what cities are close, we may wish to ensure that the resulting network has certain properties:

1. We may want the network to be connected, since it should be possible to travel between any two cities.

2. We may want to avoid crossings so that we are not forced to build bridges or tunnels.

3. In order to save costs, we may want to minimize the total length of the tracks that need to be built.

4. At the same time, we also want passengers' trips to be as short as possible.

Generally, there is a trade-off between the last two criteria, since adding additional tracks increases the total track length, but reduces some passengers' trip lengths. If we only consider the first three, then the resulting network is what is known as the minimum spanning tree. If we only consider the fourth criterion, then the network is the complete graph or the point visibility graph (see Himmel et al. [Him+19]). The proximity graphs we will study will be somewhere between these two extremes, sacrificing some total track length in order to reduce passengers' travel times.

(a) Point set. (b) Corresponding proximity graph.

Figure 1.1: An intuitive example of a proximity graph: The vertices in the graph in (b) correspond to points in the point set on the depicted in (a). They are connected by edges if the corresponding points might intuitively be considered "close".

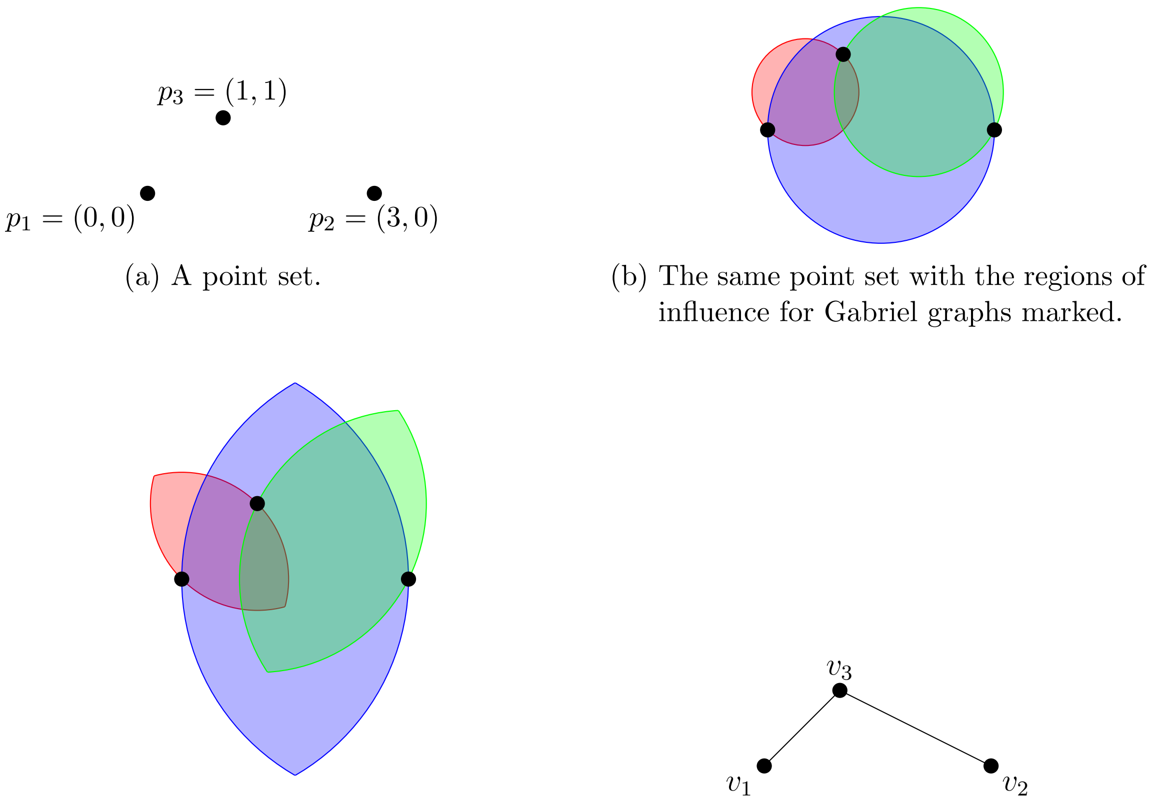
There are many different types of proximity graphs, corresponding to different approaches to defining when points are "close". We will study three types of proximity graphs:

- relative neighborhood graphs (RNGs),
- relatively closest graphs (RCGs), and
- Gabriel graphs.

All three are examples of what Cardinal, Collette, and Langerman [CCL09] called empty region graphs. An empty region graph is constructed from a set of points by defining a region of influence for every pair of points. Such a region of influence is a subset of the plane, usually a connected neighborhood of the points. Two points are deemed to be close if their region of influence does not contain any additional points. The proximity graph corresponding to a set of points is then constructed by representing points as vertices and connecting two vertices if the points they represent are close. We will limit ourselves to the two-dimensional case and the Euclidean metric, but one might also be interested in higher-dimensional spaces with different metrics.

Gabriel graphs were introduced by Gabriel and Sokal [GS69] in 1969. In a Gabriel graph, two points' region of influence is a circle whose center is midway between them and whose diameter is the distance between them. Also in 1969, Lankford [Lan69] defined relatively closest graphs. Their region of influence is the intersection of circles centered on each of the two points with a radius equal to the distance between the points. The closely related relative neighborhood graphs were introduced by Toussaint [Tou80] in 1980. Figure 1.2 illustrates how Gabriel graphs and relative neighborhood graphs are constructed using the region of influence of each pair of points. The regions of influence for relatively closest graphs are nearly identical to relative neighborhood graphs, differing only in that they include the regions' boundaries, so they are omitted from the example.

Most algorithmic research on proximity graphs has sought to devise algorithms that efficiently compute the graph from a point set. We will consider a different question.



(a) A point set.

(b) The same point set with the regions of influence for Gabriel graphs marked.

(c) The same point set with the regions of influence for RNGs marked.

(d) The resulting Gabriel graph and RNG.

Figure 1.2: Constructing Gabriel graphs and RNGs: (a) depicts a point set. (b) and (c) depict the same point set, but with the regions of influence for Gabriel graphs and RNGs, respectively, for each pair of points marked. In both cases, the regions of influence for p_1 and p_3 as well as for p_2 and p_3 do not contain any additional points, but the region for p_1 and p_2 (the blue region) contains p_3. (d) depicts the resulting Gabriel graph and RNG as they are identical. RCGs' regions of influence are nearly identical to RNGs'.

We will ask what properties can be tested for quickly in proximity graphs that are hard to test for in arbitrary graphs.

To illustrate what kind of properties we are interested in, we return to our railway example. We might want to build maintenance facilities for the tracks. Because these facilities are expensive, we wish to build them in as few cities as possible, but each track must have a maintenance facility at one of its endpoints. This is an instance of the Vertex Cover problem, which is well-known to be NP-hard to solve on arbitrary graphs. One could expect that the problem might be easier if we assume that the railway network is a proximity graph. Unfortunately, as we will show, it remains NP-hard to decide whether k such facilities suffice in order to maintain the entire network even when we consider only the proximity graphs we have described.

1.1 Related work

Relative neighborhood graphs, Gabriel graphs, and relatively closest graphs were first introduced by Toussaint [Tou80], Gabriel and Sokal [GS69], and Lankford [Lan69], respectively. Urquhart [Urq83], Matula and Sokal [MS80], and Cimikowski [Cim92] proved some basic combinatorial properties for each of the graph classes. Jaromczyk and Toussaint [JT92] surveyed early results on proximity graphs. Bose et al. [Bos+12] proved several additional results. So-called proximity trees, proximity graphs that are also trees, were analyzed by Bose, Lenhart, and Liotta [BLL96]. Lubiw and Sleumer [LS93] and Lenhart and Liotta [LL97] showed that many outerplanar graphs are relative neighborhood graphs and Gabriel graphs. Cardinal, Collette, and Langerman [CCL09] introduced the term empty region graphs, which include the three proximity graph classes under consideration, and studied what graph classes can be represented as empty region graphs.

Proximity graphs created from points in the plane drawn from some probability distribution are of particular interest in the design and analysis of mobile ad-hoc networks. Accordingly, the probabilistic combinatorial properties of such random proximity graphs have been studied extensively by Devroye [Dev88], Milic and Malek [MM06], Wan and Yi [WY07], Devroye, Gudmundsson, and Morin [DGM09], Yi et al. [Yi+10], Melchert [Mel13], Norrenbrock [Nor16], and others.

Most algorithmic research on proximity graphs has focused on devising algorithms that efficiently compute the proximity graph from a point set. Mitchell and Mulzer [MM17] gave a survey of such algorithmic results. Supowit [Sup83], Jaromczyk and Kowaluk [JK87], Katajainen, Nevalainen, and Teuhola [KNT87], Huang [Hua90], Lingas [Lin94], and Lavergne et al. [Lav+07] have given algorithms for computing relative neighborhood graphs. Matula and Sokal [MS80] have done so for Gabriel graphs. Lee [Lee85], Su and Chang [SC91], and Agarwal and Matoušek [AM92] presented algorithms for computing relative neighborhoods in higher-dimensional or non-Euclidean spaces. Cimikowski [Cim90] gave a heuristic for computing 4-colorings of Gabriel graphs. He also proposes a linear-time algorithm for computing a 4-coloring of a relative neighborhood graph, but, as we will discuss in Chapter 11, this algorithm is not correct and the question of whether such an algorithm exists remains open. Sundar and Khurd [SK17] gave parallel algorithms for computing cycle orders and cycle perimeters in relative neighborhood graphs. Hoffmann and Wanke [HW12] showed that the Metric Dimension problem is NP-complete when restricted to graphs that are both Gabriel graphs and unit disk graphs. Di Battista, Lenhart, and Liotta [DBLL94] surveyed research on the realizability of graphs as proximity graphs.

1.2 Our contributions

Our contributions are summarized in Table 1.1. We will analyze the restriction of eight problems to relative neighborhood graphs, relatively closest graphs, and Gabriel graphs. The first six problems, Dominating Set, Independent Set, Vertex Cover, 3-

	Relative neighborhood graphs	Relatively closest graphs	Gabriel graphs
DOMINATING SET (Theorem 4.13, Corollary 4.15)	NP-hard $\Delta \leq 4$ ETH $\Rightarrow 2^{o(n^{\frac{1}{4}})}$ lb	NP-hard $\Delta \leq 4$ ETH $\Rightarrow 2^{o(n^{\frac{1}{4}})}$ lb	NP-hard $\Delta \leq 4$ ETH $\Rightarrow 2^{o(n^{\frac{1}{4}})}$ lb
INDEPENDENT SET and VERTEX COVER (Theorem 5.6, Corollaries 5.7, 5.10)	NP-hard $\Delta \leq 4$ ETH $\Rightarrow 2^{o(n^{\frac{1}{4}})}$ lb	NP-hard $\Delta \leq 4$ ETH $\Rightarrow 2^{o(n^{\frac{1}{4}})}$ lb	NP-hard $\Delta \leq 4$ ETH $\Rightarrow 2^{o(n^{\frac{1}{4}})}$ lb
3-COLORABILITY (Theorems 6.6, 6.8, Corollary 6.10)	NP-hard $\Delta \leq 7$ ETH $\Rightarrow 2^{o(n^{\frac{1}{4}})}$ lb	trivial	NP-hard $\Delta \leq 7$ ETH $\Rightarrow 2^{o(n^{\frac{1}{4}})}$ lb
FEEDBACK VERTEX SET (Theorem 7.8, Corollary 7.10)	NP-hard $\Delta \leq 6$ ETH $\Rightarrow 2^{o(n^{\frac{1}{4}})}$ lb	NP-hard $\Delta \leq 6$ ETH $\Rightarrow 2^{o(n^{\frac{1}{4}})}$ lb	NP-hard $\Delta \leq 8$ ETH $\Rightarrow 2^{o(n^{\frac{1}{4}})}$ lb
HAMILTONIAN CYCLE (Theorem 8.10, Corollary 8.12)	NP-hard $\Delta \leq 4$ ETH $\Rightarrow 2^{o(n^{\frac{1}{4}})}$ lb	open	NP-hard $\Delta \leq 4$ ETH $\Rightarrow 2^{o(n^{\frac{1}{4}})}$ lb
(STRICT) TEMPORAL (s, t)-SEPARATION layer-wise class definition (Theorem 9.3, Corollaries 9.4, 9.5)	NP-hard all layers are paths ETH $\Rightarrow 2^{o(n+\sqrt{m})}$ lb	NP-hard all layers are paths ETH $\Rightarrow 2^{o(n+\sqrt{m})}$ lb	NP-hard all layers are paths ETH $\Rightarrow 2^{o(n+\sqrt{m})}$ lb
RECOGNITION (Theorem 10.1)	$\in \exists\mathbb{R}$	$\in \exists\mathbb{R}$	$\in \exists\mathbb{R}$
BOOK THICKNESS (Observation 2.12)	linear time	linear time	linear time

Table 1.1: Summary of our results: The small text beneath each complexity classification refers to additional constraints that may be placed on graphs where Δ denotes the maximum vertex degree. "ETH $\Rightarrow$" refers to lower bounds (lb) on the running time for algorithms for the respective problem based on the exponential time hypothesis. In the lower bounds, n refers to the number of vertices in a graph and m to the number of edges. Each problem is defined at the beginning of the chapter in which the results concerning this problem appear.

COLORABILITY, FEEDBACK VERTEX SET, and HAMILTONIAN CYCLE, are classical NP-complete problems. With two exceptions, namely 3-COLORABILITY and HAMILTONIAN CYCLE on RCGs, we will show that each of them remain NP-complete when restricted to these graph classes. In each case, we will also prove that if the exponential time hypothesis holds, then the problem cannot be solved in time $2^{o(n^{\frac{1}{4}})}$. Our reductions make heavy use of book embeddings, so we will also briefly discuss the complexity of computing the book thickness of the proximity graphs in question. Finally, we will turn to temporal proximity graphs. We will define temporal proximity graphs as temporal graphs in which every layer is a proximity graph. Such graphs model objects that move in the plane and whose proximity relationships therefore change over time. We will show that TEMPORAL (s,t)-SEPARATION on these graph classes is also NP-hard. For this problem an ETH-based lower bound of $2^{o(n+\sqrt{m})}$ holds. We will briefly discuss the recognition problem for these graph classes. While we cannot give any tight bounds on its complexity, we will show that it is in the complexity class known as the existential theory of the reals ($\exists\mathbb{R}$). We will conclude by discussing several questions related to our results that remain open.

2 Preliminaries

2.1 Graphs

We will now define the basic concepts of graph theory we will use throughout this thesis (see, for instance, Diestel [Die17]).

If S is a set and $k \in \mathbb{N}$, then $\binom{S}{k}$ is the set of all subsets of S containing exactly k elements. A *graph* $G = (V, E)$ is a pair consisting of any finite set V and $E \subseteq \binom{V}{2}$. The elements of V are called *vertices* and those of E are *edges*. Frequently, we will represent graphs visually by drawing the vertices as points in the plane and the edges as line segments or curves connecting them. Two vertices $u, v \in V$ are *adjacent* or *neighbors* if $\{u, v\} \in E$. If $v \in V$ is a vertex, then $N(v) \coloneqq \{u \in V \mid \{u, v\} \in E\}$ is v's *open neighborhood*, the set of all vertices adjacent to v. The vertex v's *closed neighborhood* is $N[v] \coloneqq N(v) \cup \{v\}$. When we simply refer to a vertex's neighborhood, we mean its open neighborhood. The *degree* of a vertex $v \in V$ is $\deg(v) \coloneqq |N(v)|$, the size of its open neighborhood. A vertex v is *isolated* if $\deg(v) = 0$. The *maximum degree* of a graph $G = (V, E)$ is $\Delta(G) \coloneqq \max_{v \in V} \deg(v)$. A graph is *$k$-regular* if every vertex in G has degree k.

The graph $G' = (V', E')$ is a *subgraph* of $G = (V, E)$ if $V' \subseteq V$ and $E' \subseteq E \cap \binom{V'}{2}$. It is an *induced subgraph* if $E' = E \cap \binom{V'}{2}$. If $V' \subseteq V$, then $G[V'] \coloneqq (V', E \cap \binom{V'}{2})$ is the subgraph of G *induced by* V'. We will use $G - V'$ to denote $G[V \setminus V']$. Similarly, if $G' = (V', E')$ is a subgraph of $G = (V, E)$, then we will use $G - G'$ as a shorthand for $G - V'$. If $G = (V, E)$ is a graph and $\{u, v\} \in E$ an edge, then the graph obtained by *subdividing* $\{u, v\}$ is the graph $G' \coloneqq (V \dot\cup \{w\}, (E \setminus \{\{u, v\}\}) \cup \{\{u, w\}, \{v, w\}\})$. Two graphs $G = (V, E)$ and $G' = (V', E')$ are *isomorphic* if they are structurally identical, that is, if there is a bijective map $\varphi \colon V \to V'$ such that $\{u, v\} \in E$ if and only if $\{\varphi(u), \varphi(v)\} \in E'$ for all $u, v \in V$. We will say that a graph G *contains* a graph G' or *contains a copy* of G' if G has a subgraph that is isomorphic to G'.

The *complete graph on n vertices* is $K_n \coloneqq (\{v_1, \dots v_n\}, \{\{v_i, v_j\} \mid 1 \leq i < j \leq n\})$. A *clique* of size k in a graph G is a subgraph isomorphic to K_k. The *path on n vertices* is the graph $P_n \coloneqq (\{v_1, \dots, v_n\}, \{\{v_i, v_{i+1}\} \mid 1 \leq i \leq n - 1\})$. A *path* of length $k - 1$ in a graph G is a subgraph isomorphic to P_k. We will also denote a path of length $k - 1$ as a sequence of vertices $v_1, \dots, v_k$ such that $v_i \neq v_j$ if $i \neq j$, and $\{v_i, v_{i+1}\} \in E$ for all $i = 1, \dots, k - 1$. A path's *endpoints* are the two vertices of degree one. The *distance* between two vertices is the length of the shortest path that contains both of them. A graph $G = (V, E)$ is *connected* if for any two vertices $u, v \in V$ there is a path with endpoints u and v. The *cycle on n vertices*, $n \geq 3$, is $C_n \coloneqq (\{v_1, \dots, v_n\}, \{\{v_i, v_{i+1}\} \mid 1 \leq i \leq n-1\} \cup \{\{v_1, v_n\}\})$. A *cycle* of length k in a graph G is a subgraph isomorphic to C_k.

Similarly to paths, we will also denote cycles of length k as sequences of vertices $v_1, \dots, v_k$ such that $v_i \neq v_j$ if $i \neq j$, $\{v_i, v_{i+1}\} \in E$ for all $i = 1, \dots, k-1$, and $\{v_k, v_1\} \in E$. A graph is *acyclic* if it does not contain any cycles. Note that paths and cycles are not necessarily induced subgraphs. The *complete bipartite graph with parts of size* n_1 *and* n_2 is the graph $K_{n_1,n_2} \coloneqq (\{u_1, \dots, u_{n_1}, v_1, \dots, v_{n_2}\}, \{\{u_i, v_j\} \mid 1 \leq i \leq n_1, 1 \leq j \leq n_2\})$. The *wheel graph on* n *vertices* is the graph $W_n \coloneqq (\{v_1, \dots, v_n\}, \{\{v_i, v_{i+1}\} \mid 1 \leq i \leq n-2\} \cup \{\{v_n, v_i\} \mid 1 \leq i \leq n-1\})$. The $(n_1 \times n_2)$-grid graph is $G_{n_1,n_2} \coloneqq (\{v_{i,j} \mid 1 \leq i \leq n_1, 1 \leq 2 \leq n_2\}, \{\{v_{i,j}, v_{i+1,j}\} \mid 1 \leq i \leq n_1 - 1, 1 \leq j \leq n_2\} \cup \{\{v_{i,j}, v_{i,j+1}\} \mid 1 \leq i \leq n_1, 1 \leq j \leq n_2 - 1\})$.

An *independent set* in a graph $G = (V, E)$ is a set of pairwise non-adjacent vertices $\mathcal{I} \subseteq V$. The *independence number* $\alpha(G)$ of a graph G is the size of the largest independent set in G. A set of vertices $\mathcal{S} \subseteq V$ in a graph $G = (V, E)$ is a *vertex cover* if $G - S$ does not contain any edges. The *vertex cover number* $\tau(G)$ of a graph G is the size of the smallest vertex cover in this graph. For any graph G containing n vertices, $n = \alpha(G) + \tau(G)$. A *dominating set* in a graph $G = (V, E)$ is a set of vertices $\mathcal{D} \subseteq V$ such that $\bigcup_{v \in \mathcal{D}} N[v] = V$. In other words, $\mathcal{D}$ is a dominating set if every vertex $v \in V \setminus \mathcal{D}$ has a neighbor in $\mathcal{D}$. The *domination number* $\gamma(G)$ of a graph G is the size of the smallest dominating set in G. A *feedback vertex set* in a graph $G = (V, E)$ is a set $\mathcal{F} \subseteq V$ such that $G - \mathcal{F}$ is acyclic. We will use $\varphi(G)$ to denote the size of the smallest feedback vertex set in G. A k*-coloring* of the graph $G = (V, E)$ is a map $c\colon V \to \{1, \dots, k\}$ such that $c(u) \neq c(v)$ for all $\{u, v\} \in E$. A graph G is k*-colorable* if there is a k-coloring of G. A *Hamiltonian cycle* in a graph $G = (V, E)$ consisting of $n = |V|$ vertices is a set of edges $E' \subseteq E$ such that (V, E') is isomorphic to the cycle graph C_n. Alternatively, if $V = \{v_1, \dots, v_n\}$, then we will say that the sequence of vertices $v_{i_1}, \dots, v_{i_n}$ is a *Hamiltonian cycle*, if $i_j \neq i_{j'}$ for all $j \neq j'$, v_{i_j} is adjacent to $v_{i_{j+1}}$ for all $j = 1, \dots, n-1$, and v_{i_n} is adjacent to v_{i_1}. A graph is *Hamiltonian* if it contains a Hamiltonian cycle.

2.2 Planar graphs

Intuitively, a graph is *planar* if it may be drawn in the plane in a way that no two edges intersect except possibly in their endpoints. A formal definition requires a few topological concepts. We use $d\colon \mathbb{R}^2 \to \mathbb{R}$ with $d((x_1, y_1), (x_2, y_2)) \coloneqq \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$ to denote the *Euclidean distance* between two points in the plane. If $p \in \mathbb{R}^k$ is a point in the Euclidean k-dimensional space, then the *open ball* around p with radius r is $B_r(p) \coloneqq \{q \in \mathbb{R}^k \mid d(p, q) < r\}$. The *closed ball* is $\overline{B}_r(p) \coloneqq \{q \in \mathbb{R}^k \mid d(p, q) \leq r\}$. A set of points $S \subseteq \mathbb{R}^k$ is *open* if for every $p \in S$ there is an $\varepsilon > 0$ such that $B_\varepsilon(p) \subseteq S$. Let $X \subseteq \mathbb{R}^k$ and $Y \subseteq \mathbb{R}^\ell$. A function $f\colon X \to Y$ is *continuous* if the inverse image $f^{-1}(S) \coloneqq \{p \in X \mid f(p) \in S\}$ of any open subset $S \subseteq Y$ is open. A function $f\colon X \to Y$ is a *homeomorphism* if it is bijective and continuous and f's inverse $f^{-1}\colon Y \to X$ is also continuous. The sets X and Y are *homeomorphic* if there is a homeomorphism mapping X to Y. The *straight line segment* between two points $p, q \in \mathbb{R}^k$ is $\{p + \lambda(q - p) \mid 0 \leq \lambda \leq 1\}$. An *arc* is the union of finitely many straight line segments that is homeomorphic to $[0, 1] \subseteq \mathbb{R}$. (This is actually what is generally known as a polygonal arc, but defining arcs in general requires more topological concepts and is unnecessary for our purpose.) A point $p \in A$ in an arc A is an *endpoint* of A if $f(p) = 0$ or $f(p) = 1$ where f is any homeomorphism between A and $[0, 1]$. A *plane graph* is a pair $G = (V, E)$ such that the vertex set $V \subseteq \mathbb{R}^2$ consists of finitely many points in the plane, every edge $e \in E$ is an arc whose endpoints are vertices, no two edges share both of their endpoints, and edges intersect only in their endpoints. A graph $G = (V, E)$ is *planar* if there is a plane graph $G' = (V', E')$ and a bijective map $f : V \to V'$ such that there is an edge $e = \{u, v\} \in E$ if and only if there is an edge $e' \in E'$ with endpoints $f(u)$ and $f(v)$. The plane graph G' and the function f are called a *planar embedding* or a *planar drawing* of G.

Any cycle in a planar drawing of a graph divides the plane into two regions. One of these regions is bounded and is known as the *inside* of the cycle. The other region is unbounded and is known as the *outside* of the cycle. The disjoint components of the plane created by the graph are the *faces*. There is one face, known as the *outer face*, which is outside of every cycle. A graph is *outerplanar* if it admits a planar embedding in which every vertex touches the outer face. A planar drawing of a graph is *triangulated* if every face is bounded by a 3-cycle. Any planar drawing may be triangulated by adding edges. It is well-known that K_5 and $K_{3,3}$ are not planar. It is also easy to see that if G is planar, then so is every subgraph of G. A planar graph containing $n \geq 3$ vertices may contain at most $3n - 6$ edges. Finally, by the celebrated four color theorem, every planar graph is 4-colorable.

2.3 Proximity graphs

We will now describe the three classes of proximity graphs that we will investigate. They may be characterized in several equivalent ways, which we will present. We will also describe several previously known combinatorial results on these graphs.

Definition 2.1. A *straight-line embedding* is a plane graph in which every edge is a straight line segment.

The following statement is known as Fáry's theorem:

Theorem 2.2 ([Fár48])**.** *Every planar graph admits a planar straight-line embedding.*

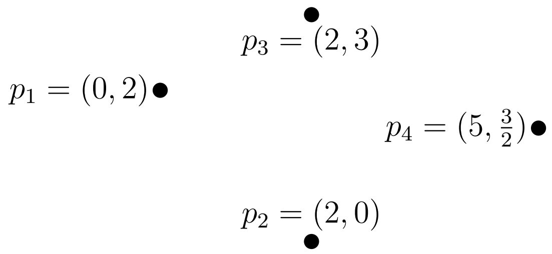
We will now define the three graph classes in which we are primarily interested. Recall that d is the Euclidean distance in the plane.

Definition 2.3. Let $P = \{p_1, \dots, p_n\} \subseteq \mathbb{R}^2$ be a finite set of points in the plane. We define a set of vertices $V \coloneqq \{v_1, \dots, v_n\}$ corresponding to these points. The *relative neighborhood graph* induced by P is $\mathrm{RNG}(P) \coloneqq (V, E_{\mathrm{RNG}})$ with:

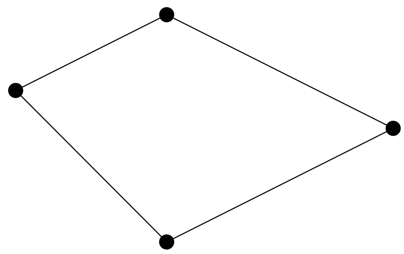
$$E_{\mathrm{RNG}} \coloneqq \{\{v_i, v_j\} \mid d(p_i, p_j) \leq \max\{d(p_i, p_k), d(p_j, p_k)\} \text{ for all } k \in \{1, \dots, n\}\}.$$

The *relatively closest graph* induced by P is $\mathrm{RCG}(P) = (V, E_{\mathrm{RCG}})$ with:

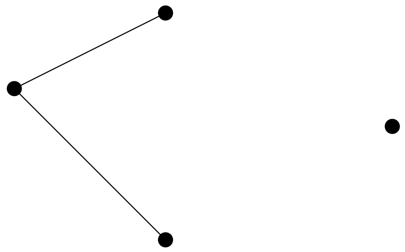
$$E_{\mathrm{RCG}} \coloneqq \{\{v_i, v_j\} \mid d(p_i, p_j) < \max\{d(p_i, p_k), d(p_j, p_k)\} \text{ for all } k \in \{1, \dots, n\}\}.$$



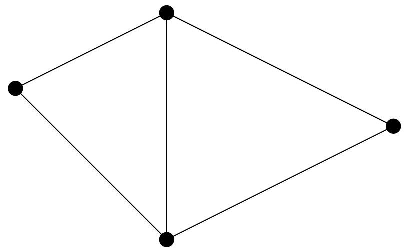
(a) A point set P consisting of four points.



(b) The relative neighborhood graph induced by P.



(c) The relatively closest graph induced by P.



(d) The Gabriel graph induced by P.

Figure 2.1: A point set P along with the relative neighborhood, relatively closest, and Gabriel graph induced by P. Each of these graphs may be computed using the distances listed in Table 2.1.

	p_1	p_2	p_3	p_4
p_1		$2\sqrt{2}$	$\sqrt{5}$	$\frac{\sqrt{101}}{2}$
p_2			3	$\frac{\sqrt{45}}{2}$
p_3				$\frac{\sqrt{45}}{2}$

Table 2.1: Pairwise distances between points in the set P pictured in Figure 2.1a.

The *Gabriel graph* induced by P is $\mathrm{GAB}(P) = (V, E_{\mathrm{GAB}})$ with:

$$E_{\mathrm{GAB}} \coloneqq \{\{v_i, v_j\} \mid d(p_i, p_j)^2 < d(p_i, p_k)^2 + d(p_j, p_k)^2 \text{ for all } k \in \{1, \dots, n\}\}.$$

A straight-line embedding $\mathrm{emb}\colon V \to \mathbb{R}^2$ is an *RNG-embedding* of $G = (V, E)$ if $\mathrm{RNG}(\mathrm{emb}(V)) = G$, a *Gabriel-embedding* of G if $\mathrm{GAB}(\mathrm{emb}(V)) = G$, and an *RCG-embedding* of G if $\mathrm{RCG}(\mathrm{emb}(V)) = G$.

Let RNG denote the class of all graphs G which admit an RNG-embedding. Similarly, let RCG denote the class of all graphs that admit an RCG-embedding and let GAB denote the class of all those which admit a Gabriel-embedding.

These definitions directly imply the following:

Lemma 2.4. *Relatively closest graphs, relative neighborhood graphs, and Gabriel graphs* *are planar and* $E_{\mathrm{RCG}} \subseteq E_{\mathrm{RNG}} \subseteq E_{\mathrm{GAB}}$ *for any point set. RNGs and Gabriel graphs are always connected. RCGs cannot contain any cliques of size three.*

Figure 2.1 illustrates the definitions of RNGs, RCGs, and Gabriel graphs. The point set P pictured in Figure 2.1a consists of four points. The pairwise distances between points are listed in Table 2.1. These distances can be used to compute the RNG, RCG, and Gabriel graph induced by the point set. Those graphs are pictured in Figures 2.1b to 2.1d.

Given an RNG-embedding emb of G, we will say that v_1 is an *RNG-blocker* for v_2 and v_3 if $d(\mathrm{emb}(v_1), \mathrm{emb}(v_2)) < d(\mathrm{emb}(v_2), \mathrm{emb}(v_3))$ and $d(\mathrm{emb}(v_1), \mathrm{emb}(v_3)) < d(\mathrm{emb}(v_2), \mathrm{emb}(v_3))$. Clearly, $\{v_i, v_j\} \in E_{\mathrm{RNG}}$ if and only there is no RNG-blocker for v_i and v_j. For an RCG-embedding emb of G, v_1 is an *RCG-blocker* for v_2 and v_3 if $d(\mathrm{emb}(v_1), \mathrm{emb}(v_2)) \leq d(\mathrm{emb}(v_2), \mathrm{emb}(v_3))$ and $d(\mathrm{emb}(v_1), \mathrm{emb}(v_3)) \leq d(\mathrm{emb}(v_2), \mathrm{emb}(v_3))$. It follows that $\{v_i, v_j\} \in E_{\mathrm{RCG}}$ if and only if emb does not have an RCG-blocker for v_i and v_j. Similarly, if emb is a Gabriel-embedding of G, then we will call v_1 a *Gabriel-blocker* for v_2 and v_3 if $d(\mathrm{emb}(v_2), \mathrm{emb}(v_3))^2 \geq d(\mathrm{emb}(v_1), \mathrm{emb}(v_2))^2 + d(\mathrm{emb}(v_1), \mathrm{emb}(v_3))^2$. Again, v_i and v_j are adjacent in G if and only if there is no Gabriel-blocker for v_i an v_j.

For three points p_1, p_2, p_3 in the plane, we will use $\measuredangle p_1p_2p_3$ to denote the angle at p_2 formed by the vectors from p_2 to p_1 and p_2 to p_3. In the following, assume that $\mathrm{emb}(v_i) = p_i$.

RNGs can also be described in terms of angles rather than distances:

Lemma 2.5 ([Urq83])**.** *The vertex* v_1 *is an RNG-blocker for* v_2 *and* v_3 *if and only if* $\measuredangle p_1p_2p_3 < \measuredangle p_2p_1p_3$ *and* $\measuredangle p_1p_3p_2 < \measuredangle p_2p_1p_3$*. In other words,* v_1 *is a blocker if the angle at* p_1 *is the uniquely largest angle in the triangle formed by* p_1*,* p_2*, and* p_3*.*

Hence,

$$E_{\mathrm{RNG}} = \{\{v_i, v_j\} \mid \measuredangle p_ip_kp_j \leq \max\{\measuredangle p_kp_ip_j, \measuredangle p_ip_jp_k\} \textit{ for all } k \in \{1, \ldots, n\}\}.$$

Relatively closest graphs can be characterized in the same way:

Lemma 2.6 ([Cim92])**.** *The vertex* v_1 *is an RCG-blocker for* v_2 *and* v_3 *if and only if* $\measuredangle p_1p_2p_3 \leq \measuredangle p_2p_1p_3$ *and* $\measuredangle p_1p_3p_2 \leq \measuredangle p_2p_1p_3$*. In other words,* v_1 *is a blocker if the angle at* p_1 *is a largest angle in the triangle formed by* p_1*,* p_2*, and* p_3*.*

Hence,

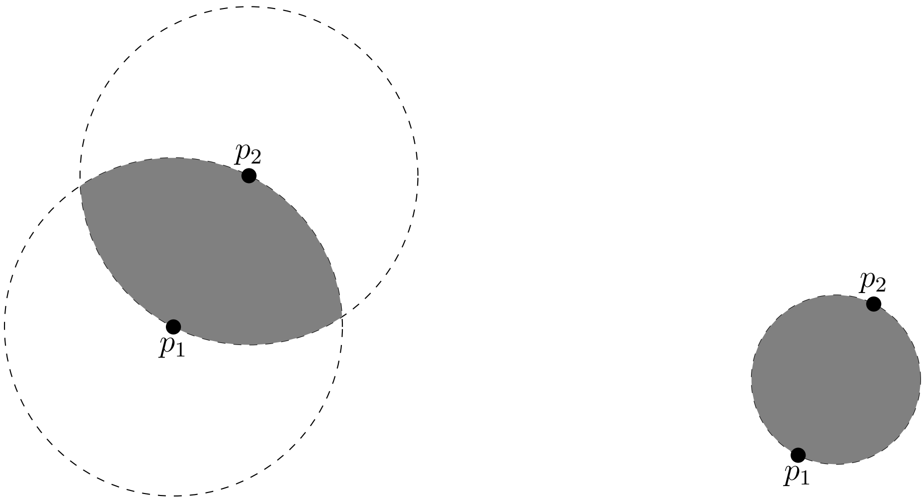
$$E_{\mathrm{RNG}} = \{\{v_i, v_j\} \mid \measuredangle p_ip_kp_j < \max\{\measuredangle p_kp_ip_j, \measuredangle p_ip_jp_k\} \textit{ for all } k \in \{1, \ldots, n\}\}.$$

Gabriel graphs can also be described in terms of angles:

Lemma 2.7 ([MS80])**.** *The vertex* v_1 *is a Gabriel-blocker for* v_2 *and* v_3 *if and only if* $\measuredangle p_2p_1p_3 \geq 90°$*.*

Hence,

$$E_{\mathrm{GAB}} = \{\{v_i, v_j\} \mid \measuredangle p_ip_kp_j < 90° \textit{ for all } k \in \{1, \ldots, n\}\}.$$



(a) The RNG-region of influence of the points $p_1 = (1, 1)$ and $p_2 = (2, 3)$ is the shaded area. It is the intersection of the two circles bounded by the dashed lines.

(b) The shaded area is the Gabriel-region of influence of the same points.

Figure 2.2: Regions of influence: The vertices corresponding to two points are adjacent if the two points' region of influence does not contain any additional points.

RNGs, RCGs, and Gabriel graphs are *empty region graphs*, that is, they can be described in terms of areas surrounding an edge, known as *regions of influence*, which may not contain any additional points. An example of an RNG-region of influence is pictured in Figure 2.2a. For two points $p_1, p_2 \in \mathbb{R}^2$, the *RNG-region of influence* consists of the intersection of the open balls with radius $r = d(p_1, p_2)$ centered on p_1 and p_2. Such an intersection of two balls is known as a lune. More formally, we will denote the RNG-region of influence as:

$$\mathrm{roi}_{\mathrm{RNG}}(p_1, p_2) \coloneqq B_r(p_1) \cap B_r(p_2).$$

Similarly the *RCG-region of influence* consists of the intersection of two closed balls of radius $r = d(p_1, p_2)$, but does not contain the points p_1 and p_2 themselves:

$$\mathrm{roi}_{\mathrm{RCG}}(p_1, p_2) \coloneqq (\overline{B}_r(p_1) \cap \overline{B}_r(p_2)) \setminus \{p_1, p_2\}.$$

For two points p and q, let $\mathrm{mid}(p, q)$ denote the point halfway between p and q. That is, if $p = (x_1, y_1)$ and $q = (x_2, y_2)$, then $\mathrm{mid}(p, q) = (\frac{x_1 - x_2}{2}, \frac{y_1 - y_2}{2})$. An example of a Gabriel-region of influence is depicted in Figure 2.2b. The *Gabriel-region of influence* is simply the open ball with radius $r \coloneqq \frac{d(p_1, p_2)}{2}$ centered on the point halfway between p_1 and p_2. That is:

$$\mathrm{roi}_{\mathrm{GAB}}(p_1, p_2) \coloneqq \overline{B}_r(\mathrm{mid}(p_1, p_2)).$$

Then, the following lemma, which has been established in the literature, holds:

Lemma 2.8 ([Urq83], [MS80]). *Let $\mathcal{C} \in \{\text{RNG}, \text{RCG}, \text{GAB}\}$. The vertex v_1 is a $\mathcal{C}$-blocker for v_2 and v_3 if and only if* $\text{emb}(v_1) \in \text{roi}_{\mathcal{C}}(\text{emb}(v_2), \text{emb}(v_3))$*.*

Hence,

$$E_{\mathcal{C}} = \{\{v_i, v_j\} \mid \text{emb}(V) \cap (\text{roi}_{\mathcal{C}}(\text{emb}(v_i), \text{emb}(v_j)) = \emptyset\}.$$

The following lemma holds equally for relative neighborhood graphs, relatively closest graphs, and Gabriel graphs. It follows directly from Lemmas 2.5 to 2.7.

Lemma 2.9 ([Urq83], [Cim92], and [MS80]). *Let $\mathcal{C} \in \{\text{RNG}, \text{RCG}, \text{GAB}\}$. Suppose that $G \in \mathcal{C}$ with a fixed $\mathcal{C}$-embedding* emb*, that u_1, u_2, and u_3 form a 3-cycle, and that v_1, v_2, v_3, and v_4 form a 4-cycle. Then, there is no w with* $\text{emb}(w)$ *inside the triangle formed by* $\text{emb}(u_1)$, $\text{emb}(u_2)$, *and* $\text{emb}(u_3)$ *or the quadrilateral generated by* $\text{emb}(v_1)$, $\text{emb}(v_2)$, $\text{emb}(v_3)$, $\text{emb}(v_4)$*.*

In other words, in an embedding of any of these proximity graphs 3-cycles and 4-cycles are empty, in that their interiors do not contain any additionally vertices. This is a fairly strong property. For instance, it can be used to show that no such proximity graphs may contain K_4 or $K_{2,3}$ as subgraphs.

2.4 Book embeddings

We will use book embeddings extensively in our NP-hardness proofs since they give us very structured representations of certain graphs. Book embeddings were first introduced in 1979 by Bernhart and Kainen [BK79]. Intuitively, a book embedding places the vertices of a graph on the spine of a book, while its edges are embedded in the pages. Such embeddings are useful in a variety of applications, including VLSI design, parallel computing, and the design of fault-tolerant systems [CLR87]. We will use them to design polynomial-time reductions, similarly to the way they are used by Fluschnik et al. [Flu+18].

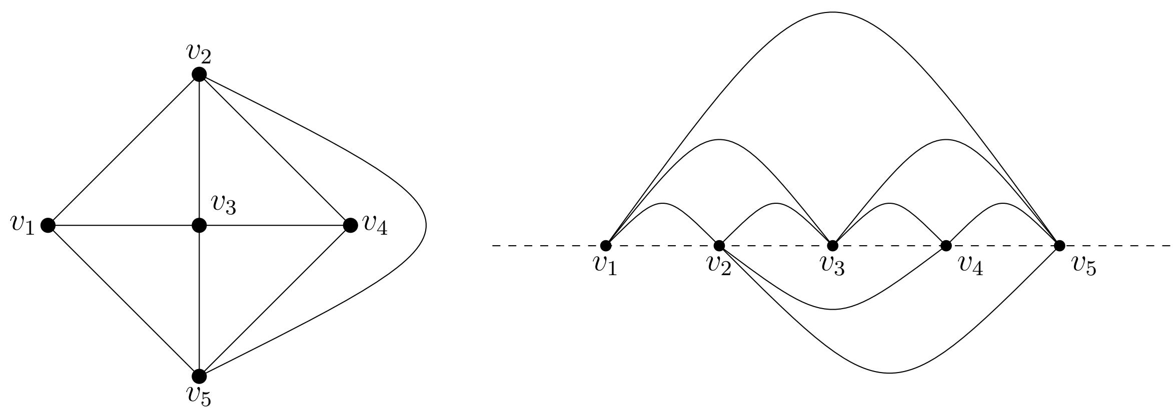
Definition 2.10. A *k-page book embedding* of a graph $G = (V, E)$ consists of:

- a partition of the edge set $E = E_1 \dot{\cup} \ldots \dot{\cup} E_k$ into k pages,
- an embedding $\text{emb} : V \to \mathbb{R}$ of the vertices onto the real line, and
- a planar embedding of each page (V, E_i) in the half-plane $\mathbb{R} \times \mathbb{R}_{\geq 0}$ which maps every $v \in V$ onto $(\text{emb}(v), 0)$.

The *book thickness* of a graph G is the minimum number k such that G is k-page book-embeddable.

Figure 2.3 pictures an example of a two-page book embedding of a graph G. A graph G is pictured in Figure 2.3a and a book embedding of G is in Figure 2.3b.

A graph is *subhamiltonian* if it is a subgraph of a planar Hamiltonian graph. Graphs with small book thickness are planar and were fully characterized by Bernhart and Kainen [BK79] as follows:



(a) A graph G.

(b) A two-page book embedding of G: The dashed line represents the spine of the book. The edges drawn above this line form E_1. Those drawn below the line form E_2.

Figure 2.3: Example of a two-page book embedding.

Lemma 2.11 ([BK79]). *A graph has:*

- *a book thickness of* 0 *if and only if it is edgeless,*
- *a book thickness of* 1 *if and only if it is outerplanar, and*
- *a book thickness of* 2 *if and only if it is subhamiltonian.*

This lemma implies that the graph pictured in Figure 2.3 requires two pages, since it is not outerplanar. Since we will be concerned with both book embeddings and proximity graphs, we briefly note the following as an aside:

Observation 2.12. *Relatively closest graphs, relative neighborhood graphs, and Gabriel graphs have a book thickness of at most* 2*. Their book thickness can be computed in linear time.*

Proof. Kainen and Overbay [KO03] define *nicely planar graphs* as planar graphs which admit an embedding in which every 3-cycle bounds a face and prove that nicely planar graphs are subhamiltonian. By Lemma 2.9, RCGs, RNGs, and Gabriel graphs are nicely planar. By Lemma 2.11, this implies that they have a book thickness of at most two.

As a result, their book thickness can be computed in linear time because it is possible to check whether they are edgeless or outerplanar in linear time [Mit79]. □

Heath [Hea85] proved that two-page book embeddings exist for all planar graphs with maximum degree at most 3 and can be computed in linear time. More recently, Bekos, Gronemann, and Raftopoulou [BGR16] extended this result:

Theorem 2.13 ([BGR16]). *Every planar graph with maximum degree at most* 4 *admits a two-page book embedding. Such an embedding can be computed in quadratic time.*

2.5 Complexity

We will prove conditional lower bounds for the running time of algorithms for the restrictions of several graph problems to proximity graphs. We will primarily focus on lower bounds derived from classical complexity theory and the theory of NP-hardness. In other words, most of our results will be based on the conjecture that P $\neq$ NP, that is, the conjecture that there are no polynomial-time algorithms for NP-complete problems. We assume that the reader is familiar with the theory of NP-hardness and will not define its terminology and central results. The reader may consult Garey and Johnson [GJ79] or Arora and Barak [AB09], for example.

Recently, newer paradigms have emerged within complexity theory that allow a more fine-grained analysis of the complexity of computational problems based on stronger conjectures than P $\neq$ NP. One such approach is the exponential time hypothesis (ETH), from which lower bounds for numerous problems may be derived.

In order to state the ETH, we must first define the 3-SAT problem. A *clause* over the variables $x_1, \ldots, x_n$ is a subset of $\{x_1, \ldots, x_n, \neg x_1, \ldots, \neg x_n\}$ ($x_1, \ldots, x_n, \neg x_1, \ldots, \neg x_n$ are called *literals*). A *Boolean conjunctive formula* over $x_1, \ldots, x_n$ is a set of clauses $C_1, \ldots, C_m$. Such a formula is *satisfiable* if there is a truth assignment $\alpha\colon \{x_1, \ldots, x_n\} \to \{0, 1\}$ such that for every clause C_i there is a variable x_j such that:

- $x_j \in C_i$ and $\alpha(x_j) = 1$ or
- $\neg x_j \in C_i$ and $\alpha(x_j) = 0$.

The 3-SAT problem is defined as follows:

3-SAT
Input: A Boolean conjunctive formula φ consisting of the clauses $C_1, \ldots, C_m$ over the variables $x_1, \ldots, x_n$ such that $|C_i| \leq 3$ for all C_i.
Question: Is φ satisfiable?

The exponential time hypothesis was introduced in 2001 by Impagliazzo and Paturi [IP01]. In a simplified form, it states:

Conjecture 2.14 ([IP01])**.** *The* 3-SAT *problem does not admit an algorithm with running time* $2^{o(n)}(n+m)^{\mathcal{O}(1)}$.

Lokshtanov, Marx, and Saurabh [LMS11] give a very useful survey of lower bounds that may be derived if one assumes the ETH.

3 Overview

In the following, we will show that several well-known NP-complete problems remain NP-complete when restricted to relative neighborhood graphs, relatively closest graphs, and Gabriel graphs. We will do so by giving polynomial-time many-to-one reductions from problems known to be NP-complete. We will consider the following problems: Dominating Set, Independent Set, Vertex Cover, 3-Colorability, Feedback Vertex Set, and Hamiltonian Cycle. Finally, we will define a notion of temporal proximity graphs and show that the Temporal (s,t)-Separation problem is NP-complete on such temporal graphs. In each case, we will also discuss ETH-based lower bounds that follow from the reduction.

We should be specific about what we mean when we refer to the restriction of a decision problem to a class of proximity graphs. We assume that the input to the decision problem consists only of the graph and, in some cases, a parameter k. The graph embedding is not part of the input. Hence, we do not have to worry about computations involving real numbers. One could suspect that the version of the computational problem where an embedding is given to the algorithm as part of the input might be easier than the version in which the embedding is not given. For the problems we consider, our reductions actually prove that this is not the case, since an embedding for the graph generated by the reduction is constructed along with the graph.

The reductions we will give for the first six problems follow the same general pattern. Before we begin, we will define some terminology that will be useful in these reductions and we will describe said pattern.

Consider a graph $G = (V, E)$ and a two-page book embedding of G. Let $v_1, \dots, v_n$ be the vertices of the graph in the order in which they appear on the spine. The book embedding induces a partition of G's edges $E = E_1 \dot\cup E_2$. We will use $N_1(v) \coloneqq \{v' \mid \{v, v'\} \in E_1\}$ to denote v's E_1-*neighborhood* and $\deg_1(v) \coloneqq |N_1(v)|$ to denote v's E_1-*degree*. We may define $N_2(v)$ and $\deg_2(v)$ in the same way with edges from E_2. For an edge $e = \{v_i, v_j\}$, $i < j$, define its *length* as:

$$\ell(e) \coloneqq j - i.$$

The *interior* of e is

$$\mathrm{int}(e) \coloneqq \{e' = \{v_r, v_s\} \mid i \leq r < s \leq j, e' \neq e, \text{ and } e \text{ and } e' \text{ are both in } E_1 \text{ or both in } E_2\}.$$

The *height* of e is

$$h(e) \coloneqq \begin{cases} 1 + \max\limits_{e' \in \mathrm{int}(e)} h(e'), & \text{if } \mathrm{int}(e) \neq \emptyset, \\ 1, & \text{if } \mathrm{int}(e) = \emptyset. \end{cases}$$

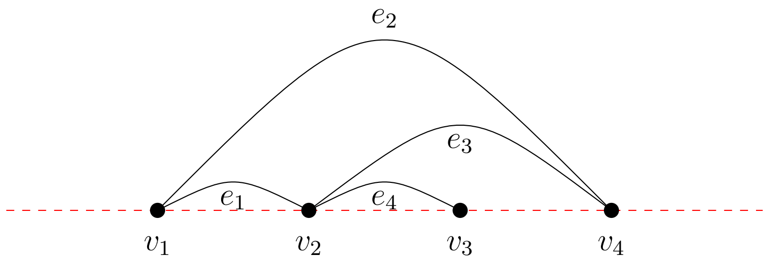


Figure 3.1: A two-page book embedding of a graph G: All edges are in E_1.

The E_1-*height* of a vertex v_i is

$$h_1(v_i) \coloneqq \begin{cases} \max\{h(e) \mid e = \{v_i, v_j\} \text{ and } e \in E_1\}, & \text{if } N_1(v_i) \neq \emptyset, \\ 0, & \text{if } N_1(v_i) = \emptyset. \end{cases}$$

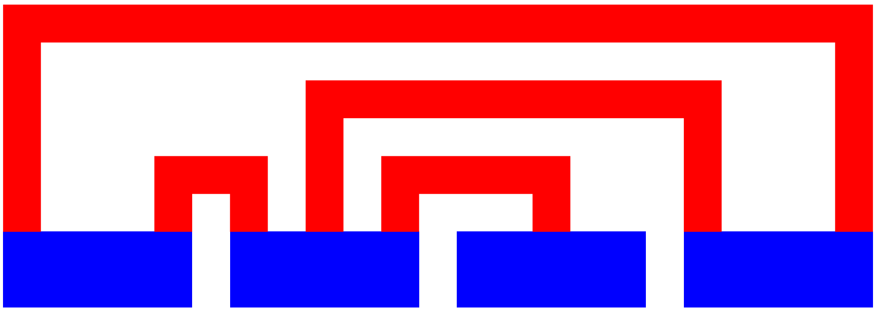
Similarly, define the E_2-*height* $h_2(v_i)$ by the maximum height of edges incident to v_i in E_2. Let $h_1(G) \coloneqq \max\{h_1(v_i) \,|\, i \in \{1, \ldots, n\}\}$, $h_2(G) \coloneqq \max\{h_2(v_i) \,|\, i \in \{1, \ldots, n\}\}$. Note that, because the height of any edge only depends on the height of shorter edges, edge height is well-defined. The length and height of an edge are both in $\mathcal{O}(n)$.

We will now describe the general idea behind each of our reductions. Figure 3.1 pictures a two-page book embedding of a graph G. Note that $\ell(e_1) = \ell(e_4) = 1$, $\ell(e_2) = 3$, $\ell(e_3) = 2$, $h(e_1) = h(e_4) = 1$, $h(e_3) = 2$, and $h(e_2) = 3$. We will use this graph to illustrate each of our reductions.

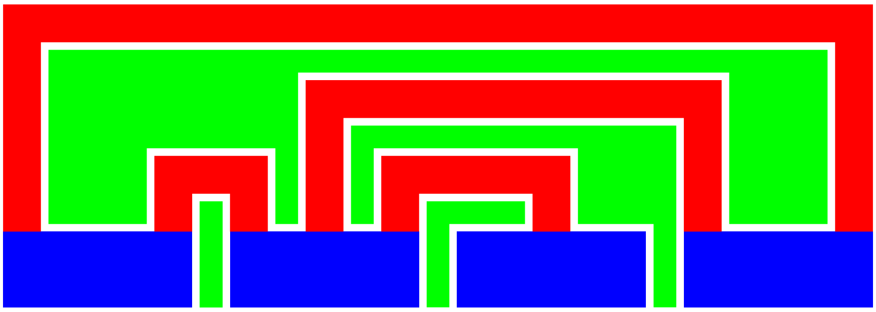
For each of the problems Independent Set, 3-Colorability, Dominating Set, Feedback Vertex Set, and Hamiltonian Cycle we will give a many-to-one polynomial-time reduction from the restriction of the problem to planar graphs with maximum degree three or four to the restriction of the problem to proximity graphs. Each reduction will start by computing a two-page book embedding in polynomial time. This is possible by Theorem 2.13.

The design of reductions for proximity graphs involves dealing with combinatorial and geometric issues. From a combinatorial perspective, the graph generated by the reduction must preserve the combinatorial properties that are relevant to the computational problem in question. When dealing with Independent Set, for example, the generated graph's independence number must depend on the original graph's independence number in a predictable way. Geometrically, we must insure that the generated graphs are, in fact, RNGs, RCGs, or Gabriel graphs. As we describe the graph that the reduction generates, we will also describe an embedding that induces this graph.

The reductions will employ three types of gadgets. There will be *vertex gadgets* representing the vertices in the input graph. These vertex gadgets will depend on the E_1 and E_2-heights and E_1 and E_2-degrees of the vertex they represent. Secondly, there will be *edge gadgets* representing the edges in the input graph. The edge gadgets will be paths or path-like in that they can elongated in some fashion. The length of these paths or like-gadgets will depend on the length and, in some cases, height of the edges they represent. The embedding of the edge gadget starts at the vertex gadget representing



(a) Addition of vertex gadgets and edge gadgets: The dark blue vertex gadgets represent vertices in the original graph. The red edge gadgets represent edges in the original graph. Their length and shape depends on the height and length of the edges they represent.



(b) Addition of filler gadgets: Filler gadgets must be added throughout the areas marked in light green.

Figure 3.2: Illustration of the general idea behind our reductions: The input is the graph G in Figure 3.1. Vertex gadgets are in blue, edge gadgets in red, and areas with filler gadgets in green.

one of the edge's endpoints. It begins with a vertical portion whose length depends on the height of the edge. The vertical portion is followed by a horizontal portion. The height at which the horizontal portion is embedded depends on the height of the edge. The length of this portion depends on the length of the edge. Finally, there is a second vertical portion that mirrors the first and goes from the end of the horizontal portion to the vertex gadget representing the second endpoint of the edge.

The general idea behind our reductions in illustrated in Figure 3.2. The addition of vertex gadgets and edge gadgets is shown in Figure 3.2a. Combinatorially, we could end the reduction at this point, but geometric issues remain. The embedding generally does not induce the graph as its proximity graph yet, because there are no blockers between the vertices in the various gadgets. The proximity graphs induced by the embedding at this point would contain additional edges which could affect the graph's combinatorial properties in unintended ways. This is why we introduce a third type of gadget, the *filler gadgets*. Filler gadgets will be added throughout the areas between the various gadgets. They must block edges between other gadgets while at the same time affecting the relevant combinatorial properties in a predictable way. In Figure 3.2b, the areas which have to be covered by filler gadgets are marked in green.

In the following chapters, we will prove the NP-hardness of the aforementioned problems.

4 Dominating Set

We begin by considering the Dominating Set problem and proving that it is NP-hard when restricted to RNGs, RCGs, and Gabriel graphs. Recall that a set of vertices $\mathcal{D} \subseteq V$ in a graph $G = (V, E)$ is a dominating set if every $v \in V \setminus \mathcal{D}$ has a neighbor in $\mathcal{D}$ and that $\gamma(G)$ denotes the size of the smallest dominating set in G. This gives rise to the Dominating Set problem:

Dominating Set
Input: A graph $G = (V, E)$ and a nonnegative integer k.
Question: Does G contain a dominating set of size at most k?

According to Garey and Johnson [GJ79] and Johnson [Joh84], Dominating Set remains NP-complete when restricted to 3-regular planar graphs, although to our knowledge no proof has been published. We will give a brief proof sketch at the end of this chapter for the following slightly weaker claim:

Theorem 4.1 ([GJ79; Joh84])**.** Dominating Set *on planar graphs with maximum degree at most* 3 *is NP-hard.*

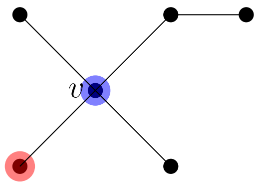
4.1 Definitions and intermediate results

We will start by proving several lemmas that we will use to show that our reduction is correct. First, we will show that, if we replace any vertex by a path consisting of four vertices while connecting the neighbors of the replaced vertex to the endpoints of the path, then we increase its domination number by exactly one. An example of such a transformation is pictured in Figure 4.1.

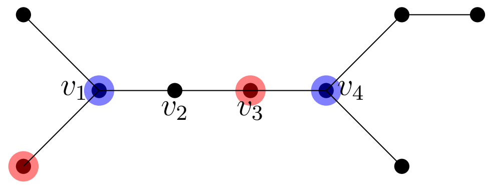
Lemma 4.2. *Let G be a graph. Suppose that G' is obtained from G by replacing the vertex v with a path consisting of the new vertices v_1, v_2, v_3, v_4 in that order and connecting every neighbor of v to either v_1 or v_4. Then, $\gamma(G') = \gamma(G) + 1$.*

Proof. Suppose that $\mathcal{D}$ is a dominating set in G. If $v \in \mathcal{D}$ (the case highlighted in blue in Figure 4.1), then $\mathcal{D}' := (\mathcal{D} \setminus \{v\}) \cup \{v_1, v_4\}$ is a dominating set of size $|\mathcal{D}| + 1$ in G'. If $v \notin \mathcal{D}$ (the case highlighted in red in Figure 4.1), then this vertex has a neighbor $u \in \mathcal{D}$. In G', u must be adjacent to either v_1 or v_4. In the first case, $\mathcal{D}' := \mathcal{D} \cup \{v_3\}$ is a dominating set of size $|\mathcal{D}| + 1$ in G'. In the second case, $\mathcal{D}' := \mathcal{D} \cup \{v_2\}$ is. Hence, $\gamma(G') \leq \gamma(G) + 1$.

Now suppose that $\mathcal{D}'$ is a dominating set in G'. First, assume that $\mathcal{D}'$ contains v_1. Then, $\mathcal{D}'$ must also contain at least one of v_2, v_3, or v_4 in order to dominate v_3.



(a) A graph G.



(b) The graph G' obtained from G by applying the the transformation.

Figure 4.1: An example for the transformation described in Lemma 4.2: The vertex v is replaced by the path consisting of v_1, v_2, v_3, and v_4.

Then, $\mathcal{D} := (\mathcal{D}' \setminus \{v_1, v_2, v_3, v_4\}) \cup \{v\}$ is a dominating set in G of size at most $|\mathcal{D}'| - 1$. If we assume that $\mathcal{D}'$ contains v_4, then an analogous argument holds. So, assume that $\mathcal{D}'$ contains neither v_1 nor v_4. It must then contain one of the following:

- v_2 and a vertex $u \neq v_3$ that is adjacent to v_4,
- v_3 and a vertex $u \neq v_2$ that is adjacent to v_1, or
- both v_2 and v_3.

In the first case, set $\mathcal{D} := \mathcal{D}' \setminus \{v_2\}$. In the second case, let $\mathcal{D} := \mathcal{D}' \setminus \{v_3\}$. In the third case, choose $\mathcal{D} := (\mathcal{D}' \setminus \{v_2, v_3\}) \cup \{v\}$. In each case, $\mathcal{D}$ is a dominating set in G of size at most $|\mathcal{D}'| - 1$. Hence, $\gamma(G') \geq \gamma(G) + 1$. □

The preceding lemma can be generalized to a path consisting of $3k + 1$ vertices for any k.

Lemma 4.3. *Suppose the graph G' is obtained from G by replacing the vertex v with a path consisting of $3k+1$ new vertices $v_1, v_2, \ldots, v_{3k+1}$ in that order and connecting every neighbor of v to v_{3i+1} for some i. Then, $\gamma(G') = \gamma(G) + k$.*

Proof. By induction on k:

If $k = 0$, then G and G' are isomorphic.

Suppose the claim holds for k and let G' be the graph obtained from G by replacing v with a path consisting of $3k+4$ vertices. Let G'' be obtained from G by replacing v with the path v_1, v_2, v_3, v_4 and connecting every u to v_1 if u is adjacent to v_1 in G' and to v_4 if u is adjacent to a v_i, $i \geq 4$, in G'. Then, G' may be obtained from G'' by replacing v_4 with a path containing $3k+1$ vertices. By Lemma 4.2, $\gamma(G'') = \gamma(G) + 1$. By induction hypothesis, $\gamma(G') = \gamma(G'') + k$, implying that $\gamma(G') = \gamma(G) + k + 1$. □

This directly implies:

Corollary 4.4. *If G' is obtained from the graph G by subdividing an edge $3k$ times, then $\gamma(G') = \gamma(G) + k$.*

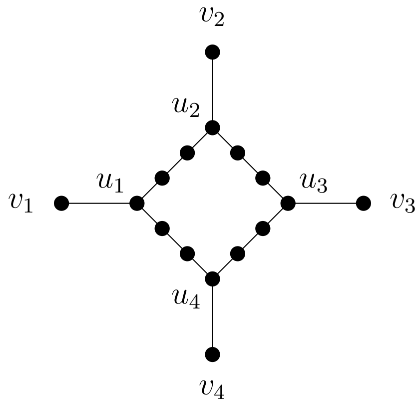


Figure 4.2: The graph G' in Lemma 4.5.

Proof. Let $\{u, v\}$ be the edge in G that is subdivided $3k$ times to obtain G'. This subdivision is tantamount to replacing u with a path of length $3k+1$, while connecting v to the last vertex on this path and all of u's other neighbors to the first vertex. By Lemma 4.3, this implies that $\gamma(G') = \gamma(G) + k$. □

Lemma 4.3 and Corollary 4.4 are the basis for the vertex and edge gadgets we will use. Next, we will consider how the addition of a certain subgraph consisting mainly of a cycle of length twelve affects the domination number. This graph will be used as the filler gadget.

Lemma 4.5. *Let G be a graph. Consider the graph G' depicted in Figure 4.2 and the vertices marked v_1, v_2, v_3, and v_4. Suppose that G' is an induced subgraph of G, that in G the vertices v_1, v_2, v_3, v_4 each have at most one neighbor in $G - G'$, and that no other vertex in G' has a neighbor in $G - G'$. Then, $\gamma(G) = \gamma(G - G') + 4$.*

The same statement is also true if any subset of $\{v_1, v_2, v_3, v_4\}$ is removed from G'.

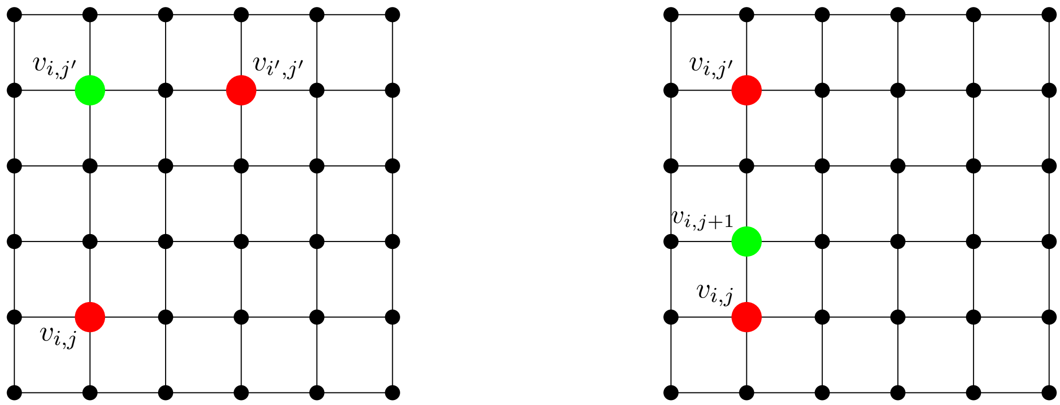
Proof. Suppose that $\mathcal{D}$ is a dominating set in $G - G'$. Then, $\mathcal{D}' := \mathcal{D} \cup \{u_1, u_2, u_3, u_4\}$ (see Figure 4.2) is a dominating set of size $|\mathcal{D}| + 4$ in G. Hence, $\gamma(G) \leq \gamma(G - G') + 4$.

Suppose now that $\mathcal{D}'$ is a dominating set in G. Then, $\mathcal{D}'$ must contain at least four vertices w_1, w_2, w_3, w_4 on the cycle in G'. We obtain $\mathcal{D}$ by removing all vertices in G' from $\mathcal{D}'$. If $v_i \in \mathcal{D}'$ and v_i has a neighbor v_i' in $G - G'$, then we add v_i' to $\mathcal{D}$. Then, $\mathcal{D}$ is a dominating set of size at most $|\mathcal{D}'| - 4$ in $G - G'$. This implies that $\gamma(G) \geq \gamma(G - G') + 4$.

The previous argument also applies if any subset of $\{v_1, v_2, v_3, v_4\}$ is removed from G'. □

Lemma 4.5 would also be correct if we replaced the 12-cycle in G' with any cycle whose length is divisible by 3 and added outer vertices v_i to every third vertex in the cycle. We will use 12-cycles in our reduction because this allows us to draw the vertices on the boundary of a square with every third vertex lying on a corner.

The previous lemmas will be used to prove the correctness of our reduction. We will now present several results that we will use to prove that the graph produced by the



(a) The vertex $v_{i,j'}$ is a Gabriel-blocker for $v_{i,j}$ and $v_{i',j'}$, because $\measuredangle(i,j)(i,j')(i',j') = 90°$.

(b) The vertex $v_{i,j+1}$ is a Gabriel-blocker for $v_{i,j}$ and $v_{i,j'}$, because it is on the line segment between them.

Figure 4.3: Each pair of non-adjacent vertices in G_{n_1,n_2} has a Gabriel-blocker. The red vertices are not adjacent and the green vertices are their blockers.

reduction is a relative neighborhood graph, a relatively closest graph, and a Gabriel graph.

Definition 4.6. The $(n_1 \times n_2)$-*grid point set* is $P_{n_1,n_2} := \{1, \dots, n_1\} \times \{1, \dots, n_2\}$.

Recall that in Section 2.1, we defined the $(n_1 \times n_2)$-grid graph as $G_{n_1,n_2} := (V_{n_1,n_2}, E_{n_1,n_2})$ with:

$$\begin{aligned} V_{n_1,n_2} &:= \{v_{i,j} \mid 1 \le i \le n_1, 1 \le 2 \le n_2\}, \\ E_{n_1,n_2} &:= \{\{v_{i,j}, v_{i+1,j}\} \mid 1 \le i \le n_1 - 1, 1 \le j \le n_2\} \\ &\quad \cup \{\{v_{i,j}, v_{i,j+1}\} \mid 1 \le i \le n_1, 1 \le j \le n_2 - 1\}. \end{aligned}$$

This is the graph induced by the P_{n_1,n_2} as its RNG, RCG, and Gabriel graph, as we will show.

Lemma 4.7. *The relative neighborhood graph, relatively closest graph, and Gabriel graph induced by the $(n_1 \times n_2)$-grid point set is the $(n_1 \times n_2)$-grid graph.*

Proof. Since $\mathrm{RCG}(P) \subseteq \mathrm{RNG}(P) \subseteq \mathrm{GAB}(P)$ for any $P \subseteq \mathbb{R}^2$, it suffices to prove that $\mathrm{GAB}(P_{n_1,n_2}) \subseteq G_{n_1,n_2} \subseteq \mathrm{RCG}(P_{n_1,n_2})$. Let $\mathrm{emb}\colon V_{n_1,n_2} \to \mathbb{R}^2$ with $v_{i,j} \mapsto (i,j)$. Clearly, $\mathrm{emb}(V_{n_1,n_2}) = P_{n_1,n_2}$.

We begin by showing that $\mathrm{GAB}(P_{n_1,n_2}) \subseteq G_{n_1,n_2}$. To this end, we must prove that every pair of non-adjacent vertices in G_{n_1,n_2} has a Gabriel-blocker. Let $v_{i,j}$ and $v_{i',j'}$ be distinct and non-adjacent vertices.

First, assume that $i \neq i'$ and $j \neq j'$. Then, $v_{i,j'}$ is distinct from both of these vertices and $\measuredangle\,\mathrm{emb}(v_{i,j})\,\mathrm{emb}(v_{i,j'})\,\mathrm{emb}(v_{i',j'}) = \measuredangle(i,j)(i,j')(i',j') = 90°$. As a result, $v_{i,j'}$ is a Gabriel-blocker for these two vertices. See Figure 4.3a.

Now assume that $i = i'$. Since $v_{i,j}$ and $v_{i',j'}$ are not adjacent, it follows that $|j-j'| > 1$. Without loss of generality, we may assume that $j' > j+1$. Then, $v_{i,j+1}$ is on the line segment between the two vertices and therefore a Gabriel-blocker. See Figure 4.3b. An analogous argument applies if we assume that $j = j'$.

We must also show that $G_{n_1,n_2} \subseteq \mathrm{RCG}(P_{n_1,n_2})$, in other words that there is no RCG-blocker for any edge in G_{n_1,n_2}. This follows from the fact that the embeddings of adjacent vertices have a distance of 1, but no third vertex has a distance of at most 1 to such vertices. □

Next, we will add points to an $(n_1 \times n_2)$-grid point set. We will add them at the position halfway between two existing points. Each added point will have one integer and one half-integer coordinate. We will prove that, if we add enough of these points, then the graph induced by the point set as its RNG, RCG, and Gabriel graph is a subdivision of the $(n_1 \times n_2)$-grid graph.

Definition 4.8. Let $n_1, n_2 \in \mathbb{N}$.

A *subdivision* (S_1, S_2) is a pair of sets $S_1 \subseteq \{1, \dots n_1 - 1\} \times \{1, \dots n_2\}$ and $S_2 \subseteq \{1, \dots, n_1\} \times \{1, \dots, n_2 - 1\}$. The subdivision (S_1, S_2) is *valid* if the following two conditions hold for all $1 \leq i \leq n_1$ and $1 \leq j \leq n_2$:

- if $(i, j) \notin S_1$, then $(i+1, j) \in S_1$ and $(i-1, j) \in S_1$ and
- if $(i, j) \notin S_2$, then $(i, j+1) \in S_2$ and $(i, j-1) \in S_2$.

The (S_1, S_2)-*subdivision* of the graph G_{n_1,n_2} is the graph obtained from G_{n_1,n_2} by subdividing each of the following edges exactly once:

- any edge $\{v_{i,j}, v_{i+1,j}\}$ if $(i, j) \in S_1$ and
- any edge $\{v_{i,j}, v_{i,j+1}\}$ if $(i, j) \in S_2$.

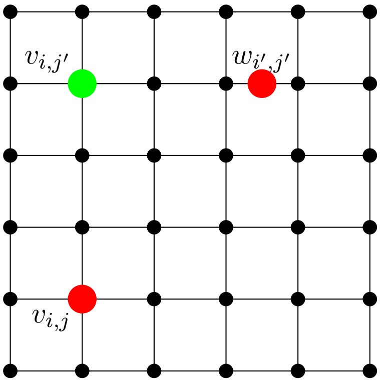
We will denote this graph as $G_{n_1,n_2}[S_1, S_2] = (V_{n_1,n_2}[S_1, S_2], E_{n_1,n_2}[S_1, S_2])$. We will use $w_{i,j}$ to refer to the vertex introduced by subdividing $\{v_{i,j}, v_{i+1,j}\}$ and $\tilde{w}_{i,j}$ for the vertex introduced by subdividing $\{v_{i,j}, v_{i,j+1}\}$. The vertices in V_{n_1,n_2} will be called *integer vertices*, while $w_{i,j}$ and $\tilde{w}_{i,j}$ will be called *half-integer vertices*.

The (S_1, S_2)-*subdivision* of the point set P_{n_1,n_2} is the point set

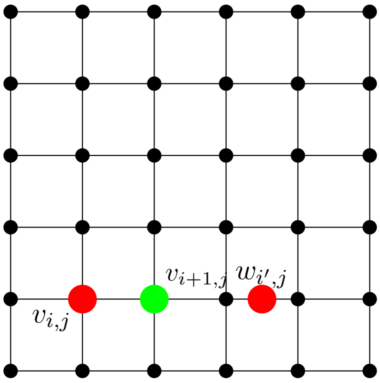
$$P_{n_1,n_2}[S_1, S_2] := P_{n_1,n_2} \cup \{(i + \frac{1}{2}, j) \mid (i, j) \in S_1\} \cup \{(i, j + \frac{1}{2}) \mid (i, j) \in S_2\}.$$

The point $(i, j) \in P_{n_1,n_2}[S_1, S_2]$ will be called an *integer point* if $i, j \in \mathbb{N}$, while $(i + \frac{1}{2}, j) \in P_{n_1,n_2}[S_1, S_2]$ and $(i, j + \frac{1}{2}) \in P_{n_1,n_2}[S_1, S_2]$ with $i, j \in \mathbb{N}$ are *half-integer points*.

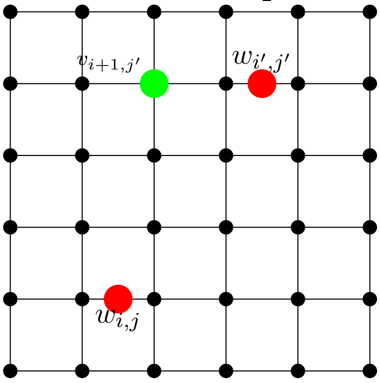
Intuitively, S_1 tells us which horizontal edges are to be subdivided and S_2 determines which vertical edges have to be subdivided. Note that every point has at least one integer coordinate. These definitions are useful due to the following:



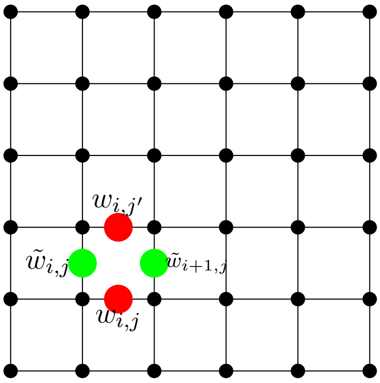
(a) The vertex $v_{i,j'}$ is a Gabriel-blocker for $v_{i,j}$ and $w_{i',j'}$, because $\measuredangle(i,j)(i,j')(i'+\frac{1}{2},j')=90°$.



(b) The vertex $v_{i+1,j}$ is a Gabriel-blocker for $v_{i,j}$ and $w_{i',j}$, because it is on the line segment between them.



(c) The vertex $v_{i+1,j'}$ is a Gabriel-blocker for $w_{i,j}$ and $w_{i',j'}$, because $\measuredangle(i,j)(i,j')(i'+\frac{1}{2},j')>90°$.



(d) The graph contains $\tilde{w}_{i,j}$ or $\tilde{w}_{i+1,j}$. One of these vertices is a Gabriel-blocker for $w_{i,j}$ and $w_{i,j'}$, because $\measuredangle(i+\frac{1}{2},j)(i,j'+\frac{1}{2})(i'+\frac{1}{2},j')\geq 90°$ and $\measuredangle(i+\frac{1}{2},j)(i+1,j'+\frac{1}{2})(i'+\frac{1}{2},j')\geq 90°$.

Figure 4.4: Each pair of non-adjacent vertices in $G_{n_1,n_2}[S_1,S_2]$ has a Gabriel-blocker. The red vertices are not adjacent and the green vertices are their blockers.

Lemma 4.9. *If the subdivision (S_1,S_2) is valid, then the (S_1,S_2)-subdivision of the point set P_{n_1,n_2} induces the (S_1,S_2)-subdivision of the graph G_{n_1,n_2} as its relative neighborhood graph, relatively closest graph, and Gabriel graph.*

Proof. Again, it suffices to prove that $\mathrm{GAB}(P_{n_1,n_2}[S_1,S_2]) \subseteq G_{n_1,n_2}[S_1,S_2] \subseteq \mathrm{RCG}(P_{n_1,n_2}[S_1,S_2])$. Let $\mathrm{emb}\colon V_{n_1,n_2}[S_1,S_2]\to\mathbb{R}^2$ with $v_{i,j}\mapsto(i,j)$, $w_{i,j}\mapsto(i+\frac{1}{2},j)$, and $\tilde{w}_{i,j}\mapsto(i,j+\frac{1}{2})$. Of course, $\mathrm{emb}(V_{n_1,n_2}[S_1,S_2])=P_{n_1,n_2}[S_1,S_2]$.

We begin by proving that $\mathrm{GAB}(P_{n_1,n_2}[S_1,S_2])\subseteq G_{n_1,n_2}[S_1,S_2]$, that is, that all pairs of non-adjacent vertices in $G_{n_1,n_2}[S_1,S_2]$ have Gabriel-blockers. Mostly, this follows along the same lines as the proof of Lemma 4.7. The only pairs of non-adjacent vertices not yet considered in that proof are pairs of integer vertices that are adjacent in G_{n_1,n_2}, but not in $G_{n_1,n_2}[S_1,S_2]$, as well as pairs of non-adjacent vertices that involve at least one

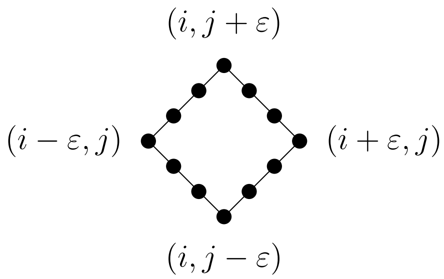


Figure 4.5: Embedding of a 12-cycle: The unlabeled vertices are spread equidistantly between the vertices whose positions are given.

half-integer vertex.

The only edges between integer vertices that are are present in G_{n_1,n_2}, but not in $G_{n_1,n_2}[S_1, S_2]$, are those that were subdivided. Clearly, they have the vertices introduced in the subdivision as Gabriel-blockers as these vertices are on the line segments between them.

We turn our attention to pairs of non-adjacent vertices that involve at least one half-integer vertex. This requires us to distinguish several cases.

First, assume that just one of the two vertices is a half-integer vertex. Suppose that $v_{i,j}$ and $w_{i',j'}$ are not adjacent. Then, $\text{emb}(v_{i,j}) = (i, j)$ and $\text{emb}(w_{i',j'}) = (i + \frac{1}{2}, j)$. If $j \neq j'$, then the vertex $v_{i,j'}$ is a Gabriel-blocker, since $\measuredangle\, \text{emb}(v_{i,j})\, \text{emb}(v_{i,j'})\, \text{emb}(w_{i',j'}) = \measuredangle(i, j)(i, j')(i' + \frac{1}{2}, j') = 90°$. See Figure 4.4a. If $j = j'$, there is a third vertex on the line segment between them. See Figure 4.4b. An analogous argument applies if we consider two non-adjacent vertices $v_{i,j}$ and $\tilde{w}_{i',j'}$.

Now consider two non-adjacent half-integer vertices. First, suppose one of them is the result of subdividing a horizontal edge and the other the product of subdividing a vertical edge, say the vertices $w_{i,j}$ and $\tilde{w}_{i',j'}$. Then, $v_{i',j}$ is a Gabriel-blocker by the same reasoning as above. Next, we deal with two half-integer vertices that were both created by subdividing a horizontal edge (for two products of a vertical edge the same argument applies). Say, the two vertices are $w_{i,j}$ and $w_{i',j'}$. If $i \neq i'$ and $j \neq j'$, we assume without loss of generality that $i < i'$. Then, $v_{i+1,j'}$ is a Gabriel-blocker. See Figure 4.4c. If $i \neq i'$ and $j = j'$, then there is again a vertex on the line segment between the two vertices. The final case is when $i = i'$ and $j \neq j'$. Without loss of generality, $j < j'$. Because the subdivision is valid, $\tilde{w}_{i,j} \in V_{n_1,n_2}[S_1, S_2]$ or $\tilde{w}_{i+1,j} \in V_{n_1,n_2}[S_1, S_2]$. Whichever of these two vertices is present in the graph is a Gabriel blocker. See Figure 4.4d.

We must also show that $G_{n_1,n_2}[S_1, S_2] \subseteq \text{RCG}(P_{n_1,n_2}[S_1, S_2])$, in other words, that no adjacent points have an RCG-blocker. Given what we proved in Lemma 4.7, we only have to show that half-integer vertices are not blockers for any edges and that the edges incident to half-integer vertices are not blocked. This follows by the same argument used in the proof of that lemma. □

We will now further modify subdivided point sets and investigate the effect this has on the proximity graphs they induce. In order to apply Lemma 4.5, we will replace

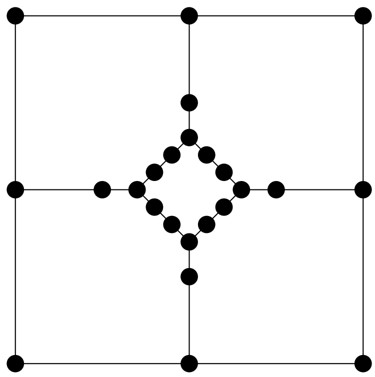


Figure 4.6: Addition of a 12-cycle to a subdivided grid

some integer vertices with cycles of length twelve. In the following, fix $0 < \varepsilon < \frac{1}{4}$. The embedding of C_{12} we will use is pictured in Figure 4.5. We will refer to this point set as $P^{12}_{i,j}$. The following observation follows from a similar argument to the one we used to prove Lemma 4.9.

Observation 4.10. *The point set $P^{12}_{i,j}$ induces C_{12} as its RNG, RCG, and Gabriel graph.*

Subdivided grids which have had some of their integer points replaced with 12-cycles will be referred to as modified subdivided grids.

Definition 4.11. Let $n_1, n_2 \in \mathbb{N}$.

A *modified subdivision* (S_1, S_2, S_3) consists of sets $S_1 \subseteq \{1, \dots n_1 - 1\} \times \{1, \dots n_2\}$, $S_2 \subseteq \{1, \dots, n_1\} \times \{1, \dots, n_2 - 1\}$, and $S_3 \subseteq \{1, \dots, n_1\} \times \{1, \dots, n_2\}$. It is *valid* if (S_1, S_2) is a valid subdivision and $(i, j), (i - 1, j) \in S_1$ and $(i, j), (i, j - 1) \in S_2$ for all $(i, j) \in S_3$. If $i = 1$, the condition $(i - 1, j) \in S_1$ is dropped and, if $j = 1$, the condition $(i, j - 1) \in S_2$ is dropped.

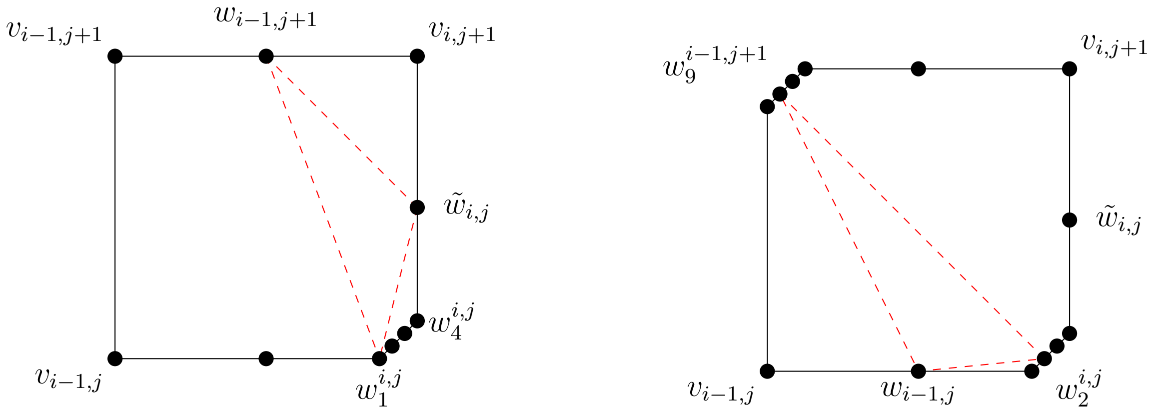
If (S_1, S_2, S_3) is a valid modified subdivision, the (S_1, S_2, S_3)-*modified subdivision* of the graph G_{n_1,n_2} is the graph obtained from $G_{n_1,n_2}[S_1, S_2]$ by removing every $v_{i,j}$ with $(i, j) \in S_3$ and adding a C_{12} in the following manner:

- add the vertices $w^{i,j}_1, \dots, w^{i,j}_{12}$ with the edges $\{w^{i,j}_k, w^{i,j}_{k+1}\}$ for $k = 1, \dots, 11$ and $\{w^{i,j}_{12}, w^{i,j}_1\}$,
- connect the C_{12} to surrounding half-integer vertices by adding the edges $\{w^{i,j}_1, w_{i-1,j}\}, \{w^{i,j}_4, \tilde{w}_{i,j}\}, \{w^{i,j}_7, w_{i,j}\}, \{w^{i,j}_{11}, \tilde{w}_{i,j-1}\}$.

Note that each of those half-integer vertices exist because (S_1, S_2, S_3) is valid. An example for the addition of a C_{12} is pictured in Figure 4.6.

The (S_1, S_2, S_3) of the point set P_{n_1,n_2} is the point set

$$P_{n_1,n_2}[S_1, S_2, S_3] \coloneqq (P_{n_1,n_2}[S_1, S_2] \setminus \{(i, j) \mid (i, j) \in S_3\}) \cup \bigcup_{(i,j) \in S_3} P^{12}_{i,j}.$$



(a) The vertex $\tilde{w}_{i,j}$ is a blocker for $\{w_{i-1,j+1}, w_1^{i,j}\}$.

(b) The vertex $\tilde{w}_{i,j}$ is a blocker for $\{w_{i-1,j+1}, w_1^{i,j}\}$.

Figure 4.7: Two subgraphs of $G_{n_1,n_2}[S_1, S_2, S_3]$ along with the Gabriel-blockers for two pairs of non-adjacent vertices.

The graph produced by the reduction that we will use to prove the NP-hardness of Dominating Set will be a modified subdivided grid graph. We will use the following lemma to prove this graph is an RNG, an RCG, and a Gabriel graph.

Lemma 4.12. *If (S_1, S_2, S_3) is valid, then the (S_1, S_2, S_3)-modified subdivision of the point set P_{n_1,n_2} induces the (S_1, S_2, S_3)-modified subdivision of the graph G_{n_1,n_2} as its relative neighborhood graph, relatively closest graph, and Gabriel graph.*

Proof sketch. The proof mostly follows along the same lines as the proof of Lemma 4.9, so we will omit many of the details. For instance, if $(i, j) \in S_3$ and (i, j) is a blocker for a pair of non-adjacent vertices in $G_{n_1,n_2}[S_1, S_2]$, then we can choose a vertex in the C_{12} which replaced (i, j) and show that it is a blocker for the same pair of vertices in $G_{n_1,n_2}[S_1, S_2, S_3]$.

In most instances, it is not hard to see that non-adjacent pairs of vertices have Gabriel-blockers and that adjacent pairs do not have RCG-blockers. We will focus on a few cases where this may not be immediately clear. Consider the excerpt of $G_{n_1,n_2}[S_1, S_2, S_3]$ pictured in Figure 4.7a. It occurs when $(i, j) \in S_3$, but $(i-1, j), (i, j-1), (i-1, j-1) \notin S_3$. The vertex $\tilde{w}_{i,j}$ is a Gabriel-blocker for $\{w_{i-1,j+1}, w_1^{i,j}\}$, since $\measuredangle \operatorname{emb}(w_{i-1,j+1}) \operatorname{emb}(\tilde{w}_{i,j}) \operatorname{emb}(w_1^{i,j}) > 90°$. Also, consider the excerpt depicted in Figure 4.7b, which occurs when $(i, j), (i+1, j+1) \in S_3$, but $(i+1, j), (i, j+1) \notin S_3$. The vertex $w_{i-1,j}$ is a Gabriel-blocker for $w_2^{i,j}$ and $w_9^{i-1,j+1}$, since $\operatorname{emb}(w_2^{i,j}) \operatorname{emb}(w_{i-1,j}) \operatorname{emb}(w_9^{i-1,j+1}) > 90°$.

□

4.2 Reduction

We will now give a polynomial-time many-to-one reduction from Dominating Set on planar graphs with maximum degree at most 3 to its restriction to relative neighborhood graphs, Gabriel graphs, and relatively closest graphs.

Let $G = (V, E)$ be a planar graph with maximum degree at most 3 and k a nonnegative integer. We compute a two-page book embedding of G in polynomial time and assume that $v_1, \ldots, v_n$ are the vertices of G in the order in which they appear on the spine of the book embedding.

Construction

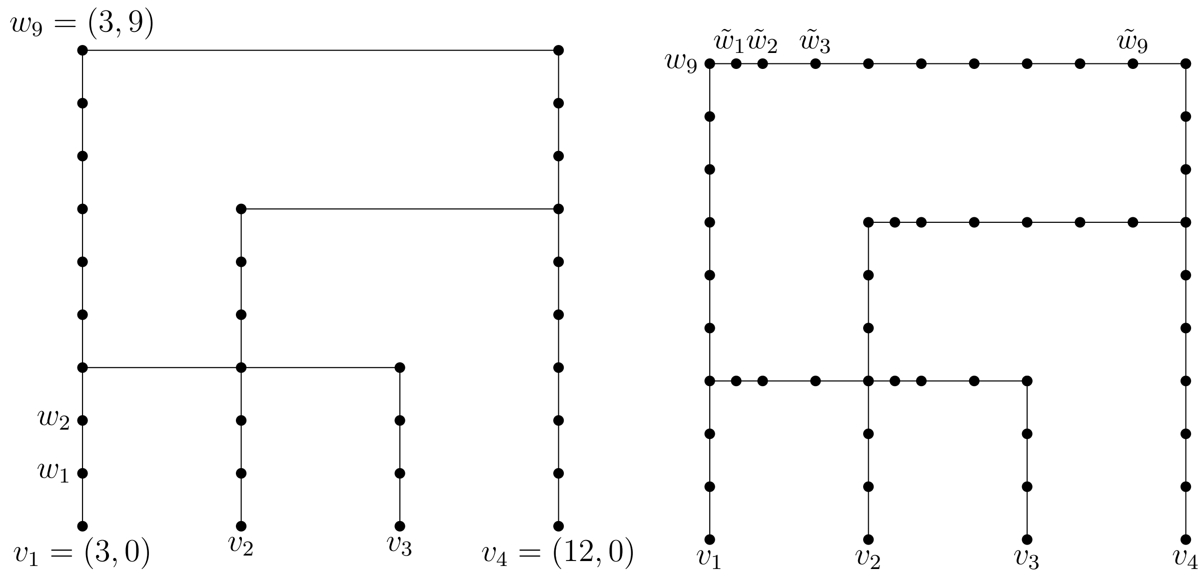
We will now construct a graph $G' = (V', E')$, which is an RNG, an RCG, and a Gabriel graph, and an integer k' such that G has a dominating set of size k if and only if G' has a dominating set of size k'. At the same time, we give an embedding emb: $V' \to \mathbb{R}$, which we will subsequently use to prove that G' is, indeed, an RCG, an RNG, and a Gabriel graph by showing that $\mathrm{RCG}(\mathrm{emb}(V')) = \mathrm{RNG}(\mathrm{emb}(V')) = \mathrm{GAB}(\mathrm{emb}(V')) = G'$. The reduction will follow the pattern described in Chapter 3 and use the terminology defined there. The vertex and edge gadgets will be paths while the filler gadgets will be copies of the graph pictured in Figure 4.2. In order to be able to describe edges between gadgets as succinctly as possible, we will refer to certain vertices or groups of vertices as (i, j)-corners for integers i and j. Every corner has up to four outlets: a top, a left, a right, and a bottom outlet. In Figures 4.8 and 4.9, we will illustrate each step of the construction using the same example graph, which is pictured in Figure 3.1.

Step 1: We start with $G' := G$, $k' := k$, and $\mathrm{emb}(v_i) := (3i, 0)$. We will call v_i the $(3i, 0)$-corner. It is simultaneously the top, left, right, and bottom outlet of this corner.

Step 2: For every $i = 1, \ldots, n$, add a path consisting of $3h_1(v_i)$ new vertices $w_1, \ldots, w_{3h_1(v_i)}$ starting in v_i. This is v_i's *up-path*. We let $\mathrm{emb}(w_j) := (3i, j)$. Increase k' by $h_1(v_i)$. We will call the j-th vertex on the up-path the $(3i, j)$-corner. It is also at the same time the top, left, right, and bottom outlet of the corner. Replace every edge $e = \{v_i, v_j\}$ with an edge from the $3h(e)$-th vertex in v_i's up-path to the $3h(e)$-th vertex in v_j's up-path. This step is illustrated in Figure 4.8a.

Step 3: For every edge $e = \{v_i, v_j\} \in E_1$ with $i < j$, replace the edge introduced in the previous step in place of e with a path consisting of $3\ell(e)$ new vertices $\tilde{w}_1, \ldots, \tilde{w}_{3\ell(e)}$ from the $3h(e)$-th vertex in v_i's up-path to the $3h(e)$-th vertex in v_j's up-path. This is e's *edge path*. We set $\mathrm{emb}(\tilde{w}_1) := (3i + \frac{1}{2}, 3h(i))$ and $\mathrm{emb}(\tilde{w}_r) := (3i + r - 1, 3h(i))$ for all $r = 2, \ldots, 3\ell(2)$. Increase k' by $\ell(e)$. For $r = 2, \ldots, 3\ell(e)$, we will call $\tilde{w}_r$ the $(3i + r - 1, 3h(e))$-corner (the first vertex is not a corner). Once again, the corner vertex is the top, left, right, and bottom outlet of the corner. This step is illustrated in Figure 4.8b.

Step 4: For every $i = 3, 4, \ldots, 3n$ and $j = 1, \ldots, 3h_1(G)$, do the following: if



(a) G' after **Step 2**: Each vertex v_i is replaced by a path consisting of $3h_1(v_i)+1$ vertices. As an example, some vertices in v_1's up-path are labeled. Since, as noted in Chapter 3, the highest edge incident to v_1 has height 3, the E_1-height of v_1 is $h_1(v_1) = 3$. Hence, the up-path consists of 9 vertices.

(b) G' after **Step 3**: Each edge e is replaced by a path consisting of $3\ell(e)$ vertices. The edge $\{v_1, v_4\}$ has height 3, so its edge path is connected to the 9th vertex in v_1's up-path. It has length 3, so the path consists of 9 vertices. Some of them are labeled.

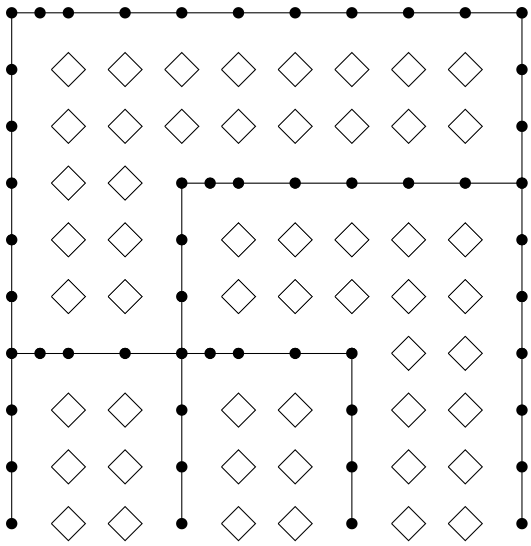
Figure 4.8: Addition of vertex and edge gadgets in **Steps 2** and **3**. We omit the distinction between a vertex and the position it is embedded at.

- i is not divisible by 3 or $3h_1(v_i) < j$ (in other words (i,j) is not covered by an up-path) and
- j is not divisible by 3 or there is no edge $e = \{v_r, v_s\}$ with $r \leq i \leq s$ and $3h(e) = j$ (in other words (i,j) is not covered by edge path),

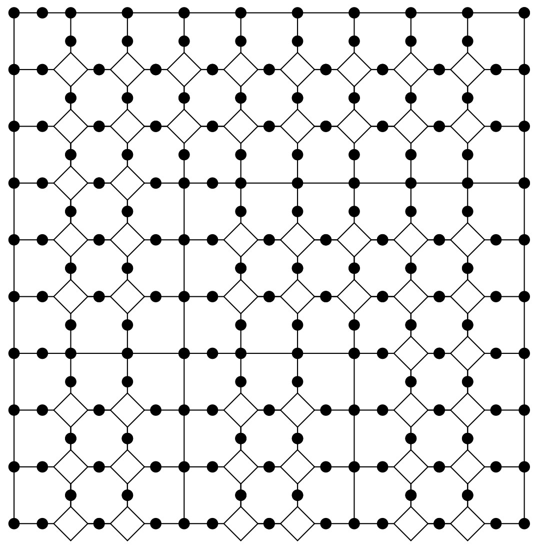
then add a cycle of length 12. Use the embedding pictured in Figure 4.5. Increase k' by 4. This cycle is the (i,j)-corner. The vertex on the top is the top outlet, the one on the left is the left outlet and so on. This step is illustrated in Figure 4.9a.

Step 5: Connect adjacent pairs of corners in the following manner: For every $3 \leq i \leq 3n-1$ add a vertex u and connect it to the right outlet of the (i,j)-corner and the left outlet of the $(i+1,j)$-corner. For every $0 \leq j \leq 3h_1(G)-1$ add a vertex u and connect it to the top outlet of the (i,j)-corner and the bottom outlet of the $(i,j+1)$-corner. In both cases, the added vertex is embedded at the midpoint of its two neighbors. This step is illustrated in Figure 4.9b.

We have only taken edges in E_1 into consideration. We must, therefore, repeat the process described in **Step 2** to **5** above for all edges in E_2, creating down-paths corresponding to the up-paths and proceeding accordingly.



(a) G' after **Step 4**: Each diamond represents a 12-cycle. Cycles are added for point (i, j) in the area in which the graph is embedded if it is not previously occupied by a vertex or edge gadget.



(b) G' after **Step 5**: The 12-cycles added in the previous step are connected to surrounding vertices via intermediary vertices, thereby creating numerous copies of the graph in Figure 4.2, some of them overlapping.

Figure 4.9: Addition of filler gadgets.

Let A denote the total number of vertices on the up-paths, down-paths, and edge paths and let B denote the number of C_{12}s added. In total, $k' = k + \frac{A}{3} + 4B$.

Running time

We must show that the reduction runs in polynomial time. The graph G' contains:

- $\mathcal{O}(n)$ copies of the vertices in G,
- $\mathcal{O}(n)$ up-paths and down-paths containing $\mathcal{O}(n)$ vertices each, for a total of $\mathcal{O}(n^2)$ vertices,
- $\mathcal{O}(n)$ edge paths containing $\mathcal{O}(n)$ vertices each, for a total of $\mathcal{O}(n^2)$ vertices, and
- $\mathcal{O}(n^2)$ C_{12}s containing a constant number of vertices each.

In total, G' contains $\mathcal{O}(n^2)$ vertices and can be computed in polynomial time by the process described above.

Correctness

We must show that $\gamma(G) \leq k$ if and only if $\gamma(G') \leq k'$. This is true after every step in the construction. For **Steps 2** and **3**, it is a consequence of Lemma 4.3 and Corollary 4.4.

These steps replace vertices with paths of length $3\ell + 1$ for some integer ℓ or subdivide existing edges 3ℓ times. For **Steps 4** and **5**, it follows from Lemma 4.5. Together, the steps add copies of the graph described in the lemma to the graph.

Graph induced by embedding

We will argue that the embedding emb given in the construction induces the graph described therein as its RNG, RCG, and Gabriel graph. This follows from Lemma 4.12, the fact that $\mathrm{emb}(V')$ is a modified subdivided grid point set, and the fact that G' is a modified subdivided grid graph. One can easily check that (S_1, S_2, S_3) is valid where S_1, S_2 represent the half-integer positions occupied by points in $\mathrm{emb}(V')$ and S_3 is the set of positions occupied by 12-cycles.

4.3 Conclusions

This concludes our reduction. Note that in the graphs generated by the reduction no vertex has a degree greater than four. It proves the following theorem:

Theorem 4.13. DOMINATING SET *is NP-hard when restricted to relative neighborhood graphs, Gabriel graphs, or relatively closest graphs each with maximum degree at most four.*

For graphs with maximum degree two, DOMINATING SET is easy to solve. It remains open whether this problem can be solved in polynomial time when restricted to proximity graphs with maximum degree three.

We will show that our reduction also implies that DOMINATING SET cannot be solved in time $2^{o(n^{\frac{1}{4}})}$ on each of our proximity graph classes (where n is the number of vertices in a graph), unless the exponential time hypothesis fails (on the ETH, see Section 2.5).

We must first establish an ETH-based lower bound of $2^{o(\sqrt{n})}$ for DOMINATING SET on planar graphs with maximum degree 3. As we mentioned at the beginning of this chapter, no proof has been published for the claim that this problem is NP-hard. We will briefly sketch a proof, which will also be the starting point for the ETH-based lower bound.

We will discuss the VERTEX COVER problem in Chapter 5. That chapter contains a definition of this problem and Theorem 5.8, which states that the planar restriction of VERTEX COVER cannot be solved in time $2^{o(\sqrt{n})}$ unless the ETH fails. Planar VERTEX COVER can be reduced to planar DOMINATING SET in the following manner. Given a graph $G = (V, E)$ and a nonnegative integer k, first delete all isolated vertices and then for every edge $\{u, v\} \in E$ add a vertex w and edges $\{u, w\}$ and $\{v, w\}$. Leave k unchanged. One can easily check that this reduction is correct and that the resulting graph is planar. Note that, if G contains n vertices and m edges, then the resulting graph contains $\mathcal{O}(m + n) = \mathcal{O}(n)$ vertices and $\mathcal{O}(m)$ edges.

Planar DOMINATING SET can then be reduced to the restriction of planar DOMINATING SET to planar graphs with maximum degree at most 3. Chen, Kanj, and Xia

[CKX09] show that if planar Dominating Set can be solved in time $2^{o(\sqrt{k})}p(n)$ for some polynomial p, then the problem's restriction to planar graphs with maximum degree at most 3 can be solved in time $2^{o(\sqrt{k})}q(n)$ where q is another polynomial. Their proof is based on what they call a vertex expansion operation, which may also be used for this reduction. It produces a graph with maximum degree three and with a number of vertices that is linear in the number of vertices of the original graph.

Taken together, these two reductions prove that Dominating Set restricted to planar graphs with maximum degree 3 is NP-hard. Additionally, combined with Theorem 5.8 they also prove the following:

Theorem 4.14. *Unless the ETH fails,* Dominating Set *restricted to planar graphs with maximum degree* 3 *cannot be solved in time* $2^{o(\sqrt{n})}$.

Our reduction from Dominating Set on planar graphs with maximum degree 3 to its restriction to our classes of proximity graphs maps instances with m edges and n vertices to graphs with $\mathcal{O}(n^2)$ vertices and edges. This along with Theorem 4.14 implies:

Corollary 4.15. *Unless the ETH fails,* Dominating Set *restricted to RNGs, RCGs, or Gabriel graphs cannot be solved in time* $2^{o(n^{\frac{1}{4}})}$.

5 Independent Set and Vertex Cover

In the following, we will show that the INDEPENDENT SET and VERTEX COVER problems are NP-hard, when restricted to relative neighborhood graphs, relatively closest graphs, or to Gabriel graphs. Recall that an independent set in a graph G is a set of vertices that are pairwise non-adjacent and that $\alpha(G)$ denotes G's independence number, the size of the largest independent set in G. The INDEPENDENT SET problem is:

INDEPENDENT SET
Input: A graph $G = (V, E)$ and a nonnegative integer k.
Question: Does G contain an independent set of size at least k?

Similarly, recall that a set $\mathcal{S} \subseteq V$ is a vertex cover if $G - \mathcal{S}$ does not contain any edges and that $\tau(G)$ is the size of the smallest vertex cover in G. The VERTEX COVER problem is:

VERTEX COVER
Input: A graph $G = (V, E)$ and a nonnegative integer k.
Question: Does G contain a vertex cover of size at most k?

The INDEPENDENT SET and VERTEX COVER problems are very closely related. This is because for any graph G containing n vertices, $n = \alpha(G) + \tau(G)$. As a result, they can each be easily reduced to one another by replacing k with $n - k$ where n is the number of vertices in a graph.

The polynomial-time many-to-one reduction, which we will use to prove that INDEPENDENT SET is NP-hard on the proximity graph classes we are concerned with, will be from INDEPENDENT SET restricted to planar graphs with maximum degree at most 3, which Garey and Johnson [GJ76] proved to be NP-complete:

Theorem 5.1 ([GJ76])**.** INDEPENDENT SET *on planar graphs with maximum degree at most* 3 *is NP-hard.*

5.1 Definitions and intermediate results

Before we begin with our reduction, we prove several lemmas concerning independent sets and define some terminology. The first lemma states that, if we partition a graph G into subgraphs, then the independence number of G is at most the sum of the independence numbers of the subgraphs.

Lemma 5.2. *If H is a subgraph of a graph G, then $\alpha(G) \leq \alpha(G - H) + \alpha(H)$.*

(a) G.

(b) G': The vertex v is replaced by the path consisting of v_1, v_2, v_3, v_4, and v_5.

Figure 5.1: The transformation described in Lemma 5.4. The circled vertices form a maximum independent set in each graph.

Proof. Let $G = (V, E)$ and $H = (V', E')$ and suppose that $\mathcal{I}$ is an independent set in G. We partition $\mathcal{I}$ into $\mathcal{I}_1 := \mathcal{I} \cap V'$ and $\mathcal{I}_2 := \mathcal{I} \setminus V'$. Then, $|\mathcal{I}_1| \leq \alpha(H)$ and $|\mathcal{I}_2| \leq \alpha(G-H)$. As a result, $|\mathcal{I}| \leq \alpha(H) + \alpha(G - H)$. □

Our second lemma states that if G contains a cycle C of even length and every other vertex in C has no neighbors outside of C, then any maximum independent set in G contains exactly half of the vertices in C.

Lemma 5.3. *Suppose that a graph G contains a cycle C consisting of the vertices $v_1, v_2, \ldots, v_{2k}$ and that $\deg(v_{2i}) = 2$ for all $i = 1, \ldots, k$. Then, $\alpha(G) = k + \alpha(G-C)$.*

Proof. If $\mathcal{I}$ is an independent set of size $\alpha(G-C)$ in $G-C$, then the set $\mathcal{I} \cup \{v_2, v_4, \ldots, v_{2k}\}$ is an independent set in G of size $k + \alpha(G - C)$ in G. So, $\alpha(G) \geq k + \alpha(G - C)$. Conversely, $\alpha(G) \leq \alpha(G-C) + \alpha(C) = \alpha(G-C) + k$, by Lemma 5.2 and because $\alpha(C_{2k}) = k$ for all $k \geq 2$. □

We will use cycles of even length as filler gadgets and Lemma 5.3 justifies this. We will use paths as vertex and edge gagets. We will show that, if we replace a vertex by a path of length $2k + 1$ connecting the vertex's neighbors to the odd-numbered vertices of the path, then we increase the graph's independence number by exactly k. An example of this transformation is depicted in Figure 5.1.

Lemma 5.4. *Let G be a graph and G' the graph obtained from G by replacing a vertex v with a path consisting of the vertices $v_1, \ldots, v_{2k+1}$ and connecting each neighbor of v with v_{2i+1} for some i. Then $\alpha(G') = \alpha(G) + k$.*

Proof. Suppose $\mathcal{I}$ is an independent set in G. If $v \in \mathcal{I}$, then $\mathcal{I}' := (\mathcal{I} \setminus \{v\}) \cup \{v_1, v_3, v_5, \ldots, v_{2k+1}\}$ is an independent set in G' with $|\mathcal{I}'| = |\mathcal{I}| + k$. If $v \notin \mathcal{I}$, then $\mathcal{I}' = \mathcal{I} \cup \{v_2, v_4, v_6, \ldots, v_{2k}\}$ is an independent set in G' with $|\mathcal{I}'| = |\mathcal{I}| + k$. Hence, $\alpha(G') \geq \alpha(G) + k$.

Now suppose that $\mathcal{I}'$ is an independent set in G'. It may contain at most $k + 1$ vertices from the introduced path. If $\mathcal{I}'$ contains $k+1$ vertices on the path, then it must contain $v_1, v_3, v_5, \ldots, v_{2k+1}$ and may not contain any vertex u that is adjacent to v in G. Hence, $\mathcal{I} := (\mathcal{I}' \setminus \{v_1, v_3, v_5, \ldots, v_{2k+1}\}) \cup \{v\}$ is an independent set in G with $|\mathcal{I}| + k = |\mathcal{I}'|$. If $\mathcal{I}'$ contains at most k vertices on the introduced path, then $\mathcal{I} \setminus \{v_1, \ldots, v_{2k+1}\}$ is an independent set in G with $|\mathcal{I}| + k \geq |\mathcal{I}'|$. Hence, $\alpha(G') \leq \alpha(G) + k$. □

This directly implies:

Corollary 5.5. *If G' is obtained from the graph G by subdividing an edge $2k$ times, then $\alpha(G') = \alpha(G) + k$.*

Proof. Let $\{u, v\}$ be the edge in G that is subdivided $2k$ times to obtain G'. This subdivision is tantamount to replacing u with a path of length $2k+1$, while connecting v to the last vertex on this path and all of u's other neighbors to the first vertex. By Lemma 5.4, this implies that $\alpha(G') = \alpha(G) + k$. □

5.2 Reduction

We will now give a polynomial-time many-to-one reduction from Independent Set on planar graphs with maximum degree at most 3 to Independent Set restricted to both relative neighborhood graphs and Gabriel graphs, thereby proving that it remains NP-complete on each of them. We will later discuss how the reduction can be modified to take RCGs into account.

Let (G, k) be an instance of Independent Set on planar graphs with maximum degree at most 3 consisting of the graph $G = (V, E)$ and an integer k. We compute a two-page book embedding of G in polynomial time and assume that $v_1, \ldots, v_n$ are the vertices of G in the order in which they appear on the spine of the book embedding.

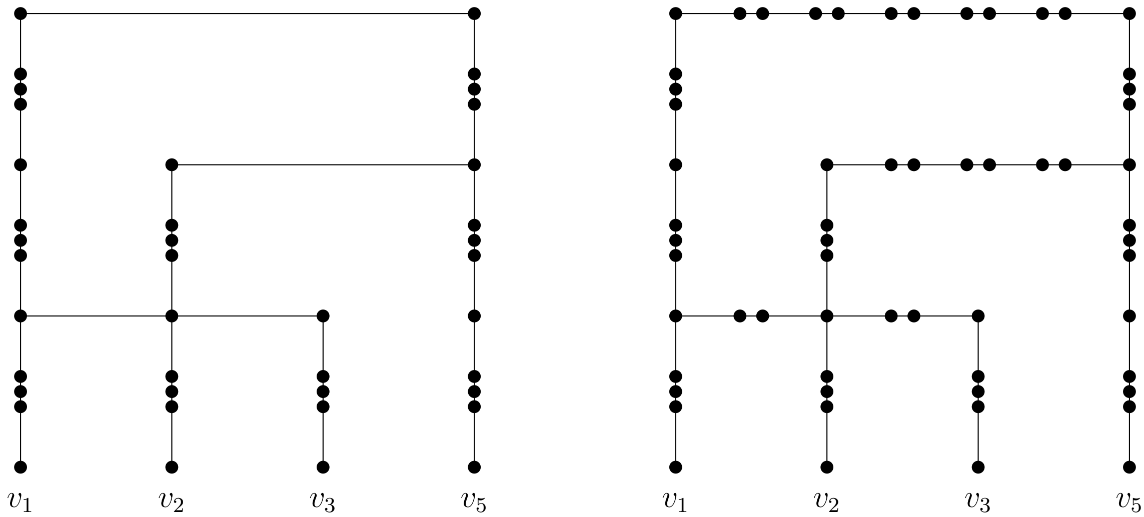
Construction

From G and k, we will now derive a graph $G' = (V', E')$ and an integer k'. At the same time, we will construct an embedding $\mathrm{emb}\colon V' \to \mathbb{R}^2$ of G' which we will later use to show that G' is an RNG and a Gabriel graph. The construction will follow the general pattern we laid out in Chapter 3 and use the terminology we defined there. The vertex and edge gadgets will consist of paths with lengths of the appropriate parity. The filler gadgets will consists of multiple cycles of even length. In order to describe edges between vertices in different gadgets we will refer to certain vertices or groups of vertices as (i, j)-corners, for integers i and j. Every corner has four outlets: the top, the left, the right, and the bottom outlet, but they may be identical. We will illustrate the construction in Figures 5.2 and 5.3 using the example graph depicted in Figure 3.1.

Fix $0 < \varepsilon < \frac{1}{4}(\sqrt{3} - 1)$.

Step 1: Start with $G' \coloneqq G$ and $k' \coloneqq k$. Set $\mathrm{emb}(v_i) \coloneqq (2i, 0)$. We will call v_i the $(2i, 0)$-corner vertex. The vertex v_i is simultaneously the top, left, right, and bottom outlet of this corner.

Step 2: For every $i = 1, \ldots, n$, add a path consisting of $4h_1(v_i)$ new vertices $w_1, \ldots, w_{4h_1(v_i)}$. Connect v_i to w_1. Increase k' by $4h_1(v_i)$. We will call this path v_i's *up-path*. We will call w_j the j-th vertex on this up-path. Embed the new vertices as



(a) G' after **Step 2**: Every vertex v_i is replaced by a path consisting of $4h_1(v_i))$ vertices. Edges incident to v_i are connected to the $4h(e)$-th vertex on this path.

(b) G' after **Step 3**: Every edge e is replaced by a path consisting of $4\ell(e)-2$ new vertices.

Figure 5.2: Addition of vertex and edge gadgets.

follows:

$$\operatorname{emb}(w_j) := \begin{cases} (2i, 2r+1-\varepsilon), & \text{if } j = 4r+1, \\ (2i, 2r+1), & \text{if } j = 4r+2, \\ (2i, 2r+1+\varepsilon), & \text{if } j = 4r+3, \\ (2i, 2r+2), & \text{if } j = 4r+4. \end{cases}$$

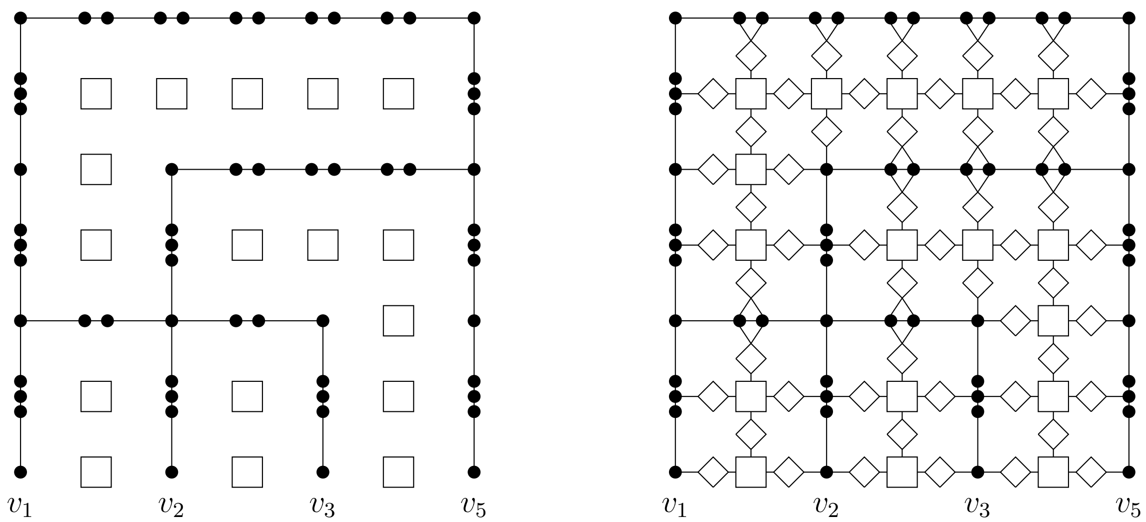
For all $j = 1, \ldots, h_1(v_i)$, we will call w_{2j} the $(2i, j)$-corner vertex and w_{2j} is again simultaneously the top, left, right, and bottom outlet of the corner. For every edge $e = \{v_i, v_j\}$, delete e and replace it by connecting the $4h(e)$-th vertex in v_i's up-path to the $4h(e)$-th vertex in v_j's up-path. This step is illustrated in Figure 5.2a.

Step 3: For every edge $e = \{v_i, v_j\} \in E_1$, $i < j$, do the following: subdivide the edge corresponding to e with the $4\ell(e) - 2$ new vertices $w_1, \ldots, w_{4\ell(e)-2}$. We will call the resulting path e's *edge path* and call w_j the j-th vertex on the path. For every $r = 1, \ldots, \ell(e) - 1$, we will refer to the pair (w_{2r-1}, w_{2r}) as the $(2i + r, 2h(e))$-corner pair. The two vertices are simultaneously the top, left, right, and bottom outlet of the corner. We set

$$\operatorname{emb}(w_r) := \begin{cases} (2i + 2s - \varepsilon, 2h(e)), & \text{if } r = 2s - 1, \\ (2i + 2s + \varepsilon, 2h(e)), & \text{if } r = 2s. \end{cases}$$

Increase k' by $\ell(e)$. This step is illustrated in Figure 5.2b.

Step 4: For every $i = 2, \ldots, 2n$ and $j = 0, \ldots, 2h_1(G)$ such that:



(a) G' after **Step 4**: Every square represents an 8-cycle. Such cycles are added for every (i, j) in the area of the plane occupied by the graph if there is no vertex or edge gadget passing through (i, j).

(b) G' after step **5**: Every diamond represents a 4-cycle. These cycles are used to connect 8-cycles to other 8-cycles or vertices in the vertex or edge gadgets.

Figure 5.3: Addition of filler gadgets.

- i is odd or $h_1(v_{\frac{i}{2}}) < j$ (in other words, there is no (i, j)-corner in an up-path) and
- j is odd or there is no edge $e = \{v_r, v_s\}$ with $r < i < s$ and $h(e) = \frac{j}{2}$ (in other words, there is no (i, j)-corner in an edge path,

do the following: Add a cycle consisting of eight new vertices. The coordinates of the vertices in the embedding are given in Figure 5.4. We will call this cycle the (i, j)-corner square. The top, right, bottom, and left outlets of this corner are the respective vertices in the cycle. Increase k' by 4. Note that for every $i = 1, \dots, 2n$ and $j = 0, \dots, 2h_1(G)$ the graph now contains either an (i, j)-corner vertex, an (i, j)-corner pair, or an (i, j)-corner square. This step is illustrated in Figure 5.3a.

Step 5: For every $i = 2, \dots, 2n$ and $j = 0, \dots, 2h_1(G)$, if the (i, j)-corner is a square, then connect it to the surrounding corners in the following manner: If $i > 2$, then add a cycle consisting of four new vertices. Embed this 4-cycle as depicted in Figure 5.5. In the following, we refer to the top vertex as w_1, to the right vertex as w_2, to the bottom vertex as w_3, and to the left vertex as w_4. Connect w_2 to the left outlet of the (i, j)-corner square and w_4 to the right outlet of the $(i-1, j)$-corner. If $i < 2n$, then similarly connect right outlet of the (i, j)-corner square to the left outlet of the $(i, j+1)$ corner (unless they are already connected). If $j > 0$, then connect the bottom outlet in the (i, j)-corner square to the top outlet in the $(i, j-1)$ corner in the same way. If $j < 2h_1(G)$, then connect the top outlet in the (i, j)-corner square to the bottom outlet in the $(i, j+1)$ corner in the same manner (unless they are already connected). This step is illustrated

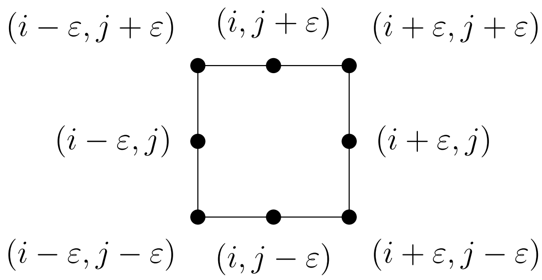


Figure 5.4: Coordinates of the vertices in the embedding of the 8-cycle.

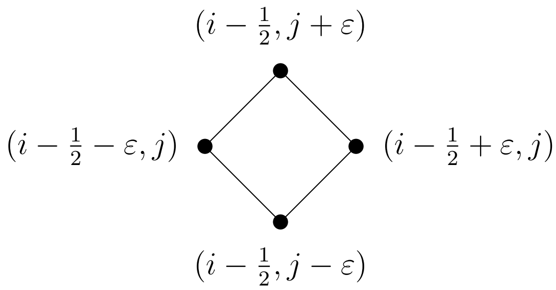


Figure 5.5: Coordinates of the vertices in the embedding of the 4-cycle.

in Figure 5.3b.

We have only taken edges in E_1 into consideration. We must, therefore, repeat the process described in **Steps 2** to **5** above for all edges in E_2, creating down-paths corresponding to the up-paths and proceeding accordingly.

Let A denote the number of 8-cycles, B the number of 4-cycles added, and C the sum of the length of all paths added. Then $k' = k + 4A + 2B + \frac{C}{2}$ by the construction above.

This concludes the construction of G' and k'.

Running time

We must show that the reduction runs in polynomial time. We have added the following vertices:

- $\mathcal{O}(n)$ copies of the original vertices,
- each up-path and down-path contains $\mathcal{O}(n)$ vertices for a total of $\mathcal{O}(n^2)$,
- $\mathcal{O}(n)$ edge paths (since G is planar it only has $\mathcal{O}(n)$ edges), each containing $\mathcal{O}(n)$ vertices for a total of $\mathcal{O}(n^2)$, and
- $\mathcal{O}(n^2)$ squares each containing $\mathcal{O}(1)$ vertices.

In total, G' contains $\mathcal{O}(n^2)$ vertices and can be computed in polynomial time.

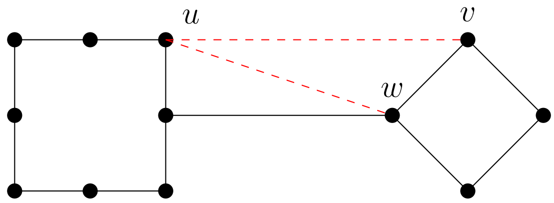


Figure 5.6: The vertex w is a Gabriel-blocker for $\{u, v\}$, since $\measuredangle\,\mathrm{emb}(u)\,\mathrm{emb}(w)\,\mathrm{emb}(v) \geq 90°$.

Correctness

We must show that $\alpha(G) \geq k$ if and only if $\alpha(G') \geq k'$. This property holds after each individual step. For **Steps 2** and **3**, this is a direct consequence of Lemma 5.4 and Corollary 5.5, respectively. For **Steps 4** and **5**, it follows from Lemma 5.3.

Graph induced by embedding

Proving that the embedding emb given in the construction induces the graph described therein as its RNG and Gabriel graph requires a similar analysis to the one given in Chapter 4 for the reduction for Dominating Set. This reduction, like the one for Dominating Set, generates a graph that is in a way grid-like. We will omit most of the details of this analysis and focus on a few aspects that may not be immediately obvious.

We only have to show that $\mathrm{GAB}(\mathrm{emb}(V')) \subseteq G' \subseteq \mathrm{RNG}(\mathrm{emb}(V'))$. Recall that $0 < \varepsilon < \frac{1}{4}(\sqrt{3} - 1)$.

To show that $\mathrm{GAB}(\mathrm{emb}(V')) \subseteq G'$, we must prove that every pair of non-adjacent vertices u, v has a Gabriel blocker. This would require checking every such pair, but in most cases it is not hard to see. It may be less obvious in the case of a corner vertex of an 8-cycle and the nearest vertex in a 4-cycle. An excerpt of G' is pictured in Figure 5.6 with the two vertices marked as u and v. Their Gabriel-blocker is marked as w. It is a Gabriel-blocker, because $\measuredangle\,\mathrm{emb}(u)\,\mathrm{emb}(w)\,\mathrm{emb}(v) = 135° - \tan^{-1}(\frac{\varepsilon}{\frac{1}{2}-2\varepsilon})$. Since $\varepsilon < \frac{1}{4}(\sqrt{3}-1) < \frac{1}{6}$, it is the case that $\tan^{-1}(\frac{\varepsilon}{\frac{1}{2}-2\varepsilon}) < 45°$ and therefore $\measuredangle\,\mathrm{emb}(u)\,\mathrm{emb}(w)\,\mathrm{emb}(v) > 90°$.

To prove that $G' \subseteq \mathrm{RNG}(\mathrm{emb}(V'))$, we must show that there is no RNG-blocker w for any adjacent vertices u, v. Again, in most cases this is not difficult to see. The subgraph pictured in Figure 5.7, which occurs along all edge paths, may be an exception. The vertex marked as w is not a blocker for u and v. The distance between u and v is 2ε, but the distance from w to both u and v is $\sqrt{(\frac{1}{2} - \varepsilon)^2 + \varepsilon^2} > 2\varepsilon$.

We will now briefly discuss relatively closest graphs. With one exception, the point set given in the reduction induces the same graph as its relatively closest graph. This exception is the subgraph mentioned in the previous paragraph and depicted in Figure 5.7. The vertex u is an RCG-blocker for $\{v, w\}$ and v is also an RCG-blocker for $\{u, w\}$. So, the relatively closest graph for this point set contains neither of the edges $\{u, w\}$

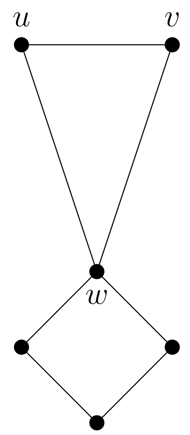


Figure 5.7: The vertex w is not an RNG-blocker for $\{u, v\}$ since $d(\mathrm{emb}(u), \mathrm{emb}(w)) = d(\mathrm{emb}(v), \mathrm{emb}(w)) \geq d(\mathrm{emb}(u), \mathrm{emb}(u))$.

and $\{v, w\}$. However, our correctness argument based on Lemma 5.3 is not dependent on the existence of these edges. Hence, this reduction must only be modified slightly to produce RCGs.

5.3 Conclusions

This concludes our reduction. Note that the graphs produced by the reduction contain only vertices with degree at most four. In all, this reduction proves:

Theorem 5.6. Independent Set *is NP-hard when restricted to relative neighborhood graphs, when restricted to Gabriel graphs, and when restricted to relatively closest graphs, each with maximum degree at most four.*

As we noted above, Independent Set and Vertex Cover can easily be reduced to one another, implying that:

Corollary 5.7. Vertex Cover *is NP-hard when restricted to relative neighborhood graphs, when restricted to Gabriel graphs, and when restricted to relatively closest graphs, each with maximum degree at most four.*

For graphs with maximum degree two, Independent Set and Vertex Cover are easy to solve. It remains open whether these problems can be solved in polynomial time when restricted to proximity graphs with maximum degree three.

We will show that our reduction also implies that Independent Set and Vertex Cover cannot be solved in time $2^{o(n^{\frac{1}{4}})}$ on each of our proximity graph classes (where n is the number of vertices in a graph), unless the exponential time hypothesis fails (on the ETH, see Section 2.5).

First, we must establish an ETH-based lower bound for these two problems on planar graphs with maximum degree 3. As Lokshtanov, Marx, and Saurabh [LMS11] point out, VERTEX COVER cannot be solved in $2^{o(\sqrt{n})}$ on planar graphs unless the ETH fails. This follows from the NP-hardness proof given by Garey, Johnson, and Stockmeyer [GJS76].

Theorem 5.8 ([GJS76; LMS11])**.** *Unless the ETH fails, planar* VERTEX COVER *and planar* INDEPENDENT SET *cannot be solved in time* $2^{o(\sqrt{n})}$.

Garey and Johnson [GJ76] give a reduction from planar VERTEX COVER to VERTEX COVER restricted to planar graphs with maximum degree 3. The central step in this reduction is replacing each vertex by a cycle of length $2n$ and one additional vertex. This implies that the resulting graph contains $O(n^2)$ vertices. However, with a slight modification we can devise a reduction that produces graphs containing $O(n)$ vertices. The cycle replacing each vertex only needs to contain vertices representing its neighbors rather than all other vertices. Rather than replacing each vertex v with a cycle containing $2n$ vertices and an additional vertex, we replace it with a cycle containing $2\deg(v)$ vertices and an additional vertex. Then, the resulting graph contains

$$\sum_{v \in V} 2\deg(v) + 1 = n + 2\sum_{v \in V} \deg(v) = n + 4|E| \in \mathcal{O}(n)$$

vertices. Then, an algorithm for VERTEX COVER or INDEPENDENT SET restricted to planar graphs with maximum degree 3 running in time $2^{o(\sqrt{n})}$ would imply an algorithm with the same running time for the same problem on arbitrary planar graphs. Together with Theorem 5.8 this proves:

Theorem 5.9. *Unless the ETH fails,* VERTEX COVER *and* INDEPENDENT SET *restricted to planar graphs with maximum degree* 3 *cannot be solved in time* $2^{o(\sqrt{n})}$.

As we mentioned before, our reduction for INDEPENDENT SET restricted to RNGs, RCGs, and Gabriel graphs produces a graph containing $\mathcal{O}(n^2)$ vertices. Hence, Theorem 5.9 implies:

Corollary 5.10. *Unless the ETH fails,* INDEPENDENT SET *and* VERTEX COVER *restricted to RNGs, RCGs, or Gabriel graphs cannot be solved in time* $2^{o(n^{\frac{1}{4}})}$.

6 3-Colorability

We will now prove that the 3-COLORABILITY problem is also NP-hard when restricted to relative neighborhood graphs or Gabriel graphs. Recall that a graph $G = (V, E)$ is 3-colorable if there is a function $c\colon V \to \{1, 2, 3\}$ such that $c(u) \neq c(v)$ for all $\{u, v\} \in E$. The 3-COLORABILITY problem is:

3-COLORABILITY
Input: A graph $G = (V, E)$.
Question: Is G 3-colorable?

Our polynomial-time many-to-one reduction will be from 3-COLORABILITY on planar graphs with maximum degree 4. Garey, Johnson, and Stockmeyer [GJS76] showed that 3-COLORABILITY remains NP-hard when restricted to planar graphs with maximum degree at most 4.

Theorem 6.1 ([GJS76]). 3-COLORABILITY *on planar graphs with maximum degree at most four is NP-hard.*

6.1 Definitions and intermediate results

We will make use of what we will call coloring paths. A coloring path essentially allows us to copy the color of a vertex. This makes them useful as vertex and edge gadgets.

Definition 6.2. The *coloring path* of length k from u_0^1 to u_k^1 is the graph $\tilde{P}_k := (V_k, E_k)$ with:

$$V_k := \{u_0^1\} \cup \{u_i^1, u_i^2, u_i^3 \mid i \in \{1, \ldots, k\}\} \text{ and}$$
$$E_k := \{\{u_{i-1}^1, u_i^2\}, \{u_{i-1}^1, u_i^3\}, \{u_i^2, u_i^3\}, \{u_i^2, u_i^1\}, \{u_i^3, u_i^1\} \mid i \in \{1, \ldots, k\}\}.$$

An example of a coloring path is pictured in Figure 6.1. We will call u_i^1 the i-th *center vertex*, u_i^2 the i-th *left vertex*, and u_i^3 the i-th *right vertex*. Coloring paths can be used as vertex and edge gadgets due to the following:

Lemma 6.3. *Any 3-coloring of a coloring path assigns the same color to all center vertices.*

Proof. Let $\tilde{P}_k = (V_k, E_k)$ be a coloring path and $c\colon V_k \to \{1, 2, 3\}$ a 3-coloring of $\tilde{P}_k$.

Without loss of generality, $c(u_0^1) = 1$. Then, $\{c(u_1^2), c(u_1^3)\} = \{2, 3\}$. Since u_1^1 is adjacent to u_1^2 and u_1^3, it follows that $c(u_1^1) = 1$. By induction then, $c(u_i^1) = 1$ for all $i = 1, \ldots, k$. □

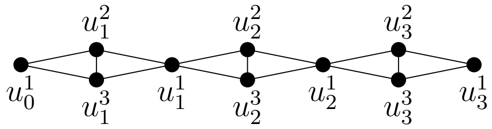


Figure 6.1: A coloring path of length 3.

This directly implies:

Lemma 6.4. *Suppose that $G = (V, E)$ is a graph and $G' = (V', E')$ is obtained from G by replacing a vertex $v \in V$ with a coloring path of any length and connecting each of v's neighbors to one of the center vertices on the introduced coloring path. Then, G is 3-colorable if and only if G' is.*

Proof. Suppose that $c\colon V \to \{1, 2, 3\}$ is a 3-coloring of G. Without loss of generality, assume that $c(v) = 1$. Then, $c'\colon V' \to \{1, 2, 3\}$, with $c'(u) := c(u)$ for all $u \in V \setminus \{v\}$, $c'(u^1) = 1$ for all center vertices u^1, $c'(u^2) := 2$ for all left vertices u^2, and $c'(u^3) := 3$ for all right vertices u^3, is a valid 3-coloring of G'.

Now suppose that $c'\colon V' \to \{1, 2, 3\}$ is a 3-coloring of G'. By Lemma 6.3, c' assigns the same color to all center vertices in the coloring path, without loss of generality the color 1. Hence, $c'(u) \neq 1$ for all vertices $u \in V \setminus \{v\}$ that are adjacent to v in g, since they are adjacent to a center vertex in G'. Then, $c\colon V \to \{1, 2, 3\}$ with $c(v) := 1$ and $c(v) := c'(u)$ for all $u \in V \setminus \{v\}$ is a valid 3-coloring of G. □

Adding a high-degree vertex to a 3-colorable graph can make it non-3-colorable. However, if we additionally subdivide every edge incident to the new vertex, then its addition does not affect the 3-colorability of the graph:

Lemma 6.5. *Let $G = (V, E)$ be a graph and $v_1, \ldots, v_k \in V$ vertices. Suppose that G' is obtained from G by adding $k+1$ vertices $w, u_1, \ldots, u_k$ and the edges $\{v_i, u_i\}$ and $\{u_i, w\}$ for every $i = 1, \ldots, k$. Then, G is 3-colorable if and only if G' is.*

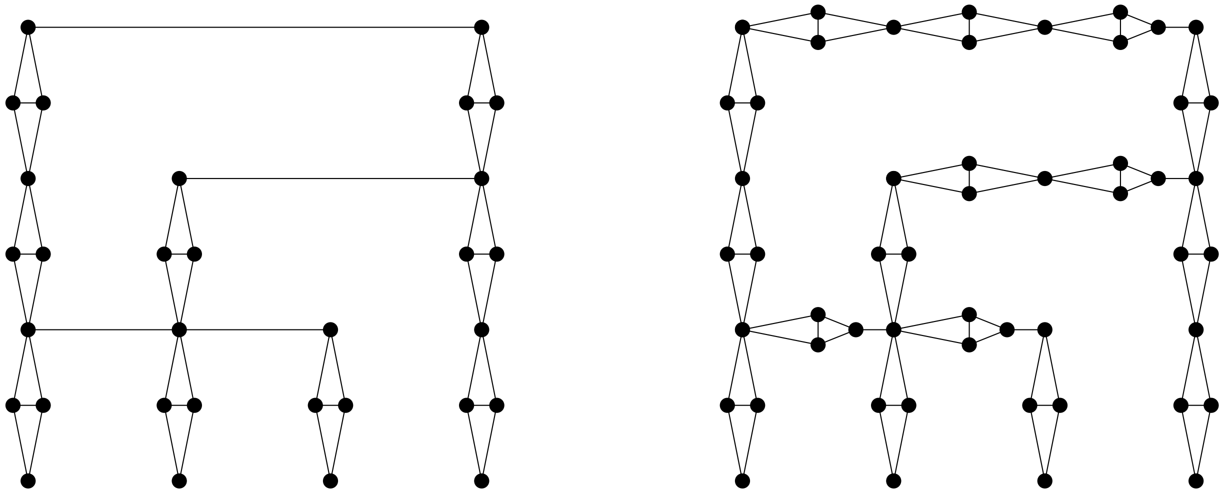
Proof. Suppose that $c\colon V \to \{1, 2, 3\}$ is a 3-coloring of G. Let $c'\colon V' \to \{1, 2, 3\}$ with $c'(v) := c(v)$ for all $v \in V$, $c(w) := 3$, and:

$$c'(u_i) := \begin{cases} 1, & \text{if } c(v_i) \neq 1, \\ 2, & \text{if } c(v_i) = 1. \end{cases}$$

Then, c' is a valid 3-coloring of G'.

If G' is 3-colorable, then so is G, since G is a subgraph of G'. □

Lemma 6.5 will be the justification for our using single vertices connected to the rest of the graph via intermediate vertices of degree two as filler gadgets.



(a) G' after **Step 2**: Each vertex v_i is replaced by a coloring coloring path of length $h_1(v_i)$. The edges incident to v_i are connected to the center vertices of the path.

(b) G' after **Step 3**: Each edge e is replaced by a coloring path of length $\ell(e)$.

Figure 6.2: Addition of vertex and edge gadgets.

6.2 Reduction

We will now give a polynomial-time many-to-one reduction from 3-Colorability on planar graphs with maximum degree at most four to 3-Colorability on relative neighborhood graphs and Gabriel graphs, thereby proving that the problem is NP-complete on both of these graph classes. We will subsequently discuss relatively closest graphs.

Let $G = (V, E)$ be a planar graph of maximum degree at most 4. We again compute a two-page book embedding of G in polynomial time and assume that $v_1, \dots, v_n$ are the vertices of G in the order in which they appear on the spine of the book embedding.

Construction

We will now give a graph $G' = (V', E')$, which is both an RNG and a Gabriel graph, such that G is 3-colorable if and only if G' is. We will also give an embedding emb: $V' \to \mathbb{R}^2$, which we will use to show that G' is an RNG and a Gabriel graph. The reduction will follow the general pattern we described in Chapter 3 and use the terminology defined there. We will use coloring paths as vertex and edge gadgets and single vertices as filler gadgets. In order to describe the edges between vertices in different gadgets we will refer to certain vertices or groups of vertices as (i, j)-corners, for integers i and j. Every corner has up to four outlets: a top, a left, a right, and a bottom outlet, but they may be identical. As in the previous reduction, we will use the graph G pictured in Figure 3.1, to illustrate the reduction in Figures 6.2a, 6.2b, and 6.3.

Let $0 < \varepsilon < \frac{1}{4}$.

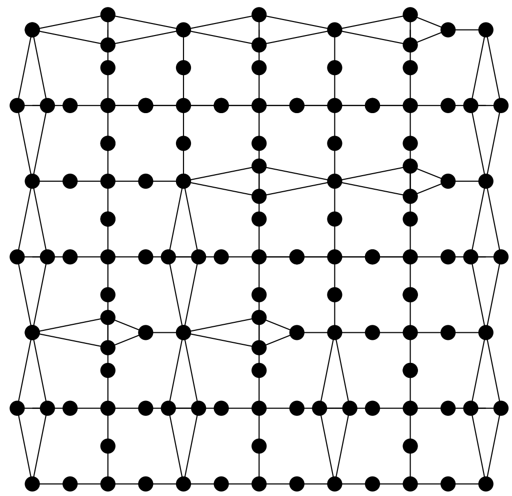


Figure 6.3: G' after **Step 4**: Filler gadgets consisting of single vertices connected to the rest of the graph via intermediate vertices of degree two are added to the graph.

Step 1: Start with $G' \coloneqq G$.

Step 2: Replace every vertex v_i with a coloring path of length $h_1(v_i)$. We will call this coloring path v_i's *up-path*. The vertices on the up-paths are embedded in the following manner: Suppose $u_0^1, \dots, u_{h_1(v_i)}^1$ are the center vertices, $u_1^2, \dots, u_{h_1(v_i)}^2$ are the left vertices, and $u_1^3, \dots, u_{h_1(v_i)}^3$ are the right vertices on that path. Then, $\mathrm{emb}(u_j^1) \coloneqq (2i, 2j)$ for all $j = 0, \dots, h_1(v_i)$ and $\mathrm{emb}(u_j^2) \coloneqq (2i - \varepsilon, 2j - 1)$ and $\mathrm{emb}(u_j^3) \coloneqq (2i + \varepsilon, 2j - 1)$ for all $j = 1, \dots, h_1(v_i)$. The center vertex u_j^1 is the $(2i, 2j)$-corner. The left and right vertices u_j^2 and u_j^3 jointly form the $(2i, 2j-1)$-corner, of which u_j^2 is the left outlet and u_j^3 is the right outlet. Replace every edge $e = \{v_i, v_j\}$ with an edge from the $h(e)$-th center vertex on v_i's up-path to the $h(e)$-th vertex on v_j's up-path. This step is illustrated in Figure 6.2a.

Step 3: Replace the edge introduced in the previous step to represent the edge $e = \{v_i, v_j\}$, $i < j$, $e \in E_1$, in the following manner: add a coloring path of length $\ell(e)$ from the $h(e)$-th center vertex in v_i's up-path to a new vertex $u_{\ell(e)}^1$. We will call this path e's *edge path*. Again, suppose $u_0^1, \dots, u_{\ell(e)}^1$ are the center vertices, $u_1^2, \dots, u_{\ell(e)}^2$ are the left vertices, and $u_1^3, \dots, u_{\ell(e)}^3$ are the right vertices on that path. Connect $u_{\ell(e)}^1$ to the $h(e)$-th center vertex of v_j's up-path. We embed the edge path with $\mathrm{emb}(u_r^1) \coloneqq (2i + 2r, 2h(e))$ for every $r = 1, \dots, \ell(e) - 1$, $\mathrm{emb}(u_r^2) \coloneqq (2i + 2r - 1, 2h(e) + \varepsilon)$ and $\mathrm{emb}(u_r^3) \coloneqq (2i + 2r - 1, 2h(e) - \varepsilon)$ for all $r = 1, \dots, \ell(e)$. The $\ell(e)$-th center vertex is embedded with $\mathrm{emb}(u_{\ell(e)}^1) \coloneqq (2j - \frac{1}{2}, 2h(e))$. For $r = 0, \dots, \ell(2) - 1$, the center vertex u_r^1 is the $(2i + 2r, 2h(e))$-corner. The left and right vertices u_r^2 and u_r^3 are jointly the $(2i + 2r - 1, 2h(e))$-corner, with u_r^2 the top outlet and u_r^3 the bottom outlet. The last center vertex $u_{\ell(e)}^1$ is not a corner. This step is illustrated in Figure 6.2b.

Step 4: For every $i = 2, \dots, 2n$ and $j = 0, \dots, 2h_1(G)$ add a vertex $w_{i,j}$ if there is no (i, j)-corner. The position of this vertex is $\mathrm{emb}(w_{i,j}) \coloneqq (i, j)$. If $i > 2$, then connect $w_{i,j}$ to the right outlet of the $(i-1,j)$-corner in the following manner: add a new intermediate vertex $\tilde{w}$ and edges from $\tilde{w}$ to both $w_{i,j}$ and to the right outlet of the $(i-1,j)$-corner. The immediate vertex is embedded by $\text{emb}(\tilde{w}) := (i-\frac{1}{2}, j)$. If $i < 2n$, then connect $w_{i,j}$ to the left outlet of the $(i+1,j)$-corner in the same manner. If $j > 0$, then $w_{i,j}$ is connected to the bottom outlet of the $(i, j-1)$-corner, and if $j < 2h_1(G)$, to the top outlet of the $(i, j+1)$-corner. This step is illustrated in Figure 6.3.

We have only taken edges in E_1 into consideration. We must, therefore, repeat the process described in **Steps 2** to **4** above for all edges in E_2, creating down-paths corresponding to the up-paths and proceeding accordingly.

Running time

We must show that the reduction runs in polynomial time. The graph G' contains:

- $\mathcal{O}(n)$ up-paths and down-paths containing $\mathcal{O}(n)$ vertices each, for a total of $\mathcal{O}(n^2)$ vertices,
- $\mathcal{O}(n)$ edge paths containing $\mathcal{O}(n)$ vertices each, for a total of $\mathcal{O}(n^2)$ vertices, and
- $\mathcal{O}(n^2)$ vertices introduced in **Step 4**.

In total, G' contains $\mathcal{O}(n^2)$ vertices and can be computed in polynomial time.

Correctness

We must show that G is 3-colorable if and only if G' is. This is true after each step. For **Step 2**, it follows from Lemma 6.4, because in this step vertices are replaced by coloring paths and their neighbors are connected to that path's center vertices. For **Step 3**, it is also a consequence of Lemma 6.4. In this step, an edge $\{u, v\}$ is replaced with a coloring path from u to a new vertex w and w is connected to v. This is tantamount to replacing u with a coloring path from u to w. For step **Step 4**, our claim follows from Lemma 6.5, since each new vertex added in this step is connected via intermediate vertices of degree 2 to existing vertices.

Graph induced by embedding

Proving that the embedding emb given in the construction induces the graph described therein as its RNG and Gabriel graph requires a similar analysis to the one given in Chapter 4 for the reduction for Dominating Set. This reduction, like the one for Dominating Set, generates a graph that is in a way grid-like. We will omit most of the details of this analysis and focus on a few aspects that may not be immediately obvious.

As in the previous section, we only have to argue that $\text{GAB}(\text{emb}(V')) \subseteq G' \subseteq \text{RNG}(\text{emb}(V'))$.

To show that $\text{GAB}(\text{emb}(V')) \subseteq G'$, we must prove that every pair of non-adjacent vertices u, v has a Gabriel blocker. We will only consider the non-obvious cases. For

(a) The vertex w is a Gabriel-blocker for $\{u, v\}$, since $\measuredangle \operatorname{emb}(u) \operatorname{emb} w \operatorname{emb}(v) = 90°$.

(b) The vertex v is not an RNG-blocker for $\{u_1, u_2\}$ or $\{w_1, w_2\}$.

Figure 6.4: The graph G' is induced by emb.

instance, consider the vertices labeled u and v in the subgraph of G' pictured in Figure 6.4a. This subgraph occurs as a result of the vertices added in **Step 4**. The blocker for these vertices is the vertex labeled as w since $\measuredangle \operatorname{emb}(u) \operatorname{emb}(w) \operatorname{emb}(v) = 90°$

To prove that $G' \subseteq \operatorname{RNG}(\operatorname{emb}(V'))$, we have to show that there is no RNG-blocker for any edge $\{u, v\} \in E'$. Again, we restrict our attention to the potentially controversial cases. Consider the subgraph pictured in Figure 6.4b, which occurs at the ends right end of each edge path. We will show that the vertex marked v is not an RNG-blocker for the edge $\{u_1, u_2\}$. We recall that $0 < \varepsilon < \frac{1}{4}$. The distance between u_1 and u_2 is $d(\operatorname{emb}(u_1), \operatorname{emb}(u_2)) = 2\varepsilon$, while

$$\begin{aligned} d(\operatorname{emb}(u_1), \operatorname{emb}(v))^2 &= d(\operatorname{emb}(u_2), \operatorname{emb}(v))^2 \\ &= \varepsilon^2 + \left(\frac{1}{2}\right)^2 \\ &= 4\varepsilon^2 + \frac{1}{4} - 3\varepsilon^2 \\ &> 4\varepsilon^2 + \frac{1}{4} - \frac{3}{16} \\ &> 4\varepsilon^2 = d(\operatorname{emb}(u_1), \operatorname{emb}(u_2))^2. \end{aligned}$$

This implies that $d(\operatorname{emb}(u_1), \operatorname{emb}(v)) > d(\operatorname{emb}(u_1), \operatorname{emb}(u_2))$ and $d(\operatorname{emb}(u_2), \operatorname{emb}(v)) > d(\operatorname{emb}(u_1), \operatorname{emb}(u_2))$, so v does not block the edge $\{u_1, u_2\}$.

Now we will show that the vertex marked v is also not an RNG-blocker for the edge $\{w_1, w_2\}$:

$$\begin{aligned} d(\operatorname{emb}(v), \operatorname{emb}(w_2))^2 &= (\frac{1}{2} - \varepsilon)^2 + 1 \\ &= \frac{1}{4} - \varepsilon + \varepsilon^2 + 1 \\ &> \varepsilon^2 + 1 \\ &= d(\operatorname{emb}(w_1), \operatorname{emb}(w_2))^2. \end{aligned}$$

This implies that $d(\operatorname{emb}(v), \operatorname{emb}(w_2)) > d(\operatorname{emb}(w_1), \operatorname{emb}(w_2))$, so v is also not an RNG-blocker for $\{w_1, w_2\}$.

6.3 Conclusions

This concludes the reduction. Note that in the graphs generated by the reduction no vertex has a degree greater than seven. It proves the following:

Theorem 6.6. 3-Colorability *is NP-complete both when restricted to relative neighborhood graphs and when restricted to Gabriel graphs, each with maximum degree at most seven.*

This theorem confirms a conjecture by Cimikowski [Cim89].

The 3-Colorability problem is trivial when restricted to proximity graphs with maximum degree at most three. This follows from Brooks' theorem:

Theorem 6.7 (Brooks' theorem [Lov75])**.** *Any graph G with $\Delta(G) \geq 3$ that does not contain a clique of size $\Delta(G) + 1$ is $\Delta(G)$-colorable.*

As we mentioned in Section 2.3, RNGs, RCGs, and Gabriel graphs do not contain cliques of size 4. So, by Brooks' theorem, all RNGs, RCGs, and Gabriel graphs with maximum degree three are 3-colorable. Additionally, all graphs with maximum degree less than three are, of course, also 3-colorable. It remains open whether 3-Colorability can be solved in polynomial time when restricted to proximity graphs with maximum degree between four and six.

As Cimikowski [Cim92] remarks, all relatively closest graphs are also 3-colorable. This is because RCGs do not contain any 3-cycles and planar graphs without 3-cycles are 3-colorable by Grötzsch's theorem [Grü63].

Theorem 6.8 ([Cim92])**.** *The* 3-Colorability *problem is trivial on relatively closest graphs as well as on relative neighborhood graphs and Gabriel graphs with maximum degree at most* 3*.*

We will show that our reduction also implies that 3-Colorability cannot be solved in time $2^{o(n^{\frac{1}{4}})}$ on each of our proximity graph classes (where n is the number of vertices in a graph), unless the exponential time hypothesis fails (on the ETH, see Section 2.5).

We must first establish an ETH-based lower bound of $2^{o(\sqrt{n})}$ for 3-Colorability on planar graphs with maximum degree 4. Garey, Johnson, and Stockmeyer [GJS76] prove that this problem is NP-hard with a series of reductions. First, 3-SAT is reduced to 3-Colorability on arbitrary graphs. This reduction maps formulas with m clauses and n variables to graphs with $\mathcal{O}(m+n)$ vertices and $\mathcal{O}(m+n)$ edges. 3-Colorability on on arbitrary graphs is then reduced to planar 3-Colorability, mapping graphs with n vertices and m edges to graphs with $\mathcal{O}(n+m^2)$ vertices and $\mathcal{O}(m^2)$ edges. Then, 3-Colorability on arbitrary planar graphs is reduced to 3-Colorability on planar graphs with maximum degree 4. This reduction replaces each vertex v with a subgraph containing $\mathcal{O}(\deg(v))$ vertices and edges. Hence, this reduction maps graphs containing n vertices and m edges to graphs with $\mathcal{O}(m)$ vertices and edges. The composition of these reductions yields a reduction from 3-SAT to 3-Colorability on planar graph with maximum degree 4 that maps a formula with m clauses and n variables to a graph containing $\mathcal{O}(m^2+n^2)$ vertices and edges. Hence:

Theorem 6.9. *Unless the ETH fails,* 3-COLORABILITY *restricted to planar graphs with maximum degree* 4 *cannot be solved in time* $2^{o(\sqrt{n})}$.

Our reduction from 3-COLORABILITY on planar graphs with maximum degree 4 to its restriction to our classes of proximity graphs maps instances with m edges and n vertices to graphs with $\mathcal{O}(n^2)$ vertices and edges. This along with Theorem 6.9 implies:

Corollary 6.10. *Unless the ETH fails,* 3-COLORABILITY *restricted to RNGs or Gabriel graphs cannot be solved in time* $2^{o(n^{\frac{1}{4}})}$.

7 Feedback Vertex Set

We now turn our attention to the FEEDBACK VERTEX SET problem in order to show that it remains NP-complete when restricted to relative neighborhood graphs, relatively closest graphs, or Gabriel graphs. Recall that a set of vertices $\mathcal{F} \subseteq V$ in a graph $G = (V, E)$ is a feedback vertex set if $G - \mathcal{F}$ is acyclic and $\varphi(G)$ denotes the size of a smallest feedback vertex set in G. The FEEDBACK VERTEX SET problem is defined as:

FEEDBACK VERTEX SET
Input: A graph $G = (V, E)$ and a nonnegative integer k.
Question: Does G contain a feedback vertex set of size at most k?

We will initially describe a reduction only for RNGs and then discuss how it can be adapted to the other two graph classes. Speckenmeyer [Spe83] proved that FEEDBACK VERTEX SET is NP-hard when restricted to planar graphs with maximum degree at most 4.

Theorem 7.1 ([Spe83])**.** FEEDBACK VERTEX SET *on planar graphs with maximum degree* 4 *is NP-hard.*

7.1 Definitions and intermediate results

Before we begin with our reduction, we will prove several intermediate results and define some terminology that will be useful in proving the correctness of the reduction.

Observation 7.2. *Let G be a graph and $C_1, \ldots, C_k$ pairwise vertex-disjoint cycles in G. Then, $\varphi(G) \geq k$.*

Proof. Any feedback vertex set in G must contain at least one vertex from each C_i. □

Lemma 7.3. *Let G be graph and $G_1, \ldots, G_k$ pairwise disjoint subgraphs in G. Then, $\varphi(G) \geq \varphi(G_1) + \ldots + \varphi(G_k)$. Moreover, if every G_i contains a minimum feedback vertex set $\mathcal{F}_i$ such that $G - \mathcal{F}_i$ contains no cycles that pass through a vertex in $G_i - \mathcal{F}_i$, then $\varphi(G) = \varphi(G_1) + \ldots + \varphi(G_k) + \varphi(G - (G_1 \cup \ldots \cup G_k))$.*

Proof. For the first claim, any feedback vertex set in G must contain at least $\varphi(G_i)$ vertices from each G_i. For the second claim, if $\mathcal{F}$ is a minimum feedback vertex set in $G - (\mathcal{F}_1 \cup \ldots \cup \mathcal{F}_k)$, then $\mathcal{F} \cup \mathcal{F}_1 \cup \ldots \cup \mathcal{F}_k$ is a minimum feedback vertex set in G containing $\varphi(G_1) + \ldots + \varphi(G_k) + \varphi(G - (G_1 \cup \ldots \cup G_k))$ vertices. □

Lemma 7.4. *If G' is obtained from G by subdividing edges, then $\varphi(G) = \varphi(G')$.*

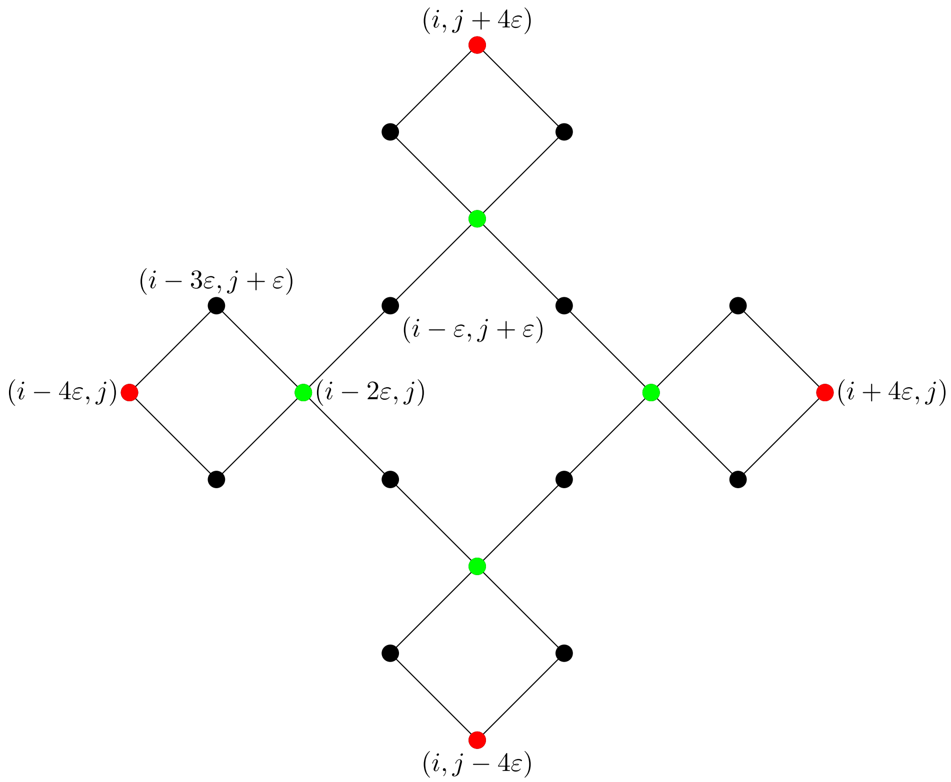


Figure 7.1: The buffer graph: The red vertices are outlets. The green vertices form a minimal feedback vertex set. This feedback vertex set disconnects the outlets from one another.

Proof. Every feedback vertex set in G is a feedback vertex set in G'. Any vertex in G' that is not part of G is on a path of vertices with degree 2 and both endpoints of this path are part of G. Replacing any such vertex in any feedback vertex set in G' by one of those endpoints yields a feedback vertex set in G containing no more vertices. □

We will call the graph pictured in Figure 7.1 a buffer and we will use it as a filler gadget and within vertex gadgets. The vertices marked in red are its outlets. Given $i, j \in \mathbb{R}$ and $\varepsilon > 0$, the embeddings of several vertices are given in the picture. All other vertices' positions may be deduced based on the graph's symmetry. The point (i, j) is the center of the cycle in the middle of the graph and ε scales the size of the embedding. The following lemma states how a buffer affects the size of a feedback vertex set of a graph.

Lemma 7.5. *Let G be a graph. Suppose that G contains a buffer B, the only vertices in B that have neighbors outside of B are the outlets, and each outlet has at most one neighbor outside of B. Then, $\varphi(G) = \varphi(G - B) + 4$.*

Proof. The subgraph B contains four disjoint cycles, so $\varphi(B) \geq 4$ by Observation 7.2. Let $\mathcal{F}$ be the set containing the four vertices marked in green in Figure 7.1. Since $\mathcal{F}$ is a feedback vertex set, $\varphi(B) = 4$. Removing $\mathcal{F}$ disconnects all outlets from one another, so, by Lemma 7.3, $\varphi(G) = \varphi(G - B) + \varphi(B) = \varphi(G - B) + 4$. □

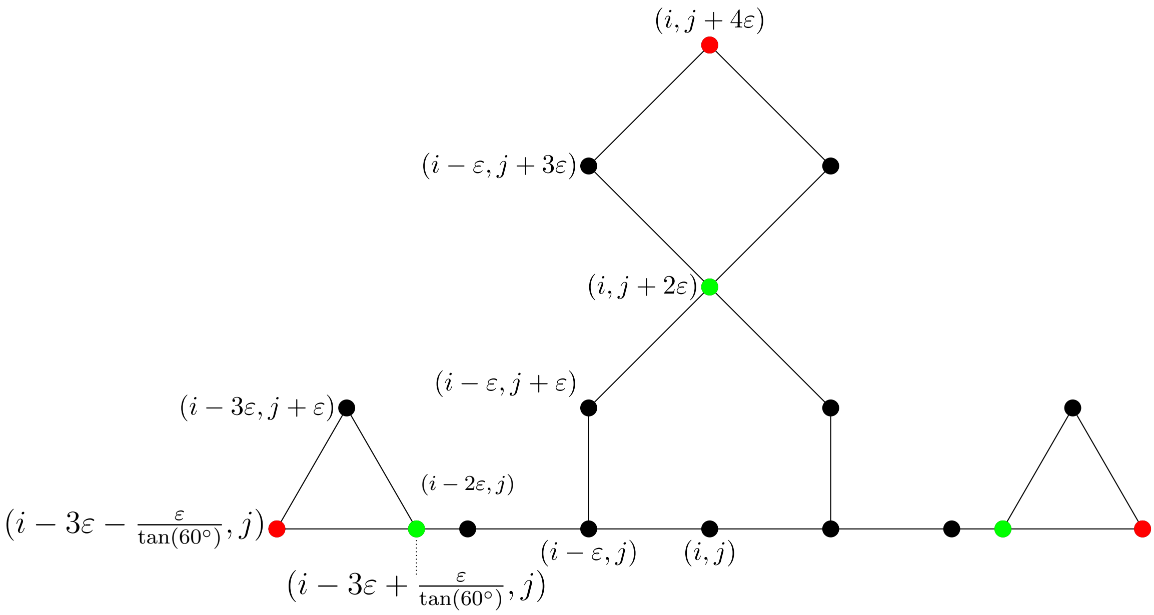


Figure 7.2: The half-buffer graph: The red vertices are the outlets. The green vertices form a minimal feedback vertex set. This feedback vertex set disconnects the outlets from one another.

The graph depicted in Figure 7.2 will be referred to as an upward-pointing half-buffer. It has three outlets, marked in red in the picture. The positions of points corresponding to the vertices are given in the image. The embedding may be rotated to create half-buffers that point left, right, or downward.

Lemma 7.6. *Suppose that G contains a half-buffer B', the only vertices in B' that have neighbors outside of B' are the outlets, and each outlet has at most one neighbor outside of B'. Then, $\varphi(G) = \varphi(G - B') + 3$.*

Proof. Identical to the proof for Lemma 7.5. □

Now we will describe the vertex gadgets. Suppose that $\deg_1(v) = a$ and $\deg_2(v) = b$. Then, we will use what we will call the (a, b)-vertex gadget. The $(4, 0)$, $(3, 1)$, and $(2, 2)$-vertex gadgets are pictured in Figures 7.3 to 7.5. In these figures, every diamond represents a buffer and every triangle represents a half-buffer facing in the appropriate direction. These gadgets each have several outlets, highlighted in red, and one central vertex, highlighted in green. The outlets above the central vertex are top outlets and those below are bottom outlets. We will refer to the leftmost top outlet as the first top outlet, to the one immediately to its right as the second top outlet, and so on. We have only given the gadgets for vertices of degree 4. Gadgets for vertices with a smaller degree can be created by combining the top and bottom halves of the gadgets we have given.

The vertex gadgets are designed to have the following property: If all buffers and half-buffers are removed from any vertex gadget, then the remaining graph consists only

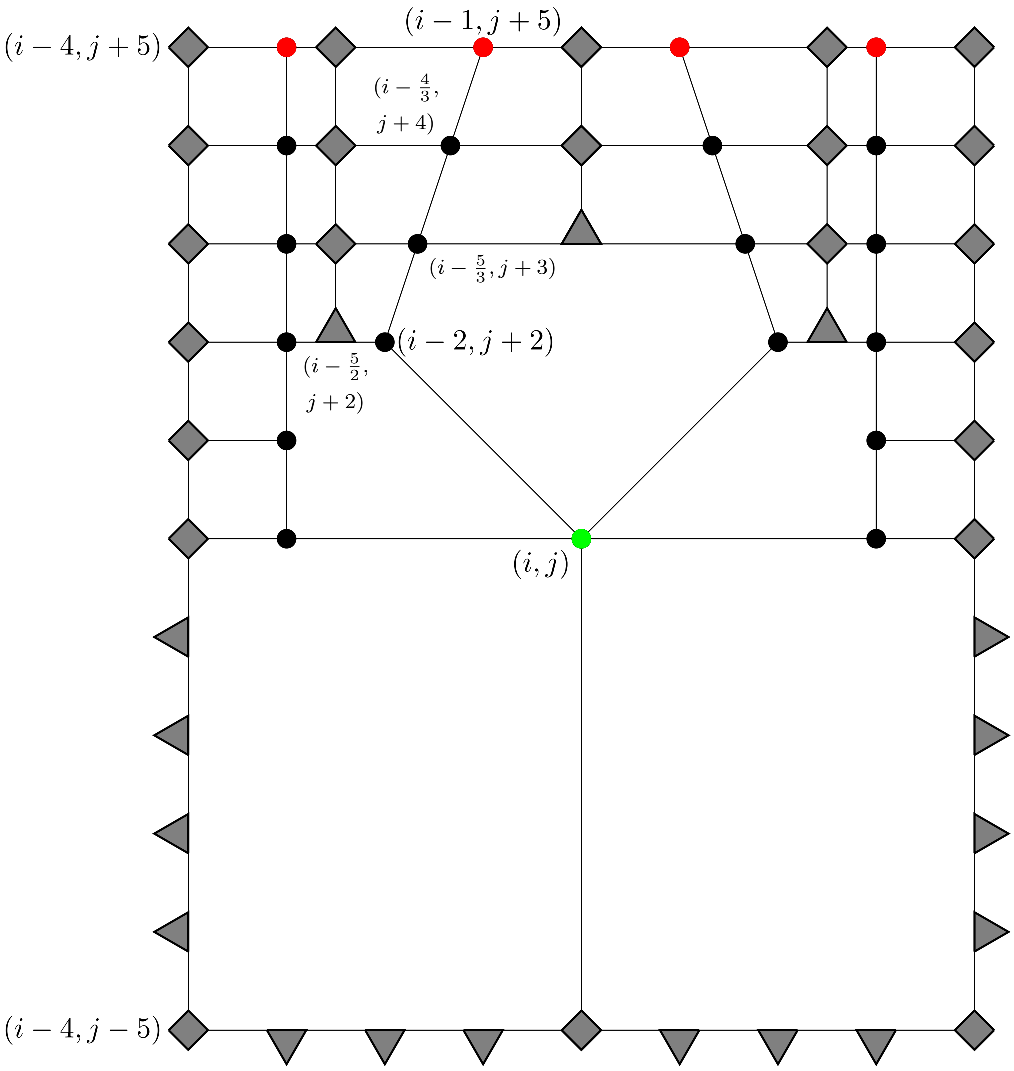


Figure 7.3: The $(4, 0)$-vertex gadget: Every diamond represents a buffer and every triangle a half-buffer. The green vertex is the central vertex and the red vertices are the outlets. Note that if all buffers and half-buffers are removed from the graph, only the central vertex, the outlets, and pairwise vertex-disjoint paths from the central vertex to each outlet remain.

of the central vertex, the outlets, and paths from the former to the latter. As a result, the following holds:

Lemma 7.7. *Let G be a graph and v a vertex in G with $a + b = \deg(v) \leq 4$ for any $a, b \in \{0, \ldots, 4\}$. Suppose G' is obtained from G by replacing v with the (a, b)-vertex gadget and connecting each of v's neighbors to a different outlet in the vertex gadget. Then, $\varphi(G') = \varphi(G) + 4F + 3F'$ where F is the number of buffers in the vertex gadget and F' the number of half-buffers.*

Proof. Follows from Lemmas 7.3 to 7.6. □

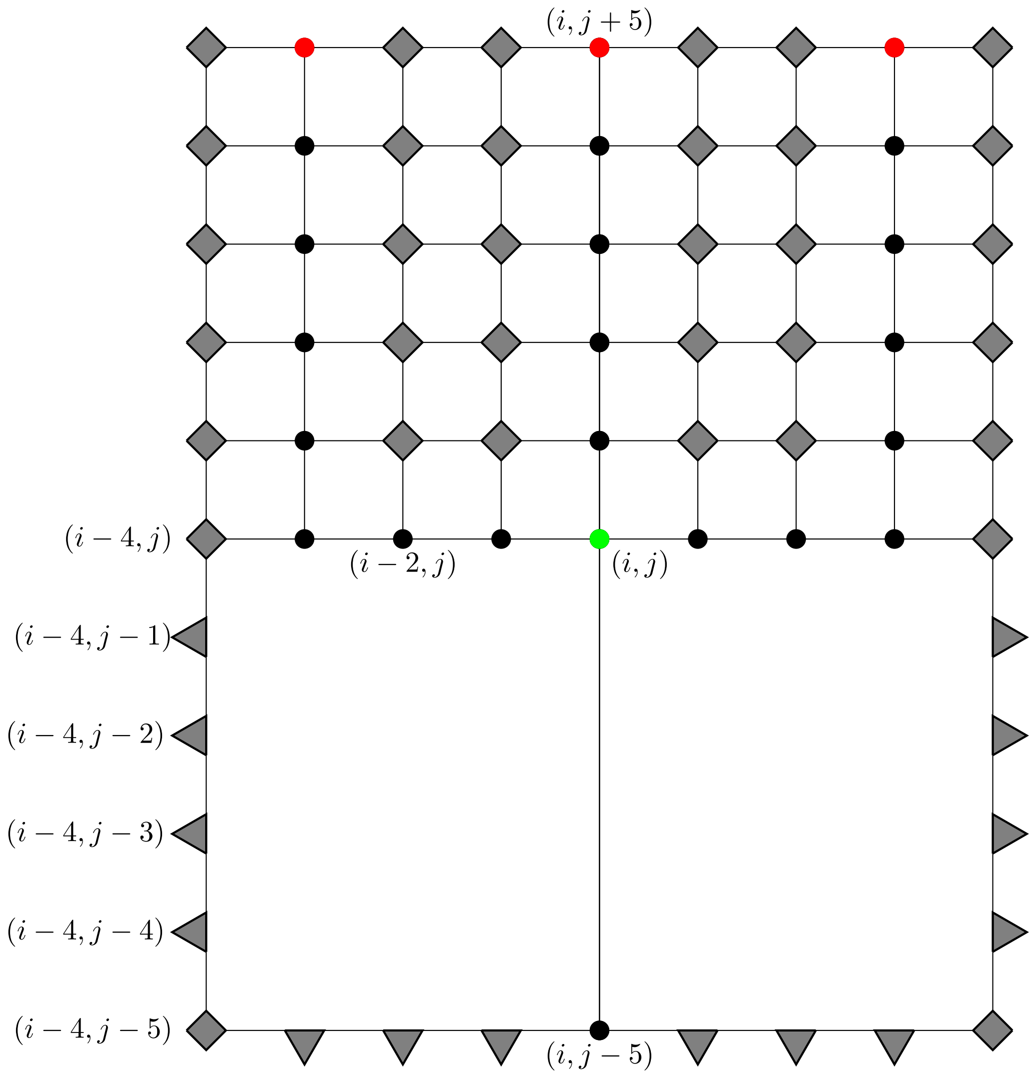


Figure 7.4: The $(3, 1)$-vertex gadget.

7.2 Reduction

We will now describe our reduction from FEEDBACK VERTEX SET on planar graphs with maximum degree at most 4 to the restriction of this problem to relative neighborhood graphs. Let $G = (V, E)$ be a planar graph with maximum degree at most 4 and k a nonnegative integer. As in our previous reductions, we compute a two-page book embedding for G in polynomial time and assume that $v_1, \ldots, v_n$ are the vertices of G in the order in which they appear on the spine of the book embedding. In the reduction, we will need an ordering of the edges incident to a certain vertex v_i. Suppose that $\{v_i, v_{j_1}\}, \ldots, \{v_i, v_{j_r}\} \in E_1$, $r = \deg_1(v_i)$, with $j_1 < \ldots < j_s < i < j_{s+1} < \ldots < j_r$, are the edges in E_1 incident to v_i. Then, the edges are ordered as follows: $\{v_i, v_{j_1}\}, \ldots, \{v_i, v_{j_s}\}, \{v_i, v_{j_r}\}, \ldots, \{v_i, v_{j_{s+1}}\}$.

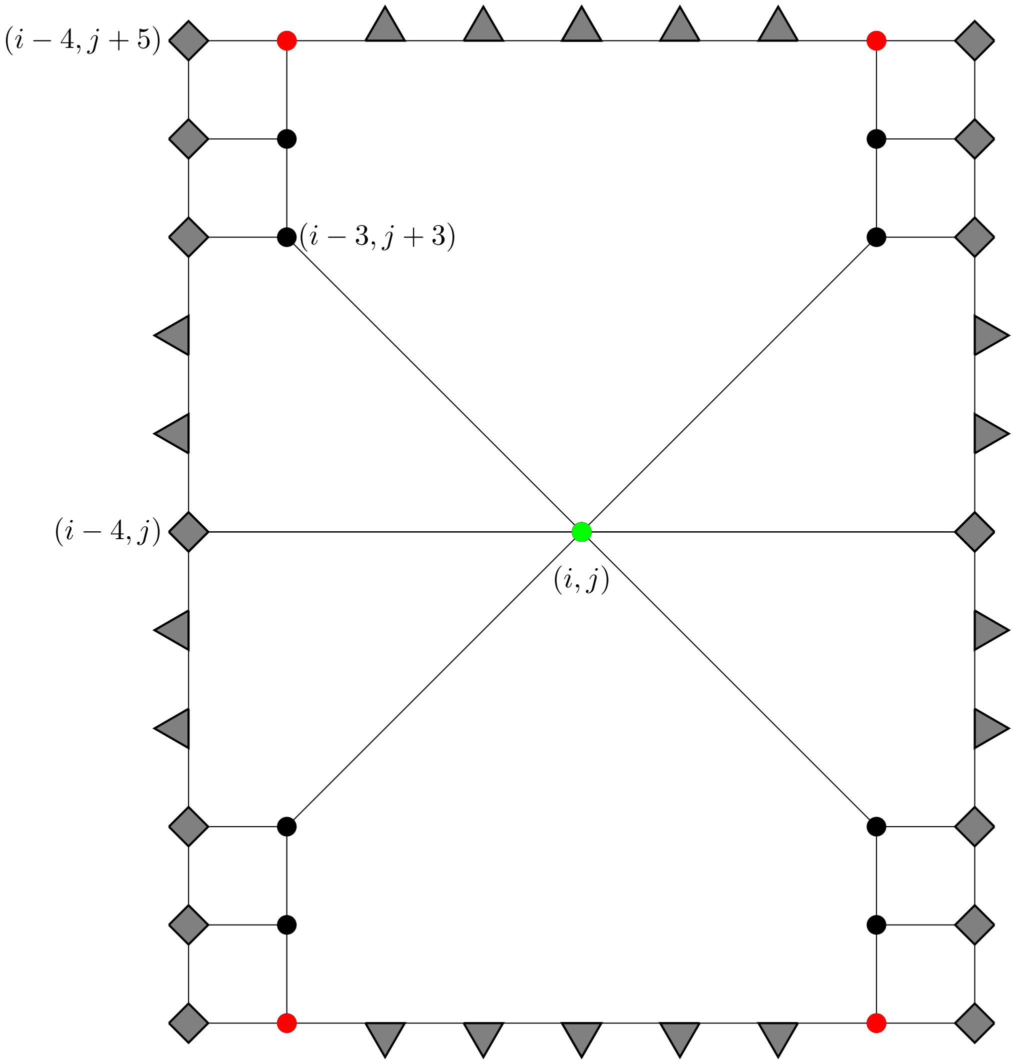


Figure 7.5: The (2, 2)-vertex gadget.

Construction

We will construct a relative neighborhood graph $G' = (V', E')$ and an integer k' such that G has a feedback vertex set of size k if and only if G' has a feedback vertex set of size k'. At the same time, we give an embedding emb, which we will subsequently use to prove that G' is an RNG by showing that $\mathrm{RNG}(\mathrm{emb}(V')) = G'$. The reduction will follow the general pattern described in Chapter 3 and use the terminology defined there. We have already described the vertex gadgets. The edge gadgets will simply be paths and the filler gadgets will be buffers. We will illustrate the construction in Figures 7.6 to 7.8 utilizing the same graph already used in previous sections and pictured in Figure 3.1. In the figures, vertex gadgets are represented by shaded areas, but their outlets are pictured.

Fix $0 < \varepsilon < \frac{1}{12}$.

Step 1: Start with $G' := G$ and $k' := k$.

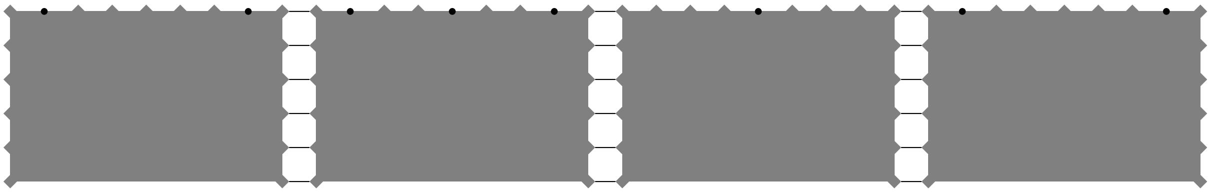


Figure 7.6: The graph G' after **Step 2**: Each vertex v_i is replaced with a copy of the $(\deg_1(v_i), \deg_2(v_i))$-vertex gadget.

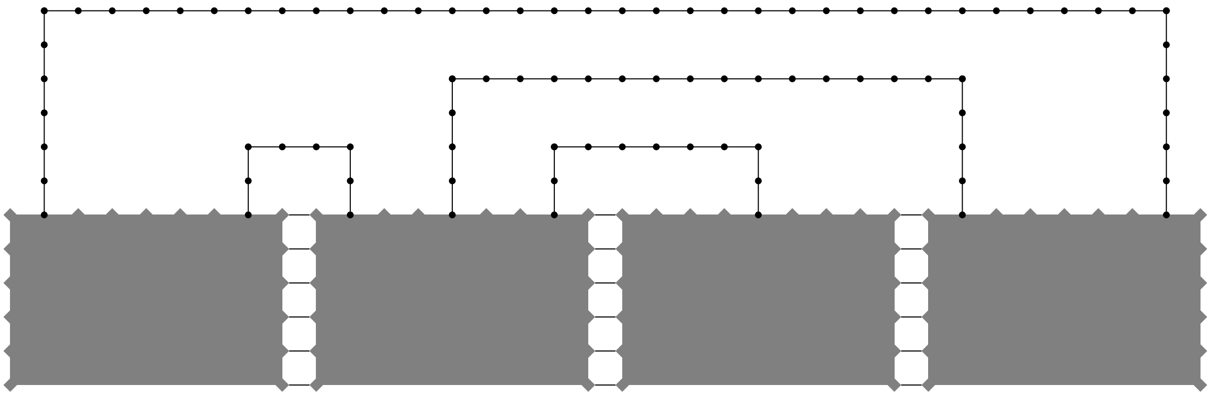


Figure 7.7: The graph G' after **Step 3**: Each edge $\{v_i, v_j\}$ is replaced by a path of the appropriate length from an outlet in v_i's vertex gadget to an outlet in v_j's vertex gadget.

Step 2: Replace every vertex v_i with the $(\deg_1(v_i), \deg_2(v_i))$-vertex gadget. Embed this gadget centered on $(9i, 0)$. Increase k' by $4F + 3F'$ where F is the number of buffers and F' the number of half-buffers in the $(\deg_1(v_i), \deg_2(v_i))$-vertex gadget. For every $i > 1$, connect the left outlets of the buffers and half buffers on the left boundary of v_i's vertex gadget with the right outlet of the buffer or half-buffer in v_{i-1}'s vertex gadget that is directly across from it. This step is illustrated in Figure 7.6.

Step 3: For every edge $e = \{v_i, v_j\} \in E_1$, $i < j$ do the following: If e is v_i's r-th edge in E_1 and v_j's s-th edge in E_1, then replace it with an edge e' from the r-th top outlet of v_i's vertex gadget to the s-th outlet of v_j's vertex gadget. Let:

$$\alpha := \begin{cases} 2r - 5, & \text{if } \deg_1(v_i) = 4, \\ 3r - 6, & \text{if } \deg_1(v_i) = 3, \\ 6r - 9, & \text{if } \deg_1(v_i) = 2, \\ 0, & \text{if } \deg_1(v_i) = 1. \end{cases}$$

This value represents the difference between the horizontal position of the central vertex of v_i's vertex gadget and r-th top outlet of the same gadget. Subdivide e', the newly created edge, $2h(e)$ times and let $w_1, \ldots, w_{2h(e)}$ be the vertices created by the subdivision. Embed these vertices with $\mathrm{emb}(w_t) := (9i + \alpha, 6 + t)$. Define β in the same way as α but for the s-th top outlet in v_j's vertex gadget. We subdivide e' an additional $2h(e)$

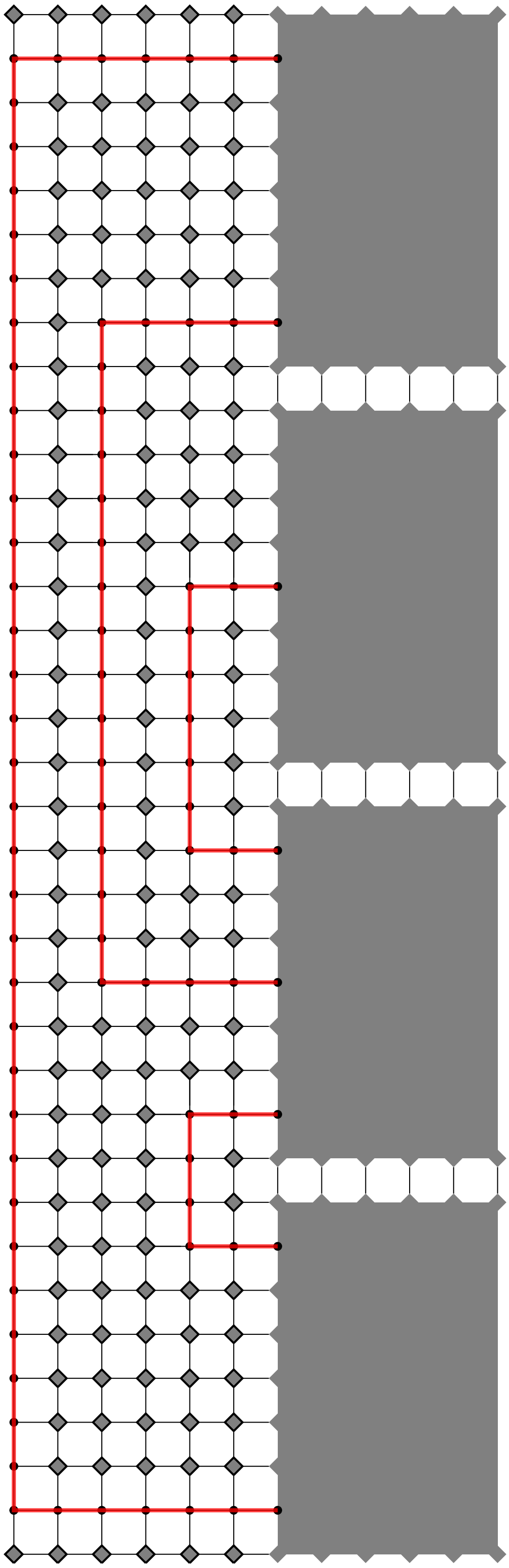


Figure 7.8: The graph G' after **Step 4**: Buffers are added as filler gadgets. Edge paths are highlighted in red in this figure.

times embed the new vertices $w'_1, \dots, w'_{2h(e)}$ with $\text{emb}(w'_t) \coloneqq (9j+\beta, 6+t)$. This creates two vertical portions on the path representing e. Finally, we create a horizontal portion. Subdivide the edge another $\gamma \coloneqq 9\ell(e)+\alpha-\beta-1$ times, creating the vertices $\tilde{w}_1, \dots, \tilde{w}_\gamma$. The positions of the new vertices are $\text{emb}(\tilde{w}_t) \coloneqq (9i - \alpha + t, 5 + 2h(e))$. There is, however, one exception: Note that in the $(4,0)$-vertex gadget there are two buffers with non-integer positions on the upper boundary of the gadget. Vertices on edge paths located above such a buffer must then also be shifted accordingly. More precisely, if the vertex v_i has $\deg_1(v_i) = 4$, then the $(4,0)$-vertex gadget representing v_i, which is embedded centered on $(9i, 0)$, has vertex buffers whose embedding is centered on $(9i - \frac{5}{2}, 5)$ and $(9i + \frac{5}{2}, 5)$ instead of $(9i-2, 5)$ and $(9i+2, 5)$. As a result, any vertex in an edge path embedded at $(9i-2, s)$, $s \geq 6$, must be shifted to $(9i - \frac{5}{2}, s)$ and correspondingly any such vertex embedded at $(9i+2, s)$ must be shifted to $(9i + \frac{5}{2}, s)$. This step is illustrated in Figure 7.7.

Step 4: For every $i = 5, \dots, h_1(G)$ and $j = 5, \dots, 9n$ if G' does not contain a vertex w with $\text{emb}(w) \coloneqq (i,j)$ (and if it would not contain such a vertex but for the aforementioned shift), add a buffer with its embedding centered on (i,j) and increase k' by 4. If this buffer is above a $(4,0)$-vertex gadget, then its position may also need to be shifted. If G' contains a buffer or an half-buffer B centered on $(i, j-1)$, then connect the bottom outlet of the newly added buffer to the top outlet of B. If G' contains a vertex whose embedding is $(i, j-1)$, then connect this vertex to the bottom outlet of the newly added buffer. Similarly, if G' contains a buffer or a half-buffer B centered on $(i-1, j)$, then connect the right outlet in B with the left outlet of the newly added buffer. If G contains a vertex u with $\text{emb}(u) \coloneqq (i-1, j)$, then add an edge from u to the left outlet of the newly added buffer. This step is illustrated in Figure 7.8.

As in previous reductions, we have only discussed the edges in E_1. The same steps must be analogously performed for edges in E_2.

Running time

We must show that the reduction runs in polynomial time. The graph G' contains:

- $\mathcal{O}(n)$ vertex gadgets each containing a constant number of vertices,
- $\mathcal{O}(n)$ edge paths containing $\mathcal{O}(n)$ vertices each, for a total of $\mathcal{O}(n^2)$ vertices, and
- $\mathcal{O}(n^2)$ additional buffers each containing a constant number of vertices.

In total, G' contains $\mathcal{O}(n^2)$ vertices and can be computed in polynomial time.

Correctness

Let F denote the number of buffers that are part of the vertex gadgets in G' or were added in **Step 4** and F' the number of half-buffers. Then, $k' = k+4F+3F'$. If G'' is the graph obtained from G' by removing all buffers and half-buffers, then $\varphi(G') = \varphi(G'')+4F+3F'$ as a result of Lemmas 7.5 to 7.7. The graph G'' is a subdivision of G. Hence, by

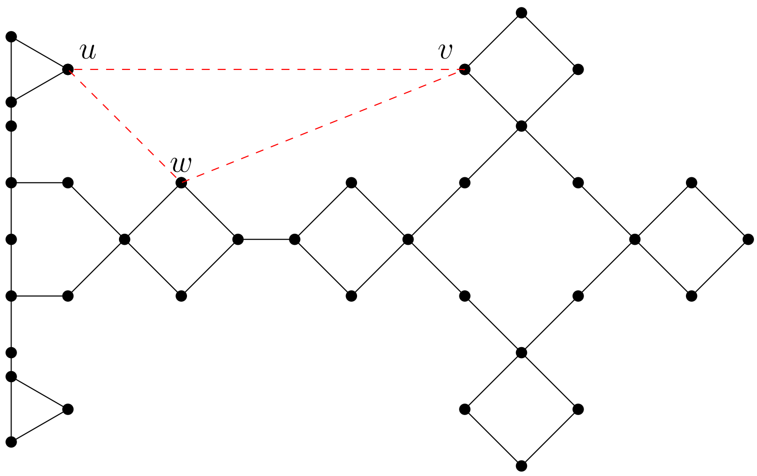


Figure 7.9: An excerpt of the graph G' generated by the reduction, which occurs wherever a buffer and a half-buffer are next to each other: The vertex w is an RNG-blocker for $\{u, v\}$.

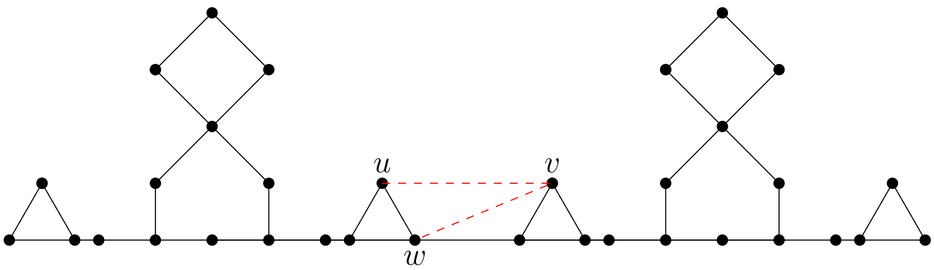


Figure 7.10: An excerpt of the graph G' generated by the reduction, which occurs wherever two half-buffers are beside one another: The vertex w is an RNG-blocker for $\{u, v\}$.

Lemma 7.4, $\varphi(G) = \varphi(G'')$. So, $\varphi(G') = \varphi(G) + 4F + 3F'$. Hence, G contains a feedback vertex set of size k if and only if G' contains one of size $k' = k + 4F + 3F'$.

Graph induced by embedding

Proving that the embedding emb given in the construction induces the graph described therein as its RNG requires a similar analysis to the one given in Chapter 4 for the reduction for Dominating Set. This reduction generates a graph that, outside of the vertex gadgets, is in a way grid-like. We will omit most of the details of this analysis and focus on a few aspects that may not be immediately obvious.

That emb induces G' as its relative neighborhood graph can be verified by checking that there is no blocker for any edge and there is a blocker for every pair of non-adjacent vertices. We will only discuss three cases in which it is not obvious that non-adjacent vertices have a blocker. Recall that $0 < \varepsilon < \frac{1}{12}$.

Consider u, the central vertex in a $(4, 0)$-vertex gadget, and v, the central vertex in the half-buffer above it. Their positions are $\text{emb}(u) = (i, j)$ and $\text{emb}(v) = (i, j+3)$. Hence, $d(\text{emb}(u), \text{emb}(v)) = 3$. Their blocker is the vertex w with $\text{emb}(i-2, j+2)$, since $d(\text{emb}(u), \text{emb}(w)) = 2\sqrt{2} < 3$ and $d(\text{emb}(v), \text{emb}(w)) = \sqrt{5} < 3$. Now consider the subgraph depicted in Figure 7.9 and the vertices marked u, v, and w. This subgraph occurs wherever a buffer connects to the top outlet of a half-buffer. Then,

$$\begin{aligned}
d(\text{emb}(u), \text{emb}(v))^2 &= (1-2\varepsilon)^2 = 1 - 4\varepsilon + 4\varepsilon^2 > 1 - 4\varepsilon > \frac{1}{2},\\
d(\text{emb}(u), \text{emb}(w))^2 &= 8\varepsilon^2 < \varepsilon < \frac{1}{2} < d(\text{emb}(u), \text{emb}(v))^2, \text{ and}\\
d(\text{emb}(v), \text{emb}(w))^2 &= 4\varepsilon^2 + (1-4\varepsilon)^2\\
&= 1 - 8\varepsilon + 20\varepsilon^2\\
&= 1 - 4\varepsilon + 4\varepsilon^2 + \varepsilon(16\varepsilon - 4) < 1 - 4\varepsilon + 4\varepsilon^2 = d(\text{emb}(u), \text{emb}(v))^2.
\end{aligned}$$

So, w is a blocker for u and v. Next, consider the subgraph pictured in Figure 7.10 and the vertices marked u, v, and w. The subgraph occurs wherever two half-buffers are next to one another. We compare the distances again, noting that $\tan(60°) = \sqrt{3}$:

$$\begin{aligned}
d(\text{emb}(u), \text{emb}(v))^2 &= (1-6\varepsilon)^2 = 1 - 12\varepsilon + 36\varepsilon^2 > 36\varepsilon^2,\\
d(\text{emb}(u), \text{emb}(w))^2 &= \varepsilon^2 + \frac{\varepsilon^2}{\tan(60°)^2} = \frac{4\varepsilon^2}{3} < 36\varepsilon^2 < d(\text{emb}(u), \text{emb}(v))^2, \text{ and}\\
d(\text{emb}(v), \text{emb}(w))^2 &= (1 - 6\varepsilon - \frac{\varepsilon}{\tan(60°)})^2 + \varepsilon^2\\
&= 1 - (12 + \frac{2}{\sqrt{3}})\varepsilon + (36 + \frac{2}{\sqrt{3}} + \frac{1}{3})\varepsilon^2 + \varepsilon^2\\
&= 1 - 12\varepsilon + 36\varepsilon^2 + \varepsilon(\frac{4+2\sqrt{3}}{3}\varepsilon - \frac{2\sqrt{3}}{3})\\
&< 1 - 12\varepsilon + 36\varepsilon^2 = d(\text{emb}(u), \text{emb}(v))^2.
\end{aligned}$$

The vertex w is again a blocker for u and v.

Gabriel graphs and relatively closest graphs

The reduction above produces relative neighborhood graphs, but not Gabriel graphs or relatively closest graphs. However, it can easily be adjusted for the latter graph classes.

We start with relatively closest graphs. Only the half-buffer must be adjusted. The embedding that induces the half-buffer as its relative neighborhood graph induces the graph pictured in Figure 7.11 as its relatively closest graph, since the two triangles on each side are equilateral. Lemma 7.6 still holds for this modified half-buffer with the following adjustment: It increases the feedback vertex number by 1 rather than 3. We may, therefore, employ the modified half-buffer in the reduction for relatively closest graphs, but the calculation of k' must be adjusted accordingly.

For Gabriel graphs, the $(4, 0)$ and $(2, 2)$-vertex gadgets must be adjusted. In the case of $(4, 0)$-vertex gadgets, consider the vertices u, v, w discussed in the previous section.

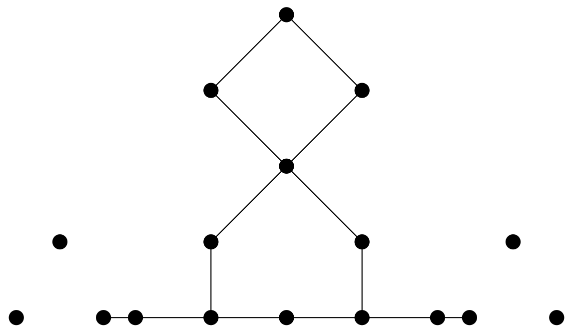
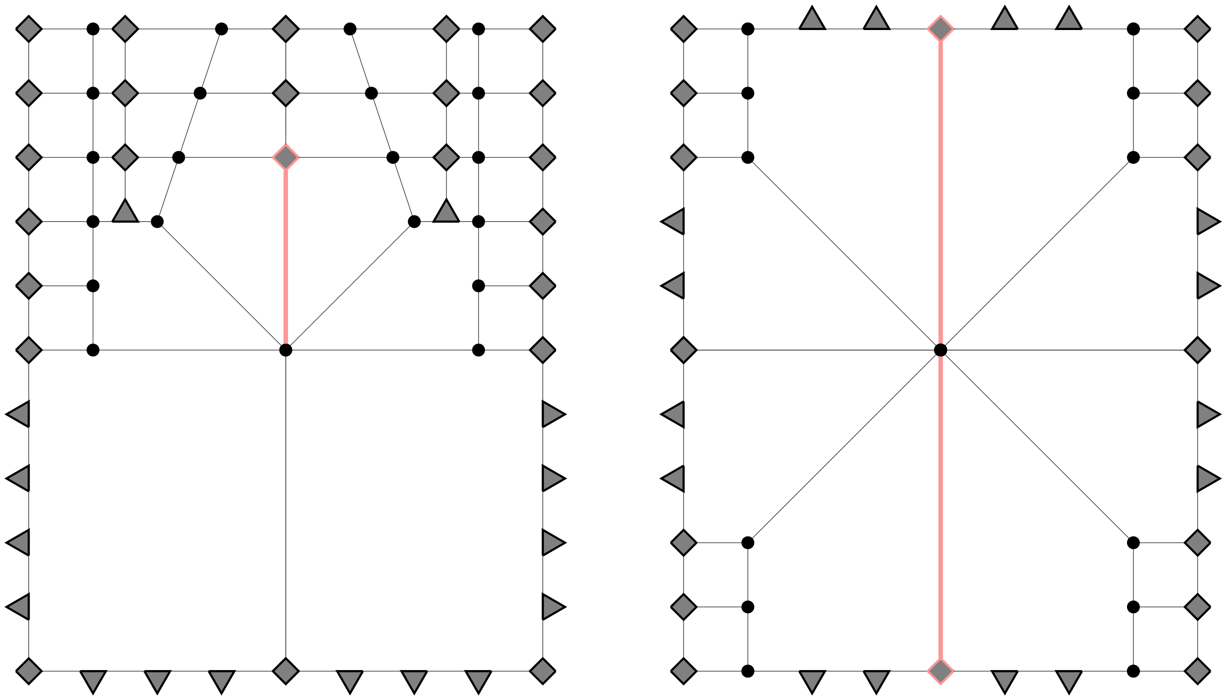


Figure 7.11: The half-buffer graph for relatively closest graphs.



(a) The $(4,0)$-vertex gadget for Gabriel graphs.

(b) The $(2,2)$-vertex gadget for Gabriel graphs

Figure 7.12: Modified vertex gadgets for Gabriel graph: The modified parts of the graph are highlighted in red.

We showed that w is an RNG-blocker for u and v. It is not, however, a Gabriel-blocker, since: $\measuredangle\, \mathrm{emb}(u)\, \mathrm{emb}(w)\, \mathrm{emb}(v) = 45° + \arctan(\frac{1}{2}) < 90°$. Hence, we modify the $(4,0)$-vertex gadget and use the graph pictured in Figure 7.12a. There is a similar issue with the $(2,2)$-vertex gadget and we replace it with the graph in Figure 7.12b. Lemma 7.7 holds for these modified vertex gadgets. Showing that pairs of non-adjacent vertices have Gabriel-blockers is slightly more involved than with RNG-blockers in the two other cases we discussed before, but the same two vertices are also Gabriel-blockers. Hence, the rest of the reduction can be adapted.

7.3 Conclusions

This concludes the reduction. Note that in the graphs generated by the reduction for RNGs and RCGs no vertex has a degree greater than six, while the reduction for Gabriel graphs does not produce any vertices with degree greater than eight. In all, we have proved the following:

Theorem 7.8. Feedback Vertex Set *is NP-hard when restricted to relative neighborhood graphs with maximum degree six, relatively closest graphs with maximum degree six, or Gabriel graphs with maximum degree eight.*

Ueno, Kajitani, and Gotoh [UKG88] showed that Feedback Vertex Set can be solved in polynomial time on planar graphs with maximum degree three. Whether the problem can be solved in polynomial time on RNGs or RCGs with maximum degree four or five or on Gabriel graphs with maximum degree between four and seven remains open.

We will show that our reduction also implies that Feedback Vertex Set cannot be solved in time $2^{o(n^{\frac{1}{4}})}$ on each of our proximity graph classes (where n is the number of vertices in a graph), unless the exponential time hypothesis fails (on the ETH, see Section 2.5).

We must first establish an ETH-based lower bound of $2^{o(\sqrt{n})}$ for Feedback Vertex Set on planar graphs with maximum degree 4. Speckenmeyer [Spe83] proves this problem is NP-hard with a series of reductions starting from Vertex Cover on planar graphs with maximum degree 3. Each of these reductions changes the number of vertices and edges only linearly. This along with Theorem 5.9 proves:

Theorem 7.9. *Unless the ETH fails,* Feedback Vertex Set *restricted to planar graphs with maximum degree* 4 *cannot be solved in time* $2^{o(\sqrt{n})}$.

Our reduction from Feedback Vertex Set on planar graphs with maximum degree 4 to its restriction to our classes of proximity graphs maps instances with m edges and n vertices to graphs with $\mathcal{O}(n^2)$ vertices and edges. This along with Theorem 7.9 implies:

Corollary 7.10. *Unless the ETH fails,* Feedback Vertex Set *restricted to RNGs, RCGs, or Gabriel graphs cannot be solved in time* $2^{o(n^{\frac{1}{4}})}$.

8 Hamiltonian Cycle

We will now consider the HAMILTONIAN CYCLE problem. Recall that, as we defined in Section 2.1, a Hamiltonian cycle in a graph $G = (V, E)$ with $|V| = n$ is a set of edges $E' \subseteq E$ such that (V, E') is isomorphic to the cycle graph C_n. Alternatively, we will also denote a Hamiltonian cycle as a sequence of vertices $v_i, \ldots, v_n$ such that $v_i \neq v_j$ if $i \neq j$, v_i is adjacent to v_{i+1} for all $i = 1, \ldots, n-1$, and v_n is adjacent to v_1. This gives rise to the HAMILTONIAN CYCLE problem:

HAMILTONIAN CYCLE
Input: A graph $G = (V, E)$.
Question: Does G contain a Hamiltonian cycle?

Garey, Johnson, and Tarjan [GJT76] prove that HAMILTONIAN CYCLE is NP-hard when restricted to 3-regular, 3-connected planar graphs:

Theorem 8.1 ([GJT76]). HAMILTONIAN CYCLE *on 3-regular planar graphs is NP-hard.*

8.1 Definitions and intermediate results

Before we describe our polynomial-time many-to-one reduction, we will again define some terminology and prove several lemmas that will simplify the description of the reduction and the proof of its correctness.

Definition 8.2. An edge in a graph G is *permissible* if there is a Hamiltonian cycle in G that passes through this edge.

Subdividing an edge only preserves Hamiltonicity if the edge is permissible:

Lemma 8.3. *The graph $G = (V, E)$ is Hamiltonian and the edge $e \in E$ is permissible if and only if the graph obtained from G by subdividing e is also Hamiltonian.*

Proof. Suppose that G' is the graph obtained by subdividing $e = \{v_1, v_2\}$ and that w is the vertex introduced in the subdivision. If $\{v_1, v_2\}$ is permissible, then G contains a Hamiltonian cycle $v_1, v_2, v_3, \ldots, v_n$ and as a result $v_1, w, v_2, v_3, \ldots v_n$ is a Hamiltonian cycle in G'. Now suppose that G' is Hamiltonian. Any Hamiltonian cycle in G' must visit w, so the cycle must contain v_1, w, v_2 or v_2, w, v_1. Replacing this segment of the cycle by v_1, v_2 or v_2, v_1 respectively yields a Hamiltonian cycle in G. □

The next two lemmas show that how we can easily determine that certain edges are not permissible.

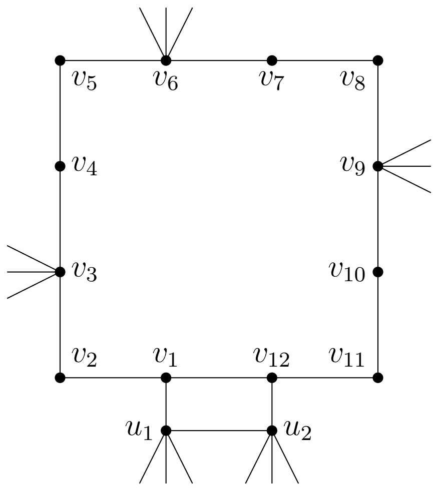


Figure 8.1: Illustration of Lemma 8.6 with a cycle C of length 12: The vertices v_1 and v_{12} each have one neighbor outside of the cycle and they are adjacent to one another. No two other consecutive vertices in the cycle have neighbors outside of the cycle.

Lemma 8.4. *Suppose $v_1, \ldots, v_n$ is a Hamiltonian cycle in G. Then, for any $1 \leq i \leq j \leq n$, the graph $G - \{v_i, \ldots, v_j\}$ cannot contain a vertex with a degree of zero or one, except possibly v_{i-1} and v_{j+1}.*

Proof. Any v_k, $k \notin \{i-1, \ldots, j+1\}$, is adjacent to both $v_{((k-1) \bmod n)+1}$ and $v_{((k+1) \bmod n)+1}$, both of which are part of $G - \{v_i, \ldots, v_j\}$. □

Lemma 8.5. *Suppose that G contains a vertex u which is adjacent to both u_1 and u_2 with* $\deg(u_1) = \deg(u_2) = 2$*. Then, no edge $\{u, v\}$ with $v \notin \{u_1, u_2\}$ is permissible.*

Proof. Suppose that $\{u, v\}$ with $v \notin \{u_1, u_2\}$ is part of a Hamiltonian cycle. Then, u_1 and u_2 both have a degree of one in $G - \{u, v\}$. By Lemma 8.4, this implies that u_1 that the Hamiltonian cycle must contain u_1, u, v, u_2 or u_2, u, v, u_1. Without loss of generality, it is the former. The vertex u_1 cannot have any additional neighbors besides u and v, so it cannot have a predecessor. □

One can add a cycle with certain properties to a graph without affecting its Hamiltonicity, but only if a certain edge is known to be permissible if the graph is Hamiltonian, as the next lemma states. The lemma is illustrated in Figure 8.1.

Lemma 8.6. *Suppose that G contains a cycle C consisting of the vertices $v_1, \ldots, v_k$ in that order with the following properties:*

- *$k \geq 4$,*
- *v_1 and v_k have neighbors u_1 and u_2, respectively, which are not part of C,*
- *u_1 and u_2 are adjacent*

- *v_1 and v_k have no neighbors outside of C besides u_1 and u_2,*
- $\deg(v_2) = \deg(v_{k-1}) = 2$, *and*
- *for all $i = 3, 4, \ldots, k-2$, if v_i has a neighbor outside of C, then* $\deg(v_{i-1}) = \deg(v_{i+1}) = 2$.

Then, G is Hamiltonian if and only if the edge between u_1 and u_2 is permissible in $G-C$. Moreover, if G is Hamiltonian, then all of the edges in C are permissible.

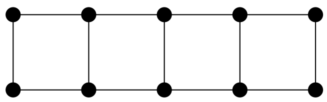
Proof. Suppose that G is Hamiltonian. By Lemma 8.5, none of the edges between C and $G-C$ except for $\{u_1, v_1\}$ and $\{u_2, v_k\}$ are permissible. This implies that any Hamiltonian cycle in G, which we may assume to start in u_1 without loss of generality, enters C through v_1 or v_k, visits every other vertex in C, and then exits C through v_1 or v_k before visiting every other vertex in G in the order $w_1, w_2, w_3, \ldots$ where $w_1 = u_2$. Then, $u_1, u_2, w_2, w_3, \ldots$ is a Hamiltonian cycle in $G-C$ which uses $\{u_1, u_2\}$. Hence, this edge is permissible.

Now suppose that $\{u_1, u_2\}$ is permissible in $G-C$. Then, there is some Hamiltonian cycle in $G-C$ which uses $\{u_1, u_2\}$. Replacing this edge with the vertices in C yields a Hamiltonian cycle in G. □

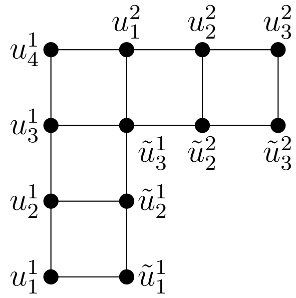
We will make use of what we call *ladder paths* to represent edges in the reduction. For any $k \geq 1$, we will refer to the grid graph $G_{2,k}$ as the *simple k-ladder path* and denote it by L_k. As an example, the graph L_5 is pictured in Figure 8.2a. We will refer to the two pairs of adjacent vertices of degree two as the two *ends* of the simple ladder path. There are also *ladder paths with bends* L_{k_1,k_2}. They consist of the vertices $u^1_1, \ldots, u^1_{k_1}$, $\tilde{u}^1_1, \ldots, \tilde{u}^1_{k_1-1}$, $u^2_1, \ldots, u^2_{k_2+1}$, and $\tilde{u}^2_2, \ldots, \tilde{u}^2_{k_2+1}$. The indices in the designations of the vertices may appear counter-intuitive, but they will simplify our proofs. In lieu of a formal list of the edges in the ladder path with bends, we refer the reader to Figure 8.2b, which pictures $L_{4,4}$. The two pairs of adjacent vertices of degree two are again called the *ends*.

We will also make use of complex ladder paths. The *complex ladder path* L_{k_1,k_2,k_3} consists of the disjoint union of two simple ladder paths with bends, L_{k_1,k_2} and L_{k_1,k_3}, and an additional vertex which is adjacent to the vertices $u^2_{k_2+1}$ and $\tilde{u}^2_{k_2+1}$ at one end of the copy of L_{k_1,k_2} and to the vertices $u^2_{k_3+1}$ and $\tilde{u}^2_{k_3+1}$ at one end of the copy of L_{k_1,k_3}. As an example $L_{6,2,3}$ is pictured in Figure 8.2c. We will refer to the vertices in the graph by the designations marked in the figure. The vertices in the copy of L_{k_1,k_2} will be referred to as the *first half* and the vertices in the copy of L_{k_1,k_3} as the *second half* of the complex ladder path. The vertex w that connects them is the *transitional vertex*. As with simple ladder paths, we will refer to the two pairs of vertices of degree two as the *ends* of the complex ladder path. That is, u^1_1 and $\tilde{u}^1_1$ form one end and u^4_1 and $\tilde{u}^4_1$ are the other. The edges $\{u^i_j, u^{i'}_{j'}\}$ will be called *outside edges*, while the edges $\{\tilde{u}^i_j, \tilde{u}^{i'}_{j'}\}$ are *inside edges*. An edge $\{u^i_j, u^i_{j+1}\}$ or $\{\tilde{u}^i_j, \tilde{u}^i_{j+1}\}$ is called *even* if j is even.

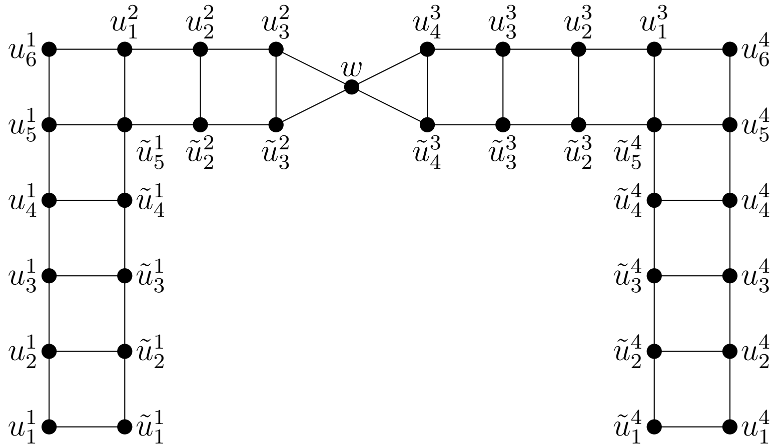
We will now investigate ladder paths in Hamiltonian graphs. A *traversal* of a complex ladder path is a path that begins in either vertex at one end of the ladder path, terminates



(a) The simple ladder path L_5 is isomorphic to the grid graph $G_{2,5}$.



(b) The simple ladder path with a bend $L_{4,2}$.

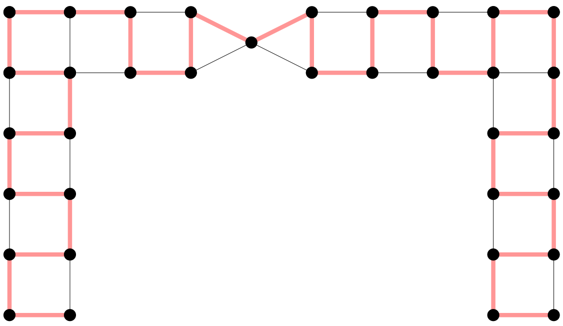


(c) The complex ladder path $L_{6,2,3}$.

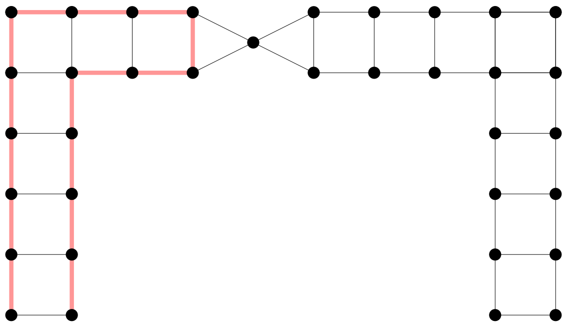
Figure 8.2: Ladder paths will be used as gadgets to represent edges in the reduction.

in either vertex at the other end of the ladder path, and visits every vertex on the ladder path and no other vertex. It is easy to see that complex ladder paths have exactly four traversals, as they may begin in either vertex at one end and may terminate in either vertex at the other end. An example of a traversal is pictured in Figure 8.3a. A *partial cover* of one half of a complex ladder path is a path that begins in one of the two vertices at the end of the half, terminates in the other vertex at that end, and visits every vertex of that half, but no other vertex. A *full cover* of a half additionally visits the transitional vertex. Examples of a partial cover and a full cover are pictured in Figures 8.3b and 8.3c, respectively. By a simple case analysis, one can make the following observation:

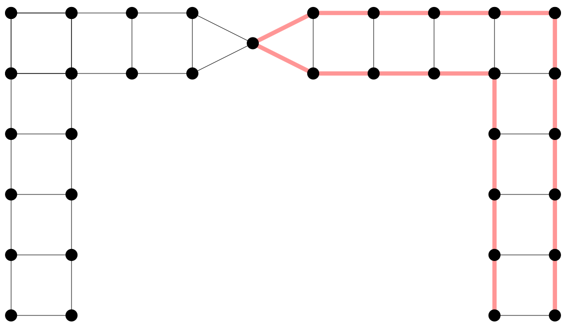
Observation 8.7. *Suppose that k_1 and k_2 are even. If a traversal of L_{k_1,k_2,k_3} begins in u_1^1, then it contains every even outside edge before the bend and every even inside edge after the bend in the first half. If such a traversal begins in $\tilde{u}_1^1$, then it contains every even inside edge before the bend and every even outside edge after the bend in the first half. In either case, it contains both an even inside and an even outside edge in the first half. By the same reasoning, it also contains an even inside and even outside edge in the second half. Furthermore, covers contain every inside and every outside edge.*



(a) A traversal of $L_{6,2,3}$: A traversal of a complex ladder path is a path that begins in one end, visits every vertex in the ladder path, and terminates in the end of the other half.



(b) A partial cover of the first half of $L_{6,2,3}$: A partial cover of a half is a path that begins and terminates in the end of that half and visits every vertex in the half, but no other vertex.



(c) A full cover of the second half of $L_{6,2,3}$: A full cover of a half is a path that begins and terminates in the end of that half and visits every vertex in the half as well as the transitional vertex, but no other vertex.

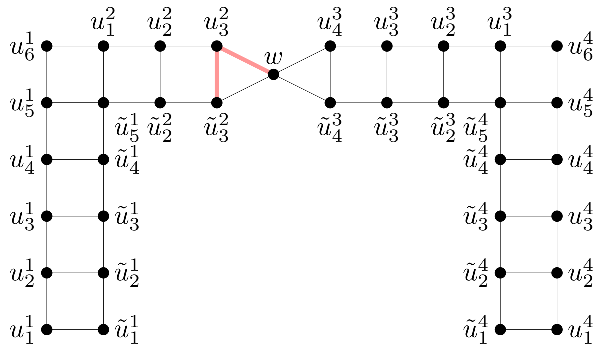
Figure 8.3: Traversals and covers: The thick, red edges are part of the respective path.

The following lemma clarifies how Hamiltonian cycles relate to complex ladder paths:

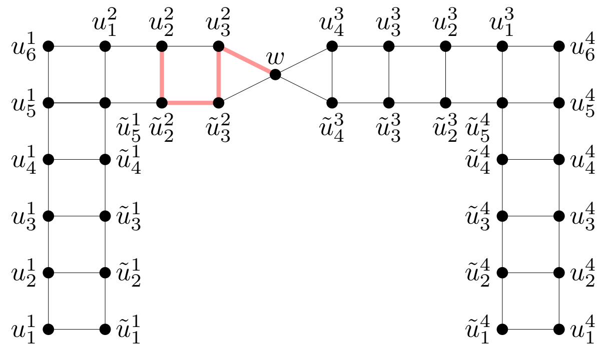
Lemma 8.8. *Suppose that the Hamiltonian graph $G = (V, E)$ contains the complex ladder path L_{k_1,k_2,k_3}, that the only vertices on the ladder path with neighbors outside of the ladder path are on its ends, and that the vertices on the ends each have no more than one neighbor outside of the ladder path. Then, any Hamiltonian cycle in G contains either:*

- *a traversal of the complex ladder path or*
- *a partial cover of one of its halves and a full cover of the other half.*

Proof. Consider any Hamiltonian cycle in G. In this Hamiltonian cycle, the transitional vertex w is succeeded by $u^2_{k_1-1}$, $\tilde{u}^2_{k_1-1}$, $u^3_{k_1-1}$, or $\tilde{u}^3_{k_1-1}$. By symmetry, we may assume without loss of generality that the successor is $u^2_{k_2-1}$ or $\tilde{u}^2_{k_2-1}$. We will only deal with the first of these two cases as the argument in the other case is very similar. The successor of $u^2_{k_2-1}$ is either $\tilde{u}^2_{k_2-1}$ or $u^2_{k_2-2}$.

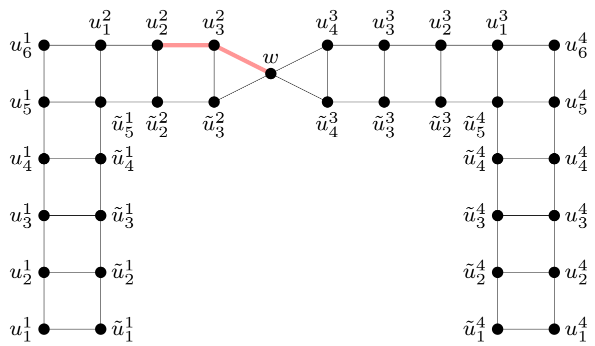


(a) The successor of u_3^2 may be $\tilde{u}_3^2$ or u_2^2. Here, we assume the former.

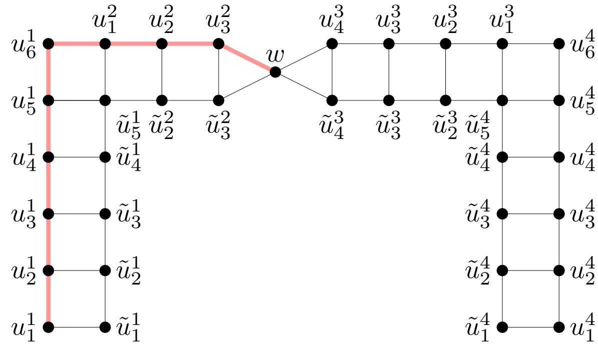


(b) The Hamiltonian cycle must continue to $\tilde{u}^2_{k_2-2}$ and then $u^2_{k_2-2}$.

Figure 8.4: The first case: The successor of $u^2_{k_2-1}$ is $\tilde{u}^2_{k_2-1}$. Then, the Hamiltonian cycle contains a traversal of the ladder path.



(a) The successor of u_3^2 may be $\tilde{u}_3^2$ or u_2^2. Here, we assume the latter.



(b) The path must continue until it reaches u_1^1.

Figure 8.5: The second case: The successor of $u^2_{k_2-1}$ is $\tilde{u}^2_{k_2-1}$. Then, the Hamiltonian cycle contains a cover of each half of the ladder path.

First, assume that this successor is $\tilde{u}^2_{k_2-1}$, as in Figure 8.4a. Since its only other neighbor is w, $\tilde{u}^2_{k_2-1}$ must be succeeded by $\tilde{u}^2_{k_2-2}$. By Lemma 8.4, $\tilde{u}^2_{k_2-2}$'s successor must then be $u^2_{k_2-2}$ (see Figure 8.4b). By iterating this argument, we can show that the Hamiltonian cycle visits every vertex u_i^2 and $\tilde{u}_i^2$ until it reaches$\tilde{u}^2_{k_1-1}$ or $u^1_{k_1}$ from where it must proceed to $u^1_{k_1}$. Applying the same argument as before, the Hamiltonian cycle must then visit every vertex u_i^1 and $\tilde{u}_i^1$ and finally reach u_1^1 or $\tilde{u}_1^1$, forming one half of a traversal of the ladder path. The transitional vertex w is preceded by either $u^3_{k_1-1}$ or $\tilde{u}^3_{k_1-1}$. By the same argument employed above, the Hamiltonian cycle reaches w by a path from u_1^4 or $\tilde{u}_1^4$ that visits every vertex in this half along the way. In all, this is a traversal of the ladder path.

Now, we assume that $u^2_{k_2-1}$'s successor is $u^2_{k_2-2}$, as in Figure 8.5a. This vertex's successor must then be $u^2_{k_2-3}$ because otherwise $\tilde{u}^2_{k_2-1}$ would be unreachable. This argument can be applied repeatedly until the Hamiltonian cycle reaches u_1^1, as in Figure 8.5b. Also,

(a) The $(3,0)$-vertex gadget with the outlet vertices enlarged and marked in red.

(b) The $(2,1)$-vertex gadget with the outlet vertices enlarged and marked in red.

Figure 8.6: There are four types of vertex gadgets, each with three pairs of distinguished adjacent vertices we will call the outlets. The four vertex gadgets are structurally identical, but have different outlets.

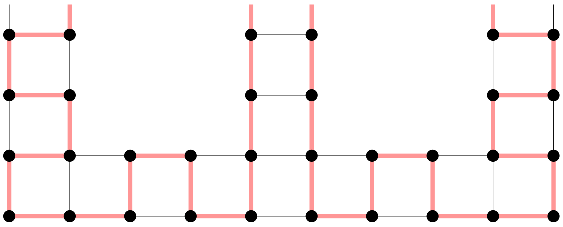
by Lemma 8.4, w's predecessor must then be $\tilde{u}^2_{k_2-1}$, which is then preceded by $\tilde{u}_{k_2-2}$, and so until $\tilde{u}^1_1$ is reached. Together, these two paths then form a full cover of this half of the ladder path. By a similar argument, the Hamiltonian cycle must also contain a partial cover of the second half of the ladder path. □

Complex ladder paths will be used to replace edges in our reduction. We will now turn our attention to the gadgets we will use to replace vertices. In light of Theorem 8.1, we will only consider 3-regular graphs. We will use copies of the simple ladder graph L_{10} as vertex gadgets. They will have particular vertex pairs designated as outlets to which the complex ladder paths representing the edges will be connected. We will use two different vertex gadgets (which one is used for a particular vertex will depend on the book embedding). Two vertex gadgets, the $(3,0)$-vertex gadget and the $(2,1)$-vertex gadget, are pictured in Figure 8.6 with the outlet vertices highlighted in red. We may obtain $(0,3)$-vertex gadgets and $(1,2)$-vertex gadgets from the pictured gadgets by flipping them. We will refer to an outlet as a *top* or *bottom outlet* depending on where the vertices in the outlet are depicted in the figure. So, a $(3,0)$-vertex gadget has three top outlets and no bottom outlets, a $(2,1)$-vertex gadget has two top outlets and one bottom outlet, and so on. We will order the outlets from left to right. So, we may refer to an outlet as the first top outlet or the third bottom outlet with the obvious meaning. The two outlets that contain a vertex of degree two are *outer outlets* and the other outlet is a *middle outlet*.

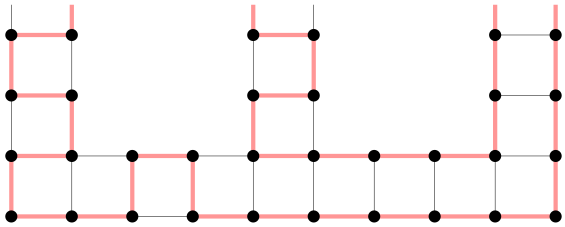
Our use of complex ladder paths and the given vertex gadgets is justified by the following lemma:

Lemma 8.9. *Suppose that G is a 3-regular graph and that G' is obtained from G by the following:*

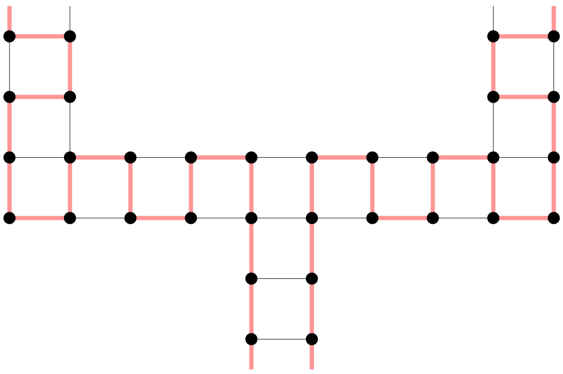
- *Replace every vertex in G with a $(3,0)$, $(2,1)$, $(1,2)$, or $(0,3)$-vertex gadget.*
- *Replace every edge in G with an arbitrary complex ladder path.*
- *Connect one end of the complex ladder path that represents the edge $e = \{u, v\}$ to an outlet of the vertex gadget representing u by adding an edge from each vertex in the end of the ladder path to a different vertex in the outlet. Connect the other end to an outlet of the vertex gadget representing v in the same way. No two distinct ladder paths may be connected to the same outlet.*



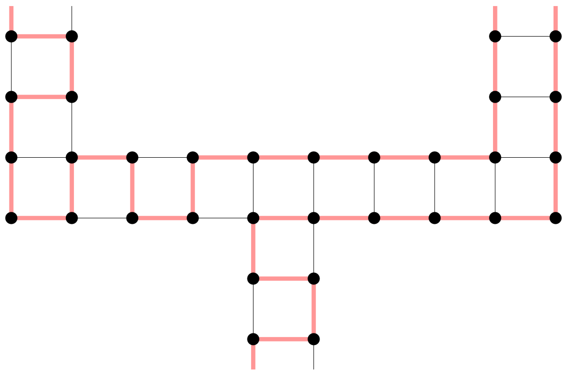
(a) The first type of vertex gadget: a $(3,0)$-vertex gadget representing a vertex that is entered and exited through edges whose corresponding ladder paths are connected to the outer outlets.



(b) The second type of vertex gadget: a $(3,0)$-vertex gadget representing a vertex that is entered and exited through edges whose corresponding ladder paths are connected to an outer outlet and the central outlet.



(c) The third type of vertex gadget: a $(2,1)$-vertex gadget representing a vertex that is entered and exited through edges whose corresponding ladder paths are connected to the outer outlets.



(d) The second type of vertex gadget: a $(2,1)$-vertex gadget representing a vertex that is entered and exited through edges whose corresponding ladder paths are connected to an outer outlet and the central outlet.

Figure 8.7: Construction of the Hamiltonian cycle in G': There are four types of vertex gadgets, depending on which ladder paths are traversed and which are covered. The edges added to the Hamiltonian cycle in G' are thick and highlighted in red.

Then, G is Hamiltonian if and only if G' is.

Proof. First assume that G' is Hamiltonian and consider any Hamiltonian cycle in G'. By Lemma 8.8, the Hamiltonian cycle contains either a traversal or a partial cover and a full cover of every complex ladder path in G'. We will say that a complex ladder path is *traversed* if the Hamiltonian cycle contains a traversal of the path. We may construct a Hamiltonian cycle in G by including every edge in G whose corresponding ladder path is traversed.

Now assume that G is Hamiltonian and consider any Hamiltonian cycle in G. The cycle contains exactly two of the three edges incident to any vertex. We will construct a Hamiltonian cycle in G'. This Hamiltonian cycle contains a traversal of every ladder path representing an edge in G's Hamiltonian cycle as well as a partial and a full cover of every other ladder path. Which half of the ladder path is covered fully and which partially is irrelevant. We will construct the Hamiltonian cycle in G' by considering each vertex gadget in G' individually. We will distinguish four types of vertex gadgets depending on which of our two vertex gadgets it is and through which of the corresponding edges the vertex in G is entered and left.

First, consider a $(3,0)$-vertex gadget representing a vertex that is entered and left through edges whose corresponding edges are connected to the two outer outlets of the vertex gadget. Then the vertex gadget is entered through a traversal of the ladder path connected to one of the outer outlets. The Hamiltonian cycle then visits some of the vertices in the vertex gadget followed by a full or partial cover of the ladder path connected to the middle outlet. The Hamitlonian cycle then visits the rest of the vertices in the vertex gadget and exits the vertex gadget by a traversal of the last ladder path connected to it. In all, we add the edges marked in red in Figure 8.7a.

The second type is a $(3,0)$-vertex gadget representing a vertex that is entered and left through edges whose corresponding edges are connected to an outer outlet and the central outlet of the vertex gadget. The edges used in this case are marked in Figure 8.7b.

Thirdly, there is a $(2,1)$-vertex gadget representing a vertex that is entered and left through edges whose corresponding edges are connected to the two outer outlets of the vertex gadget. We add the edges marked in red in Figure 8.7c.

The fourth and final case involves a $(2,1)$-vertex gadget representing a vertex, which is entered and left through edges whose corresponding edges are connected to an outer outlet and the lower outlet of the vertex gadget. We add the edges marked in red in Figure 8.7d.

Other cases may be dealt with by symmetry using the given cases.

By adding these edges in the vertex gadgets as well as the aforementioned traversals and covers in the complex ladder paths representing edges, we construct a Hamiltonian cycle in G', proving that G' is Hamiltonian. □

8.2 Reduction

We will now describe our reduction from Hamiltonian Cycle on 3-regular planar graphs to the restriction of this problem to relative neighborhood graphs and Gabriel graphs. Let $G = (V, E)$ be a 3-planar graph. As in our previous reductions, we compute a two-page book embedding for G in polynomial time and assume that $v_1, \ldots, v_n$ are the vertices of G in the order in which they appear on the spine of the book embedding. The vertices incident to a certain edge will be ordered in the same way as in the reduction for Feedback Vertex Set in Chapter 7.

Construction

We will construct a relative neighborhood graph $G' = (V', E')$ such that G has a Hamiltonian cycle if and only if G' does. At the same time, we give an embedding emb,

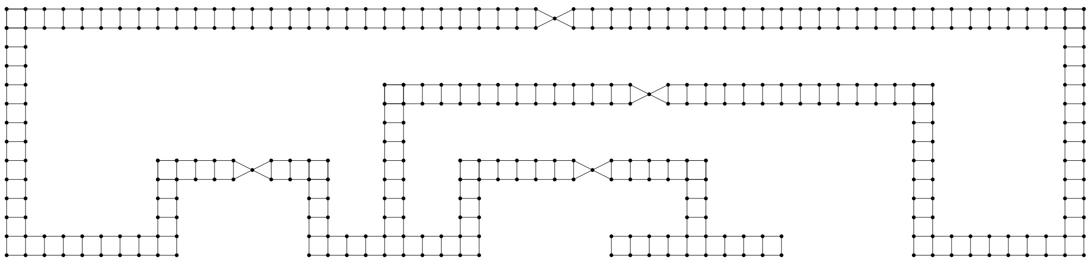


Figure 8.8: The graph G' after **Step 3**: Each vertex is replaced by a vertex gadget. Each edge is replaced by a complex ladder path.

which we will subsequently use to prove that G' is, in fact, an RNG by showing that $\mathrm{RNG}(\mathrm{emb}(V')) = G'$. The construction will follow the general pattern we laid out in Chapter 3 and use the terminology we defined there. The vertex gadgets will be the $(3,0)$ and $(2,1)$-vertex gadgets described above. The edge gadgets will be complex ladder paths. We will use cycles that fulfill the conditions in Lemma 8.6 as filler gadgets. As with the previous problems, the reduction will be illustrated using the graph pictured in Figure 3.1 (this graph is not 3-regular, but in order to have a sufficiently simple example, we will disregard this or suppose that additional edges in E_2 make the graph 3-regular). In the description we will only take into account edges in the top page E_1. All steps but the first two must be repeated for edges in E_2.

Step 1: Start with $G' \coloneqq G$.

Step 2: Replace every vertex v_i with the $(\deg_1(v_i), \deg_2(v_i))$-vertex gadget. Suppose that $u_0, \dots u_9, u'_0, \dots, u'_9$ are the vertices in the gadget. They are embedded as $\mathrm{emb}(u_r) \coloneqq (16i + r, 0)$ and $\mathrm{emb}(u'_r) \coloneqq (16i + r, 1)$ for $r = 0, \dots, 9$.

Step 3: Replace every edge $e = \{v_i, v_j\}$ in G, $i < j$, with a complex ladder path L_{k_1,k_2,k_3} connected to an outlet in v_i's vertex gadget and an outlet in v_j's vertex gadget. Recall that we defined the ordering of the edges incident to a certain vertex to be the same as in the reduction for Feedback Vertex Set in Chapter 7. The ladder path is connected to the outlets as in Lemma 8.9. The lengths k_1, k_2, k_3 are computed in the following manner: First, $k_1 \coloneqq 4h(e)$. The values k_2 and k_3 represent the horizontal length of the edge in the embedding. We need values α and β which represent the distances over v_i's and v_j's vertex gadget, respectively. These values depend on the outlet in the vertex gadget to which the ladder path is connected. The first value α is defined by:

$$\alpha \coloneqq \begin{cases} 4, & \text{if } \deg_1(v_i) = 1 \text{ or } \deg_1(v_i) = 3 \text{ and } e \text{ is } v_i\text{'s second edge in } E_1, \\ 8, & \text{if } \deg_1(v_i) \geq 2 \text{ and } e \text{ is } v_i\text{'s first edge in } E_1, \\ 0, & \text{if } \deg_1(v_i) \geq 2 \text{ and } e \text{ is } v_i\text{'s last edge in } E_1. \end{cases}$$

Similarly, β is:

$$\beta \coloneqq \begin{cases} 4, & \text{if } \deg_1(v_j) = 1 \text{ or } \deg_1(v_j) = 3 \text{ and } e \text{ is } v_j\text{'s second edge in } E_1, \\ 0, & \text{if } \deg_1(v_j) \geq 2 \text{ and } e \text{ is } v_j\text{'s first edge in } E_1, \\ 8, & \text{if } \deg_1(v_j) \geq 2 \text{ and } e \text{ is } v_j\text{'s last edge in } E_1. \end{cases}$$

Note that number of horizontal positions taken up by L_{k_1,k_2,k_3} is $k_2 + k_3 + 5$, since four positions are taken up by the vertical portions of the ladder path and one position is taken up by the transitional vertex. The actual number of positions between the outlets to be connected is $\alpha+\beta+16(j-i-1)+10$. Hence, we need $k_2+k_3 = \alpha+\beta+16(j-i-1)+5 =: \gamma$. So, we let $k_2 := \lceil \frac{\gamma}{2} \rceil$ and $k_3 := \lfloor \frac{\gamma}{2} \rfloor$. Finally, we will give the positions of the added vertices in the embedding. We will use the designations for the vertices introduced in Figure 8.2c. The vertices' embeddings are:

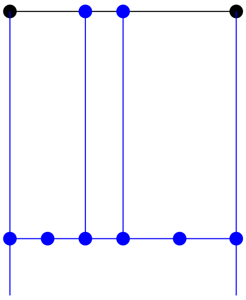
- $\mathrm{emb}(u_r^1) := (16i + 8 - \alpha, r + 1)$ for $r = 1, \ldots, k_1$,
- $\mathrm{emb}(\tilde{u}_r^1) := (16i + 9 - \alpha, r + 1)$ for $r = 1, \ldots, k_1 - 1$,
- $\mathrm{emb}(u_r^2) := (16i + 8 - \alpha + r, k_1 + 1)$ for $r = 1, \ldots, k_2 - 1$,
- $\mathrm{emb}(\tilde{u}_r^2) := (16i + 8 - \alpha + r, k_1)$ for $r = 2, \ldots, k_2 - 1$, and
- $\mathrm{emb}(w) := (16i + 8 - \alpha + k_2 + 1, k_1 + \frac{1}{2})$.

These values correspond to the position depicted in Figure 8.2c. The positions of the vertices in the second half of the ladder path are analogous to the first half, so we will omit them. This step is illustrated in Figure 8.8.

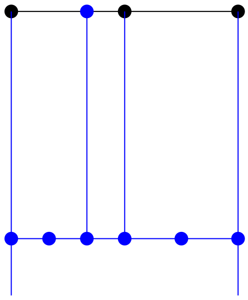
Step 4: The graph G' resulting from the first three steps is not yet an RNG or a Gabriel graph, because in the graph induced by the embedding we have described so far there are additional edges between the vertices in the ladder paths and vertex gadgets. Such edges would prevent us from applying Lemma 8.9. We will add cycles with the properties described in Lemma 8.6 in order to block such edges while making sure the resulting graph is Hamiltonian if and only if the original graph is.

Before describing the addition of the cycles in detail, we must consider the structure of the graph G' resulting from the first three steps. This graph has two types of faces. There are faces within the ladder paths and vertex gadgets. All other faces correspond to faces in the original graph G. In the following we will only consider the second type of face when referring to faces in G'. By Lemma 8.9, G' is Hamiltonian if and only if G is, since the first three steps perform the transformation described in the lemma. We will say that an edge in G' is *dockable* if this edge is permissible or G' is not Hamiltonian. In other words, if G' is Hamiltonian, then any dockable edge is permissible. We claim that every face in G' is bordered by an edge in a ladder path that is even and dockable. We find this edge in the following manner: Choose any ladder path that borders the face. By Observation 8.7, any traversal of a ladder path L_{k_1,k_2,k_3} with even k_1 contains an even inside and an even outside edge in the ladder path and any cover of either half of the ladder path contains every inside and every outside edge. In the first case, this edge may be found if we know at what vertex the traversal starts. In the second case, it does not matter which edge is chosen, so the edge from the first case is sufficient. The proof of Lemma 8.9, particularly Figure 8.7, shows that which vertex a traversal starts at depends only on whether the vertex gadget is a $(3, 0)$ or a $(2, 1)$-vertex gadget.

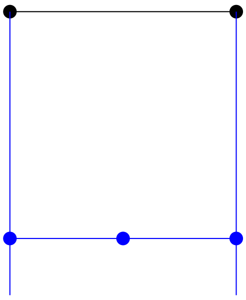
Cycles are added for every even r and even s such that (r, s) is in a face other than the outside face of G'. The cycles will be embedded as squares with corners at (r, s), $(r +$



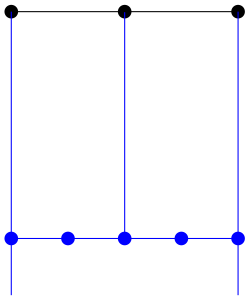
(a) A docking side adjoined to an edge.



(b) A docking side adjoined to a previously existing non-docking side.



(c) A non-docking side adjoined to an edge.



(d) A non-docking side adjoined to an edge.

Figure 8.9: Construction of the sides of a cycle: The number and positions of the vertices in a side depend on whether it is a docking side and what this side adjoins. In each picture, a side of the newly added cycle is on the bottom and it adjoins an edge or previously existing side on the top. The vertices and edges that result from the addition of the cycle are marked in blue while previously existing vertices and edges are in black.

$1, s)$, $(r+1, s+1)$, and $(r, s+1)$. We will call the cycle with these corners, the (r, s)-*cycle*. The cycles have four sides: one *docking side* and three *non-docking sides*. We will say that sides *adjoin* existing sides of other cycles or edges in ladder paths or vertex gadgets. For instance, the lower side of a cycle placed at (r, s) adjoins the edge or side running from $(r, s-1)$ to $(r+1, s-1)$. In order for Lemma 8.6 to be applicable, the docking side must adjoin a dockable edge. As we noted above, every face is adjacent to such an edge. By Lemma 8.6, all edges on the cycles added are also permissible if G' is Hamiltonian. So, we may adjoin the docking side of the first cycle in a face to the edge found in the ladder path in the manner described above and then adjoin the docking side of each new cycle to an edge in a previously added cycle.

We add the cycles in any order, as long as the docking side of the newly added cycle is adjoined to an existing edge that is dockable. The addition of the sides is illustrated in Figure 8.9. The (r, s)-cycle is added as follows: Suppose that the upper side, which goes from $(r, s+1)$ to $(r+1, s+1)$, of the cycle is the docking side. The docking side contains six vertices, including the corners. They are embedded at $(r+\delta, s+1)$ with $\delta \in \{0, \frac{1}{6}, \frac{1}{3}, \frac{1}{2}, \frac{3}{4}, 1\}$. The docking side adjoins the edge or side which runs from $(r, s+2)$

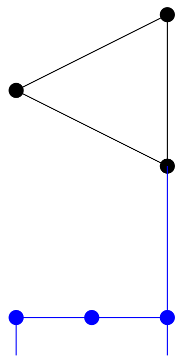
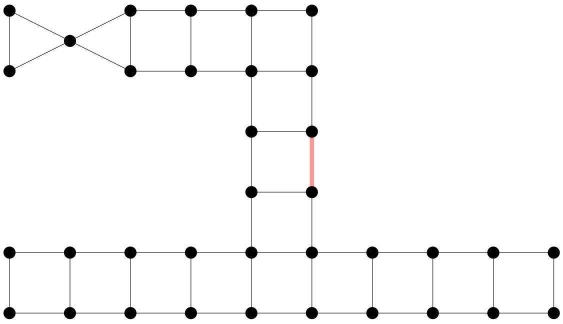


Figure 8.10: If a non-docking side adjoins an edge to a transitional vertex, one of the corners of that side is not adjacent to a vertex on that edge.

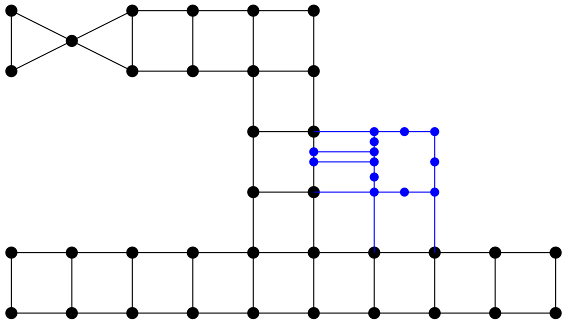
to $(r+1, s+2)$. If this is an edge, then it must be subdivided twice with new vertices introduced at $(r+\frac{1}{3}, s+2)$ and $(r+\frac{1}{2}, s+2)$. These new vertices are connected with edges to the vertices at $(r+\frac{1}{3}, s+1)$ and $(r+\frac{1}{2}, s+1)$. This case is illustrated in Figure 8.9a. If the docking side adjoins a side of a previously added cycle, then this must be a non-docking side. It has a vertex at $(r+\frac{1}{2}, s+2)$ in addition to the corners at $(r, s+2)$ and $(r+1, s+2)$. The edge between $(r, s+2)$ and $(r+\frac{1}{2}, s+2)$ must be subdivided once, with the new vertex embedded at $(r+\frac{1}{3}, s+2)$. Edges are added between the new vertex and the vertex at $(r+\frac{1}{3}, s+1)$ as well as between the existing vertex at $(r+\frac{1}{2}, s+2)$ and the vertex at $(r+\frac{1}{2}, s+1)$. This case is illustrated in Figure 8.9b. If the docking side is not the upper side, but one of the three others, then its embedding and its connections to its adjoining side or edge are analogous. This concludes the description of the docking side.

We will now describe the non-docking sides. When the cycle is added, the non-docking sides may adjoin a non-docking side of an existing cycle, an edge in a ladder path or a vertex gadget, or it may initially not adjoin anything. If it does not initially adjoin anything, then the non-docking side contains three vertices: the two corners and third vertex embedded halfway between them. If it adjoins an edge of a ladder path or a vertex gadget, then it consists of the same three vertices, but edges must be added between the two corners of the side and the endpoints of the edge (see Figure 8.9c). If it adjoins a non-docking side of an existing cycle, then the new non-docking side consists of five vertices, including the corners. Suppose the new non-docking side is the upper side of the (r, s)-cycle. Then the five vertices are embedded at $(r+\delta, s+1)$ with $\delta \in \{0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1\}$. The existing side contains the vertices (r, s), $(r+\frac{1}{2}, s)$, $(r+1, s)$, since it had not previously adjoined anything. Edges are added between the vertices at (r, s) and $(r, s-1)$, at $(r+\frac{1}{2}, s)$ and $(r+\frac{1}{2}, s-1)$, as well as at $(r+1, s)$ and $(r+1, s-1)$. This case is illustrated in Figure 8.9d.

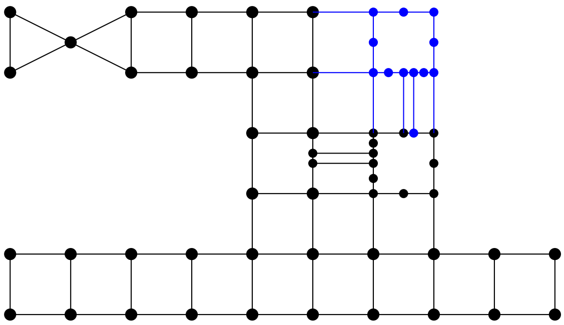
There is a special case when a non-docking side of a cycle adjoins the edge between a transitional vertex and one of the two final vertices of a half of a ladder path. Say the upper side of the (r, s)-cycle, which goes from $(r, s+1)$ to $(r+1, s+1)$, adjoins such an edge running from $(r, s+\frac{5}{2})$ to $(r+1, s+2)$. Then, just as when it adjoins a regular



(a) Excerpt of the graph pictured in Figure 8.8: The thicker edge marked in red is dockable, so the docking side of a cycle can be adjoined to it.



(b) Addition of a cycle: The left side is the docking side. It adjoins an edge in the ladder path, which is subdivided twice. The lower side adjoins an edge in the vertex gadget. The other two sides do not adjoin anything.



(c) Addition of another cycle: Its lower side adjoins a side of the previously added cycle. Its left side adjoins an edge in the ladder path. The other two sides do not adjoin anything.

Figure 8.11: Addition of cycles: Two cycles are added to a face in accordance with the construction. The newly added cycle is highlighted in blue.

edge in a ladder path, the upper side of the cycle consists of three vertices embedded at $(r, s+1)$, $(r+\frac{1}{2}, s+1)$, and $(r+1, s+1)$. The vertex at $(r, s+1)$ is adjacent to the vertex at $(r, s+2)$, but the vertex at $(r+1, s+1)$ does not have a neighbor in the edge this side adjoins. This situation is pictured in Figure 8.10.

We will illustrate the addition of cycles with an example. Figure 8.11a pictures an excerpt of the graph G' produced by **Step 2** and **3** (pictured in Figure 8.8). The even edge highlighted in red is dockable. So, we may add a cycle with its docking side adjoining this edge. This cycle is marked in blue in Figure 8.11b. Its left side is the docking side. Its lower side adjoins an edge in the vertex gadget. Its other two sides do not adjoin anything, but they are dockable, so the docking side of another cycle may adjoin these sides. In Figure 8.11c, another cycle, highlighted in blue, is added. Its docking side is the lower side and it adjoins the upper side of the previously added cycle. Its left side adjoins an edge in the ladder path, while its right and upper side do not adjoin anything.

An (r, s)-cycle must be added for every even r and s if (r, s) is located within a face

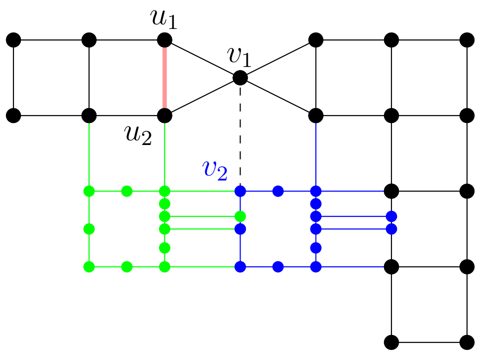


Figure 8.12: An excerpt of the graph G' produced by the reduction: The vertex v_1 is not an RNG-blocker for $\{u_1, u_2\}$, but u_2 is an RNG-blocker for $\{v_1, v_2\}$.

not contained within a vertex gadget or a ladder path.

Running time

We must show that the reduction runs in polynomial time. The graph G' contains:

- $\mathcal{O}(n)$ vertex gadgets each containing a constant number of vertices,
- $\mathcal{O}(n)$ ladder paths containing $\mathcal{O}(n)$ vertices each, for a total of $\mathcal{O}(n^2)$ vertices, and
- $\mathcal{O}(n^2)$ cycles each containing $\mathcal{O}(1)$ vertices.

In total, G' contains $\mathcal{O}(n^2)$ vertices and can be computed in polynomial time.

Correctness

We show that G is Hamiltonian if and only if G' is. This is true after every step of the construction. **Steps 2** and **3** perform the transformation described in Lemma 8.9, so this claim is a consequence of that lemma for these steps. **Step 4** involves subdividing an edge which is permissible if the graph is Hamiltonian. This is correct by Lemma 8.3. This step then adds a cycle fulfilling the conditions of Lemma 8.6. Note that, after a cycle is added, this cycle may be modified during the addition of subsequent cycles. However, each subdivision and each addition of a cycle preserve the Hamiltonicity of the graph by the aforementioned lemmas. Hence, the graph after all cycles have been added is Hamiltonian if and only if the original graph is.

Graph induced by embedding

Proving that the embedding emb given in the construction induces the graph described therein as its RNG requires a similar analysis to the one given in Chapter 4 for the reduction for Dominating Set. This reduction, like the one for Dominating Set, generates a graph that is in a way grid-like. We will omit most of the details of this analysis and focus on a few aspects that may not be immediately obvious.

Figure 8.12 pictures an excerpt of the graph produced by the reduction with parts of a ladder path, including the transitional vertex, and two cycles, which are highlighted in blue and green. One edge is thicker and highlighted in red. We will argue it is not blocked. Specifically, we will show that v_1 is not a blocker for $\{u_1, u_2\}$. The distance between u_1 and u_2 is $d(\text{emb}(u_1), \text{emb}(u_2)) = 1$, while $d(\text{emb}(u_i), \text{emb}(v_1)) = \frac{\sqrt{5}}{2} > 1$ for $i \in \{1, 2\}$, so $\{u_1, u_2\}$ is not blocked by v_1. We also show that u_2 is a blocker for $\{v_1, v_2\}$. This is because $d(\text{emb}(v_1), \text{emb}(v_2)) = \frac{3}{2}$, while $d(\text{emb}(u_2), \text{emb}(v_2)) = \sqrt{2} < \frac{3}{2}$ and $d(\text{emb}(u_2), \text{emb}(v_1)) = \frac{\sqrt{5}}{2} < \frac{3}{2}$.

Gabriel graphs

We will now discuss Gabriel graphs. This reduction can, be adjusted to Gabriel graphs with one minor modification. In the excerpt of G' pictured in Figure 8.12, u_2 is not a Gabriel-blocker for $\{v_1, v_2\}$ since $\measuredangle\, \text{emb}(v_1)\, \text{emb}(u_2)\, \text{emb}(v_2) < 90°$. As a result, **Step 4** of the construction, specifically the special case for sides adjoining an edge to a transitional vertex, has to be adjusted. Say the upper side of the (r, s)-cycle, which goes from $(r, s+1)$ to $(r+1, s+1)$, adjoins such an edge running from $(r, s+2)$ to $(r+1, s+\frac{5}{2})$. Just as when it adjoins a regular edge in a ladder path, the upper side of the cycle consists of three vertices embedded at $(r, s+1)$, $(r+\frac{1}{2}, s+1)$, and $(r+1, s+1)$. The vertex at $(r, s+1)$ is adjacent to the vertex at $(r, s+2)$. In the reduction for RNGs the vertex at $(r+1, s+1)$ does not have a neighbor in the edge this side adjoins, in the case of Gabriel graphs there is an edge from the vertex at $(r+1, s+1)$ to the transitional vertex at $(r+1, s+\frac{5}{2})$. So, in the reduction for Gabriel graphs an edge must be added along the dashed line in Figure 8.12. This does not affect the correctness of the reduction. Lemma 8.6 can still be applied, since when adding this cycle, the vertex connected to the transitional vertex is not adjacent to any other vertex on the cycle that has a neighbor outside of the cycle.

8.3 Conclusions

This concludes the reduction. The approach above fails for relatively closest graphs. It requires the use of 3-cycles to separate the two halves of ladder paths and RCGs cannot contain 3-cycles. We will leave open whether Hamiltonian Cycle on RCGs can be solved in polynomial time, but we suspect that the problem is probably also NP-hard on RCGs.

While the problem remains open for RCGs, we have shown that it is NP-hard for relative neighborhood graphs and Gabriel graphs. In fact, the reduction for RNGs and Gabriel graphs only creates vertices with degree at most four.

Theorem 8.10. Hamiltonian cycle *is NP-hard when restricted to relative neighborhood graphs or Gabriel graphs, each with maximum degree at most four.*

For graphs with maximum degree two, Hamiltonian Cycle is easy to solve. However, it remains open whether this problem can be solved in polynomial time when restricted to proximity graphs with maximum degree three.

We will show that our reduction also implies that Hamiltonian Cycle cannot be solved in time $2^{o(n^{\frac{1}{4}})}$ on each of our proximity graph classes (where n is the number of vertices in a graph), unless the exponential time hypothesis fails (on the ETH, see Section 2.5).

We must first establish an ETH-based lower bound of $2^{o(\sqrt{n})}$ for Hamiltonian Cycle on planar graphs with maximum degree 3. Lokshtanov, Marx, and Saurabh [LMS11] prove this lower bound for planar Hamiltonian Cycle. Their proof is based on the reduction given by Garey, Johnson, and Tarjan [GJT76]. This reduction produces graphs with maximum degree 3. Thus:

Theorem 8.11. *Unless the ETH fails,* Hamiltonian Cycle *restricted to planar graphs with maximum degree* 3 *cannot be solved in time* $2^{o(\sqrt{n})}$.

Our reduction from Hamiltonian Cycle on planar graphs with maximum degree 3 to its restriction to our classes of proximity graphs maps instances with m edges and n vertices to graphs with $\mathcal{O}(n^2)$ vertices and edges. This along with Theorem 8.11 implies:

Corollary 8.12. *Unless the ETH fails,* Hamiltonian Cycle *restricted to RNGs or Gabriel graphs cannot be solved in time* $2^{o(n^{\frac{1}{4}})}$.

9 Temporal separators

Proximity graphs model objects in the plane. Their positions are assumed to be static. In order to model objects in motion, we will use temporal proximity graphs. Temporal graphs are graphs that change over time. They have found applications in modeling a wide range of real-world phenomena. Kempe, Kleinberg, and Kumar [KKK02] showed that finding small separators in such graphs, unlike the static variant of the problem, is NP-hard. We will prove a similar result for temporal proximity graphs. Before formally defining temporal graphs and this problem, however, we will describe a variant of 3-SAT, which we will use to prove NP-hardness.

9.1 A 3-SAT variant

Recall the definition of the 3-SAT problem in Section 2.5. Given a variable x_j and a clause C_i, we will say that x_j *occurs* in C_i if $x_j \in C_i$ or $\neg x_j \in C_i$. It *occurs positively* if $x_j \in C_i$ and *negatively* if $\neg x_j \in C_i$. According to Garey and Johnson [GJ79], 3-SAT remains NP-hard even when restricted to formulas in which each variable occurs at most three times. We will briefly sketch a proof of this claim in order to be able to argue that the ETH implies a lower bound of $2^{o(m+n)}$ for this restriction of 3-SAT where m is the number of clauses in a formula and n the number of variables.

Theorem 9.1. 3-SAT *with the restriction that each variable occurs at most three times is NP-hard. Unless the ETH fails, it cannot be solved in time $2^{o(m+n)}$ where m is the number of clauses in a formula and n the number of variables.*

Proof. The general version of 3-SAT can be reduced in polynomial time to the restricted version in the following manner. Suppose that the variable x_i occurs k times in a formula. Then, we may replace these k occurrences by the new variables $x_i^1, \ldots, x_i^k$. We additionally add the clauses $\{x_i, \neg x_i^j\}$ and $\{\neg x_i, x_i^j\}$ for $j = 1, \ldots, k$. These additional clauses ensure that any truth assignment that satisfies the formula must assign the same value to x_i^j for all $j = 1, \ldots, k$. The resulting formula contains at most $7m$ clauses since for each clause in the original formula we add two clauses for each of the at most three variables that occur in the clause. Moreover, it contains at most $n + 3m$ variable since three new variables are added for each clause. As a result, any algorithm solving the restricted form of 3-SAT in time $2^{o(m+n)}$ could be used to solve general 3-SAT in time $2^{o(n)}$. □

We will now prove the NP-hardness of a further restricted version of 3-SAT.

Lemma 9.2. 3-SAT *remains NP-hard when the following restrictions are imposed on input instances:*

- *every variable occurs positively at most twice and negatively at most twice and*
- *for every variable x_j one of the following holds true: Every positive occurrence of x_j precedes every negative occurrence of x_j (in other words, if $x_j \in C_i$ and $\neg x_j \in C_{i'}$, then $i < i'$) or every negative occurrence of x_j precedes every positive occurrence (that is, if $\neg x_j \in C_i$ and $x_j \in C_{i'}$, then $i < i'$).*

Moreover, unless the ETH fails, this restricted version of 3-SAT *cannot be solved in time $2^{o(m+n)}$ where m is the number clauses in a formula and n the number of variables.*

Note that the second condition, of course, depends on a certain ordering of the clauses. We will refer to this restriction as RESTRICTED 3-SAT. To our knowledge, the NP-hardness of this problem has not been established in the literature.

Proof. We will show that any 3-SAT instance in which every variable occurs at most three times can be transformed in polynomial time into an equivalent RESTRICTED 3-SAT instance. Given a formula φ compute ψ by performing the following modifications:

1. If there is a clause C_i and a variable x_j such that $x_j \in C_i$ and $\neg x_j \in C_i$, then remove C_i.
2. If a variable occurs only positively or only negatively, then remove every clause in which it occurs.
3. If a variable x_j occurs negatively twice, then replace every positive occurrence of this variable by $\neg x_j$ and every negative occurrence by x_j.
4. If there are clauses $C_{i_1}, C_{i_2}, C_{i_3}$, with $i_1 < i_2 < i_3$, and a variable x_j such that $x_j \in C_{i_1}$, $\neg x_j \in C_{i_2}$, and $x_j \in C_{i_3}$, then do the following:
   - Replace x_j with a new variable x'_j in C_{i_3}
   - Add the clause $\{x_j, \neg x'_j\}$, placing it immediately after C_{i_1}
   - Add the clause $\{\neg x_j, x'_j\}$, placing it immediately before C_{i_3}

It is easy to see that none of the previous steps changes the satisfiability of φ.

The first three steps do not increase the number of variables or clauses in the formula. The final step of the transformation adds at most n variables and $2n$ clauses where n is the number of variables in the original formula. This implies the ETH-based lower bound. □

9.2 Temporal graphs, paths, and separators

We now turn our attention to temporal graphs. A temporal graph $G = (V, \tau, E_1, \ldots, E_\tau)$ consists of a vertex set V, a lifespan $\tau \in \mathbb{N}$, and edge sets $E_1, \ldots, E_\tau$. The static graph $(V, E_1 \cup \ldots \cup E_\tau)$ is its underlying graph and (V, E_i) is its i-th layer. The number of edges in G is $|E_1| + \ldots + |E_\tau|$. A path from v_1 to v_k in G is an alternating sequence of vertices and timestamps $v_1, \tau_1, v_2, \tau_2, \ldots, \tau_{k-1}, v_k$ such that:

- $\{v_i, v_{i+1}\} \in E_{\tau_i}$,
- $\tau_i \leq \tau_{i+1}$, and
- $v_j \neq v_{j'}$

for all $i = 1, \ldots, k-1$ and $j, j' = 1, \ldots, k$, $j \neq j'$. It is a strict path if $\tau_i < \tau_{i+1}$ for all $i = 1, \ldots, k-1$. If $s, t \in V$ and $s \neq t$, then $S \subseteq V \setminus \{s, t\}$ is a (strict) (s, t)-separator if every (strict) path from s to t passes through a vertex in S. This leads to the following computational problem:

(Strict) Temporal (s,t)-Separation
Input: A temporal graph $G = (V, \tau, E_1, \ldots, E_\tau)$, a nonnegative integer k, and vertices $s, t \in V$, $s \neq t$.
Question: Does G contain a (strict) (s, t)-separator $S \subseteq V \setminus \{s, t\}$ with $|S| \leq k$?

As we mentioned before, Kempe, Kleinberg, and Kumar [KKK02] showed that this problem is NP-complete. Fluschnik et al. [Flu+20] and Zschoche et al. [Zsc+20] analyzed it in more detail. They define planar temporal graphs in two different ways: A temporal graph may be considered planar if its underlying graph is planar or it may be considered planar if each of its layers is planar. Since subgraphs of planar graphs are planar, planarity in the first sense implies planarity in the second sense. The same does not hold for the classes of proximity graphs we have considered, if we define temporal proximity graphs in the same two senses.

In the following, we will consider temporal proximity graphs only in the second sense. Such graphs may be thought of as points that move in the plane and whose proximity relationships therefore change over time. We will show that (Strict) Temporal (s,t)-Separation remains NP-hard when restricted to temporal graphs all of whose layers are path graphs. Path graphs are obviously Gabriel graphs, relative neighborhood graphs, and relatively closest graphs: One may embed such graphs by placing all vertices in the graph on a line segment in the same order in which they appear on the path. So, this reduction implies that the problem is also NP-complete when restricted to temporal graphs whose layers are proximity graphs.

Since all layers of the temporal graph produced by our reduction are paths, we will describe the edge sets E_i by simply giving the order in which the vertices appear in this path. To this end, let:

$$E(w_1, \ldots, w_\ell) \coloneqq \{\{w_i, w_{i+1}\} \mid i = 1, \ldots, \ell - 1\}.$$

We will denote the concatenation of two paths in the following manner:

$$E(w_1, \dots, w_r) \cdot E(w'_1, \dots, w'_s) \coloneqq E(w_1, \dots, w_r, w'_1, \dots, w'_s).$$

Finally, if $A = \{w_1, \dots, w_r\}$ and $B = \{w'_1, \dots, w'_s\}$ are two disjoint sets of vertices and $r \leq s$, then we will denote the path in which vertices from each set alternate (in any order) followed by all remaining vertices in B (also in any order) as follows:

$$E(A, B) \coloneqq E(w_1, w'_1, w_2, w'_2, \dots, w'_{r-1}, w_r, w'_r) \cdot E(w'_{r+1}, \dots, w'_s).$$

9.3 Reduction

We will now give a reduction from Restricted 3-SAT to Temporal (s,t)-Separation on temporal graphs each of whose layers are paths.

Construction

Suppose that φ is a Boolean conjunctive formula over the variables $x_1, \dots, x_n$ consisting of the clauses $C_1, \dots, C_m$ with the aforementioned restrictions. We set $\tau \coloneqq 8n + m + 1$ and let $V_1 \coloneqq \{v_i, v'_i, \overline{v_i}, \overline{v_i}' \mid j = 1, \dots, n\}$, $V_2 \coloneqq \{u_j \mid j = 1, \dots, 4n + 1\}$, and $V \coloneqq \{s, t\} \cup V_1 \cup V_2$. Moreover, let $k \coloneqq 6n + 1$. Intuitively, the vertices v_j and v'_j will represent the at most two occurrences of x_j and $\overline{v_j}$ and $\overline{v_j}'$ the at most two occurrences of $\neg x_j$. The vertices u_j are blockers that will be used to separate the other vertices from one another.

We will now describe the edge sets of the graph produced by the reduction. There are three types of layers: one blocker layer for each blocker, four variable layers for each variable, and one clause layer for each clause. The blocker layers ensure that every blocker must be part of any (s,t)-separator. Let:

$$E_i^B \coloneqq E(V_1, V_2 \setminus \{u_i\}) \cdot E(s, u_i, t)$$

for all $i = 1, \dots, 4n + 1$. The four variable layers ensure that for every variable x_i either both v_i and v'_i or both $\overline{v_i}$ and $\overline{v_i}'$ are part of any (s,t)-separator. In φ either every positive occurrence of x_i precedes every negative occurrence or every negative occurrence precedes every positive occurrence. In the first case, we let:

$$\begin{aligned}
E_1^{x_i} &\coloneqq E(V_1 \setminus \{v_i, \overline{v_i}\}, V_2) \cdot E(s, v_i, \overline{v_i}, t), \\
E_2^{x_i} &\coloneqq E(V_1 \setminus \{v_i, \overline{v_i}'\}, V_2) \cdot E(s, v_i, \overline{v_i}', t), \\
E_3^{x_i} &\coloneqq E(V_1 \setminus \{v'_i, \overline{v_i}\}, V_2) \cdot E(s, v'_i, \overline{v_i}, t), \text{ and} \\
E_4^{x_i} &\coloneqq E(V_1 \setminus \{v'_i, \overline{v_i}'\}, V_2) \cdot E(s, v'_i, \overline{v_i}', t).
\end{aligned}$$

In the second case we switch the order of the vertices representing x_i with those representing $\neg x_i$ and set:

$$\begin{aligned}
E_1^{x_i} &\coloneqq E(V_1 \setminus \{v_i, \overline{v_i}\}, V_2) \cdot E(s, \overline{v_i}, v_i, t),\\
E_2^{x_i} &\coloneqq E(V_1 \setminus \{v_i, \overline{v_i}'\}, V_2) \cdot E(s, \overline{v_i}', v_i, t),\\
E_3^{x_i} &\coloneqq E(V_1 \setminus \{v_i', \overline{v_i}\}, V_2) \cdot E(s, \overline{v_i}, v_i', t), \text{ and}\\
E_4^{x_i} &\coloneqq E(V_1 \setminus \{v_i', \overline{v_i}'\}, V_2) \cdot E(s, \overline{v_i}', v_i', t)
\end{aligned}$$

Finally, the clause layer for C_i ensures that a vertex corresponding to one of the literals in C_i must be part of any (s,t)-separator. Suppose that $C_i = \{\ell_1, \ell_2, \ell_3\}$ where the ℓ_j are literals. Let w_1, w_2, and w_3 be the corresponding vertices. That is:

$$w_j \coloneqq \begin{cases} v_r, & \text{if } \ell_j = x_r \text{ and this is the first positive occurrence of } x_r,\\ v_r', & \text{if } \ell_j = x_r \text{ and this is the second positive occurrence of } x_r,\\ \overline{v_r}, & \text{if } \ell_j = \neg x_r \text{ and this is the first negative occurrence of } x_r,\\ \overline{v_r}', & \text{if } \ell_j = \neg x_r \text{ and this is the second negative occurrence of } x_r. \end{cases}$$

We then use the following as the clause layer for C_i:

$$E^{C_i} \coloneqq E(V_1 \setminus \{w_1, w_2, w_3\}, V_2) \cdot E(s, w_1, w_2, w_3, t).$$

To complete the description of the reduction, we must give the order in which the layers appear. The blocker layers appear first, so the first $4n+1$ layers are $E_1^B, \ldots, E_{4n+1}^B$. They are followed by the clause layers. We then intersperse the variable layers in the following manner. If every positive occurrence of x_i precedes every negative occurrence, then we place x_i's variable layers between the clause layer of the last clause in which x_i occurs positively and the clause layer of the first clause in which it appears negatively. If every negative occurrence of x_i precedes every positive occurrence, then we place x_i's variable layers between the clause layer of the last clause in which x_i occurs negatively and the clause layer of the first clause in which it appears positively.

Correctness

We must show that φ is satisfiable if and only if the temporal graph produced by the reduction has an (s,t)-separator S of size at most $k = 6n+1$.

Suppose that $\alpha\colon \{x_1, \ldots, x_n\} \to \{0,1\}$ is a satisfying assignment for φ. We let $S \coloneqq V_2 \cup \{v_i, v_i' \mid \alpha(x_i) = 1\} \cup \{\overline{v_i}, \overline{v_i}' \mid \alpha(x_i) = 0\}$. Of course, $|S| = 6n+1$. We must show that S is an (s,t)-separator. Suppose towards a contradiction that $w_1, \tau_1, \ldots, \tau_{r-1}, w_r$ with $s = w_1$ and $t = w_r$ is an (s,t)-path which does not pass through a vertex in S. The layer $E_{\tau_{r-1}}$ cannot be a blocker layer, since t's only neighbor in a blocker layer is a blocker and S contains all blockers. Suppose that $E_{\tau_{r-1}}$ is a variable layer. Then w_{r-1} must be v_i, v_i', $\overline{v_i}$, or $\overline{v_i}'$ for some i. In all preceding layers, however, that vertex is only adjacent to blockers. Hence, this is also not a possibility. Finally, suppose that $E_{\tau_{r-1}}$ is the clause layer for C_i. Without loss of generality, we may assume that $C_i = \{x_1, x_2, x_3\}$.

Since α satisfies φ, $\alpha(x_j) = 1$ for a $j \in \{1, 2, 3\}$. Hence, $v_j, v_j' \in S$. Then, $w_{r-1} = v_{j'}$ or $w_{r-1} = v_{j'}'$ with $j' > j$. Again without loss of generality, assume that $w_{r-1} = v_{j'}$. If every positive occurrence of $x_{j'}$ precedes every negative occurrence, then in all layers preceding this one $v_{j'}$ is only adjacent to blockers. Hence, every positive occurrence of $x_{j'}$ precedes every negative occurrence. Then, the only layers preceding this one in which $v_{j'}$ is adjacent to vertex which is not a blocker are $x_{j'}$'s variable layers. However, in these layers $v_{j'}$ is only adjacent to t, $\overline{v_{j'}}$, $\overline{v_{j'}}'$, and blockers. Since $v_{j'} \notin S$, it follows that $\overline{v_{j'}}, \overline{v_{j'}}' \in S$. The same is true of all blockers. Hence, there is no possible vertex that could be w_{r-2}. So, $E_{\tau_{r-1}}$ also cannot be a clause layer. the So, S is, in fact, an (s,t)-separator.

Now suppose that S, $|S| \leq 6n + 1$, is an (s,t)-separator. Because of the blocker layers, S must contain all blockers. Because of the variable layers, S must contain both v_i and v_i' or both $\overline{v_i}$ and $\overline{v_i}'$ for every variable x_i. Because $|S| \leq 6n+1$, S cannot contain any further vertices. Hence, the assignment α with:

$$\alpha(x_i) \coloneqq \begin{cases} 1, & \text{if } v_i, v_i' \in S, \\ 0, & \text{if } \overline{v_i}, \overline{v_i}' \in S, \end{cases}$$

is well-defined. Because of the clause layers, α must satisfy φ, making φ satisfiable.

9.4 Conclusions

This concludes our reduction. We have only discussed the non-strict version of Temporal (s,t)-Separation. However, the reduction described above can easily be adapted to the strict case. One must only repeat layers sufficiently often, so that any path becomes a strict path. We have proved the following:

Theorem 9.3. (Strict) Temporal (s,t)-Separation *is NP-complete when restricted to temporal graphs in which all layers are path graphs.*

This implies:

Corollary 9.4. (Strict) Temporal (s,t)-Separation *is NP-complete when restricted to temporal graphs in which all layers are relative neighborhood graphs, relatively closest graphs, or Gabriel graphs.*

We leave open the question of whether Temporal (s,t)-Separation is polynomial-time solvable on temporal graphs whose underlying graph is a proximity graph, but we suspect that it is probably also NP-hard.

We will show that our reduction also implies that (Strict) Temporal (s,t)-Separation cannot be solved in time $2^{o(n+\sqrt{m})}$ on the temporal versions of each of our proximity graph classes (where n is the number of vertices in a graph and m the total number of edges in all layers), unless the exponential time hypothesis fails (on the ETH, see Section 2.5).

Our reduction from Restricted 3-SAT produces a temporal graph containing $5n+3$ vertices, $8n + m + 1$ layers, and $(5n + 2)(8n + m + 1) \in \mathcal{O}((n + m)^2)$ edges. Due to Lemma 9.2, this implies:

Corollary 9.5. *Unless the ETH fails,* (Strict) Temporal (s, t)-Separation *cannot be solved in time* $2^{o(n+\sqrt{m})}$ *on temporal graphs all of whose layers are path graphs where* n *is the number of vertices in the temporal graph and* m *the total number of edges in the graph. Accordingly, the same holds for temporal RNGs, RCGs, and Gabriel graphs.*

10 Recognition

The recognition problem for a graph class $\mathcal{C}$ asks whether $G \in \mathcal{C}$ for a given graph G. In the context of geometric graphs, this problem is also frequently referred to as the realizability or drawability problem and one is often also interested in constructing the geometric structure that witnesses the graph's membership in $\mathcal{C}$.

The recognition problem has been shown to be NP-complete for several classes that are closely related to the proximity graphs we have studied here. For example, Eades and Whitesides proved NP-hardness for the recognition of nearest neighbor graphs [EW95] and Euclidean minimum spanning trees [EW96]. Cardinal and Hoffmann [CH17] proved that the recognition of point visibility graphs, which may be thought of as empty region graphs with the line segment between two points as their region of influence, is complete for the existential theory of the reals ($\exists\mathbb{R}$). Di Battista, Lenhart, and Liotta [DBLL94] survey results on the recognition problem for geometric graph classes. Bose, Lenhart, and Liotta [BLL96] showed that the recognition problem for RNGs, RCGs, and Gabriel graphs can be decided in linear time when restricted to trees.

One typical approach to proving NP-hardness for such problems is the so-called logic engine (see [EW95] for a detailed description). This approach utilizes "rigidity" in the realizability of certain graphs, that is, that realizations of certain graphs are unique up to rotation, scaling, translation, and reflection. The problem with this approach is that RNGs, RCGs, and Gabriel graphs are typically very non-rigid. Fekete, Houle, and Whitesides [FHW97] introduced the "wobbly" logic engine, which relaxes the rigidity requirement to some extent, but it is unclear how it could be applied to the graph classes we have studied.

The complexity of the recognition problem for RNGs, RCGs, and Gabriel graphs remains open. It is not even clear whether the problem is in NP, let alone whether it is polynomial-time solvable. The best we are able to prove is that this problem is in $\exists\mathbb{R}$.

The existential theory of the reals is defined as the set of all problems that may be reduced to the following problem: The input is an existential first-order formula $\exists x_1 \ldots \exists x_k \varphi(x_1, \ldots, x_k)$ where φ is a quantifier-free formula using 0, 1, addition, multiplication, strict and non-strict comparison, Boolean operators, and the variables $x_1, \ldots, x_k$. One is asked to decide whether this formula is true in the field of the reals. It is known that NP $\subseteq \exists\mathbb{R} \subseteq$ PSPACE. Schaefer [Sch13] gives a survey on $\exists\mathbb{R}$.

We will now briefly describe how the proximity graph realizability of a graph can be expressed as an instance to the aforementioned $\exists\mathbb{R}$-complete problem. Given three points $(x_1, y_1), (x_2, y_2), (x_3, y_3) \in \mathbb{R}^2$, (x_1, y_1) is an RNG-blocker for (x_2, y_2) and (x_3, y_3) if and only $(x_1 - x_2)^2 + (y_1 - y_2)^2 < (x_2 - x_3)^2 + (y_2 - y_3)^2$ and $(x_1 - x_3)^2 + (y_1 - y_3)^2 <$

$(x_2 - x_3)^2 + (y_2 - y_3)^2$. Consequently, we use the formula

$$\psi^{\text{RNG}}_{(x_1,y_1),(x_2,y_2),(x_3,y_3)} :=(x_1 - x_2)^2 + (y_1 - y_2)^2 < (x_2 - x_3)^2 + (y_2 - y_3)^2 \\ \wedge (x_1 - x_3)^2 + (y_1 - y_3)^2 < (x_2 - x_3)^2 + (y_2 - y_3)^2$$

to express that (x_1, y_1) is an RNG-blocker for (x_2, y_2) and (x_3, y_3). Strictly speaking, taking the square is not allowed but may be expressed using multiplication in the obvious manner. Similar formulas $\psi^{\text{RCG}}_{(x_1,y_1),(x_2,y_2),(x_3,y_3)}$ and $\psi^{\text{GAB}}_{(x_1,y_1),(x_2,y_2),(x_3,y_3)}$ exist to describe RCG and Gabriel-blockers. Using these formulas, we may express that a given graph $G = (V, E)$ with $V = \{v_1, \ldots, v_n\}$ possesses a $\mathcal{C}$-embedding for $\mathcal{C} \in \{\text{RNG}, \text{RCG}, \text{GAB}\}$ using the following formula:

$$\varphi^{\mathcal{C}}_G := \exists x_1 \ldots \exists x_n \exists y_1 \ldots \exists y_n \bigwedge_{i=1}^{n} \bigwedge_{j=i+1}^{n} (x_i \neq x_j \vee y_i \neq y_j) \\ \wedge \bigwedge_{\{v_i,v_j\} \in E} \bigwedge_{v_k \in V \setminus \{v_i,v_j\}} \neg\psi^{\mathcal{C}}_{(x_k,y_k),(x_i,y_i),(x_j,y_j)} \\ \wedge \bigwedge_{\{v_i,v_j\} \notin E} \left(\bigvee_{v_k \in V \setminus \{v_i,v_j\}} \psi^{\mathcal{C}}_{(x_k,y_k),(x_i,y_i),(x_j,y_j)} \right).$$

The variables represent the coordinates of a possible embedding of each vertex. The first part of the formula states that no two vertices may be embedded in the same point. The second part expresses that no edge may be blocked. The third part states that there must be a blocker for any two non-adjacent vertices. Note that the size this formula is cubic and therefore polynomial in the number of vertices in G. This proves:

Theorem 10.1. *The recognition problem for relative neighborhood graphs, relatively closest graphs, and Gabriel graphs is a member of the existential theory of the reals.*

Although this is the only bound on the complexity of the recognition problem for our graph classes that we are able to prove, we are not convinced that it is $\exists\mathbb{R}$-complete. Without much conviction, we conjecture that it is in NP, if not polynomial-time solvable.

11 Conclusion

We have investigated the computational complexity of several algorithmic problems when restricted to three classes of proximity graphs: relative neighborhood graphs, relatively closest graphs, and Gabriel graphs. We have found that the classical NP-complete problems Independent Set, Vertex Cover, 3-Colorability, Dominating Set, Feedback Vertex Set, and Hamiltonian Cycle remain NP-hard when restricted to these graph classes (except for 3-Colorability and possibly Hamiltonian Cycle on RCGs). Our reductions are based in large part on book embeddings that are useful to enforce a certain uniform structure on input graphs. We have further studied the Temporal (s, t)-Separation restricted to temporal graphs that are layer-wise proximity graphs and proved that it, too, remains NP-hard. We have also shown that the recognition problem for RNGs, RCGs, and Gabriel graphs is in the existential theory of the reals ($\exists\mathbb{R}$). We will now conclude by describing interesting open questions that could be the subject of further research.

Open questions In Chapter 8, we mentioned that the complexity of the Hamiltonian Cycle problem on RCGs remains open. For the classical NP-complete problems we have investigated, we have show an ETH-based lower bound of $2^{o(n^{\frac{1}{4}})}$ for their restrictions to proximity graphs. This lower bound is not necessarily tight. In fact, we suspect that it is probably not. The corresponding lower bounds for arbitrary planar graphs are $2^{o(\sqrt{n})}$ and these are mostly tight [LMS11]. For these problems, we have shown that that they remain NP-complete on proximity graphs even when their maximum degrees are restricted to a value between 4 and 8. In each case, a gap remains between the maximum degree restriction for which the problem has been shown to be NP-complete and the maximum degree restriction for which polynomial-time algorithms are known. As mentioned in Chapter 9, the question of whether temporal separators can be found in polynomial time in temporal graphs whose underlying graph is an RNG, RCG, or Gabriel graph also remains open. As discussed in Chapter 10, the precise complexity of the recognition problem for these graph classes remains an open question. We will now discuss two further open questions.

4-Coloring As we proved in Chapter 6, determining whether a Gabriel graph or a relative neighborhood graph is 3-colorable is an NP-complete problem. By contrast, every planar graph can be colored with four colors. This is the famous four color theorem. The fastest known algorithm for actually computing a 4-coloring of a planar graph runs in quadratic time and is due to Robertson et al. [Rob+96; Rob+97].

Cimikowski [Cim90] proposes an algorithm that computes a 4-coloring of any relative neighborhood graph in linear time. This algorithm is based on a similar algorithm for computing 5-colorings of arbitrary planar graphs in linear time, which is due to Williams [Wil85]. However, Cimikowski's algorithm is based on an erroneous claim made by Urquhart [Urq83].

Cimikowski's algorithm is based on the fact that planar graphs are 5-degenerate. This means that every planar graph contains a vertex of degree at most five. The core of the algorithm consists in finding a vertex v of degree at most 5 and performing one of the following three operations.

1. If $\deg(v) \leq 3$, then delete v and color the remaining graph. Color v with a color not used on any of its three neighbors.

2. If $\deg(v) = 4$, then v's neighborhood must contain two nonadjacent vertices u_1 and u_2. Identify u_1 and u_2, creating a new vertex u that is adjacent to all vertices that are adjacent to u_1 or u_2. Delete v. Color the remaining graph. Assign u's color to both u_1 and u_2 and assign a color to v that is not used in v's neighborhood.

3. If $\deg(v) = 5$, then v's neighborhood must contain three pairwise nonadjacent vertices u_1, u_2, and u_3. Identify u_1, u_2, and u_3 in the aforementioned manner and delete v. Color the remaining graph. Assign u's color to u_1, u_2, and u_3 and assign a color to v that is not used in v's neighborhood.

The third operation is based on the claim that the wheel graph W_6 cannot occur as a subgraph of an RNG. This is Urquhart's Lemma 4.2. However, as Bose et al. [Bos+12] demonstrated, this claim is incorrect. Moreover, they also proved that there are, in fact, RNGs with a minimum degree of five, implying that the third case may indeed occur.

Although we have shown that Cimikowski's algorithm does not solve the problem of computing a 4-coloring of relative neighborhood graph in linear time, we leave the question of whether relative neighborhood graphs can be 4-colored more efficiently than arbitrary planar graphs, that is in subquadratic time, open. It also remains open for Gabriel graphs.

Treewidth Treewidth is a well-known graph parameter (for a definition, see Diestel [Die17], for instance). In the TREEWIDTH problem, one is given a graph G and an integer k, and one is asked to decide whether G's treewidth is at most k. For the restriction of this problem to planar graphs, no polynomial-time algorithm or NP-hardness proof is known [Bod12], so the question of whether it can solved in polynomial time remains open. This question has received considerable attention and is likely very difficult to resolve.

We also leave open the question of whether the TREEWIDTH problem can be solved in polynomial time on the proximity graph classes we have considered here. Since an NP-hardness proof for these classes would also prove hardness for planar graphs in general, it would also probably be very difficult. A polynomial-time algorithm for these specific classes, by contrast, would not resolve the more general problem, yet would still be of independent interest.

Literature

[AB09] S. Arora and B. Barak. *Computational Complexity: A Modern Approach.* Cambridge, New York: Cambridge University Press, 2009 (cit. on p. 20).

[Ada09] A. Adamatzky. "Developing proximity graphs by *Physarum polycephalum*: Does the plasmodium follow the Toussaint hierarchy?" In: *Parallel Processing Letters* 19.01 (2009), pp. 105–127 (cit. on p. 6).

[AM92] P. K. Agarwal and J. Matoušek. "Relative neighborhood graphs in three dimensions". In: *Computational Geometry* 2.1 (1992), pp. 1–14 (cit. on p. 9).

[BCJ10] E. Baccelli, J. A. Cordero, and P. Jacquet. "Using relative neighborhood graphs for reliable database synchronization in MANETs". In: *Fifth IEEE Workshop on Wireless Mesh Networks (WiMesh '10).* 2010, pp. 1–6 (cit. on p. 6).

[BGR16] M. A. Bekos, M. Gronemann, and C. N. Raftopoulou. "Two-page book embeddings of 4-planar graphs". In: *Algorithmica* 75.1 (2016), pp. 158–185 (cit. on p. 19).

[BK79] F. Bernhart and P. C. Kainen. "The book thickness of a graph". In: *Journal of Combinatorial Theory, Series B* 27.3 (1979), pp. 320–331 (cit. on pp. 18, 19).

[BLL96] P. Bose, W. Lenhart, and G. Liotta. "Characterizing proximity trees". In: *Algorithmica* 16.1 (1996), pp. 83–110 (cit. on pp. 9, 93).

[Bod12] H. L. Bodlaender. "Fixed-parameter tractability of treewidth and pathwidth". In: *The Multivariate Algorithmic Revolution and Beyond: Essays Dedicated to Michael R. Fellows on the Occasion of His 60th Birthday.* Ed. by H. L. Bodlaender, R. Downey, F. V. Fomin, and D. Marx. Berlin, Heidelberg: Springer, 2012, pp. 196–227 (cit. on p. 96).

[Bos+12] P. Bose, V. Dujmović, F. Hurtado, J. Iacono, S. Langerman, H. Meijer, V. Sacristán, M. Saumell, and D. R. Wood. "Proximity graphs: E, δ, Δ, χ and ω". In: *International Journal of Computational Geometry & Applications* 22.05 (2012), pp. 439–469 (cit. on pp. 9, 96).

[CCL09] J. Cardinal, S. Collette, and S. Langerman. "Empty region graphs". In: *Computational Geometry* 42.3 (2009), pp. 183–195 (cit. on pp. 7, 9).

[CH17] J. Cardinal and U. Hoffmann. "Recognition and complexity of point visibility graphs". In: *Discrete & Computational Geometry* 57.1 (2017), pp. 164–178 (cit. on p. 93).

[Chi+15] C. Chilipirea, A.-C. Petre, C. Dobre, and M. Van Steen. “Proximity graphs for crowd movement sensors”. In: *10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC)*. 2015, pp. 310–314 (cit. on p. 6).

[Cim89] R. Cimikowski. “Coloring proximity graphs”. In: *Memoranda in Computer and Cognitive Science: Proceedings of the First Workshop on Proximity Graphs*. 1989, pp. 141–156 (cit. on p. 54).

[Cim90] R. Cimikowski. “Coloring certain proximity graphs”. In: *Computers & Mathematics with Applications* 20.3 (1990), pp. 69–82 (cit. on pp. 9, 95).

[Cim92] R. J. Cimikowski. “Properties of some Euclidean proximity graphs”. In: *Pattern Recognition Letters* 13.6 (1992), pp. 417–423 (cit. on pp. 9, 16, 18, 54).

[CKX09] J. Chen, I. A. Kanj, and G. Xia. “On parameterized exponential time complexity”. In: *Theoretical Computer Science* 410.27-29 (2009), pp. 2641–2648 (cit. on p. 37).

[CLR87] F. R. K. Chung, F. T. Leighton, and A. L. Rosenberg. “Embedding graphs in books: A layout problem with applications to VLSI design”. In: *SIAM Journal on Algebraic Discrete Methods* 8.1 (1987), pp. 33–58 (cit. on p. 18).

[DBLL94] G. Di Battista, W. Lenhart, and G. Liotta. “Proximity drawability: A survey (extended abstract)”. In: *International Symposium on Graph Drawing (GD '94)*. 1994, pp. 328–339 (cit. on pp. 9, 93).

[Dev88] L. Devroye. “The expected size of some graphs in computational geometry”. In: *Computers & Mathematics with Applications* 15.1 (1988), pp. 53–64 (cit. on p. 9).

[DGM09] L. Devroye, J. Gudmundsson, and P. Morin. “On the expected maximum degree of Gabriel and Yao graphs”. In: *Advances in Applied Probability* 41.4 (2009), pp. 1123–1140 (cit. on p. 9).

[Die17] R. Diestel. *Graph Theory*. Berlin, Heidelberg: Springer, 2017 (cit. on pp. 12, 96).

[Dim+18] S. Dimitriadis, L. Brindley, L. Evans, D. Linden, and K. Singh. “A novel, fast, reliable and data-driven method for simultaneous single-trial mining and amplitude–latency estimation based on proximity graphs and network analysis”. In: *Frontiers in Neuroinformatics* 12 (2018) (cit. on p. 6).

[EW95] P. Eades and S. Whitesides. “Nearest neighbour graph realizability is NP-hard”. In: *Latin American Symposium on Theoretical Informatics (LATIN '95)*. 1995, pp. 245–256 (cit. on p. 93).

[EW96] P. Eades and S. Whitesides. “The realization problem for Euclidean minimum spanning trees is NP-hard”. In: *Algorithmica* 16.1 (1996), pp. 60–82 (cit. on p. 93).

[Fár48] I. Fáry. “On straight-line representation of planar graphs”. In: *Acta Scientiarum Mathematicarum* 11 (1948), pp. 229–233 (cit. on p. 14).

[FHW97] S. P. Fekete, M. E. Houle, and S. Whitesides. “The wobbly logic engine: Proving hardness of non-rigid geometric graph representation problems”. In: *International Symposium on Graph Drawing (GD ’97)*. 1997, pp. 272–283 (cit. on p. 93).

[Flu+18] T. Fluschnik, D. Hermelin, A. Nichterlein, and R. Niedermeier. “Fractals for kernelization lower bounds”. In: *SIAM Journal on Discrete Mathematics* 32.1 (2018), pp. 656–681 (cit. on p. 18).

[Flu+20] T. Fluschnik, H. Molter, R. Niedermeier, M. Renken, and P. Zschoche. “Temporal graph classes: A view through temporal separators”. In: *Theoretical Computer Science* 806 (2020), pp. 197–218 (cit. on p. 88).

[GJ76] M. R. Garey and D. S. Johnson. “The rectilinear Steiner tree problem is NP-complete”. In: *SIAM Journal on Applied Mathematics* 32 (1976), pp. 826–834 (cit. on pp. 39, 47).

[GJ79] M. R. Garey and D. S. Johnson. *Computers and Intractability. A Guide to the Theory of NP-Completeness.* San Francisco: W. H. Freeman, 1979 (cit. on pp. 20, 25, 86).

[GJS76] M. R. Garey, D. S. Johnson, and L. Stockmeyer. “Some simplified NP-complete graph problems”. In: *Theoretical Computer Science* 1.3 (1976), pp. 237–267 (cit. on pp. 47, 48, 54).

[GJT76] M. R. Garey, D. S. Johnson, and R. Tarjan. “The planar Hamiltonian circuit problem is NP-complete”. In: *SIAM Journal on Computing* 5.4 (1976), pp. 704–714 (cit. on pp. 69, 85).

[Grü63] B. Grünbaum. “Grötzsch’s theorem on 3-colorings”. In: *Michigan Mathematical Journal* 10.3 (1963), pp. 303–310 (cit. on p. 54).

[GS15] A. C. Gyllensten and M. Sahlgren. “Navigating the semantic horizon using relative neighborhood graphs”. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP ’15)*. 2015, pp. 2451–2460 (cit. on p. 6).

[GS69] K. R. Gabriel and R. R. Sokal. “A new statistical approach to geographic variation analysis”. In: *Systematic Biology* 18.3 (1969), pp. 259–278 (cit. on pp. 7, 9).

[Hea85] L. S. Heath. “Algorithms for embedding graphs in books”. PhD thesis. University of North Carolina at Chapel Hill, 1985 (cit. on p. 19).

[Him+19] A.-S. Himmel, C. Hoffmann, P. Kunz, V. Froese, and M. Sorge. “Computational complexity aspects of point visibility graphs”. In: *Discrete Applied Mathematics* 254 (2019), pp. 283–290 (cit. on p. 6).

[Hua90] N. F. Huang. “A divide-and-conquer algorithm for constructing relative neighborhood graph”. In: *BIT* 30.2 (1990), pp. 196–206 (cit. on p. 9).

[HW12] S. Hoffmann and E. Wanke. "Metric dimension for Gabriel unit disk graphs is NP-complete". In: *International Symposium on Algorithms and Experiments for Sensor Systems, Wireless Networks and Distributed Robotics (ALGOSENSORS '12)*. 2012, pp. 90–92 (cit. on p. 9).

[IP01] R. Impagliazzo and R. Paturi. "On the Complexity of k-SAT". In: *Journal of Computer and System Sciences* 62.2 (2001), pp. 367–375 (cit. on p. 20).

[JK87] J. W. Jaromczyk and M. Kowaluk. "A note on relative neighborhood graphs". In: *Proceedings of the Third Annual Symposium on Computational Geometry (SCG '87)*. 1987, pp. 233–241 (cit. on p. 9).

[Joh84] D. S. Johnson. "The NP-completeness column: An ongoing guide". In: *Journal of Algorithms* 5.1 (1984), pp. 147–160 (cit. on p. 25).

[JT92] J. W. Jaromczyk and G. T. Toussaint. "Relative neighborhood graphs and their relatives". In: *Proceedings of the IEEE* 80.9 (1992), pp. 1502–1517 (cit. on p. 9).

[KKK02] D. Kempe, J. Kleinberg, and A. Kumar. "Connectivity and inference problems for temporal networks". In: *Journal of Computer and System Sciences* 64.4 (2002), pp. 820–842 (cit. on pp. 86, 88).

[KNT87] J. Katajainen, O. Nevalainen, and J. Teuhola. "A linear expected-time algorithm for computing planar relative neighbourhood graphs". In: *Information Processing Letters* 25.2 (1987), pp. 77–86 (cit. on p. 9).

[KO03] P. C. Kainen and S. Overbay. "Book embeddings of graphs and a theorem of Whitney". Tech. rep. Georgetown University, 2003 (cit. on p. 19).

[Lan69] P. M. Lankford. "Regionalization: Theory and alternative algorithms". In: *Geographical Analysis* 1.2 (1969), pp. 196–212 (cit. on pp. 7, 9).

[Lav+07] J. Lavergne, H. Azzag, C. Guinot, and G. Venturini. "Incremental construction of neighborhood graphs using the ants self-assembly behavior". In: *19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI '07)*. Vol. 1. 2007, pp. 399–406 (cit. on p. 9).

[Lee85] D. Lee. "Relative neighborhood graphs in the L_i-metric". In: *Pattern Recognition* 18.5 (1985), pp. 327–332 (cit. on p. 9).

[Lin94] A. Lingas. "A linear-time construction of the relative neighborhood graph from the Delaunay triangulation". In: *Computational Geometry* 4.4 (1994), pp. 199–208 (cit. on p. 9).

[LL97] W. Lenhart and G. Liotta. "Proximity drawings of outerplanar graphs (extended abstract)". In: *International Symposium on Graph Drawing (GD '96)*. 1997, pp. 286–302 (cit. on p. 9).

[LMS11] D. Lokshtanov, D. Marx, and S. Saurabh. "Lower bounds based on the exponential time hypothesis". In: *Bulletin of EATCS* 105 (2011), pp. 41–71 (cit. on pp. 20, 46, 47, 85, 95).

[Lov75] L. Lovász. “Three short proofs in graph theory”. In: *Journal of Combinatorial Theory, Series B* 19.3 (1975), pp. 269–271 (cit. on p. 54).

[LS93] A. Lubiw and N. Sleumer. “Maximal outerplanar graphs are relative neighbourhood graphs”. In: *Proceedings of the Fifth Canadian Conference on Computational Geometry (CCCG ’93)*. 1993, pp. 198–203 (cit. on p. 9).

[Mel13] O. Melchert. “Percolation thresholds on planar Euclidean relative-neighborhood graphs”. In: *Physical Review E* 87 (4 2013), p. 042106 (cit. on p. 9).

[Mit79] S. L. Mitchell. “Linear algorithms to recognize outerplanar and maximal outerplanar graphs”. In: *Information Processing Letters* 9.5 (1979), pp. 229–232 (cit. on p. 19).

[MM06] B. Milic and M. Malek. “Dropped edges and faces’ size in Gabriel and relative neighborhood graphs”. In: *IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS ’06)*. 2006, pp. 407–416 (cit. on p. 9).

[MM17] J. S. Mitchell and W. Mulzer. “Proximity algorithms”. In: *Handbook of Discrete and Computational Geometry*. Ed. by C. D. Toth, J. O’Rourke, and J. E. Goodman. New York: Chapman and Hall/CRC, 2017, pp. 849–874 (cit. on p. 9).

[MS80] D. W. Matula and R. R. Sokal. “Properties of Gabriel graphs relevant to geographic variation research and the clustering of points in the plane”. In: *Geographical Analysis* 12.3 (1980), pp. 205–222 (cit. on pp. 9, 16, 18).

[Nor16] C. Norrenbrock. “Percolation threshold on planar Euclidean Gabriel graphs”. In: *The European Physical Journal B* 89.5 (2016), p. 111 (cit. on p. 9).

[PKV16] C. Papageorgiou, P. Kokkinos, and E. Varvarigos. “Energy-efficient unicast and multicast communication for wireless ad hoc networks using multiple criteria”. In: *Mobile Ad Hoc Networks: Current Status and Future Trends*. Ed. by J. Loo, J. L. Mauri, and J. H. Ortiz. Boca Raton, London, New York: CRC Press, 2016, pp. 201–230 (cit. on p. 6).

[Rob+96] N. Robertson, D. Sanders, P. Seymour, and R. Thomas. “Efficiently four-coloring planar graphs”. In: *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC ’96)*. 1996, pp. 571–575 (cit. on p. 95).

[Rob+97] N. Robertson, D. Sanders, P. Seymour, and R. Thomas. “The four-colour theorem”. In: *Journal of Combinatorial Theory, Series B* 70.1 (1997), pp. 2–44 (cit. on p. 95).

[SC91] T.-H. Su and R.-C. Chang. “On constructing the relative neighborhood graphs in Euclidean K-dimensional spaces”. In: *Computing* 46.2 (1991), pp. 121–130 (cit. on p. 9).

[Sch13] M. Schaefer. “Realizability of graphs and linkages”. In: *Thirty Essays on Geometric Graph Theory*. Ed. by J. Pach. New York, Heidelberg: Springer, 2013, pp. 461–482 (cit. on p. 93).

[SK17] H. Sundar and P. Khurd. “Parallel algorithms for the computation of cycles in relative neighborhood graphs”. In: *46th International Conference on Parallel Processing (ICPP '17)*. 2017, pp. 191–200 (cit. on p. 9).

[Spe83] E. Speckenmeyer. “Untersuchungen zum Feedback Vertex Set Problem in ungerichteten Graphen”. PhD thesis. Universität Paderborn, 1983 (cit. on pp. 56, 68).

[Sup83] K. J. Supowit. “The relative neighborhood graph, with an application to minimum spanning trees”. In: *Journal of the ACM* 30.3 (1983), pp. 428–448 (cit. on p. 9).

[Tou80] G. T. Toussaint. “The relative neighbourhood graph of a finite planar set”. In: *Pattern Recognition* 12.4 (1980), pp. 261–268 (cit. on pp. 7, 9).

[UKG88] S. Ueno, Y. Kajitani, and S. Gotoh. “On the nonseparating independent set problem and feedback set problem for graphs with no vertex degree exceeding three”. In: *Discrete Mathematics* 72.1-3 (1988), pp. 355–360 (cit. on p. 68).

[Urq83] R. B. Urquhart. “Some properties of the planar Euclidean relative neighbourhood graph”. In: *Pattern Recognition Letters* 1.5 (1983), pp. 317–322 (cit. on pp. 9, 16, 18, 96).

[Vri+16] N. J. de Vries, A. S. Arefin, L. Mathieson, B. Lucas, and P. Moscato. “Relative neighborhood graphs uncover the dynamics of social media engagement”. In: *International Conference on Advanced Data Mining and Applications (ADMA '16)*. 2016, pp. 283–297 (cit. on p. 6).

[Wat10] D. Watanabe. “A study on analyzing the grid road network patterns using relative neighborhood graph”. In: *Ninth International Symposium on Operations Research and Its Applications (ISORA '10)*. 2010, pp. 112–119 (cit. on p. 6).

[Wil85] M. H. Williams. “A linear algorithm for colouring planar graphs with five colours”. In: *The Computer Journal* 28.1 (1985), pp. 78–81 (cit. on p. 96).

[WY07] P. Wan and C. Yi. “On the longest edge of Gabriel graphs in wireless ad hoc networks”. In: *IEEE Transactions on Parallel and Distributed Systems* 18.1 (2007), pp. 111–125 (cit. on p. 9).

[Xu+19] X. Xu, I. W. Tsang, X. Cao, R. Zhang, and C. Liu. “Learning image-specific attributes by hyperbolic neighborhood graph propagation”. In: *28th International Joint Conference on Artificial Intelligence (IJCAI '19)*. 2019, pp. 3989–3995 (cit. on p. 6).

[Yi+10] C. Yi, P. Wan, L. Wang, and C. Su. “Sharp thresholds for relative neighborhood graphs in wireless ad hoc networks”. In: *IEEE Transactions on Wireless Communications* 9.2 (2010), pp. 614–623 (cit. on p. 9).

[Zsc+20] P. Zschoche, T. Fluschnik, H. Molter, and R. Niedermeier. “The complexity of finding small separators in temporal graphs”. In: *Journal of Computer and System Sciences* 107 (2020), pp. 72–92 (cit. on p. 88).