TU WIEN Informatics

Erweiterung des Planaren Storyplan Problems

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Logic and Computation

eingereicht von

Maximilian Holzmüller, BSc.
Matrikelnummer 11770953

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Univ.Prof. Dipl.-Inform. Dr.rer.nat. Martin Nöllenburg
Mitwirkung: Univ.Ass. Dipl.-Ing. Alexander Dobler, BSc.

Wien, 24. Jänner 2025

Maximilian Holzmüller	Martin Nöllenburg

TU WIEN Informatics

Expanding the Planar Storyplan Problem

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Logic and Computation

by

Maximilian Holzmüller, BSc.
Registration Number 11770953

to the Faculty of Informatics

at the TU Wien

Advisor: Univ.Prof. Dipl.-Inform. Dr.rer.nat. Martin Nöllenburg
Assistance: Univ.Ass. Dipl.-Ing. Alexander Dobler, BSc.

Vienna, 24th January, 2025

Maximilian Holzmüller	Martin Nöllenburg

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

Erklärung zur Verfassung der Arbeit

Maximilian Holzmüller, BSc.

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.
Ich erkläre weiters, dass ich mich generativer KI-Tools lediglich als Hilfsmittel bedient habe und in der vorliegenden Arbeit mein gestalterischer Einfluss überwiegt. Im Anhang „Übersicht verwendeter Hilfsmittel“ habe ich alle generativen KI-Tools gelistet, die verwendet wurden, und angegeben, wo und wie sie verwendet wurden. Für Textpassagen, die ohne substantielle Änderungen übernommen wurden, habe ich jeweils die von mir formulierten Eingaben (Prompts) und die verwendete IT-Anwendung mit ihrem Produktnamen und Versionsnummer/Datum angegeben.

Wien, 24. Jänner 2025

Maximilian Holzmüller

Danksagung

Ich möchte all den Menschen danken, die mich auf meinem Weg begleitet und unterstützt haben. Sie haben es mir ermöglicht, dorthin zu gelangen, wo ich heute stehe.

Mein tiefster Dank gilt meiner Familie. Vor allem meinem verstorbenen Großvater, der meinen Werdegang stets mit großem Interesse und sichtbarem Stolz verfolgt hat. Meinen Eltern, die mir alle Möglichkeiten eröffnet und mich in jeder Hinsicht unterstützt haben. Meiner Großmutter, mit der ich selbst über die tiefgründigsten Themen diskutieren konnte, und meiner Schwester, durch die ich wertvolle Lektionen fürs Leben gelernt habe.

Von Herzen danke ich auch meiner geliebten Frau, die mich so annimmt, wie ich bin, und mich mit ihrer Liebe und Geduld jeden Tag aufs Neue stärkt.

Mein Dank gilt zudem meinem Betreuer, Prof. Martin Nöllenburg, sowie Alexander Dobler, die mir mit ihrem konstruktiven Feedback und ihren wertvollen Ratschlägen geholfen haben, dass diese Arbeit ihre heutige Form annehmen konnte.

Ohne die Zeit, Aufmerksamkeit und Unterstützung dieser Menschen wäre es mir nicht möglich gewesen, all dies zu erreichen. Ihnen allen gilt mein aufrichtiger Dank.

Acknowledgements

I want to express my heartfelt gratitude to everyone who has supported and guided me along the way, helping me get to where I am today.

My deepest thanks go to my family. To my late grandfather, who followed my journey with great pride and interest. To my parents, who have always supported me and given me every opportunity to succeed. To my grandmother, who shared countless deep and meaningful conversations with me, and to my sister, who has taught me invaluable life lessons.

A special thanks goes to my beloved wife, who loves and accepts me just as I am and supports me every day with her patience and kindness.

I am sincerely grateful to my supervisor, Prof. Martin Nöllenburg, and to Alexander Dobler for their constructive feedback and valuable insights, which have been instrumental in shaping this thesis.

None of this would have been possible without the time, effort, and support of these amazing people. To all of you, thank you from the bottom of my heart.

Kurzfassung

Diese Diplomarbeit untersucht Methoden zur Verbesserung des Verständnisses komplexer Graphen durch die Einführung neuer Varianten des Storyplan Problems. Es werden das Planare Geometrische Storyplan, das Planare Topologische k-StoryPlan, das Planare Geometrische k-StoryPlan, das Minimale Planare Topologische StoryPlan und das Minimale Planare Geometrische StoryPlan Problem untersucht, wobei die NP-Schwere all dieser Probleme nachgewiesen wird.

Ein Storyplan ist eine Methode, welche einen Graphen $G = (V, E)$ in mehrere Frames aufteilt, sodass jedes einen induzierten Teilgraphen von G repräsentiert. Die Frames sind chronologisch angeordnet, wobei in jedem Frame ein neuer Knoten erscheint. Diese Praxis erleichtert das Verständnis der Graphen, indem diese in kleinere, überschaubare Teile zerlegt werden. Allerdings ist das Aufteilen in mehrere Teilgraphen nicht ausreichend. Jedes Frame muss auch einfach zu interpretieren sein. Das motiviert die Entwicklung neuer Varianten des Storyplan Problems.

Unsere Arbeit baut auf der Forschung von Binucci et al. [2] auf. Diese haben das Planare Topologische StoryPlan Problem (PTOP-SP) eingeführt, wo jedes Frame eine planare Zeichnung sein muss und alle Kanten als Jordankurven dargestellt werden. Wir erweitern diese Grundlage durch die Einführung fünf neuer Varianten.

Beim Planaren Geometrischen StoryPlan Problem (PGEO-SP) muss die Zeichnung jedes Frames planar sein und alle Kanten müssen geradlinig dargestellt werden, um für ein besseres Verständnis der einzelnen Frames zu sorgen.
Das Planare Topologische k-StoryPlan Problem (PTOP-k-SP) und das Planaren Geometrischen k-StoryPlan Problem (PGEO-k-SP) haben die selben Einschränkungen für die Zeichnungen der einzelnen Frames wie PTOP-SP und PGEO-SP, erlauben jedoch, dass mehrere Knoten in einem Frame auftauchen dürfen, um die Gesamtanzahl der benötigten Frames zu reduzieren.
Das Minimale Planare Topologische StoryPlan Problem (MIN-PTOP-SP) und das Minimale Planaren Geometrischen StoryPlan Problem (MIN-PGEO-SP) wollen die Anzahl der Frames minimieren, um jeweils den kleinstmöglichen planaren topologischen beziehungsweise planaren geometrischen Storyplan zu erhalten.

Der Hauptbeitrag dieser Arbeit ist ein Beweis durch Gegenbeispiel, dass PGEO-SP und PTOP-SP zwei distinkte Probleme sind. Weiters zeigen wir, dass alle fünf Probleme NP-schwer sind und PTOP-k-SP sogar NP-vollständig ist.

Abstract

This thesis investigates methods for improving the comprehension of complex graphs by introducing new variations of the storyplan problem. We explore the Planar Geometric Storyplan, the Planar Topological k-StoryPlan, the Planar Geometric k-StoryPlan, the Minimal Planar Topological StoryPlan, and the Minimal Planar Geometric StoryPlan problems, providing novel insights into graph representation and establishing the NP-hardness of each of these problems.

A storyplan is a technique in which a complex graph $G = (V, E)$ is split into multiple frames, each representing an induced subgraph of G. The frames are ordered chronologically, with one new vertex appearing in each frame. This process helps simplify graph comprehension by breaking it into smaller, more manageable parts. However, simply dividing the graph into subgraphs may not be sufficient. Each frame must also be easy to interpret, which motivates the development of new variations of the storyplan problem.

We build on the work of Binucci et al. [2], who introduced the Planar Topological StoryPlan problem (PTOP-SP), where each frame must be a planar drawing and all edges are represented by Jordan arcs. Our contributions expand on this foundation by proposing five novel variations.

In the Planar Geometric StoryPlan problem (PGEO-SP), the drawing of each frame must be planar, and all edges must be embedded as straight-lines for better clarity of the single frames.
The Planar Topological k-StoryPlan problem (PTOP-k-SP) and the Planar Geometric k-StoryPlan problem (PGEO-k-SP) have the same restrictions on the embedding of each frame as PTOP-SP and PGEO-SP, respectively, but allow multiple vertices to appear in the same frame to reduce the total number of frames needed.
The Minimal Planar Topological StoryPlan problem (MIN-PTOP-SP) and the Minimal Planar Geometric StoryPlan problem (MIN-PGEO-SP) seek to minimize the number of frames to obtain the smallest possible planar topological or planar geometric storyplan, respectively.

The main contribution of this thesis is a proof by counterexample showing that PGEO-SP is distinct from PTOP-SP. We then prove that PGEO-SP is NP-hard. In addition, we show that PTOP-k-SP is NP-complete and PGEO-k-SP is NP-hard, for general values of k. Finally, we demonstrate that the minimization variants MIN-PTOP-SP and MIN-PGEO-SP are NP-hard.

Contents

CHAPTER 1

Introduction

1.1 Problem statement and motivation

Graphs are powerful tools for representing and abstracting relationships and structures in various domains, from social networks to biological systems. However, visualizing the entire graph structure at once can be overwhelming when dealing with complex graphs. A practical method to make these graphs more comprehensible is to break them down into smaller, more manageable components. The concept of a storyplan provides a solution to this problem, where a graph is decomposed into a series of frames, each representing an induced subgraph on the vertices of the original graph. The frames have a chronological order, each introducing one new vertex, thus simplifying the visualization process.

In a storyplan of a given graph $G = (V, E)$, each vertex and edge of G must appear in at least one frame. The frames follow a strict total order, with the restriction that vertices can only appear once and must remain present until all edges incident to them have been introduced. Furthermore, the embeddings of vertices and edges over multiple frames cannot change. While this technique significantly aids comprehension, it is not enough to merely split a graph into subgraphs and draw them separately. The challenge lies in ensuring that each individual frame is easy to understand while maintaining the overall structure of the graph. To better understand the concept of storyplans, see Figure 1.1, which shows an example of a storyplan for the well-known graph $K_{3,3}$ and illustrates how a graph is revealed incrementally across multiple frames.

The original Planar StoryPlan problem, introduced by Binucci et al. [2], imposes additional constraints: every frame must be planar, and edges must be represented as Jordan arcs. This results in a simplified representation of complex graphs that generalizes planarity. However, we argue that further improvements can be made. In this thesis, we present five unexplored variations of the Planar StoryPlan problem intending to simplify graph representation further and examine the computational complexity of these five problems.

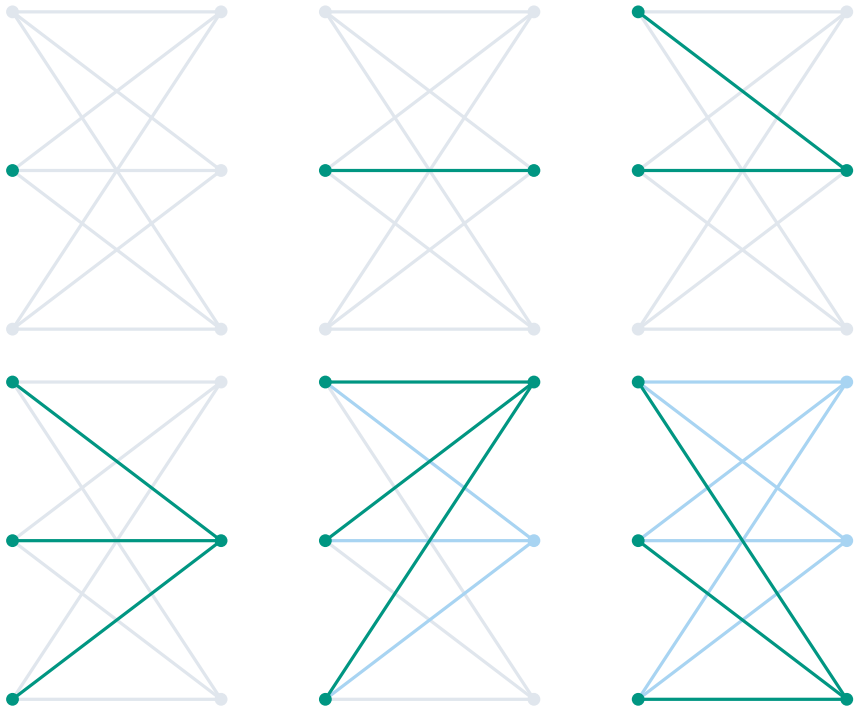


Figure 1.1: An example of a planar storyplan of the graph $K_{3,3}$, from left to right, top to bottom. Each frame shows one new vertex getting activated (green). Vertices and edges in light grey are yet to be activated, while light blue represents those that have already disappeared.

From now on, we will refer to the original planar storyplan problem as the Planar Topological StoryPlan problem (PTOP-SP) to achieve a clear and intuitive distinction between it and the new problem variants we introduce in this thesis. These variations are the following:

- The Planar Geometric StoryPlan problem (PGEO-SP): A stricter version of PTOP-SP where all edges are drawn as straight-lines, instead of as Jordan arcs. The simplification of the edges leads to better readable frames. This problem is the main subject of this thesis.
- The Planar Topological k-StoryPlan problem and the Planar Geometric k-StoryPlan problem: Versions of PTOP-SP and PGEO-SP that limit the total number of frames by letting multiple vertices appear at the same time.
- The Minimal Planar Topological StoryPlan problem and the Minimal Planar Geometric StoryPlan problem: Versions of PTOP-SP and PGEO-SP that ask for the minimum number of frames required to represent the graph accurately as a planar topological or planar geometric storyplan, respectively.

1.2 State of the art

The Planar StoryPlan problem is a relatively new problem; so far, only a handful of important results and variants have been established.

The primary influence on this thesis is the noteworthy paper by Binucci et al. [2], which introduced the Planar StoryPlan problem. They showed that the Planar StoryPlan problem is NP-complete, using a reduction from One-In-Three 3SAT to show NP-hardness. Furthermore, they provided two fixed-parameter tractable (FPT) algorithms, one based on the vertex cover number and the other based on the feedback edge set number. The vertex cover number refers to the smallest set of vertices that covers all edges in the graph, while the feedback edge set number is the smallest number of edges whose removal makes the graph acyclic.

Furthermore, Binucci et al. established that for partial 3-trees (i.e., graphs with treewidth at most 3), a planar storyplan always exists and can be computed in linear time. Finally, they introduced a variant of the Planar StoryPlan problem where the total order of vertices is predetermined and showed that this variant is also NP-complete.

Multiple open problems are also mentioned in the paper by Binucci et al. as well, among these other versions of the StoryPlan problem. One of them is the restriction of the edges to only allow straight-line drawings for them instead of Jordan arcs. Precisely, the main problem that we explore in this thesis.

Another notable contribution to the research of storyplans is from Fiala et al. [7], who introduced and studied forest and outerplanar storyplans. In these storyplan types, each frame must be a drawing of a planar forest (a collection of trees) or an outerplanar drawing (a planar embedding of a graph where all vertices lie on the outer face of the drawing), respectively. They provided efficient algorithms for the construction of geometric storyplans for certain graph families, where such storyplans always exist. Furthermore, Fiala et al. established and proved a chain of strict containment relationships between different graph classes. In particular, the classes that admit forest, outerplanar, and planar storyplans, as well as the class that includes all graphs:

$$G_{forest} \subsetneq G_{outerplanar} \subsetneq G_{planar} \subsetneq G$$

They also identified specific graph classes that always admit a geometric outerplanar storyplan or a geometric forest storyplan, respectively.

In a similar line of research, Borrazzo et al. [4] introduced the concept of graph stories, which bear similarities to storyplans in that they also use multiple drawings of induced subgraphs of a given graph G but differ in that they have a fixed window given by an integer w, where one vertex is active. Precisely, this fixed window entails that some edges of the graph G may never appear in any of the drawings of a graph story. Borrazzo et al. focused on finding upper bounds for grid sizes that allow all drawings of graph stories of paths, respectively, trees to be planar and straight-line.

The same graph story definition was extended by Di Battista et al. [6] in their paper. They formulate the problem of drawing a graph story by mapping the vertices only to $w + k$ given points, where w is again an integer representing the fixed window of drawings in which each vertex is active and k is an optimization integer that should be as small as possible. They showed that even for constant values of k, the problem remains NP-hard and FPT when parameterized by $w + k$. They also defined different families of graph stories and established several important properties for these families.

Schaefer's work [12] delves into various notions of planarity, including outerplanarity and simultaneous planarity. A simultaneous drawing is a drawing of two or more graphs, which might share a common subset of vertices and edges that contains all the vertices and edges of all these graphs. If the drawings of the individual graphs maintain planarity, while the only line crossings in the simultaneous drawing are those between edges of different graphs, then the graphs are simultaneously planar. This problem is of particular interest in the context of the StoryPlan problem, as it involves multiple independent graphs and focuses on identifying commonalities between them. This idea is similar to the relationship between neighboring frames in a storyplan.

Lastly, Fink et al. [8] explored the complexity of simultaneous planarity, proving its NP-hardness and providing several FPT algorithms for drawing simultaneous planar graphs.

1.3 Contribution

In this thesis, we expand upon existing research by introducing new variants of the StoryPlan problem and investigating their computational complexity. Our primary contribution is the introduction and study of the Planar Geometric StoryPlan problem, which serves as the central focus of this work.

The core research question we explore is whether the Planar Geometric StoryPlan problem is distinct from the Planar Topological StoryPlan problem, mainly if they define the same set of yes-instances. For planar graphs, it is well known through Fáry's theorem [9] that every planar graph has a planar straight-line embedding. However, whether this also applies to planar storyplans is not immediately apparent. To answer this, we investigate in Chapter 3 (*Subset relation of the Planar Geometric StoryPlan problem*) whether every graph that admits a planar topological storyplan also admits a planar geometric storyplan. The reverse direction is trivial, as any geometric drawing is inherently a valid topological drawing. So, every graph that admits a planar geometric storyplan also automatically admits a planar topological storyplan, as any geometric drawing of a frame is also a valid topological drawing of the same frame. Therefore, our research focuses primarily on determining whether the relation between the set of yes-instances of both problems is a proper subset relation.

We demonstrate that the Planar Geometric StoryPlan problem is indeed distinct by providing a concrete example of a graph that is a yes-instance for the Planar

Topological StoryPlan problem but a no-instance for the Planar Geometric StoryPlan problem. This shows that these two problems have different sets of graphs as their affirmative instances.

Once we establish that the Planar Geometric StoryPlan problem defines a unique set of yes-instances, we turn to its computational complexity. In Chapter 4 (*NP-hardness of PGEO-SP*), we show that the Planar Geometric StoryPlan problem remains NP-hard. This result is attained by an adaptation of the NP-hardness proof of the Planar Topological StoryPlan problem that was presented in [2] to contain only geometric constructions.

In Chapter 5 (*Further Results*), the final part of this thesis, we introduce optimization variants of the StoryPlan problem. We first define the Planar k-StoryPlan problem, which asks, given a graph G and an integer k, with $k \leq n$, whether G admits a storyplan with exactly k frames. This question is feasible as a k-storyplan allows multiple vertices to appear in the same frame. Subsequently, we define the Minimal StoryPlan problem as the problem that asks for the smallest integer k such that a given graph G still admits a k-storyplan. Based on these two definitions, we define the Planar Geometric k-StoryPlan problem, the Planar Topological k-StoryPlan problem, the Minimal Planar Topological StoryPlan problem and the Minimal Planar Geometric StoryPlan problem as variants of the Planar k-StoryPlan problem and the Minimal StoryPlan problem using storyplans, where all drawings are planar and the edges are drawn as Jordan arcs (for the topological variants) or straight-lines (for the geometric variants), respectively. We establish that for general values of k, the Planar Topological k-StoryPlan problem is NP-complete and the Planar Geometric k-StoryPlan problem is NP-hard, while the optimization variants, the Minimal Planar Topological StoryPlan problem and the Minimal Planar Geometric StoryPlan problem, are both NP-hard.

CHAPTER 2

Preliminaries

Before delving into the core of the thesis, we must first introduce several crucial formal definitions to understand the subsequent proofs and results.

Graph Drawing and Planarity. A *drawing* Γ of a graph $G = (V, E)$ is a mapping that assigns each vertex in V to a point in $\mathbb{R}^2$, and each edge in E to a Jordan arc connecting its two corresponding endpoints, without passing through any other vertices. The drawing Γ is *planar* if no edges cross each other and *geometric* if each edge is drawn as a straight-line between its endpoints.

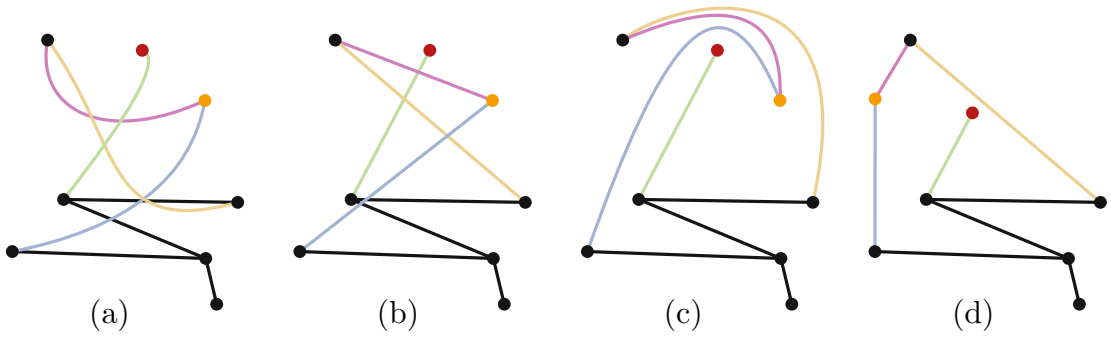


Figure 2.1: This Figure illustrates the differences between (a) a non-planar drawing, (b) a non-planar geometric drawing, (c) a planar drawing and (c) a planar geometric drawing. All four illustrations are drawings of the same graph. The colors help visualize the vertices and edges in the different drawings.

A graph G is considered *planar* if it admits a planar drawing, i.e., if some drawing Γ of G exists in which no edges cross. Each planar drawing divides the plane into disjoint regions called *faces*. The infinite region is called the *outer face*. A *planar embedding* $\mathcal{E}$ of

G is an equivalence class of planar drawings that define the same faces and share the same outer face.

Graph Terminology. We use the following terminology to describe relationships between vertices in a graph:

For a graph $G = (V, E)$ and two vertices $a, b \in V$ we say that a vertex a can *see* (or *reach*, or *is connected to*, or *share an edge with*, or *has a common edge with*) another vertex b in graph $G = (V, E)$, if G has an edge $e = \{a, b\}$.

Let $V(f)$ denote the set of vertices that lie on the boundary of a face f in a planar drawing of a graph.

Remark 2.1. The set $V(f)$ represents the vertices that any new vertex v, placed inside face f, can reach using topological edges without violating planarity. If the face f is convex, this also holds for geometric edges.

We define $[n]$ as an abbreviation for the set $\{1, 2, \ldots, n\}$.

Let $N[v]$ denote the neighborhood of a vertex v, including v itself, i.e., the set containing v and all its adjacent vertices.

Storyplan. The formal definition of a storyplan adapted from previous work by Binucci et al. and by Fiala et al.

Definition 1. *Let $G = (V, E)$ be a graph with n vertices. A planar storyplan $S = \langle \tau, \{D_i\}_{i\in[n]} \rangle$ of G is a pair defined as follows.*
The first element is a bijection $\tau : V \to [n]$ that imposes a strict total order on the vertices of G based on their appearance in the storyplan S. For each vertex $v \in V$, let $i_v = \tau(v)$ and let $j_v = max_{u\in N[v]}\tau(u)$. The interval $[i_v, j_v]$ is the lifespan of v. We say that v appears at step i_v, is visible at step i for each $i \in [i_v, j_v]$, and disappears at step $j_v + 1$. Note that a vertex disappears only once all its neighbors have appeared.
The second element of S is a sequence of drawings $\{D_i\}_{i\in[n]}$, called the frames of S. For each $i \in [n]$ the drawing D_i must satisfy the following conditions:
i) D_i is a drawing of the graph G_i induced by the vertices visible at step i, ii) D_i is planar, iii) the point representing a vertex v is the same over all drawings that contain v, and iv) the curve representing an edge e is the same over all drawings that contain e.

The Planar Topological StoryPlan problem then asks, given a graph $G = (V, E)$, does G admit a planar (topological) storyplan?

For the *planar geometric storyplan*, we add the following condition that must hold for each drawing D_i with $i \in [n]$: *v) the curve representing an edge e must be a straight-line.*

The Planar Geometric StoryPlan problem then asks, given a graph $G = (V, E)$, does G admit a planar geometric storyplan?

Complete Bipartite Graphs. A graph $G = (V, E)$ is called a *complete bipartite graph*, if its vertices can be split into two sets V_1 and V_2, such that $V_1 \cap V_2 = \emptyset$, and for all vertices holds $v \in V_i : N[v] = V_{\{1,2\}\setminus\{i\}}$ for $i \in \{1, 2\}$. So, E contains all possible edges over the vertices in V_1 and V_2, for which holds that one endpoint is in V_1 and the other in V_2, and no other edges.

To efficiently work with complete bipartite graphs in Chapter 4 (*NP-hardness of PGEO-SP*), we need the following two results from the work of Binucci et al. [2]. The proof of the lemma can be found in the same paper.

Lemma 2.1. *Let $K_{a,b} = (A \cup B, E)$ be a complete bipartite graph with $a = |A|$, $b = |B|$, and $3 \leq b \leq a$. Let $S = \langle \tau, \{D_i\}_{i\in[a+b]\rangle}$ be a storyplan of $K_{a,b}$. Then, exactly one of A or B is such that all its vertices are visible at some $i \in [a + b]$.*

Definition 2. *For a complete bipartite graph $K_{a,b}$ with $3 \leq b \leq a$ and a storyplan S of $K_{a,b}$, we call fixed the partite set of $K_{a,b}$ whose vertices are all visible at some step of S, and flexible the other partite set.*

Remark 2.2. For a complete bipartite graph $K_{a,b}$ with $3 \leq b \leq a$ and a storyplan S, precisely one side of $K_{a,b}$ must be fixed in S and the other must be flexible in S.

CHAPTER 3

Subset relation of the Planar Geometric StoryPlan problem

In this chapter, we will show that the Planar Topological StoryPlan problem and the Planar Geometric StoryPlan problem describe different sets of yes-instances. By definition, we know that the set of graphs that are yes-instances of PGEO-SP is a subset of the graphs that are yes-instances of PTOP-SP. To show that this is a proper subset relation, we will show the existence of a graph G that has a solution in PTOP-SP, but for which no valid PGEO-SP drawing can exist. We will start by defining G, then we will give an example of a PTOP-SP drawing of G, and subsequently, we will show why G cannot be drawn as an instance of the Planar Geometric StoryPlan problem. For the counterexample to the assumption that both problems are equal, we will first create a construction with the help of some assumptions that cannot be drawn with straight-lines without introducing intersections, then show why this construction must always appear in one frame of every geometric storyplan drawing of G, and at last demonstrate why we can remove each of the helping assumptions. Thereby proving that G is a no-instance of PGEO-SP.

3.1 The definition of the counterexample graph G

To establish an understanding of the counterexample graph G that will show the difference between the instances spaces of PTOP-SP and PGEO-SP, we will first describe the graph in terms of the structures that were used to build the graph, and then we will give formal definitions of G, one written and one visual.

3.1.1 The structure

Our graph $G = (V, E)$ contains two quadrangles $Q = (\{v_1, v_2, v_3, v_4\}, \{e_1, e_2, e_3, e_4\})$ and $Q' = (\{v'_1, v'_2, v'_3, v'_4\}, \{e'_1, e'_2, e'_3, e'_4\})$, where v_1, v_2, v_3, v_4 are the vertices and e_1, e_2, e_3, e_4 the edges of the quadrangle Q, and v'_1, v'_2, v'_3, v'_4 are the vertices and e'_1, e'_2, e'_3, e'_4 are the edges of the quadrangle Q'. The defining factor of the graph is the existence of eight additional structures S_i, also called the apex-structures, with the following structure: one vertex a^q_i, also called the quadrangle-apex, as it is connected to all vertices of both quadrangles and another vertex a^e_i, also called the edge-apex, that is connected to a^q_i and two vertices of each quadrangle, also called $v_{\phi(i)}$ ($v'_{\phi(i)}$) and $v_{\phi(i+1)}$ ($v'_{\phi(i+1)}$) of the quadrangles Q (Q') respectively. Both of these vertices are incident to the same edge $e_{\phi(i)}$ ($e'_{\phi(i)}$) of the quadrangle Q (Q'). The function $\phi : \mathbb{N} \to [4]$ maps the indices of the structures S_i to the indices of the vertices and edges of the quadrangles: $\phi(i) = ((i-1) \bmod 4) + 1$.

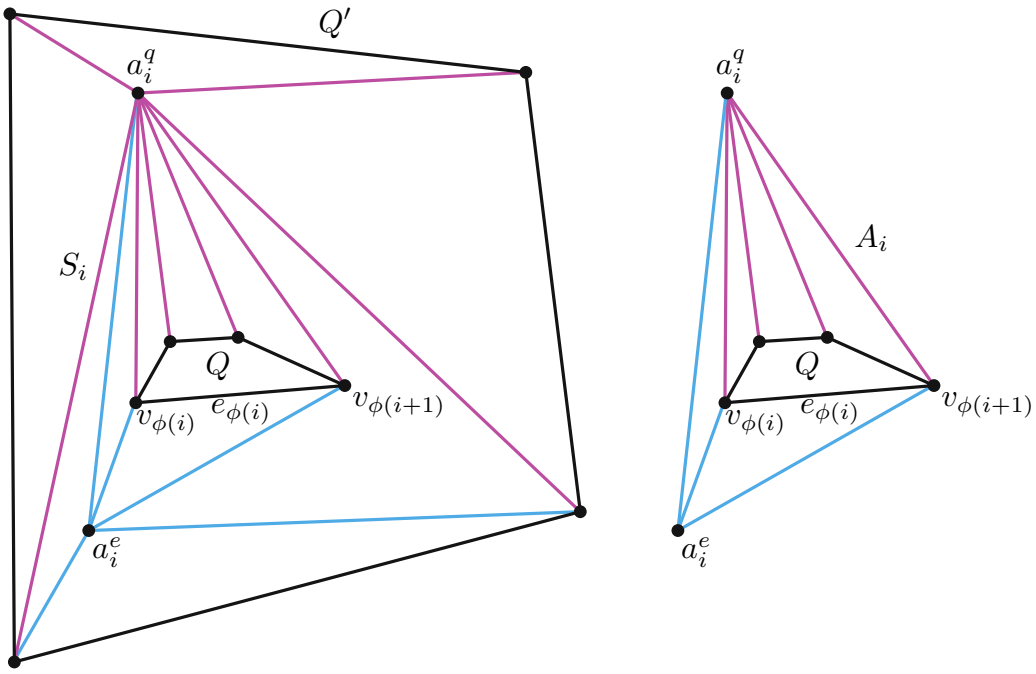
There are four individual structures S_i with $i \in \{1, 2, 3, 4\}$, each one defined by different edges $e_{\phi(i)}$ and $e'_{\phi(i)}$ of the quadrangles Q and Q'. The other four structures S_i with $i \in \{5, 6, 7, 8\}$, have the same neighborhoods of adjacent vertices in Q and Q' as the first four structures, based on the function ϕ, so the following statements hold: $N_G(a^q_i) \setminus a^e_i = N_G(a^q_{i-4}) \setminus a^e_{i-4}$ and $N_G(a^e_i) \setminus a^q_i = N_G(a^e_{i-4}) \setminus a^q_{i-4}$ for all $i \in \{5, 6, 7, 8\}$. Due to this similarity between each structure S_i and S_{i+4} for $i \in \{1, 2, 3, 4\}$, we will call these structures that have the same neighborhoods concerning the vertices of the quadrangles structure siblings. For most of our proof, we will talk about the first four structures S_i with $i \in [4]$, but we will need their siblings later in the proof.

The edges between one of the quadrangle-apex vertices a^q_i and the vertices of quadrangle Q (Q') will be annotated with E_{Q,a^q_i} (E_{Q',a^q_i}), similarly, the edges between one of the edge-apex vertices a^e_i and the vertices $v_{\phi(i)}$ ($v'_{\phi(i)}$) and $v_{\phi(i+1)}$ ($v'_{\phi(i+1)}$) will be annotated with $E_{e_{\phi(i)},a^e_i}$ ($E_{e'_{\phi(i)},a^e_i}$). Figure 3.1(a) shows an example of this structure.

The apex-triangle A_i is a planar drawing of a subgraph of G consisting only of the two apex vertices a^q_i and a^e_i and one of the quadrangles Q or Q' and all their induced edges. The drawing is of such a form that both apex vertices are part of the outer face of A_i, and they lie in different half-spaces created by the edge $e_{\phi(i)}$ or $e'_{\phi(i)}$ depending on the used quadrangle. The outer face of A_i always has a triangular shape, hence the name apex-triangle. Figure 3.1(b) depicts an example of an apex-triangle.

3.1.2 Formal representations

Based on the definition of the structures that build G, we will now define G in terms of an adjacency table 3.1 and a visual graph representation in Figure 3.2. In the visual graph representation, every structure S_i with $i \in \{1, 2, 3, 4\}$ has its unique color to make it more recognizable. S_1 is blue colored, S_2 is green colored, S_3 is purple colored, and S_4 is brown colored. The structures S_i with $i \in \{5, 6, 7, 8\}$ are not drawn explicitly but are also represented by their siblings.



(a) (b)

Figure 3.1: (a) A straight-line drawing of an induced subgraph of G, showing the two quadrangles Q and Q' and one of the structures S_i. The two quadrangles Q and Q' are black, and the edges of the structure S_i is presented in colors. The edges E_{Q,a_i^q} and E_{Q',a_i^q} between the quadrangle-apex a_i^q and the vertices of both quadrangles are magenta, and the edges incident to a_i^e are blue (those are the edges $E_{e_{\phi(i)},a_i^e}$, $E_{e'_{\phi(i)},a_i^e}$, and a_i). (b) An apex-triangle A_i consists of the apex vertices a_i^q and a_i^e, one quadrangle Q, and all their induced edges. The defining factor of the apex-triangle is that both apex vertices must lie in the outer face of this drawing.

For the representation via an adjacency table, we will define some additional vertex sets: Q_a is the set of all quadrangle-apex vertices, and Q_v is the set of all quadrangle vertices. Based on its definition, G has 24 vertices, 152 edges, a minimum degree of five, and a maximum degree of fourteen.

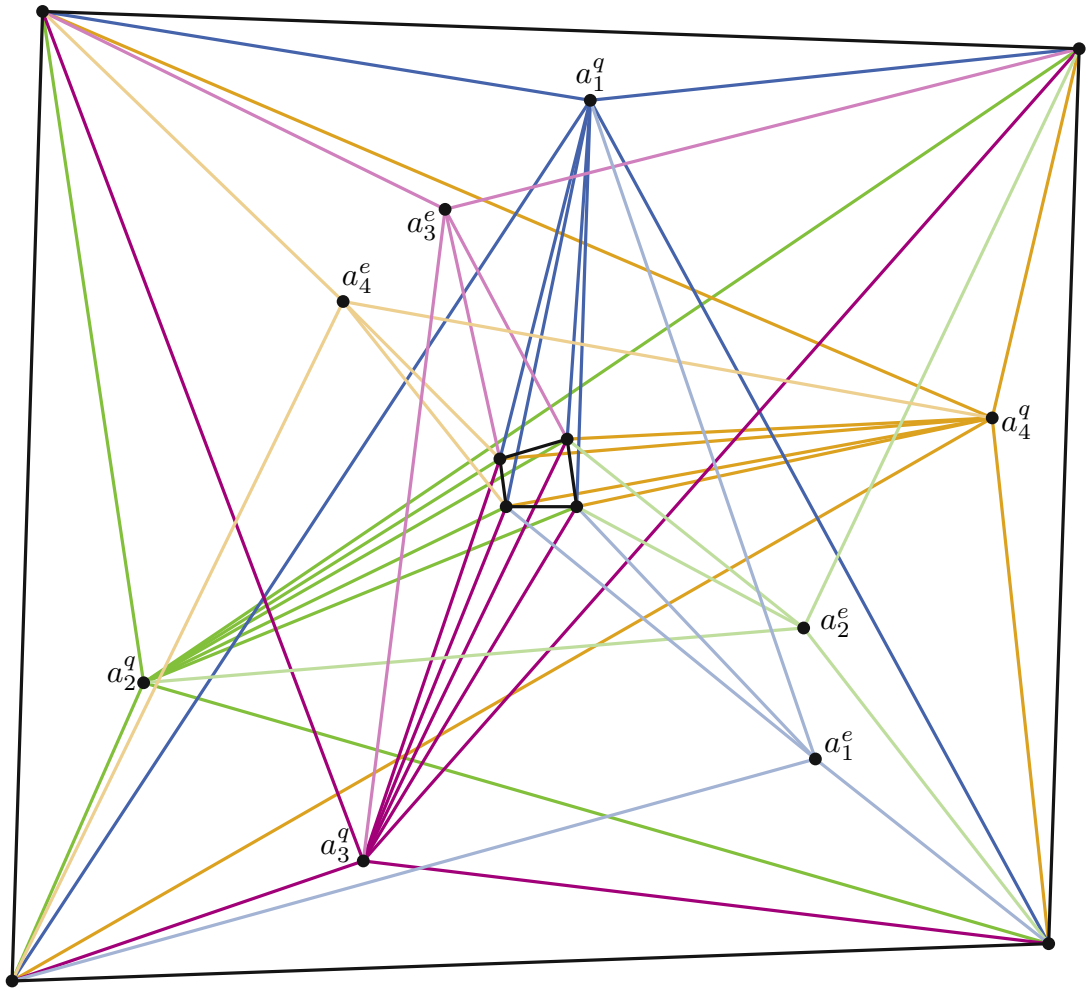


Figure 3.2: Graph G, the two quadrangles Q and Q' are black, and every structure S_i for $i \in [4]$ has its unique color. The sibling structures are omitted in the drawing, as the coordinates of the embedding of their vertices and edges can be identical to the coordinates of the embeddings of the vertices and edges of the four structures shown in this figure.

Vertex	Adjacent vertices						
v_1	v_2	v_4	Q_a	a_1^e	a_4^e	a_5^e	a_8^e
v_2	v_1	v_3	Q_a	a_1^e	a_2^e	a_5^e	a_6^e
v_3	v_2	v_4	Q_a	a_2^e	a_3^e	a_6^e	a_7^e
v_4	v_1	v_3	Q_a	a_3^e	a_4^e	a_7^e	a_8^e
v_1'	v_2'	v_4'	Q_a	a_1^e	a_4^e	a_5^e	a_8^e
v_2'	v_1'	v_3'	Q_a	a_1^e	a_2^e	a_5^e	a_6^e
v_3'	v_2'	v_4'	Q_a	a_2^e	a_3^e	a_6^e	a_7^e
v_4'	v_1'	v_2'	Q_a	a_3^e	a_4^e	a_7^e	a_8^e
a_1^q	Q_v	a_1^e					
a_1^e	v_1	v_2	v_1'	v_2'	a_1^q		
a_2^q	Q_v	a_2^e					
a_2^e	v_2	v_3	v_2'	v_3'	a_2^q		
a_3^q	Q_v	a_3^e					
a_3^e	v_3	v_4	v_3'	v_4'	a_3^q		
a_4^q	Q_v	a_4^e					
a_4^e	v_1	v_4	v_1'	v_4'	a_4^q		
a_5^q	Q_v	a_5^e					
a_5^e	v_1	v_2	v_1'	v_2'	a_5^q		
a_6^q	Q_v	a_6^e					
a_6^e	v_2	v_3	v_2'	v_3'	a_6^q		
a_7^q	Q_v	a_7^e					
a_7^e	v_3	v_4	v_3'	v_4'	a_7^q		
a_8^q	Q_v	a_8^e					
a_8^e	v_1	v_4	v_1'	v_4'	a_8^q		

Table 3.1: Table representation of G. In the leftmost column, all vertices of G are listed. To the right, all vertices adjacent to that respective vertex are listed. (Q_a is the set of all quadrangle-apex vertices, and Q_v is the set of all quadrangle vertices.)

3.2 Yes-instance of PTOP-SP

We can easily find a valid planar topological storyplan of the graph G by splitting G into eight frames. Every frame contains the quadrangles Q and Q' and exactly one of the structures S_i. this can be seen in the Figures 3.3, 3.4, 3.5, and 3.6, as well as in the respective Tables in Appendix A. Every figure shows two frames at once, as each structure and its sibling have the same neighborhood relations with the vertices of the quadrangles and, therefore, can be drawn in the same manner.

With these figures and the ordering of the frames, we have shown an example of a planar topological storyplan of G. The existence of such a storyplan proves that G is indeed a yes-instance of PTOP-SP.

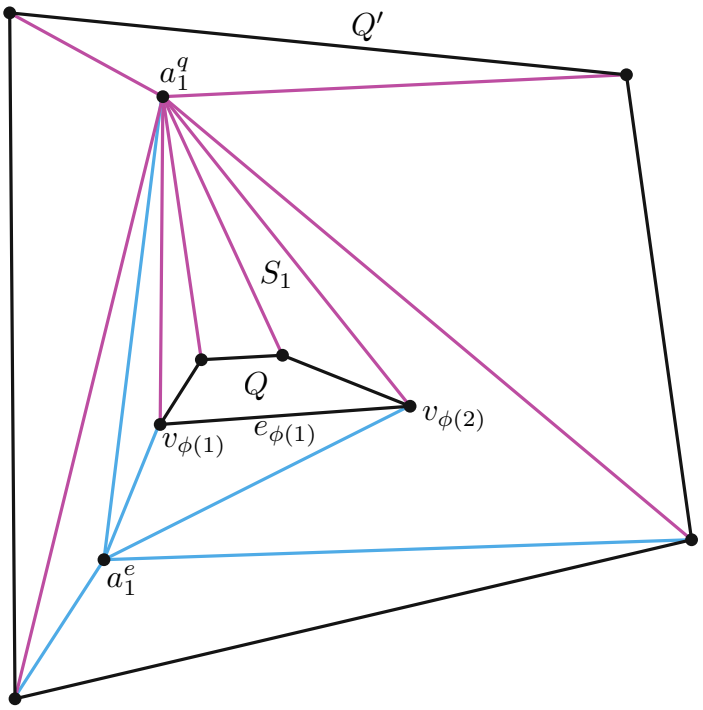


Figure 3.3: The first two frames of graph G. In the second frame, the structure S_5 can be placed in the same position as S_1. Resulting in an equivalent frame, the only difference being that the apex vertices a_5^q and a_5^e are used instead of a_1^q and a_1^e.

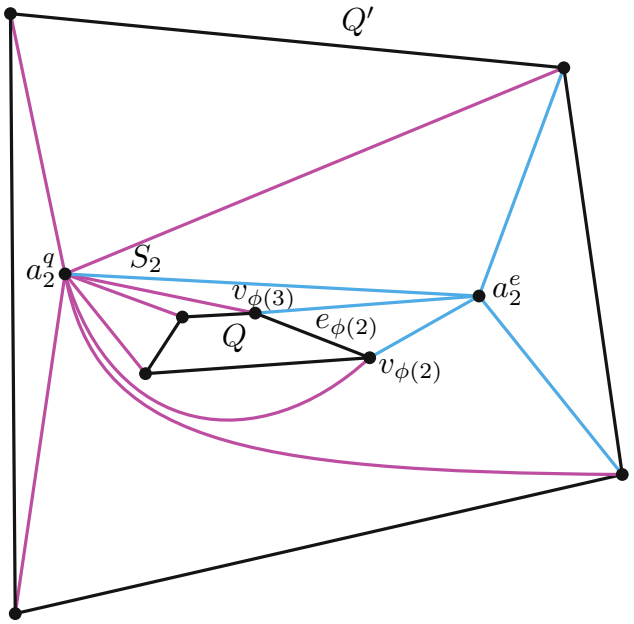


Figure 3.4: The graph's G third and fourth frame. The structure S_6 replaces S_2 in the fourth frame.

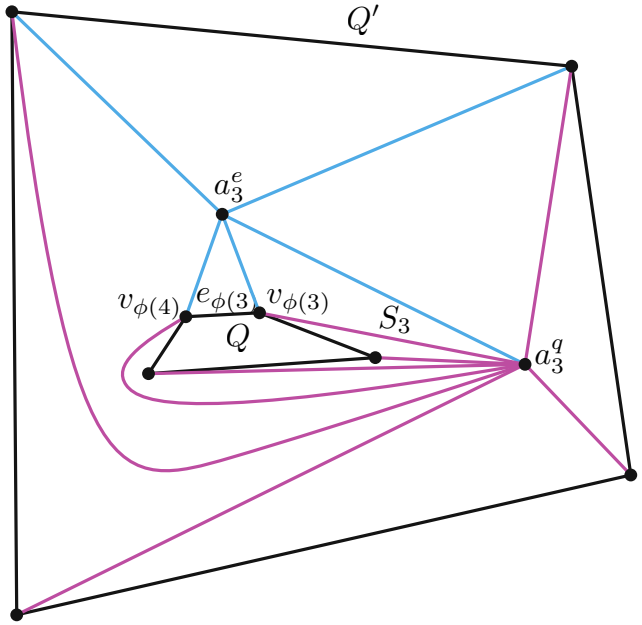


Figure 3.5: The fifth and sixth frames of the graph G. The structure S_7 replaces S_3 in the sixth frame.

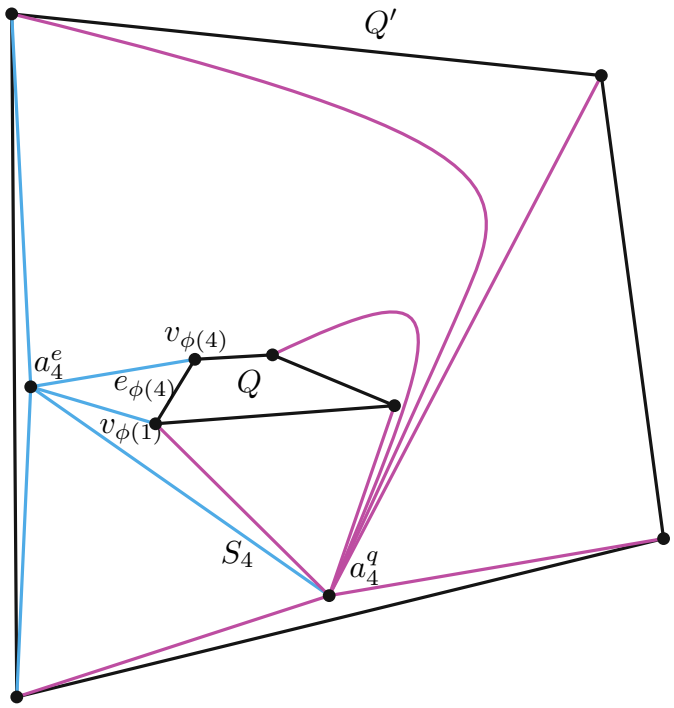


Figure 3.6: The seventh and eighth frames of the graph G. The structure S_8 replaces S_4 in the eighth frame.

3.3 No-instance of PGEO-SP

After establishing that a planar topological storyplan exists for G, we now have to show that no sequence of frames of G can exist that would form a valid planar geometric storyplan to prove that G is a no-instance of PGEO-SP. We start by creating a contradiction using a straight-line non-crossing drawing of both quadrangles Q and Q' as well as one of the apex-structures S_i in one frame, and then we argue why this construction must appear in at least one frame of every possible planar geometric storyplan of G. Finally, we will deal with assumptions and special cases that we omitted in the first part of the proof.

3.3.1 Towards a contradiction

As mentioned, our first step towards a contradiction is to show that a planar straight-line drawing of one apex-structure S_i and the quadrangles Q and Q' can create an impossible solution. Therefore, all lemmas in this subsection, from Lemma 3.1 to Lemma 3.7, only concern planar straight-line drawings.

To start arguing, we will begin with the assumptions that one of the quadrangles lies inside the other and that the inner quadrangle forms a convex polygon without any parallel edges. Later, we will show why these restrictions are unnecessary, but in the beginning, they help us make the main points of our counterexample understandable without losing ourselves in edge cases. Note that an assumption is only applied to a Lemma if explicitly mentioned. This will allow us to reuse some of the Lemmas later in this proof.

We start our construction by starting from the outside of the graph, moving inwards, going through all possible placements of the vertices, and fixing outer vertices first before moving towards inner vertices. Firstly, we will show that starting from two quadrangles placed inside each other, we will end at the apex-triangle A_i for our structure S_i and its connection to the inner quadrangle. Then, we will show that from the apex-triangle A_i, we will always have two structures S_i, for which no planar geometric embedding exists.

Due to the symmetric structure of G, we can assume w.l.o.g. that for the remainder of this proof the inner quadrangle is $Q = (\{v_1, v_2, v_3, v_4\}, \{e_1, e_2, e_3, e_4\})$ and the outer quadrangle is $Q' = (\{v'_1, v'_2, v'_3, v'_4\}, \{e'_1, e'_2, e'_3, e'_4\})$.

Lemma 3.1. *Given an embedding of two quadrangles Q and Q' that do not intersect and a structure S_i as defined above, then all vertices of S_i must be placed in the face between the two quadrangles.*

Proof. With the given embedding of Q and Q', they define three faces in the plane, as every quadrangle is a cyclic graph of four vertices that splits the face into which it is drawn into two new faces. Let us call these faces f_Q, $f_{between}$, and $f_{Q'}$, where f_Q is the face that is only incident to Q, $f_{between}$ is the face between the two quadrangles, and $f_{Q'}$ is the face only incident to Q'.

Every structure S_i contains two vertices a_i^q and a_i^e with a common edge a_i. Therefore, both of these vertices must be placed in the same face. Otherwise, the edge a_i would cross one of the edges of one of the quadrangles, making the drawing non-planar. Furthermore, both vertices of S_i have connections to both quadrangles.

We will use the vertex sets of the faces defined by the two quadrangles Q and Q' to show in which face a_i^q and a_i^e must be placed.

$$V(f_Q) = \{v_1, v_2, v_3, v_4\}$$
$$V(f_{between}) = \{v_1, v_2, v_3, v_4, v_1', v_2', v_3', v_4'\}$$
$$V(f_{Q'}) = \{v_1', v_2', v_3', v_4'\}$$

All of the vertices in the neighborhood of the quadrangle-apex a_i^q must be reachable from within the face where a_i^q is placed. Placing a_i^q in a face where one of its neighboring vertices cannot be reached must again introduce an intersection with one of the edges of one of the quadrangles, destroying the planarity property of the resulting drawing. As $V(f_{between}) \cup \{a_i^q, a_i^e\} = N[a_i^q]$, whereas $V(f_Q) \cup \{a_i^q, a_i^e\} \subset N[a_i^q]$ and also $V(f_{Q'}) \cup \{a_i^q, a_i^e\} \subset N[a_i^q]$, it becomes apparent that $f_{between}$ is the only face from where all the neighboring vertices of a_i^q can be reached without introducing line-crossings. Therefore, the structure S_i must be placed in the face $f_{between}$ between the two quadrangles Q and Q'. □

An adaption of Lemma 3.1 to our situation, where one quadrangle lies inside of the other quadrangle, is visualized in Figure 3.7. The figure shows an example of one of the structures S_i placed in the faces inside, between, and outside two quadrangles. However, the lemma already told us that the apex-structures S_i must be placed in the face inside the outer quadrangle Q' and outside the inner quadrangle Q.

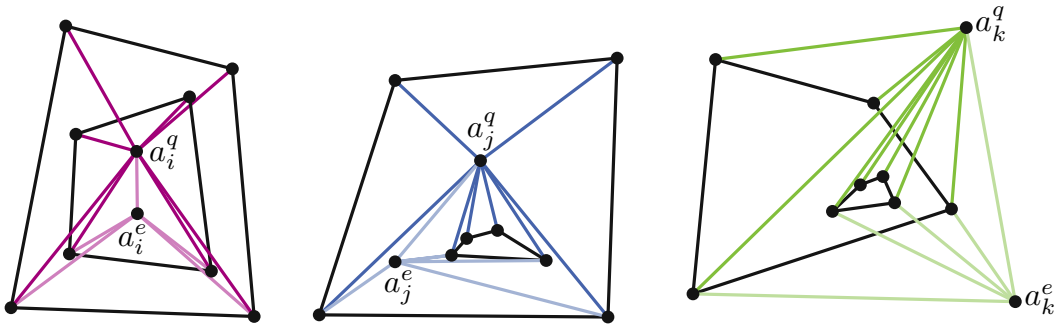


Figure 3.7: The left shows two quadrangles with one S_i inside the inner one. The middle shows the structure S_i placed between the two quadrangles, and the right has a structure S_i outside of both quadrangles. Of those three, only the graph in the middle is planar.

The core idea behind our counterexample revolves around the inner quadrangle and the structures S_i for $i \in [4]$ that we have defined previously. Neither of the quadrangles itself could create any contradiction. The outer quadrangle acts as a support structure

that imposes certain restrictions, which we will use to create the contradiction with the inner quadrangle. By itself, the outer quadrangle is relatively harmless, as each of the structures S_i can be drawn inside a quadrangle without creating any line crossings, as seen in Figure 3.8.

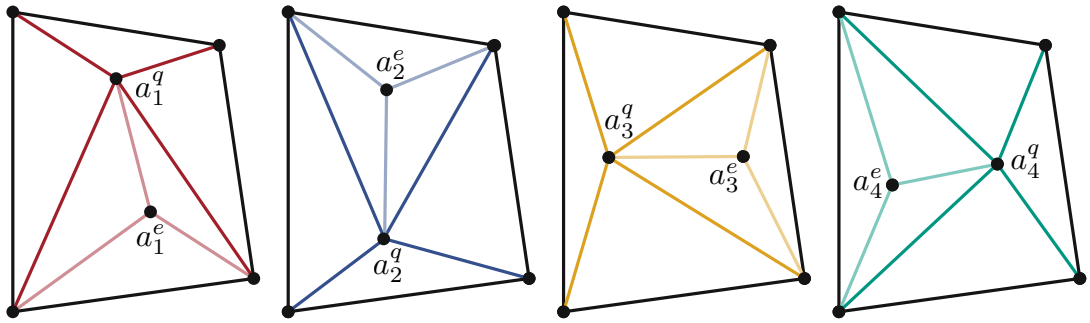


Figure 3.8: This Figure shows that each structure S_i separately can be placed inside of a quadrangle without any problems.

The construction of our graph can be done step by step. The positions of the outer quadrangle are already fixed, as those have to stay the same over all frames, and we will now discuss how we have to place the apex vertices a_i^q and a_i^e, for $i \in [4]$. The positions of the inner quadrangle must stay the same over multiple frames, but when constructing the first S_i, we can still move them around. However, they must also stay in that place for the other structures. We will show that no matter how we place the apex vertices and the inner quadrangle Q, we will always end up with structures S_i for which we cannot find a planar geometric embedding.

In this step, we take a look at the existing faces and then decide into which face an apex vertex needs to be placed, and based on the edges that apex vertex shares with the outer quadrangle, how this apex vertex must be placed in that face. New faces are created by placing an apex vertex in a face and drawing all the edges it shares with the outer quadrangle. The interesting question is, in which of these faces must the inner quadrangle lie? The next lemma will deal with those faces.

Lemma 3.2. *Given two quadrangles Q and Q' that lie inside of each other, a vertex a_i^q that is connected to all vertices of both Q and Q', as well as another vertex a_i^e that is connected to two adjacent vertices $v_{\phi(i)}$ ($v'_{\phi(i)}$) and ($v_{\phi(i+1)}$) $v'_{\phi(i+1)}$ of each quadrangle respectively and to the vertex a_i^q,*
1) a_i^q must be placed in the face between Q and Q' in such a way that Q lies in the same newly created face $f_{\phi(i)}$ as the vertices $v'_{\phi(i)}$ and $v'_{\phi(i+1)}$,
2) and a_i^e must be placed in the face $f_{\phi(i)}$ in such a way that Q lies in the same newly created face as both apex vertices a_i^q and a_i^e.

Proof. 1) Placing a_i^q in the inner face of Q' and only drawing the edges it shares with the outer quadrangle creates four new faces f_1, f_2, f_3, and f_4, with the following vertex sets:

$$V(f_1) = \{a_i^q, v'_1, v'_2\}$$
$$V(f_2) = \{a_i^q, v'_2, v'_3\}$$
$$V(f_3) = \{a_i^q, v'_3, v'_4\}$$
$$V(f_4) = \{a_i^q, v'_1, v'_4\}$$

The quadrangle-apex a_i^q can be placed so that the inner quadrangle Q could be in any of these faces. Depending on the index i of a_i^q and a_i^e, there is exactly one face, namely $f_{\phi(i)}$ that contains the vertices $(v'_{\phi(i)})$ and $v'_{\phi(i+1)}$. The edge-apex a_i^e needs to connect to these two vertices and two vertices of the inner quadrangle Q. So, let us assume that Q is in one of the other faces. Then, no face contains all adjacent vertices of a_i^e, and therefore, no matter where we place a_i^e, we would always introduce line crossings. Therefore, Q must lie in the same face as the vertices $v'_{\phi(i)}$ and $v'_{\phi(i+1)}$, which is the face $f_{\phi(i)}$. This is possible, as there is one face that contains both $v'_{\phi(i)}$ and $v'_{\phi(i+1)}$ and we stated before that Q can lie in any of these newly created faces.

2) As just established, $f_{\phi(i)}$ is the only face that contains all adjacent vertices of a_i^e, therefore, a_i^e must be placed in the face $f_{\phi(i)}$. So, again placing a_i^e in the face $f_{\phi(i)}$ and only drawing the edges it shares with the outer quadrangle creates three new faces $f_{\phi(i),1}$, $f_{\phi(i),2}$, and $f_{\phi(i),3}$ with the following vertex sets:

$$V(f_{\phi(i),1}) = \{a_i^e, v'_{\phi(i)}, v'_{\phi(i+1)}\}$$
$$V(f_{\phi(i),2}) = \{a_i^e, a_i^q, v'_{\phi(i)}\}$$
$$V(f_{\phi(i),3}) = \{a_i^e, a_i^q, v'_{\phi(i+1)}\}$$

The edge-apex a_i^e can again be placed so that the inner quadrangle Q could be in any of these faces. Both apex vertices a_i^q and a_i^e share edges with the inner quadrangle Q, therefore a_i^e must be placed in such a way that Q lies either in the face $f_{\phi(i),2}$ or $f_{\phi(i),3}$. Would Q lie in the other face $f_{\phi(i),1}$, then the quadrangle-apex a_i^q could not directly connect to Q and multiple edge crossings would be introduced into the drawing when drawing the edges between Q and a_i^q. Therefore, this cannot happen as we require our drawings to be planar. □

The just discussed Lemma 3.2, with all the newly created faces and possible placements for one specific apex vertex pair a_i^q and a_i^e, is also visualized in Figure 3.9

Based on two quadrangles that lie inside of each other and one structure S_i, we know from the Lemmas 3.1 and 3.2 that in order to be able to find a planar drawing of S_i, Q', and the edges between those two sets E_{Q',S_i}, while still being able to draw the edges between S_i and Q, we get the following intermediate result:
Q lies inside a triangular face consisting of the two apex vertices a_i^q, a_i^e, and one of the vertices of the outer quadrangle. It is still open, how the two apex vertices need to be placed in relation to Q?

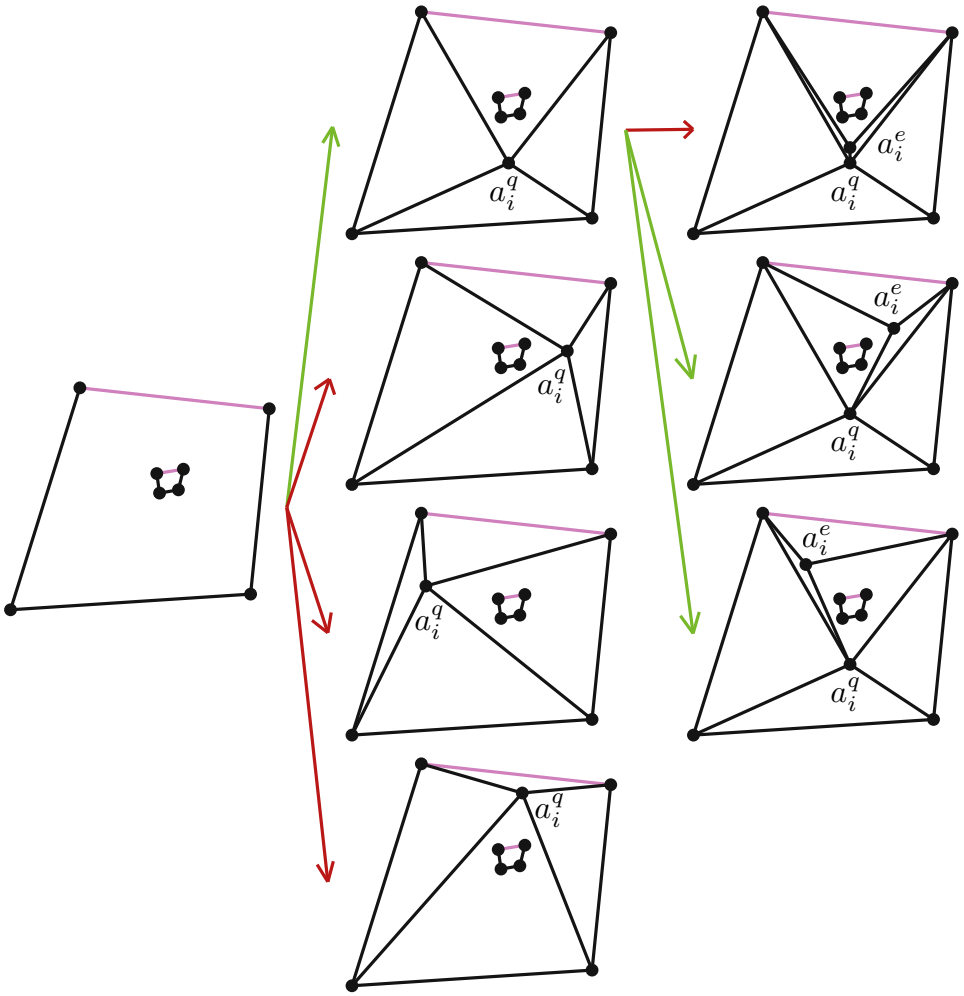


Figure 3.9: On the left, we see one fixed placement for the outer and the inner quadrangle and the edges that connect to a_i^e marked in violet. In the middle, we see the possible placements for a_i^q, where only the upper one is legitimate as all others split both violet edges so that there is no possible placement for a_i^e that could lead to a planar drawing. On the right, we see the possible placements for a_i^e. Here, the upper right drawing is invalid, as there is no way to draw edges between a_i^q and the inner quadrangle that would lead to a planar drawing. So, the two drawings in the second and third row of the right column represent the only feasible placements for a_i^q and a_i^e.

Lemma 3.3. *Given two edges $e_{a,b} = \{a, b\}$, and $e_{x,y} = \{x, y\}$, where the vertices a, b, x, and y form a clique. We are looking for a planar straight-line drawing of those edges and their connecting edges. Extending any edge e to infinity defines two half-spaces h_e^+ and h_e^-. If the vertices x and y lie in the same half-space $h_{e_{a,b}}^+$ or $h_{e_{a,b}}^-$ (w.l.o.g. let us assume these vertices lie in half-space $h_{e_{a,b}}^+$), then the vertices a and b must lie in different half-spaces $h_{e_{x,y}}^+$ and $h_{e_{x,y}}^-$.*

Proof. Let us assume both vertices a and b lie in the same half-space defined by $e_{x,y}$, w.l.o.g. let us assume this is the half-space $h^+_{e_{x,y}}$. Now we have to draw the edges $e_{a,x}$, $e_{a,y}$, $e_{b,x}$, and $e_{b,y}$. When drawing the edges $e_{a,x}$ and $e_{b,y}$, we create a closed face in the form of a convex quadrangle. The quadrangle must be convex as the vertices x and y lie in the same half-space $h^+_{e_{a,b}}$. Drawing one of the other two edges $e_{a,y}$, $e_{b,x}$ splits this quadrangle into two triangular faces, let us say w.l.o.g. that the edge $e_{a,y}$ was drawn, so now we have two faces $f_{a,x,y}$ and $f_{a,b,y}$. The remaining edge $e_{b,x}$ cannot be drawn in any of these faces but must cross the edge $e_{a,y}$ and thereby create an intersection. Therefore, this assumption must be wrong, and a and b must lie in different half-spaces defined by $e_{x,y}$. Then it is possible to draw the edges $e_{a,x}$ and $e_{a,y}$ in one half-space and the edges $e_{b,x}$, and $e_{b,y}$ in the other half-space. This results in a planar drawing of all our vertices and concludes this proof. □

To repeat, the quadrangle vertex a^q_i is connected to all vertices of Q, while a^e_i is only connected to the vertices $v_{\phi(i)}$ and $v_{\phi(i+1)}$ of Q. These vertices are incident to the edge $e_{\phi(i)}$. Now taking Lemma 3.3 and applying it to our situation, with a^q_i and a^e_i being the vertices a and b in the lemma, as well as setting the vertices $v_{\phi(i)}$ and $v_{\phi(i+1)}$ to x and y. The vertices $v_{\phi(i)}$ and $v_{\phi(i+1)}$ must lie in the same half-space defined by the edge $e_{a^q_i,a^e_i}$ as Q needs to completely lie in one face, and $e_{a^q_i,a^e_i}$ is one of the border edges of the face which contains Q. We now get that the apex vertices a^q_i and a^e_i must be placed in such a way that both of them lie in different half-spaces defined by the edge $e_{v_{\phi(i)},v_{\phi(i+1)}}$. More informally, the apex vertices must lie on different sides of the inner quadrangle Q.

Lemma 3.4. *Given a convex quadrangle Q, a vertex a^q_i that is connected to all vertices of Q, as well as another vertex a^e_i that is connected to two adjacent vertices $v_{\phi(i)}$ and $(v_{\phi(i+1)})$ of Q and to the vertex a^q_i. The vertices $v_{\phi(i)}$ and $(v_{\phi(i+1)})$ are incident to the edge $e_{\phi(i)}$. It is already established that both a^q_i and a^e_i must lie outside of Q and in different half-spaces $h^+_{e_{\phi(i)}}$ or $h^-_{e_{\phi(i)}}$ defined by the edge $e_{\phi(i)}$.*
We are looking for a planar straight-line drawing. Then a^e_i must lie in the half-space $h^+_{e_{\phi(i)}}$, the half-space that lies completely outside of Q, and a^q_i must lie in the half-space $h^-_{e_{\phi(i)}}$, the half-space that contains all vertices of Q.

Proof. Let us assume that a^e_i lies in the half-space $h^-_{e_{\phi(i)}}$, then from Lemma 3.3 we know that a^q_i must lie in half-space $h^+_{e_{\phi(i)}}$. As Q is convex and by the definition of the half-spaces defined by the edge $e_{\phi(i)}$, all vertices of Q must lie inside half-space $h^-_{e_{\phi(i)}}$. The edge-apex vertex a^e_i is connected to the vertices $v_{\phi(i)}$ and $(v_{\phi(i+1)})$, together they form a cycle. As a^e_i lies outside of Q, the remaining vertices of Q would need to be inside this cycle. The cycle is fully inside the half-space $h^-_{e_{\phi(i)}}$ and per assumption the quadrangle-apex vertex a^q_i must lie in half-space $h^+_{e_{\phi(i)}}$. However, a^q_i has edges to all vertices of Q, so also to the vertices inside this cycle built by a^e_i and the edge $e_{\phi(i)}$. This leads to a nonplanar drawing and, therefore, to an unfeasible solution. So our assumption cannot be true and per contradiction a^e_i must lie in the half-space $h^+_{e_{\phi(i)}}$ and a^q_i must lie in the half-space $h^-_{e_{\phi(i)}}$. □

The application of the Lemmas 3.3 and 3.4 to our situation is also shown in Figure 3.10.

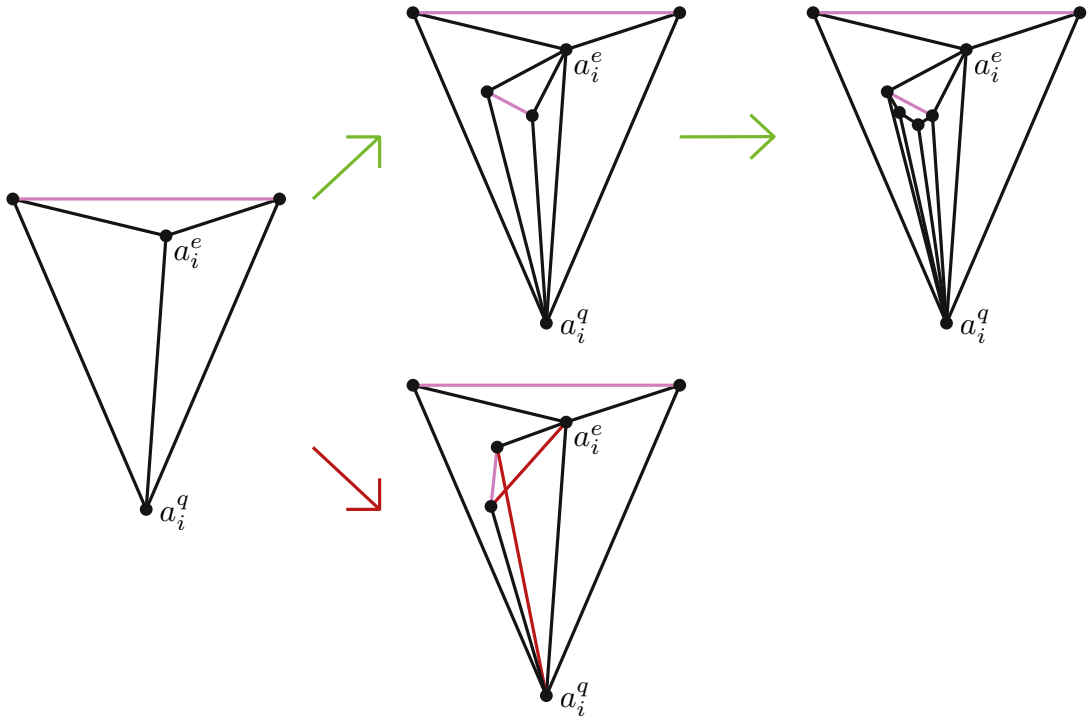


Figure 3.10: On the left, we see the face defined by a_i^q and one edge of the outer quadrangle, and a_i^e placed inside that face. In the middle graphs, we see the different possible positions of the edge of the inner quadrangle in contrast with the positions of a_i^q and a_i^e. In the lower middle graph, both these vertices lie on the same side of the edge $e_{\phi(i)}$, which leads to line crossings. On the upper middle graph, we see the vertices placed on different sides of the edge $e_{\phi(i)}$, which leads to a valid planar straight-line drawing. On the right, we see the same graph again, but the two missing vertices of the inner quadrangle are added, showing that those have to be placed on the same side of $e_{\phi(i)}$ as a_i^q.

Corollary 3.1. *Given the graph G, where the convex quadrangle Q lies inside of the quadrangle Q', and structures S_i for $i \in [4]$, Q and S_i always form the apex-triangle A_i as defined in Subsection 3.1.1 (The structure).*

Proof. Starting from the graph G, where two quadrangles lie inside each other, by applying Lemma 3.1, we get that the structures S_i must be drawn between those two quadrangles. Lemma 3.2 describes into which newly formed faces the apex vertices must be placed and in which face the inner quadrangle Q must lie. The Lemmas 3.3 and 3.4 establish, where the apex vertices a_i^q and a_i^e must lie in relation to a convex quadrangle Q. This brings us to a position where Q lies in a face that is incident to both apex vertices,

and these apex vertices lie in different half-spaces defined by the edge $e_{\phi(i)}$ of Q. So Q and the structure S_i are building precisely the apex-triangle A_i for each $i \in [4]$. □

Remark 3.1. Corollary 3.1 describes exactly the steps of the proof that we have taken so far. These steps are also illustrated in Figure 3.11. With that, we get from two quadrangles that lie inside of each other to the point that the four structures S_i and the inner quadrangle Q must form four the apex-triangles A_i, one for each $i \in [4]$.

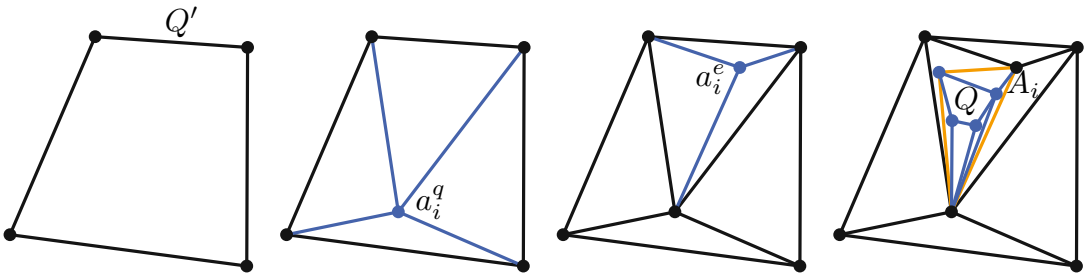


Figure 3.11: This Figure shows how we went from the outer quadrangle Q' to the apex-triangle A_i. The new edges in every step are represented in blue, and A_i is drawn in orange.

So far, we have established that when using straight-line edges, a_i^q and a_i^e must lie on different sides of the inner quadrangle. Then, there are two possible scenarios, as shown in Figure 3.12, one without any line crossings and one with line crossings. The existence of line crossings between one of the corners of the inner quadrangle and one of the quadrangle-apexes a_i^q depends solely on the position of the four vertices of the inner quadrangle in relation to the position of a_i^q. So far, we have only fixed the faces in which our vertices must lie and did not need to find fixed positions for them. Due to the existence of our four structures S_i, where their respective edge-apexes a_i^e are all connected to different sides $e_{\phi(i)}$ of the inner quadrangle Q, we would have to find a position for the four vertices of the inner quadrangle Q such that all four structures S_i can be drawn planarly using straight-line edges.

We will now look at the defining property that decides whether line crossings between the inner quadrangle Q and one a_i^q exist. To draw straight-line edges between a_i^q and the vertices of Q, a_i^q must be able to see all of the vertices of Q. This fact is formalized in the following Lemma 3.5, and this Lemma is also visualized in Figure 3.13.

Lemma 3.5. *Given a convex quadrangle Q, a vertex a_i^q that is connected to all vertices of Q, as well as another vertex a_i^e that is connected to the edge $e_{\phi(i)}$ of Q and to the vertex a_i^q. Each edge e_i of Q defines two half-spaces, where the positive half-space $h_{e_i}^+$ looks away from Q and the negative half-space $h_{e_i}^-$ contains Q. It is already established that both a_i^q and a_i^e must lie outside of Q, and that a_i^e must lie in the half-space $h_{e_{\phi(i)}}^+$ and a_i^q must lie in the half-space $h_{e_{\phi(i)}}^-$. (So Q and S_i build an apex-triangle A_i.) We are looking for a planar straight-line drawing. Then, in order for a planar straight-line*

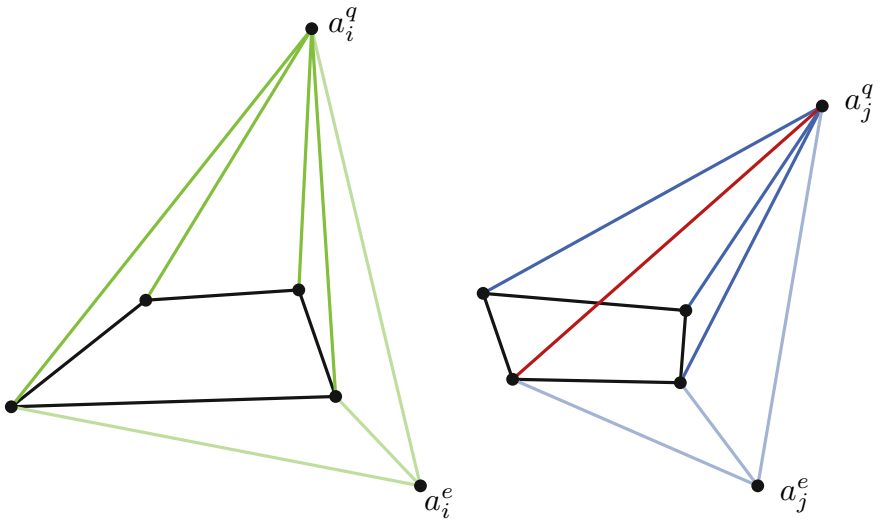


Figure 3.12: To the left in green, we see a structure where a_i^q can connect to all corners of the quadrangle, with straight-line edges. To the right in blue, we see a structure where this is impossible.

drawing to exist, a_i^q must lie in the intersection of the half-spaces $h^+_{e_{\phi(i-1)}}$, $h^+_{e_{\phi(i+1)}}$, and $h^-_{e_{\phi(i)}}$.

Proof. Each vertex $v_{\phi(i)}, i \in [4]$ of Q can only be seen from a vertex that lies in one of the positive half-spaces that are defined by one of the edges of Q that contain that vertex $v_{\phi(i)}$. These are the half-spaces $h^+_{e_{\phi(i-1)}}$ and $h^+_{e_{\phi(i)}}$. Now as a_i^q must be able to see all of the vertices of Q for each vertex $v_{\phi(i)}$, a_i^q must either lie in $h^+_{e_{\phi(i-1)}}$ and $h^+_{e_{\phi(i)}}$. As we already know a_i^q lies in the half-space $h^-_{e_{\phi(i)}}$, this means that in order for a_i^q to be able to see the vertex $v_{\phi(i)}$, a_i^q must lie in the half-space $h^+_{e_{\phi(i-1)}}$, and in order to be able to see $v_{\phi(i+1)}$, a_i^q must lie in the half-space $h^+_{e_{\phi(i+1)}}$. If a_i^q lies inside the half-spaces $h^+_{e_{\phi(i-1)}}$ and $h^+_{e_{\phi(i+1)}}$ the remaining two vertices of Q, $v_{\phi(i-2)}$ and $v_{\phi(i+2)}$ are also covered by those half-spaces. This concludes the proof. □

In Lemma 3.5 we established that a_i^q must lie in the intersection of the half-spaces $h^+_{e_{\phi(i-1)}}$, $h^+_{e_{\phi(i+1)}}$, and $h^-_{e_{\phi(i)}}$ to be able to draw straight-line edges between a_i^q and all vertices of the inner quadrangle. Every vertex placed in the intersection of the half-spaces $h^+_{e_{\phi(i-1)}}$ and $h^+_{e_{\phi(i+1)}}$ can see all vertices of the quadrangle Q. As we assumed that Q is a convex quadrangle with no parallel edges, such an intersection must always exist. The question then remains whether the intersection of these two half-spaces lies inside of the half-space $h^-_{e_{\phi(i)}}$ or not. This is the deciding property, whether a planar drawing of the structure S_i and the inner quadrangle Q can exist. Figure 3.14 shows different possibilities for the positions of these half-spaces.

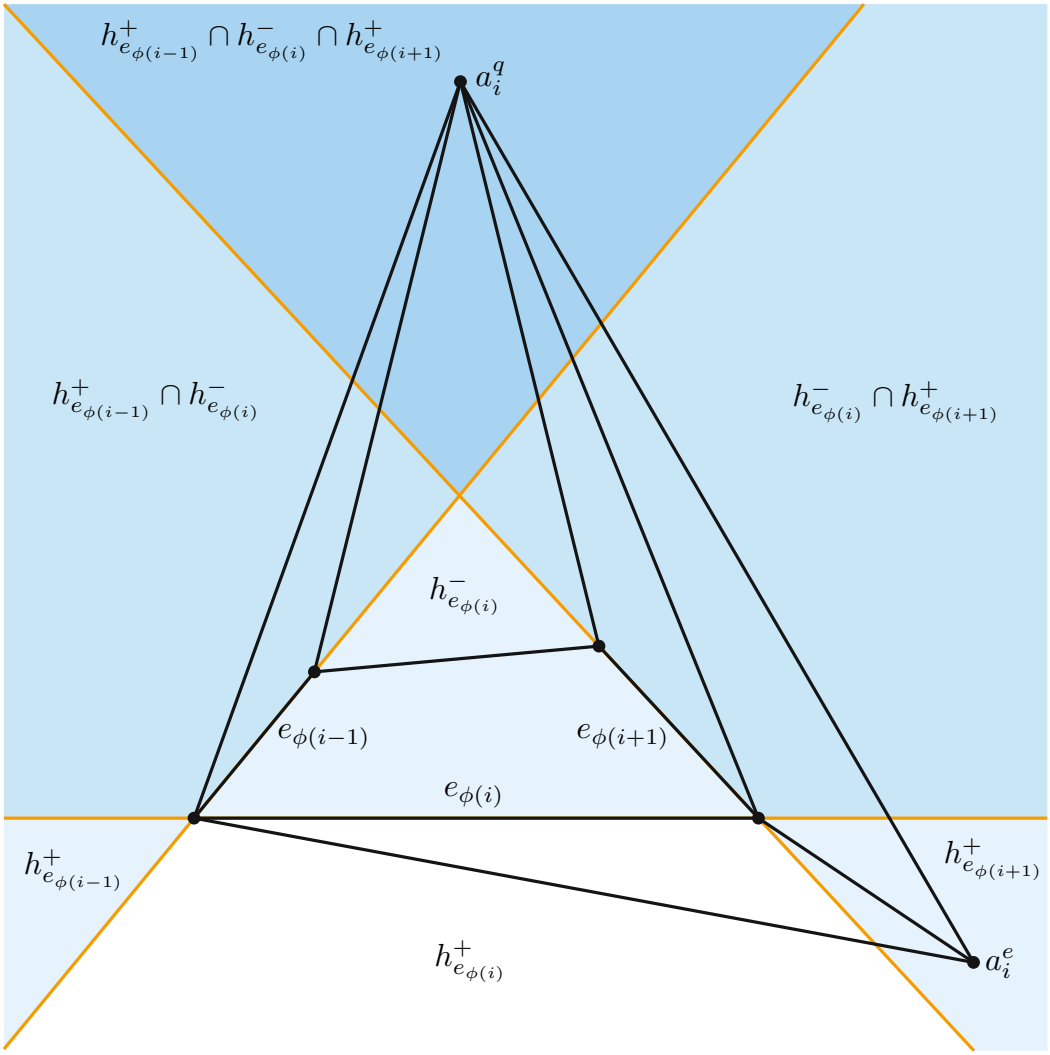


Figure 3.13: This graph shows the three defining half-planes. First, a horizontal line that splits the half-planes $h^+_{e_{\phi(i)}}$ and $h^-_{e_{\phi(i)}}$ and also separates a_i^q and a_i^e. Second and third, the half-planes $h^+_{e_{\phi(i-1)}}$ and $h^+_{e_{\phi(i+1)}}$ defined by the side edges of the inner quadrangle. a_i^q is placed inside the intersection of the latter two half-planes and, therefore, can see all vertices of the inner quadrangle.

So, we defined the defining property for one a_i^q. For the other quadrangle-apexes a_i^q this property stays the same with the caveat that the half-space $h^-_{e_{\phi(i)}}$, where the intersection of the other two half-spaces $h^+_{e_{\phi(i-1)}}$ and $h^+_{e_{\phi(i+1)}}$ must happen, is defined by another edge of the quadrangle. As a quadrangle has four sides, there are two pairs of opposed sides, and every pair has the same two neighboring edges. The first pair consists of the edges e_1 and e_3, and the second pair of the edges e_2 and e_4, where each pair has the other pair as neighboring edges.

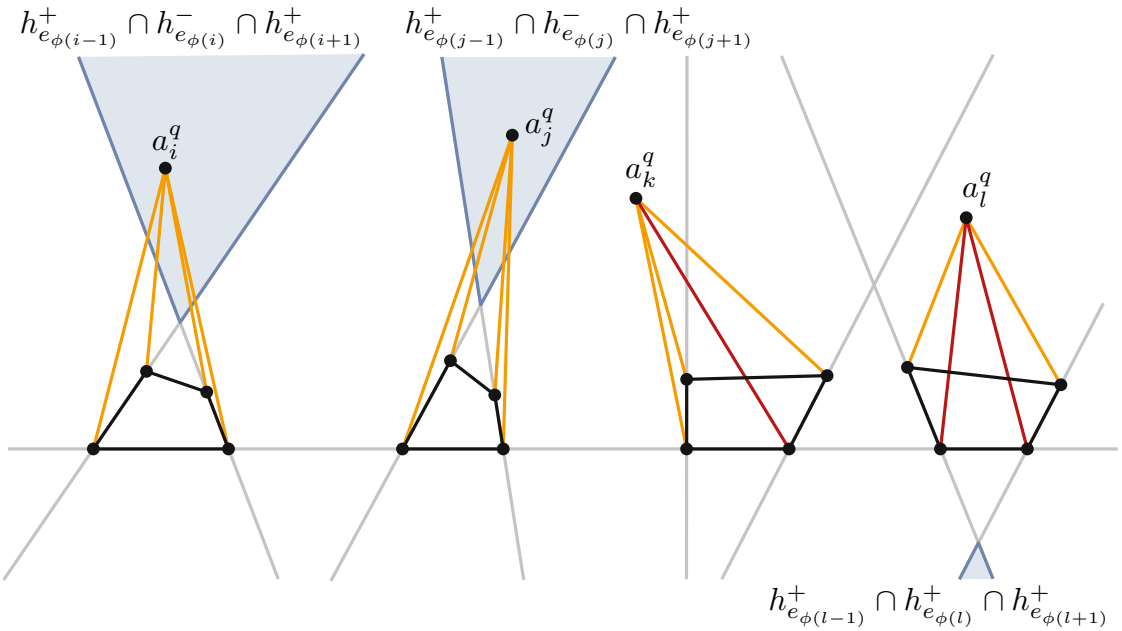


Figure 3.14: To the left, we see two quadrangles, where the half-planes defined by the side edges intersect above the grey horizontal half-plane, which allows a planar drawing of a^q_i and its edges with the inner quadrangle. To the right, we see two quadrangles, where the half-planes defined by the side edges intersect below the horizontal half-plane. As a^q_i must be placed above that horizontal half-plane, there does not exist a valid position for a^q_i that does not create line crossings. These line crossings are represented in red.

This means that for structures S_i, where $i = 1$ or $i = 3$, the half-spaces $h^+_{e_2}$ and $h^+_{e_4}$ must intersect. For $i = 1$ in the half-space $h^-_{e_1}$ and for $i = 3$ in the half-space $h^-_{e_3}$. The problem now arises as it is not possible for the half-spaces $h^+_{e_2}$ and $h^+_{e_4}$ to intersect both in half-space $h^-_{e_1}$ and in half-space $h^-_{e_3}$, but only in one of those two, as Lemma 3.6 will show.

Lemma 3.6. *Given a convex quadrangle Q with no parallel edges, the outer half-spaces defined by the edges of two opposing sides always intersect. However, this intersection of the two outer half-spaces of opposing sides of Q cannot lie in both inner half-spaces of its neighboring sides. Or formally:*

$$(h^-_{e_{\phi(i)}} \cap h^+_{e_{\phi(i+1)}} \cap h^+_{e_{\phi(i+3)}} = \emptyset) \vee (h^-_{e_{\phi(i+2)}} \cap h^+_{e_{\phi(i+1)}} \cap h^+_{e_{\phi(i+3)}} = \emptyset).$$
$$(h^-_{e_{\phi(i+1)}} \cap h^+_{e_{\phi(i)}} \cap h^+_{e_{\phi(i+2)}} = \emptyset) \vee (h^-_{e_{\phi(i+3)}} \cap h^+_{e_{\phi(i)}} \cap h^+_{e_{\phi(i+2)}} = \emptyset).$$

Proof. We know that in Euclidean Geometry, every pair of two non-parallel lines must have an intersection point. Therefore, it also holds that the half-spaces defined by two edges must have an intersection, no matter which of the two half-spaces defined by an edge one regards. This then obviously also holds for the opposing sides of any convex quadrangle.

Let us say w.l.o.g. that we are looking for the intersection of the outer half-spaces of the edges $e_{\phi(i+1)}$ and $e_{\phi(i+3)}$. By definition, we know that the intersection of the outer half-spaces must lie outside of the quadrangle Q. Further, both edges $e_{\phi(i)}$ and $e_{\phi(i+2)}$ have one of their incident vertices on the edge $e_{\phi(i+1)}$ and the other one on the edge $e_{\phi(i+3)}$. Now w.l.o.g. let us say that the edge $e_{\phi(i+2)}$ is the edge closer to the intersection point of the half-spaces $h^+_{e_{\phi(i+1)}}$ and $h^+_{e_{\phi(i+3)}}$ than the edge $e_{\phi(i)}$. As the edge $e_{\phi(i+2)}$ spans from one half-space $h^+_{e_{\phi(i+1)}}$ to the other $h^+_{e_{\phi(i+3)}}$ it must always have the same orientation in relation to the intersection point of the two half-spaces. In formal terms, the intersection of $h^+_{e_{\phi(i+1)}}$ and $h^+_{e_{\phi(i+3)}}$ will always lie in the half-space $h^+_{e_{\phi(i+2)}}$ and therefore the cut of the half-spaces $h^-_{e_{\phi(i+2)}}$, $h^+_{e_{\phi(i+1)}}$, and $h^+_{e_{\phi(i+3)}}$ will always be empty so we get: $h^-_{e_{\phi(i+2)}} \cap h^+_{e_{\phi(i+1)}} \cap h^+_{e_{\phi(i+3)}} = \emptyset$.

If, on the other hand, the edge $e_{\phi(i)}$ is closer to the intersection of the two outer half-spaces, we get by the same argumentation steps that $h^-_{e_{\phi(i)}} \cap h^+_{e_{\phi(i+1)}} \cap h^+_{e_{\phi(i+3)}} = \emptyset$. So it must always hold that:

$$(h^-_{e_{\phi(i)}} \cap h^+_{e_{\phi(i+1)}} \cap h^+_{e_{\phi(i+3)}} = \emptyset) \vee (h^-_{e_{\phi(i+2)}} \cap h^+_{e_{\phi(i+1)}} \cap h^+_{e_{\phi(i+3)}} = \emptyset). \tag{3.1}$$

Trivially, the same steps apply if the outer half-spaces are defined by the edges $e_{\phi(i)}$ and $e_{\phi(i+2)}$. Therefore, the following must also always hold:

$$(h^-_{e_{\phi(i+1)}} \cap h^+_{e_{\phi(i)}} \cap h^+_{e_{\phi(i+2)}} = \emptyset) \vee (h^-_{e_{\phi(i+3)}} \cap h^+_{e_{\phi(i)}} \cap h^+_{e_{\phi(i+2)}} = \emptyset).$$

□

Applying the above lemmas leads us to a final Lemma 3.7 and the conclusion of our proof.

Lemma 3.7. *Given a convex quadrangle Q with no parallel edges, apex-structures S_i for $i \in [4]$, containing a vertex a^q_i that is connected to all vertices of Q, as well as another vertex a^e_i that is connected to the edge $e_{\phi(i)}$ of Q and to the vertex a^q_i. The apex vertices lie outside of Q in different half-spaces of the edge $e_{\phi(i)}$. (So Q and all of the apex-structures S_i build apex-triangles A_i for $i \in [4]$.) Then for four structures S_i that are connected to all edges e_i of the quadrangle Q respectively, there are two of those structures for which no planar geometric embedding can be found.*

Proof. Let us assume a planar geometric embedding can be found for all the apex-structures S_i. Applying the Lemmas 3.4 and 3.5 gives that under the given assumptions, a planar straight-line drawing of a structure S_i can only exist when the half-spaces $h^+_{e_{\phi(i-1)}}$, $h^+_{e_{\phi(i+1)}}$, and $h^-_{e_{\phi(i)}}$ intersect. We want this to happen for all four of our apex-structures S_i. So we get that we need the following inequalities to hold:

$$I_1 : h^-_{e_{\phi(1)}} \cap h^+_{e_{\phi(2)}} \cap h^+_{e_{\phi(0)}} \neq \emptyset.$$
$$I_2 : h^-_{e_{\phi(2)}} \cap h^+_{e_{\phi(3)}} \cap h^+_{e_{\phi(1)}} \neq \emptyset.$$
$$I_3 : h^-_{e_{\phi(3)}} \cap h^+_{e_{\phi(4)}} \cap h^+_{e_{\phi(2)}} \neq \emptyset.$$
$$I_4 : h^-_{e_{\phi(4)}} \cap h^+_{e_{\phi(5)}} \cap h^+_{e_{\phi(3)}} \neq \emptyset.$$

Now, applying the ϕ function, we get the following inequalities:

$$I_1 : h^-_{e_1} \cap h^+_{e_2} \cap h^+_{e_4} \neq \emptyset.$$
$$I_2 : h^-_{e_2} \cap h^+_{e_3} \cap h^+_{e_1} \neq \emptyset.$$
$$I_3 : h^-_{e_3} \cap h^+_{e_4} \cap h^+_{e_2} \neq \emptyset.$$
$$I_4 : h^-_{e_4} \cap h^+_{e_1} \cap h^+_{e_3} \neq \emptyset.$$

As we want to find a valid drawing for all four structures, all of the above formulas need to hold. Similarly, when we want to find valid drawings for multiple structures S_i, their respective formulas must hold simultaneously.

However, from Lemma 3.6, we know that for every convex quadrangle Q the following holds:

$$(h^-_{e_{\phi(i)}} \cap h^+_{e_{\phi(i+1)}} \cap h^+_{e_{\phi(i+3)}} = \emptyset) \vee (h^-_{e_{\phi(i+2)}} \cap h^+_{e_{\phi(i+1)}} \cap h^+_{e_{\phi(i+3)}} = \emptyset).$$
$$(h^-_{e_{\phi(i+1)}} \cap h^+_{e_{\phi(i)}} \cap h^+_{e_{\phi(i+2)}} = \emptyset) \vee (h^-_{e_{\phi(i+3)}} \cap h^+_{e_{\phi(i)}} \cap h^+_{e_{\phi(i+2)}} = \emptyset).$$

Applying the ϕ function to these formulas for any $i \in \mathbb{N}$ leads to the following formulas:

$$(h^-_{e_1} \cap h^+_{e_2} \cap h^+_{e_4} = \emptyset) \vee (h^-_{e_3} \cap h^+_{e_2} \cap h^+_{e_4} = \emptyset).$$
$$(h^-_{e_2} \cap h^+_{e_1} \cap h^+_{e_3} = \emptyset) \vee (h^-_{e_4} \cap h^+_{e_1} \cap h^+_{e_3} = \emptyset).$$

Now inserting the formulas I_1, I_2, I_3 and I_4 into these formulas gives us the following:

$$\neg I_1 \vee \neg I_3.$$
$$\neg I_2 \vee \neg I_4.$$

Therefore, from the formulas I_1 and I_3, one cannot hold, and the same applies to the formulas I_2 and I_4. So we get a contradiction to our assumption that we can find a planar geometric drawing for all structures S_i.

From these equations, we get that for a convex quadrangle Q, we can find valid drawings for the structures S_i that connect to the edge e_1 or the edge e_3, but not for both. The same holds for structures S_i that connect to the edges e_2 and e_4; only one can be satisfied. So, for a convex quadrangle with no parallel edges, a planar geometric drawing can be found for exactly two structures, and two structures will always lead to non-planar drawings. □

We mentioned above that starting from two quadrangles inside each other, we get to the apex-triangle A_i. This apex-triangle A_i is the starting point of Lemma 3.7. So, in total, we get that under our assumptions that both quadrangles Q and Q' lie inside of each other, where Q is the inner quadrangle, and that Q is convex and has no parallel edges, that of the four apex-structures S_i, two do not have a planar geometric embedding. Thereby, we have shown that our construction creates two situations for which no planar straight-line drawing exists. In a PGEO-SP, every frame must be a planar straight-line drawing, so we get that, as long as there is a frame for every apex-structure S_i with $i \in [4]$ containing that S_i as well as the inner quadrangle Q and the outer quadrangle Q', then our contradicting construction must appear in two of those frames.

3.3.2 Applying the line crossings to a frame

We have shown that two structures S_i create a line crossing with the inner quadrangle Q if the structures lie outside Q. Now, we still have to show that the inner quadrangle and each respective structure must also be visible simultaneously in one of the frames of a drawing of a storyplan, such that one frame that contains a line crossing must exist.

From the definition of the storyplan problem, we know that one vertex – after it is drawn – can only disappear in one frame f_k if all its incident edges were also drawn in one of the frames f_j with $j \leq k$. For ease of understanding, we assume that the structures S_i with $i \in [8]$ will appear in their ordering in the storyplan. So for all $m \in [8]$, holds that if S_m appears in frame f_j and disappears in frame f_{j+1}, every structure S_o with $m < o$, can only appear in frames f_k with $j \leq k$.

We know that every a_i^q is connected to the corners of both quadrangles. If two of those quadrangle-apexes a_i^q were active in the same frame, we would get multiple line crossings. This situation can be seen in Figure 3.15. This must happen as one a_i^q, let us call it a_j^q, and the inner quadrangle split the plane into five faces. Now, to place another a_i^q, let us call it a_k^q, in the same frame, it must lie inside one of those faces. a_k^q cannot lie inside the inner quadrangle, as it could not have crossing free edges with the outer quadrangle if it lies inside the inner quadrangle. So four faces remain, each defined by a_j^q and two of the corners of the quadrangle. So, no matter in which of those four faces we place a_k^q, it can only connect to the two corners of the quadrangle that define that particular face without creating line crossings. However, as each frame must contain all the edges existing between its active vertices, we must also connect a_k^q with the two corners of the quadrangle that define other faces. In order to reach these corners, those connecting edges must intersect with some of the edges between a_j^q and the quadrangle. This situation can be seen in Figure 3.16. Therefore, when two vertices a_j^q and a_k^q are active in the same frame, we always get a non-planar graph and, thereby, an illegal frame. Based on this, we know that our assumption must be wrong, and therefore, every a_i^q must be active in a different frame.

Do both quadrangles have to be active simultaneously as each a_i^q? For a_i^q to disappear again, it must first see every corner of both quadrangles at least once. Now, let us assume that the two quadrangles are not active simultaneously. This would mean that w.l.o.g. the inner quadrangle is active first, all connections to a_i^q are drawn, then the inner quadrangle disappears, and the outer quadrangle is active afterward. However, as we established, different quadrangle-apexes a_i^q must be active in different frames, all with edges to all corners of both quadrangles. So, the inner quadrangle cannot disappear before the outer quadrangle is drawn as it must be active in at least one frame with each a_i^q. It could only disappear after being in a frame with the last quadrangle-apex a_8^q. The outer quadrangle must also share one frame with each quadrangle-apex a_i^q. So, both quadrangles must be active in the same frame with a_1^q; otherwise, a_1^q could not disappear. Both quadrangles Q and Q' also have to share frames with each other and all a_j^q with $1 < j < 8$, as there is still another a_i^q appearing later in the storyplan, so they cannot

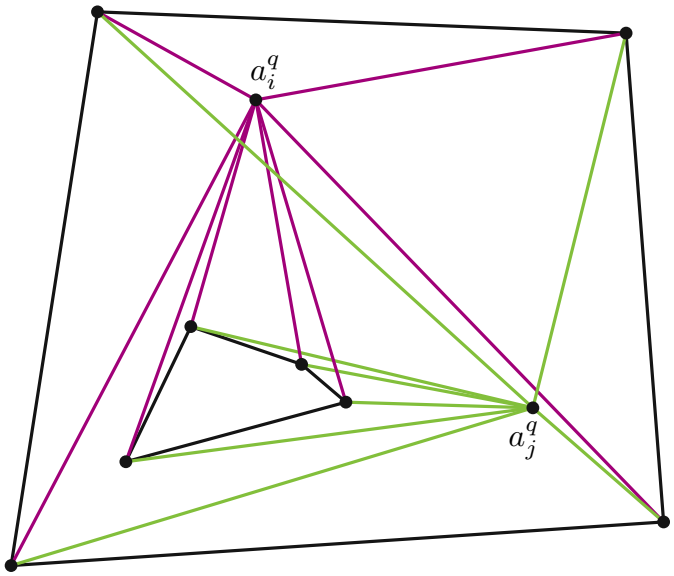


Figure 3.15: Two quadrangle-apexes a_i^q are active in the same frame. Drawn with all their connections to both quadrangles.

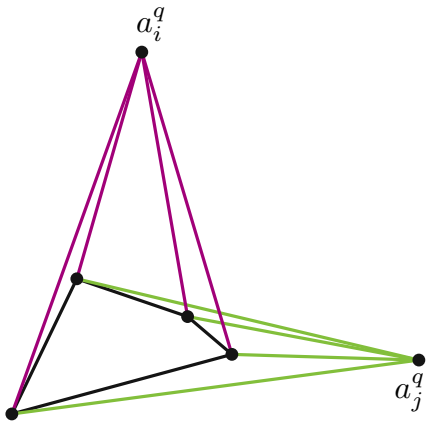


Figure 3.16: If two quadrangle-apexes a_i^q are active in the same frame, there must always be line crossings between their edges with the inner quadrangle.

yet disappear. For the last of the quadrangle-apexes a_i^q, a_8^q we have that both the outer and the inner quadrangle are active before a_8^q is drawn, and they both still need the connections to a_8^q before they can disappear. This means that both quadrangles must still be active in the frame where a_8^q appears. Therefore, every a_i^q has at least one frame that it shares with all vertices of both quadrangles.

As there is an edge between each a_i^q and its respective a_i^e, both of them must be active together in at least one frame. However, a_i^e must only be active in the same frame as the two respective vertices of both quadrangles adjacent to it. Therefore, for the first and

the last a_i^e, we can draw a_i^e before, respectively, after the other non-adjacent vertices of the quadrangles are drawn. As can be seen in Figure 3.17. Nevertheless, the edge-apexes a_i^e for $i \in \{2, 3, 4, 5, 6, 7\}$ still have to share one frame with all of the vertices of both quadrangles.

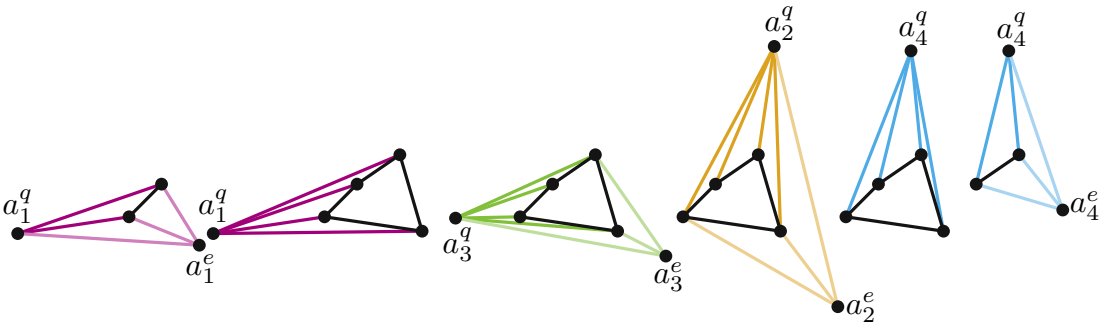


Figure 3.17: Showing six frames representing the inner quadrangle with the structures S_i for $i \in [4]$.

In total, we get that for each of the inner structures S_i for $i \in \{2, 3, 4, 5, 6, 7\}$, there must exist one frame where a_i^q, a_i^e, and both quadrangles Q and Q' are active at the same time.

Our construction can be avoided twice, once for the first and once for the last structure built. Now, if we only had four apex-structures S_i, then this would be a problem, as the two problematic structures that introduce the line crossings could be drawn as the first and the last structure, where these structures do not have to be fully represented, and the line crossings that they introduce could be avoided. That is precisely the reason why we needed eight structures. With eight structures, we have four structures that would introduce line crossings. Two can be avoided by making them the first and the last structures to be drawn. Nevertheless, the other two problematic structures must appear in the frames between, where the full apex-structure S_i and the quadrangles Q and Q' are active simultaneously. This is why, given our assumptions that both quadrangles Q and Q' lie inside of each other, where Q is the inner quadrangle, and that Q is convex and has no parallel edges, a planar geometric storyplan for our graph G cannot exist. With that, we have shown that indeed, for our assumptions, G is a yes-instance for PTOP-SP, but a no-instance for PGEO-SP, which in turn proves that under our assumptions PTOP-SP and PGEO-SP are different problems.

The argumentation of the last paragraph can be generalized into the following remark:

Remark 3.2. Given that two of the structures S_i for $i \in [4]$ have no valid planar drawing with Q, using the structure siblings, our graph G cannot have a planar geometric storyplan and is, therefore, a no-instance of PGEO-SP.

3.3.3 Dealing with our assumptions

In our proof, we made several assumptions. We will show why our proof works even when we drop those assumptions. The assumptions were that the inner quadrangle must be convex, cannot contain parallel edges, and that the two quadrangles are placed inside of each other.

Concave inner quadrangle

The first assumption we will tackle is that the inner quadrangle must be convex. We will show that using a concave quadrangle also leads to the necessity of the existence of line crossings for half of the apex-structures S_i and thereby creates the same problems when used in an attempt to create a valid PGEO-SP drawing of our graph G.

Every concave quadrangle has the same "boomerang"-shape. This "boomerang"-shape creates a distinction between the "outer" and the "inner" edges of the quadrangle in the context of the "curvature" of the "boomerang". The "inner" edges are the ones that are incident to the one concave vertex in the quadrangle – a quadrangle can, at most, have one concave vertex in a straight-line Euclidean drawing.

In the situation where a_i^e is connected to one of these "inner" edges $e_{\phi(i)}$, S_i could be drawn in two ways, as illustrated by Figure 3.18. Either the focus lies on a_i^q being able to see all vertices of the inner quadrangle, but then a_i^q and a_i^e must lie in the same half-plane $h^+_{e_{\phi(i)}}$ and line crossings between the edges that connect them to the edge $e_{\phi(i)}$ must exist, as was shown in Lemma 3.3. The other possibility would be to put a_i^q and a_i^e in different half-spaces $h^+_{e_{\phi(i)}}$ and $h^-_{e_{\phi(i)}}$. However, this leads to an intersection of the edge that connects the concave vertex of the inner quadruple with a_i^q with one of the edges of Q. A concave vertex can only be seen from vertices in the intersection of the outer half-spaces of its incident edges. As established in Lemma 3.4 a_i^q must lie in the inner half-space of the edge $e_{\phi(i)}$, which in our case is one of the incident edges of the concave vertex. Therefore, no valid placement for a_i^q can be found that allows a planar straight-line drawing of a concave Q and S_i.

So again, as this holds for both of the "inner" edges of a concave quadrangle Q, we get that we have two situations where multiple apex-structures S_i cannot be drawn together with both quadrangles with straight-line edges without introducing line crossings. That was precisely the property of our construction using a convex quadruple that we used in our proof to create a contradiction. Therefore, no matter whether we use a concave or a convex inner quadruple, no PGEO-SP drawing of our graph G can exist.

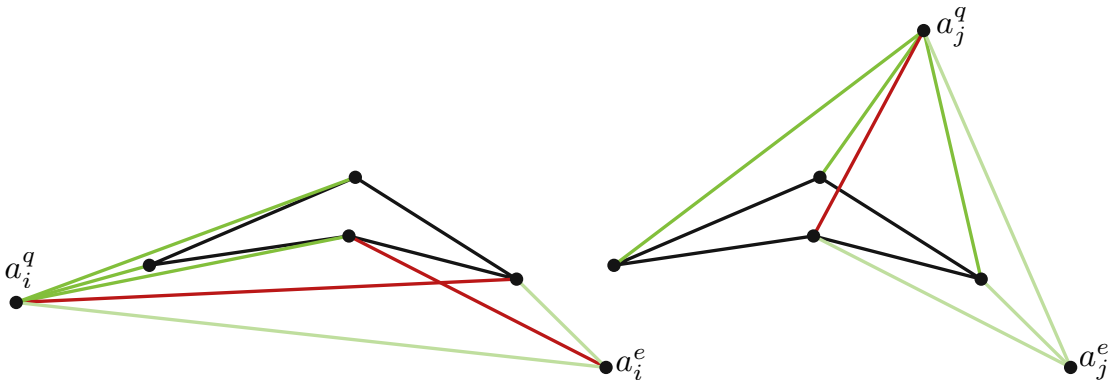


Figure 3.18: Here, we see both possible drawings of a structure S_i if the inner quadrangle is concave. On the left, a_i^q can see all vertices of the quadrangle, but a line crossing with one of the edges of a_i^e exists. On the right, the edges of a_i^q and a_i^e do not intersect, but the edge between a_i^q and the middle vertex of the "boomerang" shaped quadrangle crosses through the quadrangle. So, in both cases, we have a non-planar drawing.

Parallel inner quadrangle

If the inner quadrangle has parallel edges, it would be a parallelogram, a rectangle, or a trapezoid, and then the placement of the quadrangle-apexes a_i^q would be problematic. We discussed above in Lemma 3.5 that a_i^q must be placed in the intersection of half-planes of opposing sides of the quadrangle. (This was also visualized in Figures 3.13 and 3.14.) If two of these opposed sides of the quadrangle are parallel, they only intersect at infinity, which we cannot draw in our PGEO-SP drawing. Therefore, having an inner quadrangle with parallel sides leads to either six of the apex-structures S_i not being able to be drawn together with both quadrangles with straight-line edges without introducing line crossings in the case of a trapezoid or even all eight of the apex-structures S_i in the case of a rectangle or parallelogram.

Other positions for the quadrangles

We assumed that the quadrangles must be placed inside each other, creating an inner and an outer quadrangle. If the two quadrangles are not placed inside each other, they must be placed next to each other in the outer face. In Lemma 3.1 we showed that the apex-structures S_i must be placed in the face between the two quadrangles. This means that each S_i must lie outside of both quadrangles in the outer face of our drawing.

In our proof dealing with quadrangles inside of each other, we created the contradicting construction starting from the apex-triangles A_i. We now have to show that even when the quadrangles lie next to each other, one of the quadrangles will always form the apex-triangles A_i for $i \in [4]$ when drawn with the apex-structures S_i.

Now, when the quadrangles do not lie inside each other but next to each other, two possible scenarios can occur when drawing a structure S_i with one of the quadrangles

(w.l.o.g. with Q). Either both apex vertices are part of the outer face of the newly drawn structure S_i or the edge-apex a_i^e lies inside one of the triangular faces that the quadrangle-apex a_i^q defines with the quadrangle Q, as can be seen in Figure 3.19. Now, in the first case, S_i and Q already form an apex-triangle A_i, and the other quadrangle Q' must also lie in that outer face, so also S_i and Q' form an apex-triangle A_i'. In the second case, the other quadrangle Q' must be placed into one of the triangular inner faces, with one vertex of Q, a_i^q, and a_i^e as the vertices that defined that face before Q' was placed into it. Now we can apply the Lemmas 3.3 and 3.4 in the same manner as in the inside-outside case of our proof, and we get that now Q' and S_i also build an apex-triangle.

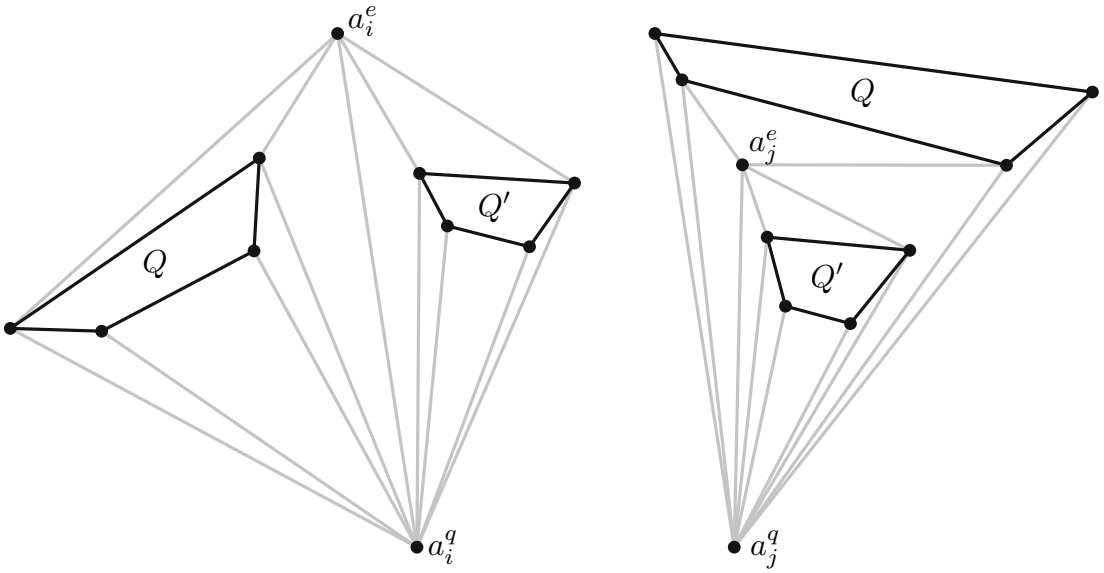


Figure 3.19: This figure shows the two possible ways to place two quadrangles next to each other and connect the apex vertices to them.

Lemma 3.7 requires that one quadrangle forms apex-triangles with all structures S_i for $i \in [4]$. So far, we have shown that when we draw two quadrangles next to each other together with the structure S_i, one of the quadrangles and the structure S_i will always form an apex-triangle. What remains to be seen is whether the same quadrangle always forms an apex-triangle. From the situations we just discussed, only once does a quadrangle Q and a structure S_i not form an apex-triangle A_i. That was where the edge-apex a_i^e lies inside one of the triangular faces that the quadrangle-apex a_i^q defines with the quadrangle Q. Now the question becomes, if for one structure S_i the edge-apex a_i^e lies inside one of the triangular faces that the quadrangle-apex a_i^q defines with the quadrangle Q, is it then possible that for another structure S_j, where $i \neq j$, the edge-apex a_j^e lies inside one of the triangular faces that the quadrangle-apex a_j^q defines with the quadrangle Q'. Or, in simpler terms, if S_i and Q define a face that includes Q', is it possible for a different S_j, that Q' and S_j define a face that includes Q?

First, we define a "smaller" relation ($\prec$) between two polygons. Imagine a convex hull over two polygons that do not intersect each other. For the $\prec$-relation, we examine the edges of the convex hull that connect the two polygons. Now, there are two possible cases: either there are two such lines, then we can use the distance of the polygons to the intersection point of these two lines to determine which polygon is bigger, or there is no such line. The last case can only happen when one polygon is concave, and the other polygon lies wholly "within" the concave part of the first polygon. Both cases of this $\prec$ relation are also represented in Figure 3.20.

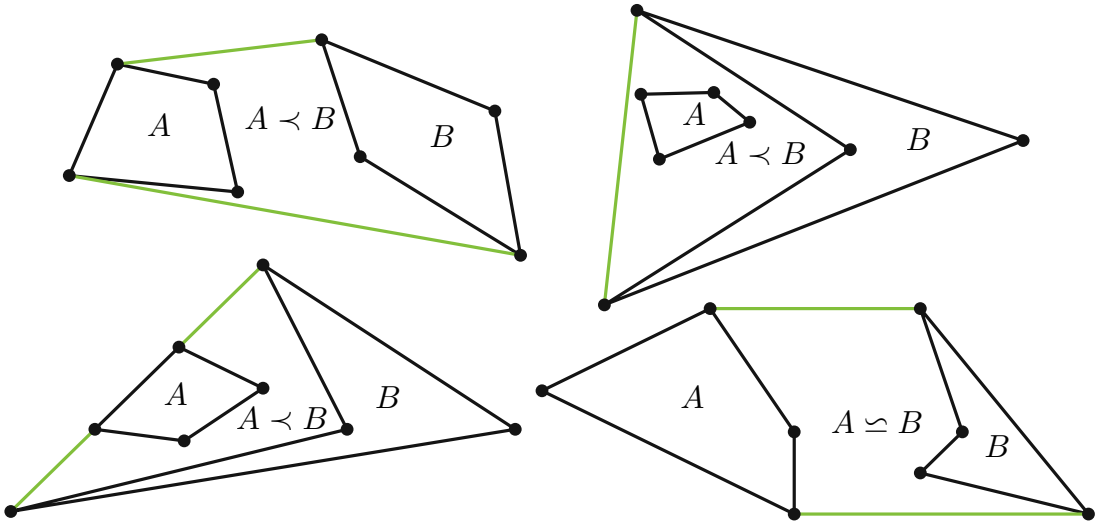


Figure 3.20: This figure shows the four possible cases of the $\prec$-relation, defined through the convex hull over two non-intersecting polygons A and B. In green are the non-input-polygon edges of the convex hull.

Formal definition of the $\prec$-relation between two polygons

In a convex hull $C_{A,B}$ over two non-intersecting polygons A and B, each of the vertices of $C_{A,B}$ belongs to either polygon A or polygon B. As we know, a convex hull always returns a clockwise order of its vertices, so we can give the vertices of the convex hull a numbering based on their order and the polygon they originate from. The vertices are called v_i^P, where P is the polygon they belong to and i is their respective index individually counted for each polygon. V_P is the set of vertices of polygon P.

We can distinguish two cases, one where all vertices of the convex hull only belong to one of the polygons, so $V_A \cap V_{C_{A,B}} = \emptyset \vee V_B \cap V_{C_{A,B}} = \emptyset$, or the case where both polygons contribute to the vertices of the convex hull, so $V_A \cap V_{C_{A,B}} \neq \emptyset \wedge V_B \cap V_{C_{A,B}} \neq \emptyset$. (If it is clear from context which convex hull is meant, we will omit its indices.)

In the first case, where only one polygon contributes to the vertices of the convex hull $C_{A,B}$, so $V_A \cap V_C = \emptyset \vee V_B \cap V_C = \emptyset$ holds, we say that polygon A is "bigger" than polygon B, written $A \succ B$ (or equally B is "smaller" than polygon A, written $B \prec A$) if no vertex of B is part of the convex hull $C_{A,B}$, so $V_B \cap V_C = \emptyset$ holds.

In the second case, where both polygons contribute vertices to the convex hull, so $V_A \cap V_C \neq \emptyset \wedge V_B \cap V_C \neq \emptyset$ holds, as the polygons A and B do not intersect each other we get that there are precisely two non-input-polygon edges (v_i^A, v_j^B) of the convex hull $C_{A,B}$, such that one vertex of the edge belongs to polygon A and the other vertex belongs to polygon B.These edges can either be parallel or not.

If these edges are not parallel, we can use an extension of those edges to determine which polygon is "bigger". Two non-parallel lines must always intersect in Euclidean space; let us call that intersection point p. Now, we can compare the distance between the points p and v_i^A to the distance between p and v_j^B. $A \succ B$ (or equally $B \prec A$), if the distance between the intersection point p and the vertex v_i^A is bigger than the distance between p and the vertex v_j^B. So it holds that $|v_i^A - p| > |v_j^B - p|$.

One polygon A is "equal sized" to another polygon B, written $A \backsimeq B$, if and only if the two non-input-polygon edges (v_i^A, v_j^B) of the convex hull $C_{A,B}$ are parallel. However, the extensions of those edges are not identical.

In the edge case, where the two non-input-polygon edges $(v_i^A, v_j^B), (v_k^A, v_l^B)$ of the convex hull $C_{A,B}$ are parallel and the extensions of those edges are identical, then $A \succ B$ (or equally $B \prec A$), if the distance between the two vertices on that line that belong to A is bigger than the distance between the two vertices on that line that belong to B, so the following equality holds: $|v_i^A - v_k^A| > |v_j^B - v_l^B|$.

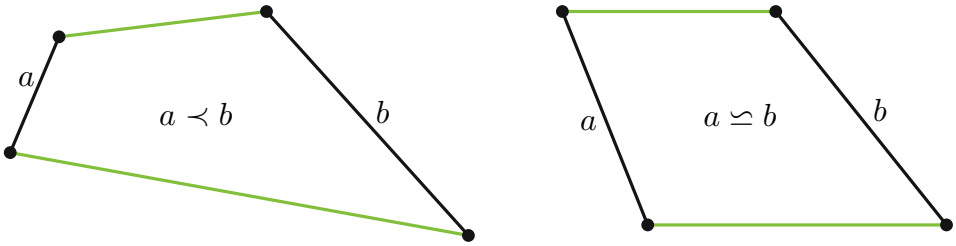


Figure 3.21: This figure shows the two possible cases of the $\prec$-relation between two line segments, defined through the convex hull over two non-crossing line segments a and b. In green are the non-input edges of the convex hull.

Formal definition of the $\prec$-relation between two line segments.
Similar to the "smaller" relation between two polygons, we can define a "smaller" relation between two line segments that are not crossing and not part of the same line. Let us call the line segments a and b, where $a = (v_1^a, v_2^a)$ and $b = (v_1^b, v_2^b)$. The convex hull $C_{a,b}$ over a and b always has the edges $\{a, (v_1^a, v_1^b), b, (v_2^b, v_2^a)\}$. Both cases of this relation are represented in Figure 3.21.

A line segment a is "equal sized" to a line segment b, written $a \backsimeq b$, if and only if the edges (v_1^a, v_1^b) and (v_2^b, v_2^a) are parallel to each other.

If the edges (v_1^a, v_1^b) and (v_2^b, v_2^a) are not parallel, their extensions form an intersection point p. Now a line segment a is "bigger" than a line segment b, written $a \succ b$ (or equally

b is "smaller" than line segment a, written $b \prec a$), if and only if the distance between p and v_1^a is bigger than the distance between p and v_1^b. So it holds that $|v_1^a - p| > |v_1^b - p|$.

Remark 3.3. We just witnessed that the cases of the formal definition for the $\prec$-relation between two line segments are a subset of the cases covered in the formal definition of the $\prec$-relation between two polygons. The missing cases are the ones that cover concave polygons. This is no issue, as line segments cannot be concave. So, a $\prec$-relation between a line segment and a polygon is also well-defined when combining both formal definitions.

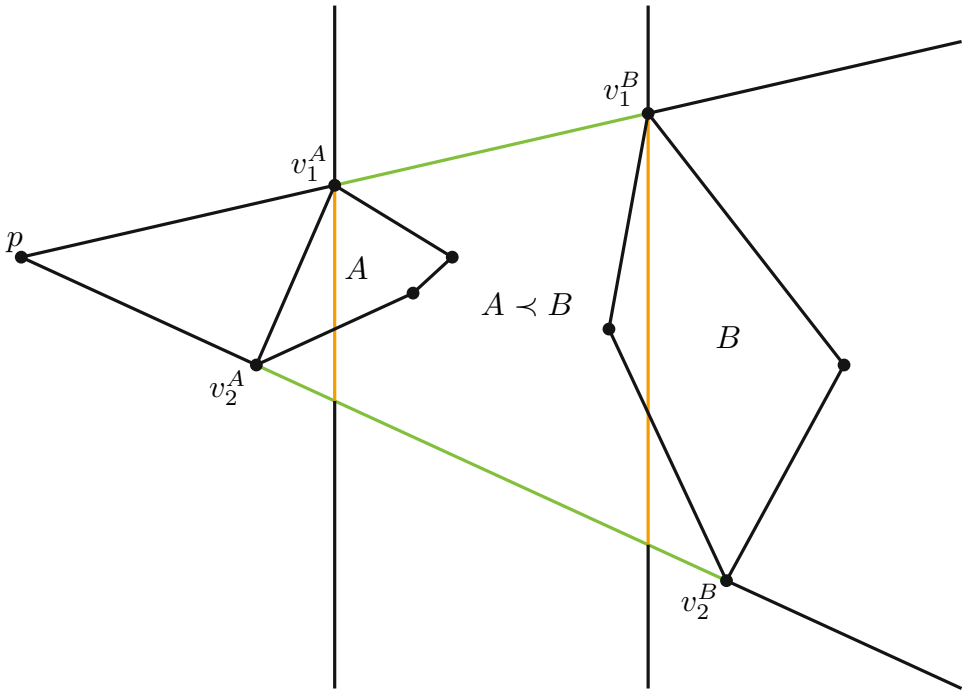


Figure 3.22: This figure shows how Thales theorem can be applied to determine which one of two polygons is bigger by comparison of two parallel lines going through the polygons.

Remark 3.4. The most common case that one polygon or line segment A is bigger than another B is when there are two non-input-polygon edges (v_i^A, v_i^B) of the convex hull $C_{A,B}$ which are not parallel. The definition used the intersection point between these lines and the distance from the intersection point to the endpoints of these edges to determine which one of the polygons is bigger. Using Thales's theorem, it is also possible to draw two parallel lines through v_i^A and v_i^B, measure the length of the segment of these lines that lies between the two edges, and use these lengths to determine which of the polygons is bigger. The polygon with the longer line segment is the bigger one. This remark is visualized in Figure 3.22.

Remark 3.5. The last case of the definition, in which the two non-input-polygon edges of the convex hull C are parallel to each other and their extended lines are identical, can only occur when one of the polygons is concave, the other polygon lies inside of the

"concave" part of the first polygon, and a vertex or edge of the second polygon lies on the convex hull of the first polygon. In this scenario, the first polygon is obviously the bigger one.

Definition 3. *If for two polygons A and B, it holds that A and B are of equal size, $A \backsimeq B$, or that A is "smaller" than B, $A \prec B$, then we can also write $A \preceq B$. This is also called A is "smaller than or equal" to B.*

Remark 3.6. The $\preceq$-relation between non-intersecting polygons and lines has the following properties: it is reflexive ($\forall p \in \mathcal{P}, p \preceq p$), antisymmetric ($\forall p, q \in \mathcal{P}, p \preceq q \wedge q \preceq p \Rightarrow p = q$), and strongly connected ($\forall p, q \in \mathcal{P}, p \preceq q \vee q \preceq p$). Interestingly, it is not a transitive relation and, therefore, does not define an order over polygons and lines. A counterexample against the transitivity of the $\preceq$-relation is shown in Figure 3.23. Proofs for these properties are omitted as these are relatively trivial.

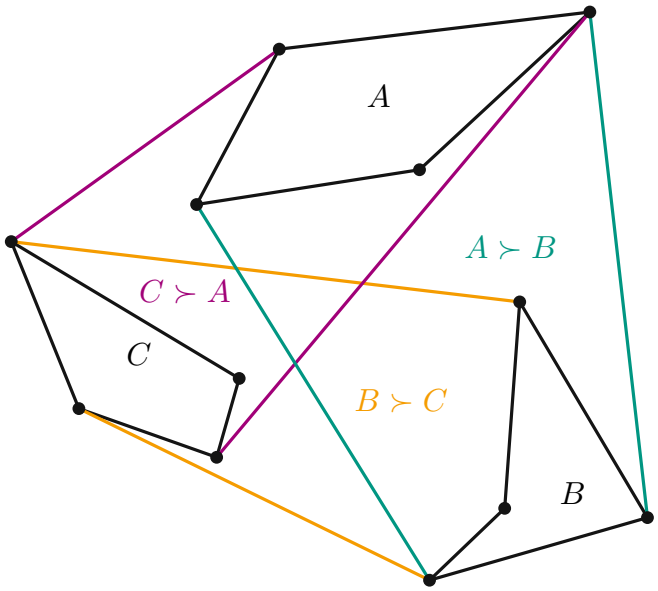


Figure 3.23: An example showing that the $\preceq$-relation is not transitive. In dark green, we see that $A \succ B$ holds; in orange, we see that $B \succ C$ holds, which, according to transitivity, should give us $A \succ C$. However, in purple, we see that the reverse is true, and $C \succ A$ holds.

We must also define a direction towards a vertex in a triangular face and a width measure inside a triangular face towards a point. This is also visually represented in Figure 3.24.

Definition 4. *The direction of a triangular face f towards one of its incident vertices v is the directional vector going from the middle of the edge e opposite to v directly to v. Taking a point p inside of f and laying a line parallel to e through p, then the width of the face at p in direction $\vec{v}$ is the distance from one side of the face to the other along this parallel line.*

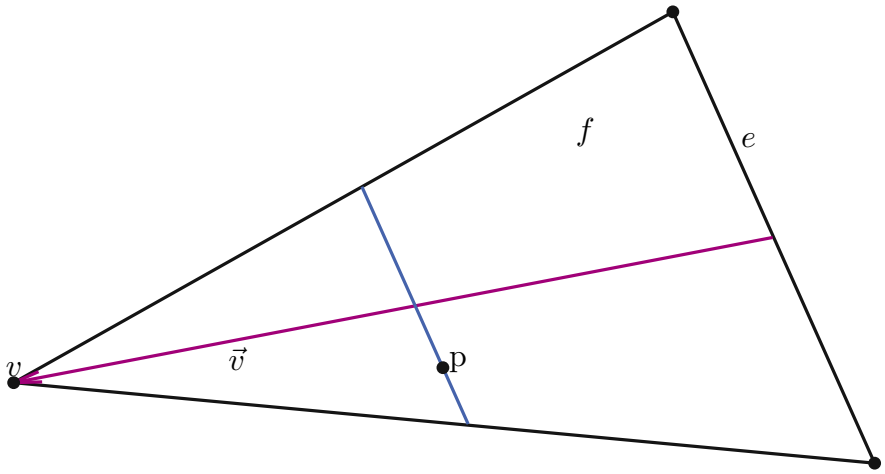


Figure 3.24: This figure shows a triangular face f, with one edge e and an opposing vertex v. Then, the direction of f towards v is represented by the purple vector $\vec{v}$. The blue line represents the width of the face at point p in direction $\vec{v}$.

We can now apply these new definitions in the following lemmas.

Lemma 3.8. *If for a single edge e_B of a polygon B it already holds that $A \prec e_B$, where A is another polygon, then $A \prec B$ must also hold.*

Proof. We already know from our assumption that $A \prec e_B$. As e_B is only an edge, we can be sure that $V_A \cap V_C \neq \emptyset$ for the convex hull C_{A,e_B} between the polygon A and the edge e_B. Also, the extensions of the non-input-polygon edges $(v_i^A, v_j^{e_B})$ of C_{A,e_B} cannot be identical, as for this scenario to occur, e_B would need to be concave, which a single edge cannot be. Therefore, only one case of the definition of the $\prec$-relation remains. So we know the edge e_B must be bigger than the polygon A because the non-input-polygon edges of the convex hull C_{A,e_B} have an intersection point p and the distance from p to v_j^A must be smaller than the distance from p to $v_i^{e_B}$. Applying Remark 3.4, we also know that drawing a line parallel to e_B through any point of A, let us take the vertex v_i^A, and measure the length from one end of the convex hull to the other will result in a line segment smaller than e_B.

Now, regarding the convex hull $C_{A,B}$ over the polygons A and B, there are two possibilities: either both endpoints of e_B are still part of $C_{A,B}$ or they are not. In the first case, the two non-input-polygon edges (v_i^A, v_j^B) of $C_{A,B}$ are the same as the two non-input-polygon edges of C_{A,e_B}. As the same edges form the same extended lines, they also share the same intersection point p, and therefore, it still holds that $|v_i^A - p| > |v_j^B - p|$. In the second case, where one or both vertices of e_B are not part of the convex hull $C_{A,B}$, this must mean that e_B lies within the convex hull $C_{A,B}$ and that the convex hull along the direction of e_B became bigger than the length of e_B. We know that the polygon A stayed the same, so the line segment going through e_B is bigger than a parallel line segment

going through v_i^A. Applying Thales' theorem and Remark 3.4, we again get $A \succ B$. So, in both cases, we get the desired result that $A \succ B$ holds. □

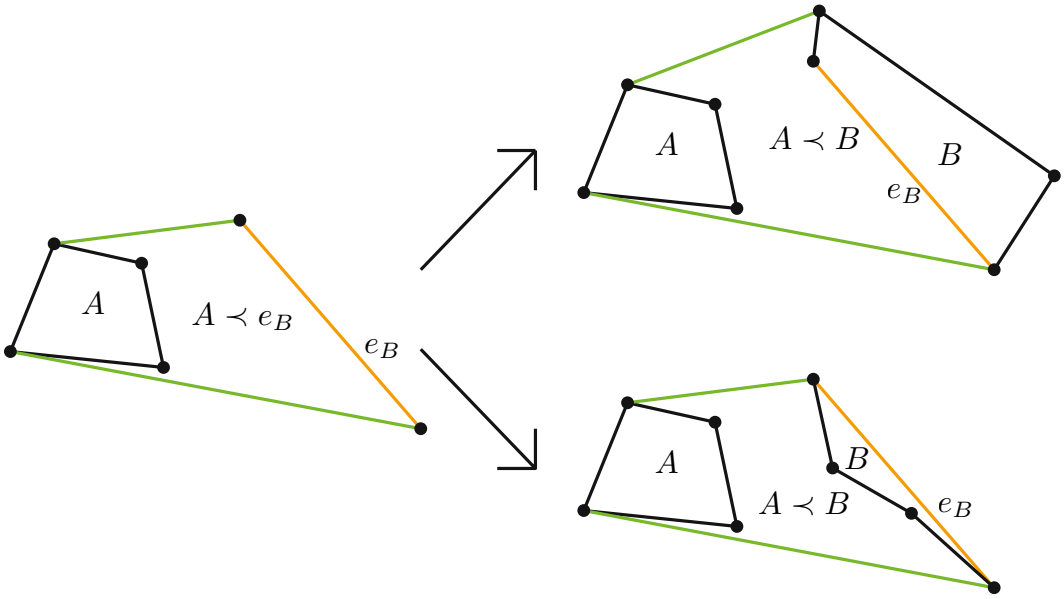


Figure 3.25: On the left, we see an edge e_B bigger than the polygon A. On the right, we see the two possibilities for a polygon B that contains the edge e_B.

Lemma 3.9. *If a polygon P' lies entirely inside a triangular face f, then if one of the edges of f belongs to a polygon P, it holds that $P' \prec P$.*

Proof. Per assumption, we know that there is a triangular face f with one edge e that belongs to polygon P. As f is triangular, one vertex of f is not part of e but lies on the opposite side of the face; let us call this vertex v. Furthermore, the width of the face f in direction $\vec{v}$ is at most the length of e. We also know that the polygon P' lies completely within the face f. This means that the width of the face f at any point of P' in direction $\vec{v}$ is always smaller than the length of e.

The $\prec$-relation between two polygons is defined through their convex hull. Let us start by considering the convex hull $C_{P',e}$ of the polygon P' and the edge e. As P' lies within the triangular face f, we can assume that the two non-input-polygon edges $(v_i^{P'}, v_j^e)$ of $C_{P',e}$ have an intersection point p. We already established that the width of f at any point of P' in direction $\vec{v}$ is smaller than e, so this also holds at vertex $v_i^{P'}$. The line segment parallel to e going through $v_i^{P'}$ that goes from one end of the convex hull to the other can at most be as big as the width of f at $v_i^{P'}$, as P' lies wholly within f. Now we can apply Thales's theorem, and by Remark 3.4, we get that $P' \prec e$ holds. Furthermore, as e is an edge of the polygon P, we get from Lemma 3.8 that also $P' \prec P$ must hold.

□

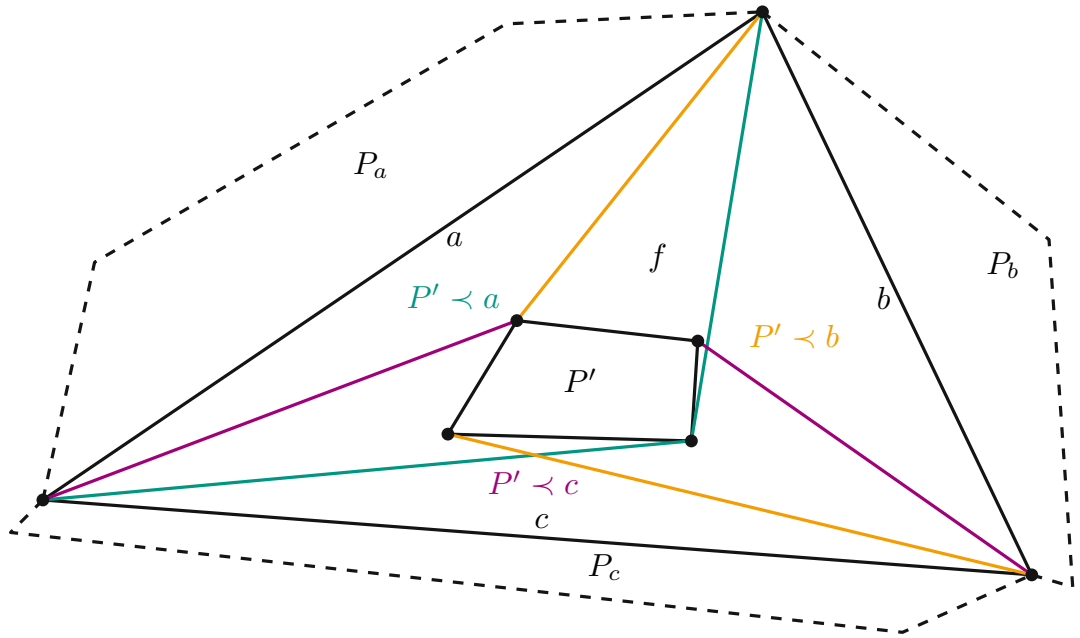


Figure 3.26: This figure shows a triangular face f, a polygon P' inside of f, and the convex hulls between P' and each of the edges of f separately. It is clear to see that the edges a, b, and c are bigger than P', and therefore by Lemma 3.8 that also any polygon P that contains one of a, b, or c as an edge is bigger than P'.

Lemma 3.10. *Given two quadrangles Q and Q' that are positioned next to each other as well as a structure S_i, where we are looking for a planar straight-line drawing, then the edge-apex a_i^e can only lie inside one of the triangular faces that the quadrangle-apex a_i^q defines with the quadrangle Q if $Q \succ Q'$.*

Proof. Every face defined by the quadrangle Q and the quadrangle-apex a_i^q is either a triangular face or the outer face. For the triangular faces, it holds that the faces reduce in width in the direction of the point a_i^q. We know that a_i^e lies inside one of these triangular faces by assumption; let us call it f. By definition of S_i, e_i of Q must then be the edge of f opposite to a_i^q. The apex structure S_i includes edges between Q' and a_i^e and edges between Q' and a_i^q, so the quadrangle must be placed in a face neighboring both apex vertices. Remember, we are looking for a planar straight-line drawing. As a_i^e was placed in the triangular face f, Q' must also be placed inside the former face f.

So we have that Q' lies inside the triangular face f, which on one border has the edge e_i of Q. Therefore, the preconditions for Lemma 3.9 are fulfilled, which means we can apply Lemma 3.9 and get that $Q \succ Q'$. This concludes the proof. □

Corollary 3.2. *Given the graph G, where the convex quadrangles Q and Q' lie next to each other and it holds that $Q \succ Q'$, then the structure S_i always forms an apex-triangle A_i with the quadrangle Q' for each $i \in [4]$.*

Proof. Based on Lemma 3.10, we know that for a structure S_i, the edge-apex a_i^e can only lie inside one of the triangular faces that the quadrangle-apex a_i^q defines with the quadrangle Q if Q is the "bigger" of the two quadrangles ($Q \succ Q'$). Now, as only one quadrangle can be "bigger" than the other, the reverse cannot be true ($Q' \not\succ Q$), and it is not possible to find a structure S_j, where $i \neq j$, such that the edge-apex a_j^e lies inside one of the faces that the quadrangle-apex a_j^q defines with the quadrangle Q'. So, if $Q \succ Q'$ holds, Q' will form an apex-triangle A_i with every structure S_i for $i \in [4]$. □

Now we can apply Lemma 3.7 to Q' and the apex-triangles A_i for $i \in [4]$ and again get that there are two apex-structures S_i for which no planar geometric embedding can be found. So, even allowing other positions for the quadrangles Q and Q' does not change the outcome of our proof. A planar geometric storyplan for G still does not exist.

3.3.4 The contradiction

Lemma 3.11. *The graph G does not admit a planar geometric storyplan.*

Proof. The graph G contains two quadrangles Q and Q' and eight structures S_i for $i \in [8]$. We can now differentiate the different possible positions of these two quadrangles. First distinction, they can either *1)* be situated next to each other or *2)* one can lie inside the other. In the case *1)*, we can again differentiate whether both quadrangles are *1.a)* of the same size or *1.b)* of different sizes. For case *1.a)*, we established in Subsubsection 3.3.3 (*Other positions for the quadrangles*) that the only way that one quadrangle (w.l.o.g. Q') does not form apex-triangles with the apex vertices is in the case that the edge-apex a_i^e lies inside one of the triangular faces that the quadrangle-apex a_i^q defines with Q'. However, by Lemma 3.10, we know this can only occur if $Q' \succ Q$ holds, which contradicts our assumption that both quadrangles are of equal size. Therefore, both quadrangles form eight apex-triangles A_i with the structures S_i, one for each $i \in [8]$. Four of these apex-triangles cannot be drawn planarly using straight-lines (see Lemma 3.7). In case *1.b)*, we have that one of the quadrangles is bigger, w.l.o.g. $Q \prec Q'$ holds. Now the smaller quadrangle Q, can either be *1.b.a)* convex or *1.b.b)* concave. Similarly, in case *2)*, the quadrangle on the inside, w.l.o.g. Q, can again either be *2.a)* convex or *2.b)* concave. For the convex cases *1.b.a)* and *2.a)*, we know by the Corollaries 3.1 (case *2.a*) and 3.2 (case *1.b.a*) that Q forms apex-triangles A_i with the structures S_i for every $i \in [8]$. Again, by Lemma 3.7, we have that four of these apex-triangles cannot be drawn planarly using straight-lines. For the concave cases *1.b.b)* and *2.b)*, we have shown in Subsubsection 3.3.3 (*Concave inner quadrangle*) that again four of the eight structures S_i cannot be drawn planarly using straight-lines.

So we have shown that no matter how we place the two quadrangles and in which form they are, always four of the structures S_i cannot be drawn planarly using straight-lines.

In Subsection 3.3.2 (*Applying the line crossings to a frame*) we have shown that for six of the eight structures, one frame must exist such that the whole structure S_i is active.

Combining the last two results, we get that at least two problematic structures must occur in any possible drawing of a storyplan of G. As these two structures S_i cannot be drawn with straight-lines without introducing line-crossings, we get that the graph G does not admit a planar geometric storyplan. □

3.4 Conclusion of the proof

Theorem 1. *The Planar Topological StoryPlan problem and the Planar Geometric StoryPlan problem are two distinct problems.*

Proof. By showing that a graph G exists (for its definition see Section 3.1) that admits a planar topological storyplan (see Section 3.2), but that does not admit a planar topological storyplan (see Section 3.3), we proved that the yes-instances of the Planar Geometric StoryPlan problem are a proper subset of the yes-instances of the Planar Topological StoryPlan problem, which in extension establishes that the Planar Topological StoryPlan problem and the Planar Geometric StoryPlan problem themselves are two distinct problems. □

3.4.1 Why choose quadrangles and not other polygons?

The same proof with an inner and an outer polygon connected by structures S_i should also work with little adaption for any n-gon with $n > 4$. The last step of constructing the contradiction would need to be adapted, as polygons with more than four sides have more cases for different structures of the inner n-gon that need to be considered. Lemma 3.5 must still be applied to find proper placements for the quadrangle-apexes a_i^q, but as long as it can be shown that at least two sides of the n-gon lead to invalid positions, a contradiction can be found. We expect this to work for any n-gon with $n > 4$, but as the proof would need to cover more variations in the structure of the inner n-gon, we decided to stick to the smallest n-gon for which such a proof could be found.

Using triangles does not work. For the structures we use, there always exists a way to draw them without any intersections when applied to a triangle. We know from Lemma 3.5 that we can always find a valid placement for a_i^q as long as the outer half-spaces defined by the incident edges of the edge $e_{\phi(i)}$ intersect in the half-space $h^-_{e_{\phi(i)}}$. As the incident edges themselves always intersect in the third vertex of the triangle, this always holds for all three edges of a triangle. An example of such a valid drawing of the apex-structures S_i and the inner triangle T is also shown in Figure 3.27.

So, quadrangles are the smallest n-gons for which our proof works, which is exactly why we used them.

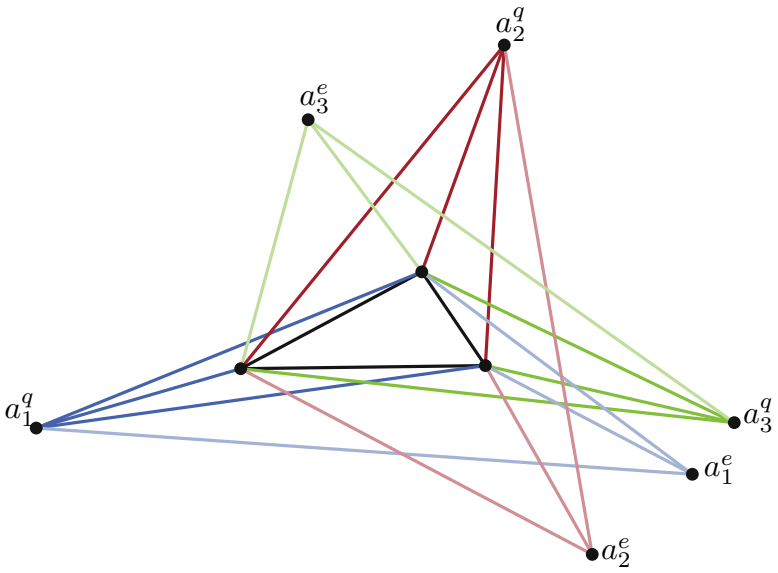


Figure 3.27: Showing a possible drawing of the apex-structures S_i for a triangle. The important part here is that there are no intersections between any of the edges connecting the quadrangle-apexes a_i^q to the triangle and the edges that define the triangle itself.

3.4.2 Why do we need two quadrangles?

As we saw in the proof above, planar geometric frames can easily be found for apex-structures S_i and one quadrangle Q. Either place the structures inside of Q or outside of Q so that the edge-apex a_i^e does not lie in the outer face of the drawing.

We needed a second quadrangle Q' to enforce that for one of the quadrangles, the apex-structures S_i must be drawn on the outside of that quadrangle in such a way that both apex vertices lie on different sides of that quadrangle. Given that situation, we then could apply the Lemma 3.7 to show that there are apex-structures S_i for which no planar geometric drawing can be found and use the knowledge of Remark 3.2, which tells us that therefore no planar geometric storyplan can exist for the graph G.

We could have also found a different construction that uses more than two quadrangles, but as two were already sufficient for our proof, we used exactly two to keep the proof as simple as possible. We needed to use two quadrangles and could not just use one quadrangle and one simpler polygon like a triangle because in a storyplan, we are concerned with the existence of a drawing, which means we cannot decide which polygon is placed where, so we needed two polygons that, under the right circumstances, each could lead to an impossible drawing and where the right circumstances are precisely the existence of another quadrangle.

CHAPTER 4

NP-hardness of PGEO-SP

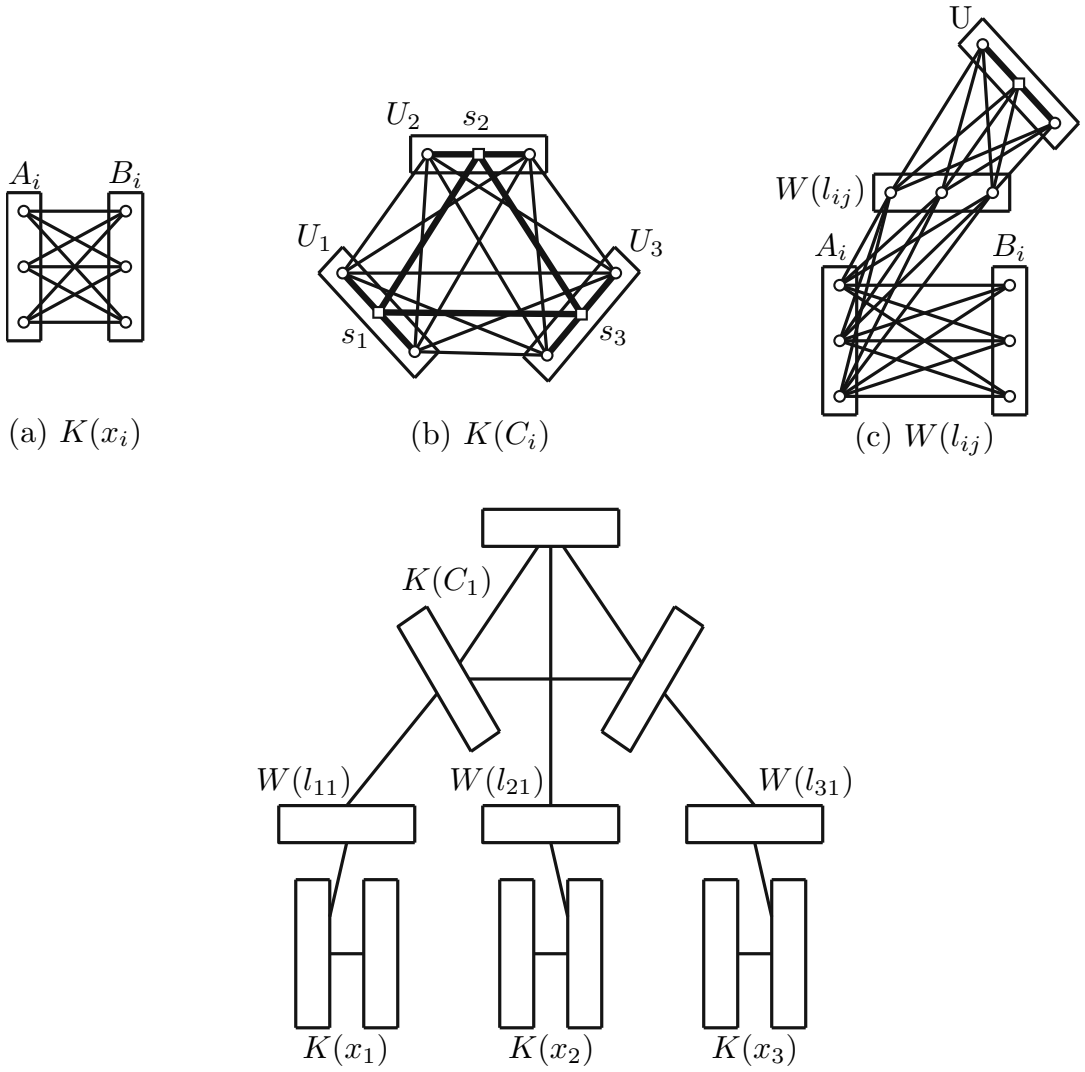
In this chapter, we look at the complexity of the Planar Geometric StoryPlan problem. Binucci et al. not only established PTOP-SP but also proved that it is NP-complete in the same paper [2]. We will prove that even though PGEO-SP is a different problem than PTOP-SP, it is still NP-hard. We will also discuss why the NP-completeness of PGEO-SP is uncertain.

To prove the NP-hardness of PGEO-SP, we will recreate the proof of the NP-hardness of PTOP-SP as presented in [2]. The original proof is a reduction proof from One-In-Three 3SAT to PTOP-SP. One-In-Three 3SAT is a variant of 3SAT, which asks whether there is a satisfying assignment in which exactly one of the three literals in each clause is true. We will now adapt this proof to a reduction from One-In-Three 3SAT to PGEO-SP. The aspects of the proof that do not concern explicit drawings, like the formal definitions, can be taken one-to-one from the original proof. However, the aspects of the proof that use gadgets to show the reduction must be adapted.

We define a formula ϕ as a 3SAT formula over N variables $\{x_i\}_{i\in[N]}$ and M clauses $\{C_i\}_{i\in[M]}$. We then construct an instance of PGEO-SP based on a satisfying assignment with exactly one true literal in each clause of that formula ϕ, i.e., a graph $G = (V, E)$ consisting of clause, variable, and wire gadgets, which are built as follows. (The definitions for the gadgets are taken verbatim from [2].) See Figure 4.1 for a visual representation of these gadgets.

Variable Gadget Each variable x_i is represented in G by a copy $K(x_i)$ of $K_{3,3}$ (see Figure 4.1(a)). Let A_i and B_i be the two partite sets of $K(x_i)$, which we call the *v-sides* of $K(x_i)$. A true (false) assignment of x_i will correspond to set A_i being flexible (fixed) in a putative storyplan of G (see Definition 2).

Clause Gadget Consider a copy of $K_{2,2,2} = (U_1 \cup U_2 \cup U_3, F)$. An extended $K_{2,2,2}$ is the graph obtained from any such a copy by adding three vertices s_1, s_2, s_3, such that these three vertices are pairwise adjacent, and each s_j is adjacent to both vertices in



(a) $K(x_i)$ (b) $K(C_i)$ (c) $W(l_{ij})$

(d) Schematization of $(x_1 \vee \neg x_2 \vee \neg x_3)$

Figure 4.1: Illustration of the various types of gadgets and how they are connected.

U_j, for $j \in \{1, 2, 3\}$. In the following, s_1, s_2, s_3 are the special vertices of the extended $K_{2,2,2}$, while the other vertices are the simple vertices. A clause C_i is represented in G by an extended $K_{2,2,2}$, denoted by $K(C_i)$ (see Figure 4.1(b)). In particular, we call each of the three sets of vertices $U_j \cup \{s_j\}$ a c-side of $K(C_i)$. The idea is that $K(C_i)$ admits a storyplan if and only if exactly one c-side is flexible (each c-side will be part of a $K_{3,3}$, see the wire gadget below).

Wire Gadget Refer to Figure 4.1(c). Let x_i be a variable having a literal l_{ij} in a clause C_j. Any such variable-clause incidence is represented in G by a set of three vertices,

which we call the w-side $W(l_{ij})$. All vertices of $W(l_{ij})$ are connected to all vertices of one of the three c-sides of $K(C_j)$, which we call U, such that the graph induced by $W(l_{ij}) \cup U$ in G contains a copy of $K_{3,3}$. Also, each vertex of $W(l_{ij})$ is connected to all vertices of the v-side A_i (B_i) if the literal is positive (negative), such that the graph induced by $W(l_{ij}) \cup A$ ($W(l_{ij}) \cup B$) in G is a copy of $K_{3,3}$. Also, note that each c-side of $K(C_j)$ is adjacent to exactly one w-side.

The following lemma and its proof is no new result but taken from the original storyplan paper [2]. We did not want to omit this part of the proof, as it helps to form a complete picture of the NP-hardness of the Planar Geometric StoryPlan problem. The wording of the proof of the following lemma is our own.

Lemma 4.1. *If graph G admits a planar geometric storyplan, then ϕ admits a satisfying assignment with exactly one true literal in each clause.*

Proof. Let $S = \langle \tau, \{D_i\}_{i \in [n]} \rangle$ be a planar geometric storyplan of G. We can create a 3SAT formula ϕ based on the clause and variable gadgets appearing in the storyplan S. As hinted at in the definition of the variable gadget above, we assign each variable a truth value based upon which of its v-sides is flexible and which is fixed. If the v-side A_i is flexible (fixed) in S and the v-side B_i is fixed (flexible) in S, then the variable x_i is assigned the value *true* ($false$).

Each variable gadget is connected to at least one w-side of a wire gadget, which is connected to exactly one c-side of a clause gadget. By definition 2, we know that each $K_{3,3}$ always has one fixed and one flexible side. As the wire gadget consists of two connected $K_{3,3}$'s, the "status" of the v-side is propagated through the wire gadget to the c-side consistently, such that the c-side is flexible (fixed) if and only if the v-side with which it is connected via the wire-gadget is also flexible (fixed). We also know that the w-side is connected to the v-side A_i if the literal is positive in the clause and to the v-side B_i if the literal is negated in the clause. Combining these two facts, we get that if the literal is positive and if we assign the variable the value *true* ($false$), as its v-side A_i is flexible (fixed), then the connected c-side must also be flexible (fixed) and corresponds to a *true* ($false$) literal. If, on the other hand, the literal is negative and if we assigned the variable the value $false$ (*true*), as its v-side B_i is flexible (fixed), then the connected c-side is flexible (fixed) and corresponds to the literal having the value $\neg false$ ($\neg true$) which is equal to the value *true* ($false$). So we get that the c-side of a clause gadget is flexible if and only if the literal itself is *true*.

To prove that our storyplan S leads to a 3SAT formula ϕ having exactly one true literal in each clause, we need to show that each clause gadget must have exactly one flexible c-side.

Let us first assume that all c-sides of a clause gadget could be fixed so that we would have a clause with no true literal. As discussed before, a clause gadget consists of a $K_{2,2,2}$ of so-called simple vertices, which are split into three partite sets U_1, U_2, and U_3 and special vertices s_1, s_2, and s_3, one for each partite set of the $K_{2,2,2}$. Each of these

special vertices s_j is connected to the vertices of its partite set U_j, and all special vertices are connected to each other. W.l.o.g., we can assume that the partite sets are ordered according to their appearance in the storyplan S. A fixed c-side $U_j \cup \{s_j\}$ means that the simple vertices in U_j and the special vertex s_j must all appear in at least one frame together.

The clause gadget contains a $K_{2,2,2}$ and in a $K_{2,2,2}$, all of the partite sets are connected, which means that the simple vertices of U_1 and U_2 must still be active when the simple vertices of U_3 appear, as both simple vertices in U_3 share edges to the other four simple vertices. Therefore, the whole $K_{2,2,2}$ must appear in at least one frame of the storyplan S.

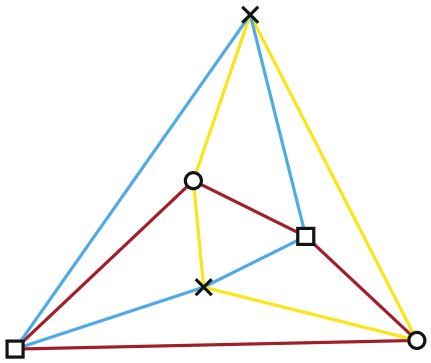


Figure 4.2: This plot shows a planar geometric embedding of a $K_{2,2,2}$. The vertices of each partite set are drawn using the same symbol (circle, square, and cross). It is visible that any pair of two partite sets creates a cycle (see the edges of the same color) that splits the vertices of the remaining partite set.

A $K_{2,2,2}$ is a maximal planar graph, meaning that each planar embedding is the same up to the choice of the outer face. Furthermore, the vertices of any two partite sets of the $K_{2,2,2}$ form a cycle. One of the vertices of the remaining partite set is always positioned inside of this cycle, and the other outside of it. This entails that vertices of the same partite set are never part of the same face. See Figure 4.2 for a visual representation of this property.

As in our current assumption, all c-sides are fixed, and all simple vertices must remain active until their respective special vertex appears in one frame. Each special vertex must share an edge with both of the simple vertices of their respective c-side. However, the simple vertices of one c-side do not share any faces. Therefore, we could not place the special vertex of the corresponding c-side anywhere without destroying planarity. As this holds for all three partite sets, we get that a clause gadget cannot be part of a planar storyplan if all three c-sides are fixed. So, our assumption must be false, and as a result, at least one c-side must be flexible.

For the argumentation from the other side, we assume that a clause gadget could have two flexible c-sides. A flexible c-side means that the two simple vertices of that c-side

cannot appear in the same frame. This follows as both simple vertices share an edge with their respective special vertex. Therefore, if both simple vertices appear together in one frame, they must stay active until their special vertex appears. At that point, all three vertices of the c-side are active in the same frame and thereby represent a fixed c-side, which contradicts the fact that this c-side was supposed to be flexible.

One of the simple vertices of the $K_{2,2,2}$ must be the last one that appears in the storyplan S, w.l.o.g. this simple vertex belongs to U_3. As this simple vertex is connected to all vertices of U_1 and U_2, these other simple vertices must all be active when the last simple vertex of U_3 appears. In return, this means that both c-sides corresponding to the partite sets U_1 and U_2 must be fixed.

Now, we got the results that at least one c-side must be flexible, and at least two c-sides must be fixed. Therefore exactly one c-side of each clause gadget must be flexible. This results in the formula ϕ having an assignment with exactly one true literal in each clause, precisely what we were looking for. □

The construction steps in the proof of the following lemma are based upon the corresponding construction for planar topological storyplans as given in [2], but were adapted by us to provide straight-line drawings.

Lemma 4.2. *If the formula ϕ admits a satisfying assignment with exactly one true literal in each clause, then graph G admits a planar geometric storyplan.*

Proof. We show this lemma by providing one possible way to construct a planar geometric storyplan based on a satisfying assignment of a One-In-Three 3SAT formula ϕ. We will draw one variable gadget for each variable in ϕ, as well as one clause gadget and three wire gadgets for each clause in ϕ. The question remains: how can we do so while adhering to the properties of a planar geometric storyplan? As a first step, before we can start drawing any gadgets, we must establish which sides of the gadgets must be fixed and which must be flexible. For each variable x_i, the v-side A_i is flexible (fixed) and the v-side B_i is fixed (flexible) if the variable x_i is true (false) in the satisfying assignment of ϕ. The wire and clause gadgets are then based on the truth-value assignments of the variables. The w-side $W(l_{ij})$ of a wire gadget is fixed (flexible) when connected to a flexible (fixed) v-side A_i (if the literal in clause C_j is positive) or B_i (if the literal in C_j is negative). Similarly, a c-side of a clause gadget C_j is flexible (fixed) when connected to a fixed (flexible) w-side $W(l_{ij})$. With that, we have determined which sides are fixed and flexible for all variable, wire, and clause gadgets so that we can start drawing the actual storyplan.

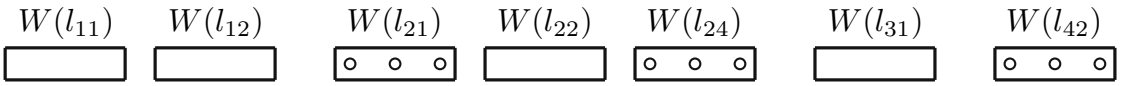
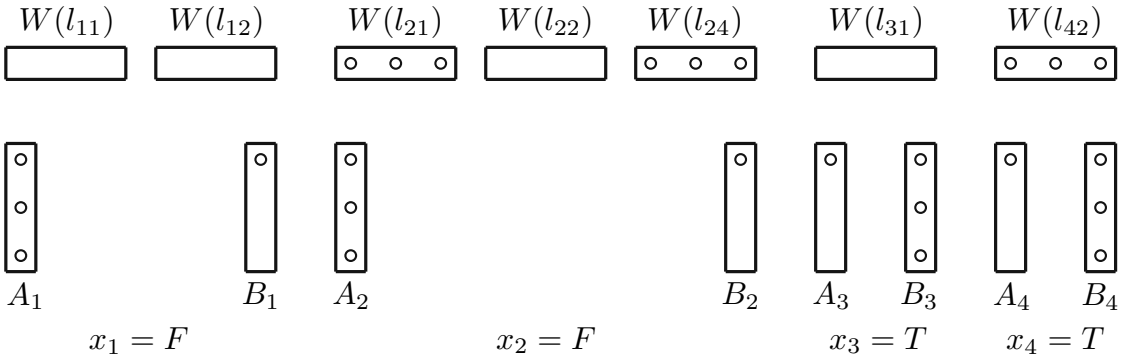


Figure 4.3: Proof of Lemma 4.2 drawing the vertices of the fixed w-sides.

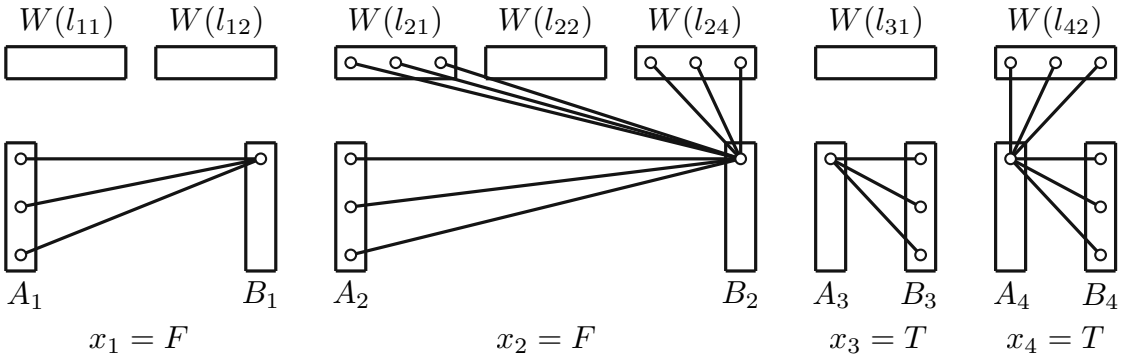
We start by drawing the wire gadgets. We reserve space for all wire gadgets on a horizontal line. (We will also refer to this line as the "wire-line".) The wire gadgets are ordered based on the variables they are connected with. So, the wire gadgets that connect to the variable x_1 are placed at the left end of the line, followed by the wire gadgets connecting to variable x_2, and so on, until the variable x_N is reached at the right end of the line. Initially, we only draw the fixed w-sides, while the space for the flexible w-sides is simply reserved, as seen in Figure 4.3.



$$(x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_2 \vee x_4 \vee \neg x_5) \wedge (x_1 \vee \neg x_5 \vee x_6) \wedge (\neg x_2 \vee \neg x_5 \vee \neg x_6)$$

Figure 4.4: Proof of Lemma 4.2 drawing the vertices of the fixed v-sides.

As a next step, the variable gadgets are drawn below the wire gadgets. The opposing v-sides of each variable gadget are drawn vertically below the starting and end points of the space reserved for their respective wire gadgets on the wire-line. By this construction, all variable gadgets are drawn horizontally next to each other. Again, only the fixed sides of each variable gadget are drawn initially, while the space for the flexible sides is just reserved at that point; see Figure 4.4.



$$(x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_2 \vee x_4 \vee \neg x_5) \wedge (x_1 \vee \neg x_5 \vee x_6) \wedge (\neg x_2 \vee \neg x_5 \vee \neg x_6)$$

Figure 4.5: Proof of Lemma 4.2 drawing the vertices of the flexible v-sides.

As flexible v-sides are only connected to fixed v-sides and fixed w-sides, which we have already placed, we can draw the flexible v-sides next; see Figure 4.5. After all flexible v-sides are drawn, the fixed v-sides not connected to any wire gadget can also disappear. (This concerns all true variables that only occur in positive literals and all false variables that only occur in negative literals).

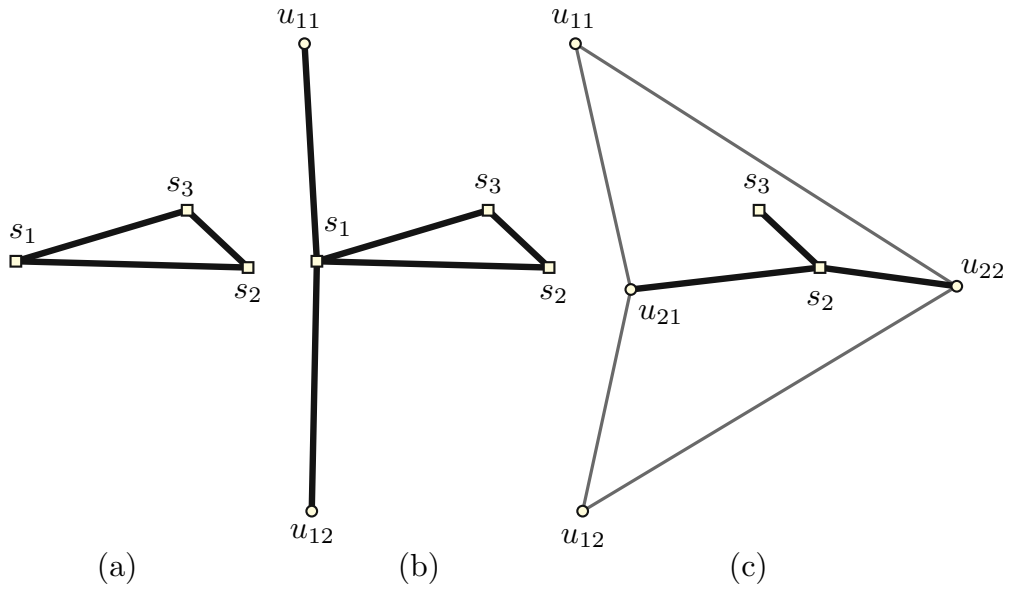


Figure 4.6: Proof of Lemma 4.2 drawing the first three steps of a clause gadget.

For the clause gadgets, we will first show how they can be drawn in five distinct steps and afterward explain how they are connected to the wire gadgets in each step. These five steps are shown in Figures 4.6 and 4.7.

The exact shape of the clause gadgets depends on the position of the variables in each clause. Generally, there is a wide range of options for the precise positions of the clause gadget. The construction will still work as long as planarity is maintained and only straight-lines are used for the edges. Therefore, rough instructions are sufficient for the construction steps.

In the first step, we draw the triangle of the three special and pairwise adjacent vertices s_1, s_2, s_3, and their connecting edges. The two special vertices that belong to the fixed c-sides or, as established, the false literals of the clause, are positioned on one horizontal level, and the special vertex belonging to the flexible c-side is positioned horizontally between but vertically slightly above them. (Figure 4.6(a)).

In the second step, we draw the two simple vertices of the first fixed c-side. Both simple vertices of that fixed c-side are drawn with some vertical distance above and below the triangle of special vertices (Figure 4.6(b)).

In the third step, the special vertex of the first fixed c-side disappears, and the two simple vertices of the second c-side appear. One is roughly positioned, where the special vertex

of the first fixed c-side was, and the other is drawn on a similar vertical level but on the other side of the two remaining special vertices. As these simple vertices are part of the $K_{2,2,2}$, they get connected to the other two simple vertices, forming a quadrangle with the two remaining special vertices positioned in the inner face of that quadrangle (Figure 4.6(c)).

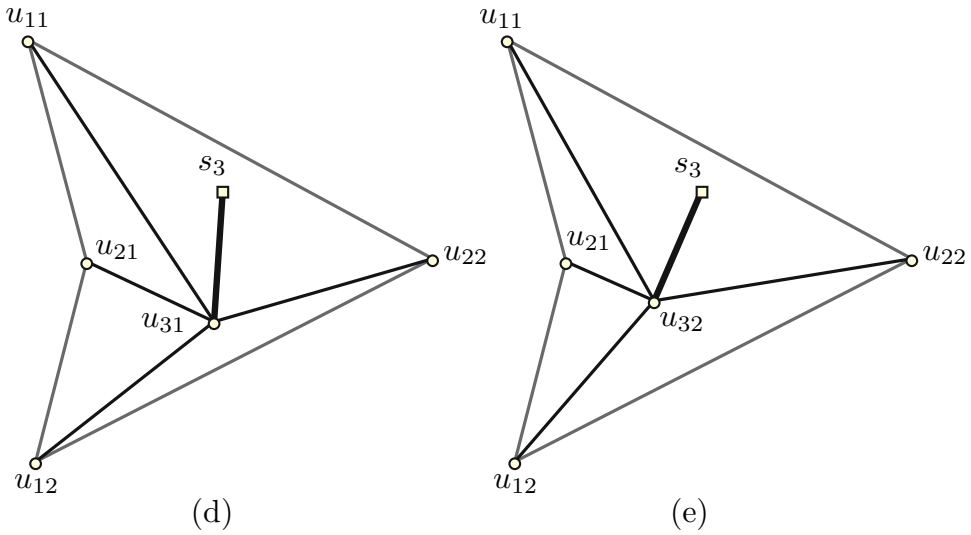


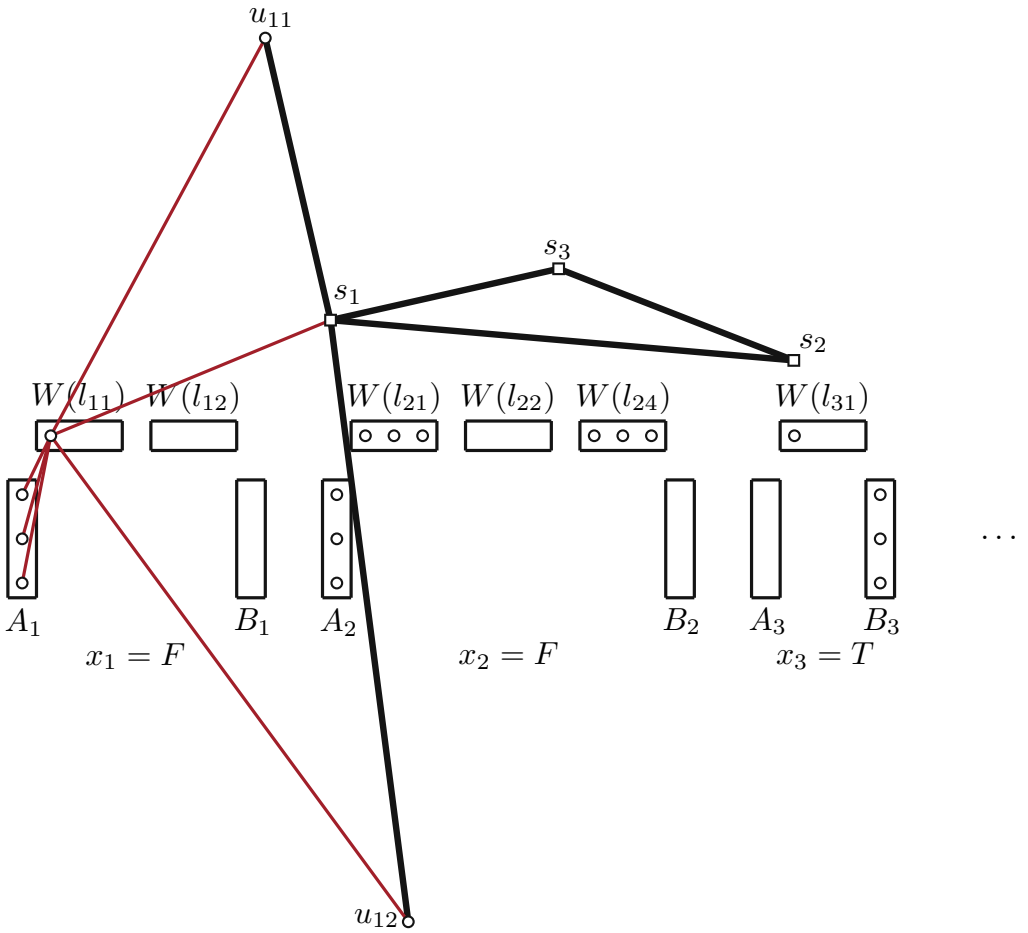
Figure 4.7: Proof of Lemma 4.2 drawing the last two steps of a clause gadget.

In the fourth step, the special vertex of the second c-side can now disappear, and the first of the two simple vertices of the flexible c-side appears. This vertex is placed somewhere inside of the face that is created by the simple vertices of the fixed c-sides, as it then can be connected to its special vertex and all of the other four simple vertices (Figure 4.7(d)).

In the fifth step, the first simple vertex of the flexible c-side disappears, and the second one appears. As it needs to be connected to the same vertices as the first one, it can simply appear at the same position from which the first one just disappeared (Figure 4.7(e)).

With these five steps, we have drawn the whole clause gadget planarly using only straight-line edges, thereby adhering to the limitations of the Planar Geometric StoryPlan problem.

We must show that the clause gadget and the wire gadgets can be combined without breaking planarity or resorting to Jordan arcs to draw the edges. We know each of the c-sides is connected to one w-side. We will review them individually in order of their appearance in the clause gadget. First, we will deal with the first fixed c-side, which has a vertical position. Placing the vertices of the flexible w-side of the connected wire gadget to the outer side of the fixed c-side, in regards to the clause gadget, allows us to draw straight-lines between the vertices of the flexible wire gadget and the fixed c-side. The vertically lower simple vertex of the fixed c-side must be positioned so as not to

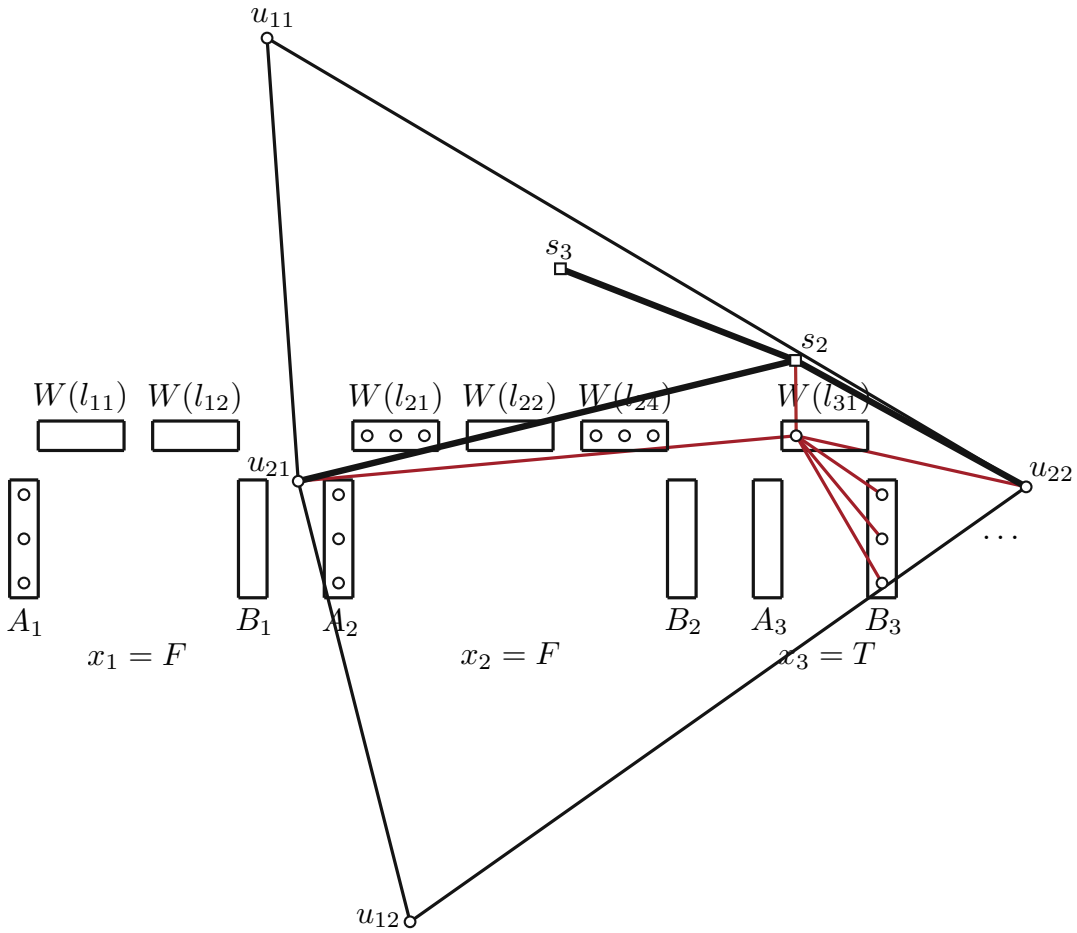


$$(x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_2 \vee x_4 \vee \neg x_5) \wedge (x_1 \vee \neg x_5 \vee x_6) \wedge (\neg x_2 \vee \neg x_5 \vee \neg x_6)$$

Figure 4.8: Proof of Lemma 4.2 drawing the first false literal.

interfere with the fixed v-side of its corresponding variable gadget that connects to the same flexible w-side. As each c-side of a clause gadget must only accommodate one variable, such a position can always be found and usually lies vertically further below the variable gadgets. When explaining the wire gadget, we mentioned that the vertices of all wire gadgets are drawn on one horizontal line. The triangle of the three special vertices of the clause gadget is positioned above that wire-line. The connection between the clause gadget and the first wire gadget is represented in Figure 4.8.

We will connect the second fixed c-side to its corresponding flexible w-side. Important to note here is that the special vertex of the second fixed c-side is inside the face created by the simple vertices of the fixed c-sides. Therefore, the w-side of its corresponding wire gadget and the fixed v-side of its corresponding variable gadget must also lie inside that face. Each vertex of the w-side appears after the other and connects to the fixed c-side and the fixed v-side of its connected variable gadget, as seen in Figure 4.9.

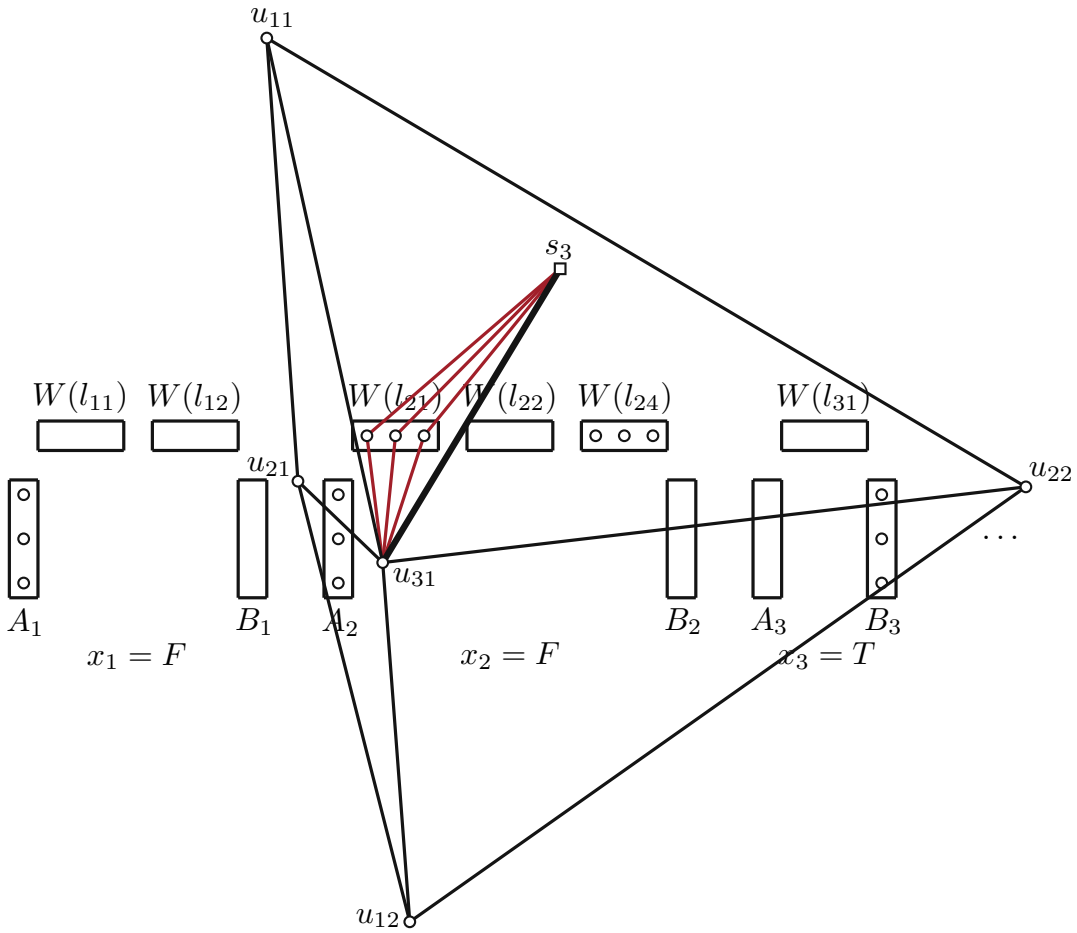


$(x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_2 \vee x_4 \vee \neg x_5) \wedge (x_1 \vee \neg x_5 \vee x_6) \wedge (\neg x_2 \vee \neg x_5 \vee \neg x_6)$

Figure 4.9: Proof of Lemma 4.2 drawing the second false literal.

Also, for the flexible c-side, the special vertex lies inside the face created by the simple vertices of the fixed c-sides, so the fixed w-side must be placed into the same face. Furthermore, the special vertex of the flexible c-side and both simple vertices must be connected to that fixed w-side, and the special vertex must be connected to each simple vertex, one after the other. The special vertex must be placed above and the simple vertices below the wire-line to make this possible. Then, all three vertices of the wire gadget can simultaneously connect to both active vertices of the flexible c-side; see Figure 4.10. As the flexible v-sides were already active, no problems with other vertices to which the wire gadgets must be connected are introduced. The same drawing can be used for both simple vertices of the flexible c-side.

So far, we have shown that connecting a clause gadget and its corresponding three wire gadgets is possible when the wire gadgets along the "wire-line" align properly. As the wire gadgets are drawn on one horizontal line, we can distinguish them as the left, the



$$(x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_2 \vee x_4 \vee \neg x_5) \wedge (x_1 \vee \neg x_5 \vee x_6) \wedge (\neg x_2 \vee \neg x_5 \vee \neg x_6)$$

Figure 4.10: Proof of Lemma 4.2 drawing the true literal.

middle, and the right wire gadget. Through our construction, we realize that one of the wire gadgets must lie outside the quadrangle while the other two must lie within it. As different clauses can contain various variables, we can not assume or enforce any ordering of the variables and, thereby, of the wire gadgets themselves. Even though the realization of the quadrangle is orientated towards the right in our sketches – and by that containing the middle and the right wire gadget – nothing in its definition prohibits it from being orientated to the left and encompassing the left and the middle wire gadget. Furthermore, in the construction of the drawing, it was also left open in which relation the two wire gadgets inside the quadrangle must lie. So if the fixed wire gadget is the right or middle wire gadget, we can use the right-orientated quadrangle, and if the fixed wire gadget is the left (or middle) wire gadget, we can use the left-orientated quadrangle. These variants are represented in Figures 4.11, 4.12, and4.13, where each figure shows a drawing with a different fixed wire gadget.

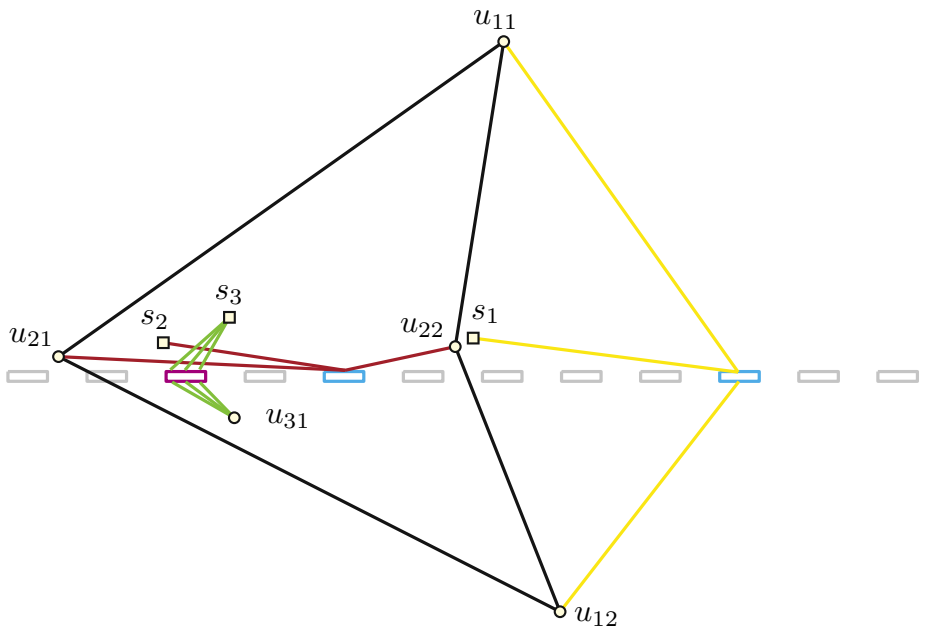


Figure 4.11: This and the following two figures show that no matter where the fixed wire gadget (purple) is positioned on the "wire-line", creating the clause gadget as described above is always possible. Here we see our construction in the situation, where the left wire gadget is fixed.

All in all, no matter which variable corresponds to the true literal in the clause, we can draw the clause gadget and its corresponding wire and variable gadgets as planar geometric storyplan drawings.

So we have shown that if the formula ϕ admits a satisfying assignment with exactly one true literal in each clause, then we can define a graph G based on ϕ and construct a planar geometric storyplan S of G. □

Theorem 2. *The Planar Geometric StoryPlan problem is NP-hard, and it has no $2^{o(n)}$ time algorithm unless the Exponential Time Hypothesis (ETH) fails.*

Proof. The construction of the graph G from the formula ϕ, as was done in Lemma 4.2, can be done in polynomial time. The correctness of the reduction from One-In-Three 3SAT to PGEO-SP follows from the Lemmas 4.1 and 4.2. Thereby proving that the Planar Geometric StoryPlan problem is indeed NP-hard.

To find a lower time boundary for possible algorithms solving the Planar Geometric StoryPlan problem, we have to take a look at the ETH and the *Sparsification Lemma*. Combining these two, we get that for 3SAT formulas consisting of N variables and M clauses no algorithm can exist that can solve these formulas in $2^{o(N+M)}$ time [3, 11]. We also know that a polynomial-time reduction from 3SAT instances with $N + M$ variables and clauses to equivalent One-In-Three 3SAT instances with $O(N + M)$ variables

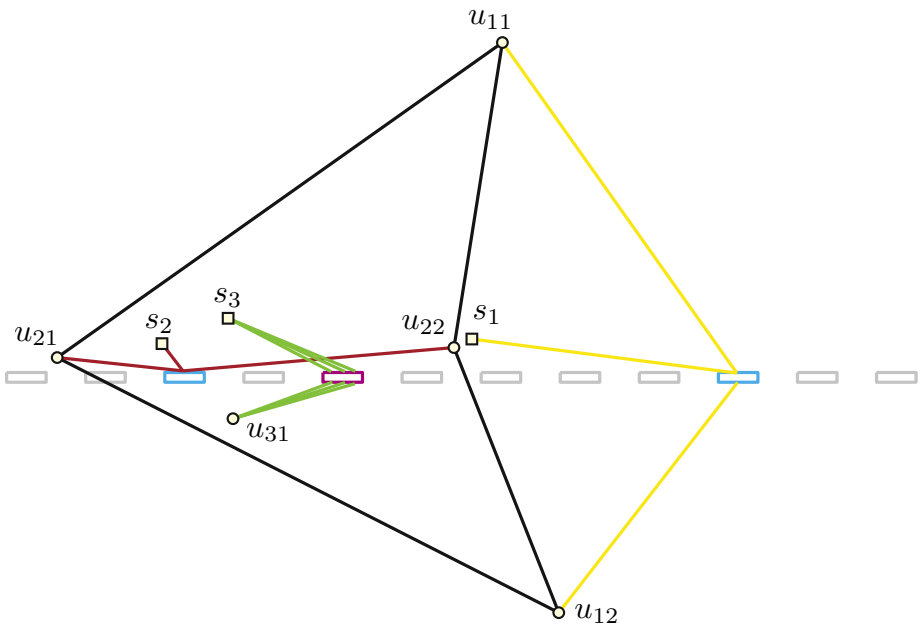


Figure 4.12: This figure shows a possible drawing of our construction when the middle wire gadget is fixed.

and clauses exists [5]. Furthermore, in Lemma 4.2, we have shown that each instance of One-In-Three 3SAT can be transformed into a graph with n vertices and m edges such that $m = O(n)$ and $n + m = O(N + M)$. To conclude, finding an algorithm that solves the Planar Geometric StoryPlan problem in $2^{o(n)}$ time would mean that we also found an algorithm to solve 3SAT formulas in $2^{o(N+M)}$ time, which would contradict ETH. □

Remark 4.1. The attentive observer might have realized that we have only shown the NP-hardness of PGEO-SP, whereas for PTOP-SP NP-completeness was shown in [2]. Whether PGEO-SP is in NP or whether it is $\exists\mathbb{R}$-hard or even $\exists\mathbb{R}$-complete is still an open question.

$\exists\mathbb{R}$, the so-called existential theory of the reals is the complexity class that contains all problems which are representable by a true sentence of the form $\exists X_1 \cdots \exists X_n \, F(X_1, \ldots, X_n)$, where $F(X_1, \ldots, X_n)$ is a quantifier-free formula over equalities and inequalities of real-valued polynomials. This complexity class lies between NP and PSPACE [1].

We suspect that PGEO-SP might be more complicated than PTOP-SP. To guess a candidate solution of PTOP-SP, it is sufficient to know the order of the vertices, the outer face, and the rotation systems of all vertices, as this information is enough to identify any planar topological storyplan (candidate) uniquely. An upper bound for these steps can be calculated using simple combinatorics; see [2]. This is not as easily possible for planar geometric storyplans, as each candidate solution of PGEO-SP also relies on

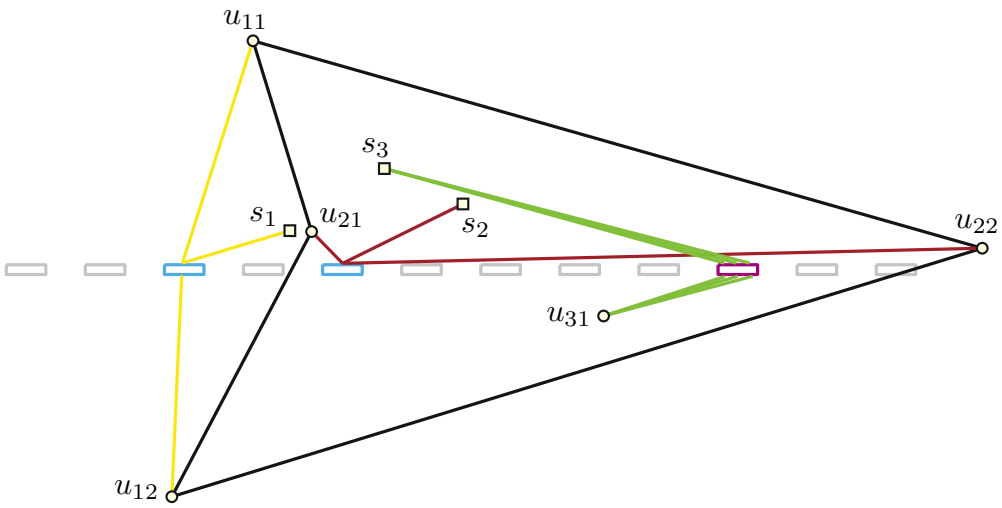


Figure 4.13: This figure shows how our construction can be mirrored when the right wire gadget is fixed.

the precise position of all of its vertices. It could be that with some constructions, the vertices of a planar geometric storyplan require even exponential precision.

CHAPTER 5

Further Results

In this chapter, we will explore some additional minor results. First, we give a definition for a storyplan with a limited number of frames, define new versions of the StoryPlan problem based on this definition, and use these to establish optimization variants of the StoryPlan problem. We will then prove complexity results for these new problems. Finally, we will list two specific results for planar geometric storyplans, one concerning planar graphs and one partial 3-trees.

5.1 On the Minimal Planar StoryPlan problem

So far, we have only looked at the Planar Topological StoryPlan problem and the Planar Geometric StoryPlan problem, two decision problems. We now want to present an optimization version of the StoryPlan problem.

With the current definition of a storyplan, a strict total order of the frames exists based on the bijective mapping of the vertex appearances to the frames. However, it becomes clear that in most cases, there are frames whose order could change, which would still result in a valid storyplan. Therefore, the order of frames could be relaxed based on a surjective mapping of the vertex appearances to the frames. So, we will adapt the definition of a storyplan to the following:

Let $G = (V, E)$ be a graph with n vertices. A planar k-storyplan $S_k = \langle \tau, \{D_i\}_{i\in[k]} \rangle$ of G is a pair defined as follows.
The first element is a surjection $\tau : V \to [k]$ that imposes a partial order on the vertices of G. For each vertex $v \in V$, let $i_v = \tau(v)$ and let $j_v = max_{u\in N[v]}\tau(u)$. The interval $[i_v, j_v]$ is the lifespan of v. We say that v appears at step i_v, is visible at step i for each $i \in [i_v, j_v]$, and disappears at step $j_v + 1$. Note that a vertex disappears only once all its neighbors have appeared, and multiple vertices may appear in each frame.
The second element of S_k is a sequence of drawings $\{D_i\}_{i\in[k]}$, called the frames of S_k.

For each $i \in [k]$ the drawing D_i must satisfy the following conditions: (i) D_i is a drawing of the graph G_i induced by the vertices visible at step i, (ii) D_i is planar, (iii) the point representing a vertex v is the same over all drawings that contain v, and (iv) the curve representing an edge e is the same over all drawings that contain e.

The Planar k-StoryPlan problem then asks, given a graph $G = (V, E)$ and an integer k, does G admit a planar k-storyplan, a planar storyplan with exactly k frames?

The Minimal Planar StoryPlan problem then asks, given a graph $G = (V, E)$, what is the minimal integer k such that G still admits a planar k-storyplan?

Remark 5.1. Every n-storyplan S_n is also a storyplan S and vice versa, as a storyplan S has exactly one frame for each of its n vertices.

Remark 5.2. Given a graph $G = (V, E)$ that admits a k-storyplan S_k, where $k < n$. Then G also admits a j-storyplan S_j, where for j holds: $k < j \wedge j \leq n$. So G also admits a storyplan S.

Definition 5. *We define the Planar Topological k-StoryPlan problem (PTOP-k-SP) and the Planar Geometric k-StoryPlan problem (PGEO-k-SP) as versions of the Planar k-StoryPlan problem, where the embedding of all edges in all frames must be Jordan arcs (PTOP-k-SP) or straight-lines (PGEO-k-SP), respectively.*

Definition 6. *The Minimal Planar Topological StoryPlan problem (MIN-PTOP-SP) and the Minimal Planar Geometric StoryPlan problem (MIN-PGEO-SP) are similarly defined as the versions of the Minimal Planar StoryPlan problem, where all edges must be Jordan arcs (MIN-PTOP-SP) or straight-lines (MIN-PGEO-SP), respectively.*

Lemma 5.1. *The Planar Topological k-StoryPlan problem and the Planar Geometric k-StoryPlan problem are NP-hard for general values of k.*

Proof. Let us start by showing that a polynomial-time many-one reduction from the Planar StoryPlan problem to the Planar k-StoryPlan problem exists. Given a graph G, we ask if a planar n-storyplan S_n exists for G. If G is a yes-instance of the Planar n-StoryPlan problem, we know that a n-storyplan S_n exists for G. Furthermore, as every n-storyplan S_n is also a storyplan S, we get that G is also a yes-instance of the Planar StoryPlan problem. If, on the other hand, G is a no-instance of the Planar n-StoryPlan problem, we know that no n-storyplan S_n exists for G and thereby also no storyplan S can exist for G, as every storyplan S of G would also be a n-storyplan S_n of G. As the yes- and the no-instances are correctly propagated from instances of the Planar k-StoryPlan problem to instances of the Planar StoryPlan problem, we get that a many-one-reduction reduction from the Planar StoryPlan problem to the Planar k-StoryPlan problem exists. As the reduction step was trivial, it is also a polynomial-time many-one reduction.

By the existence of a polynomial-time many-one reduction from the Planar StoryPlan problem to the Planar k-StoryPlan problem, we get that also polynomial-time

many-one reductions from PTOP-SP to PTOP-k-SP and from PGEO-SP to PGEO-k-SP must exist, as the restrictions on the drawings of the frames were not part of the reduction proof at all. We already established that both PTOP-SP and PGEO-SP are NP-hard problems. Therefore, by the existence of polynomial-time many-one reductions from these known NP-hard problems to the Planar Topological k-StoryPlan problem and the Planar Geometric k-StoryPlan problem, these problems must also both be NP-hard.

□

Remark 5.3. The Planar k-StoryPlan problem, and in extension PTOP-k-SP and PGEO-k-SP, are not NP-hard for specific small k. For $k = 1$, the Planar 1-StoryPlan problem is equivalent to testing whether a graph G is planar. Planarity Testing is known to be solvable in $O(n)$ time [10]. The time complexity of the Planar k-StoryPlan problem for other small values of k such as $k = 2$ or $k = 3$ is so far unknown, but might also be in P.

Lemma 5.2. *The Planar Topological k-StoryPlan problem is NP-complete for general values of k.*

Proof. A problem is in NP if, for a given instance, a candidate solution (certificate) is of polynomial size, and it can be checked in polynomial time by a certifier if this candidate is a solution for that instance of the problem or not. A candidate solution for a graph G of the Planar Topological k-StoryPlan problem is a series of graphical embeddings of subgraphs of G and consists of the following information:

Firstly, a partial order on the vertices of G, which takes linear space. This partial order defines which vertex appears in which frame. Based on the vertices and their neighbors in G, we can calculate in which frames each vertex is active and, based on that, in which frames which edges are active. We know by Euler's formula that for all connected planar embeddings holds $|E_i| \leq 3|V_i| - 6$, so a linear factor limits the possible number of neighbors of each vertex in a planar graph. This limit also holds for disconnected planar graphs, as the neighbors of a vertex, per definition, are always part of the same connected component as the vertex itself. Once we know which vertices appear in which frame, we need a rotation system for each vertex v active in one frame, which is the clockwise order of the incident edges of v active in the same frame. Furthermore, we need an outer face, and for each disconnected component, we must know into which face they must be placed. All of that information needs, again, only linear space. So the information necessary to distinctly define a candidate solution (certificate) of PTOP-k-SP takes $O(n)$ space.

To check the validity of a candidate solution, it is enough – as we are dealing with topological drawings, so all edges are Jordan arcs – to verify whether two consecutive frames are "consistent" or not. For a given partial order of the vertices of G, it is equivalent if a planar topological storyplan or a "consistent" sequence of planar embeddings exist. To verify that two planar embeddings are "consistent", we can use the fact that planar topological embeddings can be differentiated simply by their rotation systems, the choice

of the outer face, and the face in which disconnected components are placed. So, given two consecutive planar embeddings that share a common subgraph, their restrictions for this subgraph must match. This check must first compare the vertices in both embeddings to get the subgraphs, which can be done in linear time. Then, checking the rotation systems, the chosen outer face, and the position of disconnected components takes quadratic time. These checks are repeated $k-1$ times. So, the verification step takes $O(n^3)$ time.

We have shown that a candidate solution for PTOP-k-SP is of polynomial size and that verifying its validity takes polynomial time. Therefore, the Planar Topological k-StoryPlan problem is in NP.

As NP-hardness and membership in NP entail NP-completeness and we have already shown the NP-hardness of PTOP-k-SP in Lemma 5.1, we get that the Planar Topological k-StoryPlan problem must be NP-complete. □

Lemma 5.3. *The Minimal Planar Topological StoryPlan problem and the Minimal Planar Geometric StoryPlan problem are NP-hard.*

Proof. A simple algorithm to turn any decision problem with a limiting integer k into an optimization problem is to repeatedly execute the algorithm that solves the k-limited version of that problem in a binary search manner. So, in our case, to get an algorithm for a version of the Minimal Planar StoryPlan problem, we first execute an algorithm to answer the respective Planar k-StoryPlan problem and set k to n. On a *no*, we know that also no smaller storyplan can exist, and we return ∞. On a *yes*, we run the algorithm again for $k = \lfloor \frac{n}{2} \rfloor$. The same procedure is done for ever smaller k until we find the k, where a k-storyplan exists, but no $k-1$-storyplan exists anymore.

As this is the binary search procedure, we must run the Planar k-StoryPlan algorithm $log(n)$ times. As the respective Planar k-StoryPlan problems are already NP-hard, the $log(n)$ factor does not influence the complexity of their corresponding Minimal Planar StoryPlan problems. Hence, proving that both the Minimal Planar Topological StoryPlan problem and the Minimal Planar Geometric StoryPlan problem are NP-hard. □

5.2 Planar graphs and partial 3-trees

Lastly, we will list two small results for specific, restricted graph classes, one for planar graphs and one for partial 3-trees. k-trees have an iterative definition. Each k-tree G starts as a complete graph with $k+1$ vertices. Any additional vertex added to the k-tree must be a neighbor to all vertices of an existing k-clique in G. Partial k-trees are subgraphs of k-trees. Furthermore, k-trees always have a treewidth of k, and partial k-trees are all graphs with a treewidth of at most k. Treewidth is a parameter that describes how much a graph resembles a tree.

The result for partial 3-trees is from the original storyplan paper [2] for planar topological storyplans that we adapt for planar geometric storyplans. We omit the proof here because it is the same as in [2].

Remark 5.4. Every planar graph G with n vertices admits a planar geometric 1-storyplan, which can be computed in $O(n)$ time.

Proof. Every planar graph G admits a storyplan [2]. Since the graphs are already planar, they can be drawn in one frame, thereby admitting a 1-storyplan.

By Fáry's theorem [9], every planar graph also has a planar straight-line embedding. Therefore, every planar graph admits a planar geometric 1-storyplan.

Multiple algorithms are known to calculate a planar straight-line drawing for a planar graph G with n vertices in $O(n)$ time, for example, the shift algorithm [5]. □

Corollary 5.1. *Every partial 3-tree G with n vertices admits a planar geometric storyplan, which can be computed in $O(n)$ time.*

Proof. The paper [2] includes the following theorem: "Every partial 3-tree G with n vertices admits a storyplan, which can be computed in $O(n)$ time." The construction of the storyplan in the proof for this theorem already results in a planar geometric storyplan and takes only $O(n)$ time. □

CHAPTER 6

Conclusion and Open Questions

6.1 Conclusion

In this thesis, we introduced and explored several new variations of the PLANAR STORYPLAN problem. The primary focus was on the Planar Geometric StoryPlan problem, a variant of the storyplan problem containing only crossing-free, straight-line drawings for each frame. We presented two key results for this problem variant. Firstly, the Planar Geometric StoryPlan problem is more restrictive than the Planar Topological StoryPlan problem, and secondly, the Planar Geometric StoryPlan problem remains NP-hard.

We then expanded the concept of a storyplan by introducing the idea of a k-storyplan S_k, where multiple vertices can appear in the same frame, and the storyplan consists of a sequence of exactly k frames. This led to the definition of two new Planar StoryPlan problems. Namely, the Planar k-StoryPlan problem and the Minimal Planar StoryPlan problem. The Planar k-StoryPlan problem asks, given a graph G and an integer k, whether a graph G admits a k-storyplan S_k with exactly k frames, while the Minimal Planar StoryPlan problem asks, given a graph G, for the smallest integer k such that a k-storyplan still exists for G.

As final results, we were able to prove that the complexity results of the Planar Topological StoryPlan problem and the Planar Geometric StoryPlan problem propagate to their respective versions of the Planar k-StoryPlan problem. In particular, we proved that the Planar Topological k-StoryPlan problem is NP-complete and the Planar Geometric k-StoryPlan problem is NP-hard, for general values of k. Finally, we showed that both the Minimal Planar Topological StoryPlan problem and the Minimal Planar Geometric StoryPlan problem are NP-hard.

6.2 Open Questions

Our research opens up several exciting directions for future exploration and extension.

One significant question remains unanswered: the exact complexity class of the Planar Geometric StoryPlan problem, as well as its k-storyplan and minimal storyplan variants. More precisely, it is still unclear whether the Planar Geometric StoryPlan problem belongs to NP and, therefore, is NP-complete.

Alternatively, could the Planar Geometric StoryPlan problem be $\exists\mathbb{R}$-hard or even $\exists\mathbb{R}$-complete? To investigate $\exists\mathbb{R}$-hardness, a potential approach could involve finding a polynomial-time many-one reduction from a known $\exists\mathbb{R}$-hard problem to the Planar Geometric StoryPlan problem. Another avenue for proving $\exists\mathbb{R}$-hardness, could be to show the existence of a gadget that, with each use, restricts the region where a vertex can be placed by at least half.

If a new complexity result for PGEO-SP is established, a natural follow-up question is whether the same result could be extended to its variants, PGEO-k-SP and MIN-PGEO-SP?

We mentioned that for $k = 1$, PTOP-k-SP and PGEO-k-SP are in $O(n)$. Therefore, the complexity of the Planar k-StoryPlan problem for small values of k such as $k \in \{2, 3, 4, 5\}$ is still an open question. Furthermore, at what value of k does the problem turn out to be NP-hard?

Another set of open questions pertains to the time complexity upper bounds of algorithms for the storyplan problems explored in this thesis.

Namely, what is the upper bound on the time complexity of an algorithm solving PGEO-SP? Are there any fixed-parameter tractable (FPT) algorithms for PGEO-SP? If so, what upper bounds do they provide?

In addition, related questions arise regarding variants of the Planar Topological StoryPlan problem, particularly PTOP-k-SP and MIN-PTOP-SP. How efficient can algorithms for these problems be? What are their respective time complexities, and how could FPT algorithms and kernelization be used to optimize them further?

Finally, other variants of the Planar StoryPlan problem could be considered by introducing additional restrictions on the placement of vertices. For instance, could a storyplan exist when vertices are not freely placed but must be located on a grid or a set of fixed points instead? Alternatively, what happens if the vertices can only be placed on a fixed number of positions, but these can be freely chosen? Exploring such restrictions could lead to new insights and even more nuanced problem variants.

APPENDIX A

Frame tables of a PTOP-SP instance for G

Here, we present the complete set of frame tables that demonstrate that the graph G is a yes-instance of PTOP-SP, as outlined in in Section 3.2 (*Yes-instance of PTOP-SP*). Each frame table shows the adjacencies of the vertices of G and lists the active vertices in a specific frame. To minimize the number of frames and thus the number of tables, we adopt the concept of k-storyplans, allowing multiple vertices to appear in the same frame. This approach does not affect the validity of the proof, as any graph that admits a k-storyplan S_k also admits a corresponding storyplan S.

Vertex	Adjacent vertices													
v_1	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^e	a_5^e	a_8^e
v_2	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
v_1'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^e	a_5^e	a_8^e
v_2'	v_1'	v_3'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4'	v_1'	v_2'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
a_1^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_1^e					
a_1^e	v_1	v_2	v_1'	v_2'	a_1^q									
a_2^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_2^e					
a_2^e	v_2	v_3	v_2'	v_3'	a_2^q									
a_3^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_3^e					
a_3^e	v_3	v_4	v_3'	v_4'	a_3^q									
a_4^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_4^e					
a_4^e	v_1	v_4	v_1'	v_4'	a_4^q									
a_5^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_5^e					
a_5^e	v_1	v_2	v_1'	v_2'	a_5^q									
a_6^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_6^e					
a_6^e	v_2	v_3	v_2'	v_3'	a_6^q									
a_7^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_7^e					
a_7^e	v_3	v_4	v_3'	v_4'	a_7^q									
a_8^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_8^e					
a_8^e	v_1	v_4	v_1'	v_4'	a_8^q									

Table A.1: Table representation of the first frame of G as PTOP-SP. Black-colored are all vertices that are active in that frame. Magenta-colored are all the vertices that have not yet been activated.

Vertex	Adjacent vertices													
v_1	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^e	a_5^e	a_8^e
v_2	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
v_1'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^q	a_5^e	a_8^e
v_2'	v_1'	v_3'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4'	v_1'	v_2'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
a_1^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_1^e					
a_1^e	v_1	v_2	v_1'	v_2'	a_1^q									
a_2^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_2^e					
a_2^e	v_2	v_3	v_2'	v_3'	a_2^q									
a_3^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_3^e					
a_3^e	v_3	v_4	v_3'	v_4'	a_3^q									
a_4^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_4^e					
a_4^e	v_1	v_4	v_1'	v_4'	a_4^q									
a_5^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_5^e					
a_5^e	v_1	v_2	v_1'	v_2'	a_5^q									
a_6^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_6^e					
a_6^e	v_2	v_3	v_2'	v_3'	a_6^q									
a_7^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_7^e					
a_7^e	v_3	v_4	v_3'	v_4'	a_7^q									
a_8^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_8^e					
a_8^e	v_1	v_4	v_1'	v_4'	a_8^q									

Table A.2: Table representation of the second frame of G as PTOP-SP. Green-colored are all the vertices that were already active in a previous frame.

Vertex	Adjacent vertices													
v_1	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^e	a_5^e	a_8^e
v_2	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
v_1'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^q	a_5^e	a_8^e
v_2'	v_1'	v_3'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4'	v_1'	v_2'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
a_1^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_1^e					
a_1^e	v_1	v_2	v_1'	v_2'	a_1^q									
a_2^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_2^e					
a_2^e	v_1	v_2	v_1'	v_2'	a_2^q									
a_3^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_3^e					
a_3^e	v_3	v_4	v_3'	v_4'	a_3^q									
a_4^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_4^e					
a_4^e	v_1	v_4	v_1'	v_4'	a_4^q									
a_5^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_5^e					
a_5^e	v_1	v_2	v_1'	v_2'	a_5^q									
a_6^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_6^e					
a_6^e	v_2	v_3	v_2'	v_3'	a_6^q									
a_7^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_7^e					
a_7^e	v_3	v_4	v_3'	v_4'	a_7^q									
a_8^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_8^e					
a_8^e	v_1	v_4	v_1'	v_4'	a_8^q									

Table A.3: Table representation of the third frame of G as PTOP-SP.

Vertex	Adjacent vertices													
v_1	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^e	a_5^e	a_8^e
v_2	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
v_1'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^q	a_5^e	a_8^e
v_2'	v_1'	v_3'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4'	v_1'	v_2'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
a_1^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_1^e					
a_1^e	v_1	v_2	v_1'	v_2'	a_1^q									
a_2^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_2^e					
a_2^e	v_1	v_2	v_1'	v_2'	a_2^q									
a_3^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_3^e					
a_3^e	v_3	v_4	v_3'	v_4'	a_3^q									
a_4^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_4^e					
a_4^e	v_1	v_4	v_1'	v_4'	a_4^q									
a_5^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_5^e					
a_5^e	v_1	v_2	v_1'	v_2'	a_5^q									
a_6^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_6^e					
a_6^e	v_1	v_2	v_1'	v_2'	a_6^q									
a_7^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_7^e					
a_7^e	v_3	v_4	v_3'	v_4'	a_7^q									
a_8^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_8^e					
a_8^e	v_1	v_4	v_1'	v_4'	a_8^q									

Table A.4: Table representation of the fourth frame of G as PTOP-SP.

Vertex	Adjacent vertices													
v_1	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^e	a_5^e	a_8^e
v_2	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
v_1'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^q	a_5^e	a_8^e
v_2'	v_1'	v_3'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4'	v_1'	v_2'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
a_1^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_1^e					
a_1^e	v_1	v_2	v_1'	v_2'	a_1^q									
a_2^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_2^e					
a_2^e	v_1	v_2	v_1'	v_2'	a_2^q									
a_3^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_3^e					
a_3^e	v_1	v_2	v_1'	v_2'	a_3^q									
a_4^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_4^e					
a_4^e	v_1	v_4	v_1'	v_4'	a_4^q									
a_5^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_5^e					
a_5^e	v_1	v_2	v_1'	v_2'	a_5^q									
a_6^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_6^e					
a_6^e	v_1	v_2	v_1'	v_2'	a_6^q									
a_7^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_7^e					
a_7^e	v_3	v_4	v_3'	v_4'	a_7^q									
a_8^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_8^e					
a_8^e	v_1	v_4	v_1'	v_4'	a_8^q									

Table A.5: Table representation of the fifth frame of G as PTOP-SP.

Vertex	Adjacent vertices													
v_1	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^e	a_5^e	a_8^e
v_2	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
v_1'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^q	a_5^e	a_8^e
v_2'	v_1'	v_3'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4'	v_1'	v_2'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
a_1^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_1^e					
a_1^e	v_1	v_2	v_1'	v_2'	a_1^q									
a_2^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_2^e					
a_2^e	v_1	v_2	v_1'	v_2'	a_2^q									
a_3^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_3^e					
a_3^e	v_1	v_2	v_1'	v_2'	a_3^q									
a_4^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_4^e					
a_4^e	v_1	v_4	v_1'	v_4'	a_4^q									
a_5^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_5^e					
a_5^e	v_1	v_2	v_1'	v_2'	a_5^q									
a_6^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_6^e					
a_6^e	v_1	v_2	v_1'	v_2'	a_6^q									
a_7^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_7^e					
a_7^e	v_1	v_2	v_1'	v_2'	a_7^q									
a_8^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_8^e					
a_8^e	v_1	v_4	v_1'	v_4'	a_8^q									

Table A.6: Table representation of the sixth frame of G as PTOP-SP.

Vertex	Adjacent vertices													
v_1	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^e	a_5^e	a_8^e
v_2	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
v_1'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^q	a_5^e	a_8^e
v_2'	v_1'	v_3'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4'	v_1'	v_2'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
a_1^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_1^e					
a_1^e	v_1	v_2	v_1'	v_2'	a_1^q									
a_2^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_2^e					
a_2^e	v_1	v_2	v_1'	v_2'	a_2^q									
a_3^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_3^e					
a_3^e	v_1	v_2	v_1'	v_2'	a_3^q									
a_4^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_4^e					
a_4^e	v_1	v_2	v_1'	v_2'	a_4^q									
a_5^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_5^e					
a_5^e	v_1	v_2	v_1'	v_2'	a_5^q									
a_6^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_6^e					
a_6^e	v_1	v_2	v_1'	v_2'	a_6^q									
a_7^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_7^e					
a_7^e	v_1	v_2	v_1'	v_2'	a_7^q									
a_8^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_8^e					
a_8^e	v_1	v_4	v_1'	v_4'	a_8^q									

Table A.7: Table representation of the seventh frame of G as PTOP-SP.

Vertex	Adjacent vertices													
v_1	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^e	a_5^e	a_8^e
v_2	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3	v_2	v_4	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4	v_1	v_3	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
v_1'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_4^q	a_5^e	a_8^e
v_2'	v_1'	v_3'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_1^e	a_2^e	a_5^e	a_6^e
v_3'	v_2'	v_4'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_2^e	a_3^e	a_6^e	a_7^e
v_4'	v_1'	v_2'	a_1^q	a_2^q	a_3^q	a_4^q	a_5^q	a_6^q	a_7^q	a_8^q	a_3^e	a_4^e	a_7^e	a_8^e
a_1^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_1^e					
a_1^e	v_1	v_2	v_1'	v_2'	a_1^q									
a_2^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_2^e					
a_2^e	v_1	v_2	v_1'	v_2'	a_2^q									
a_3^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_3^e					
a_3^e	v_1	v_2	v_1'	v_2'	a_3^q									
a_4^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_4^e					
a_4^e	v_1	v_2	v_1'	v_2'	a_4^q									
a_5^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_5^e					
a_5^e	v_1	v_2	v_1'	v_2'	a_5^q									
a_6^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_6^e					
a_6^e	v_1	v_2	v_1'	v_2'	a_6^q									
a_7^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_7^e					
a_7^e	v_1	v_2	v_1'	v_2'	a_7^q									
a_8^q	v_1	v_2	v_3	v_4	v_1'	v_2'	v_3'	v_4'	a_8^e					
a_8^e	v_1	v_2	v_1'	v_2'	a_8^q									

Table A.8: Table representation of the eighth frame of G as PTOP-SP.

Overview of Generative AI Tools Used

Grammarly was used to check the whole thesis for grammar mistakes. It can also provide recommendations for better, more precise sentence formulations. A feature that we employed.

ChatGPT was used to help with formulations in Chapter 1 (*Introduction*), in Chapter 2 (*Preliminaries*), in Chapter 6 (*Conclusion and Open Questions*), and the Abstract. First all of these chapters were written by hand, then ChatGPT was used to reformulate and streamline these contribution-free chapters.

List of Figures

List of Tables

Glossary

$\exists\mathbb{R}$ The so-called existential theory of the reals is the complexity class that contains all problems which are representable by a true sentence of the form $\exists X_1 \cdots \exists X_n \; F(X_1, \ldots, X_n)$, where $F(X_1, \ldots, X_n)$ is a quantifier-free formula over equalities and inequalities of real-valued polynomials and lies between NP and PSPACE [1]. 61, 70

NP The complexity class that describes all problems solvable by a **N**ondeterministic turingmachine in **P**olynomial time. xi, xiii, xv, 3–5, 9, 49–52, 54, 56, 58, 60–62, 64–66, 69, 70, 89

P The complexity class that describes all problems solvable by a deterministic turingmachine in **P**olynomial time. 65

PSPACE The complexity class that describes all problems solvable by a deterministic turingmachine using **P**olynomial **SPACE**. 61, 89

Acronyms

One-In-Three 3SAT One-In-Three 3-Satisfiability problem. 3, 49, 53, 60, 61

3SAT 3-Satisfiability problem. 49, 51, 60, 61

ETH Exponential Time Hypothesis. 60, 61

FPT fixed-parameter tractable. 3, 4, 70

MIN-PGEO-SP Minimal Planar Geometric StoryPlan problem. xi, xiii, 2, 5, 64, 66, 69, 70

MIN-PTOP-SP Minimal Planar Topological StoryPlan problem. xi, xiii, 2, 5, 64, 66, 69, 70

PGEO-k-SP Planar Geometric k-StoryPlan problem. xi, xiii, 2, 5, 64, 65, 69, 70

PGEO-SP Planar Geometric StoryPlan problem. xi, xiii, xv, 2, 4, 5, 8, 9, 11, 12, 14, 16, 18–46, 49–52, 54, 56, 58, 60–63, 65, 69, 70

PTOP-k-SP Planar Topological k-StoryPlan problem. xi, xiii, 2, 5, 64–66, 69, 70

PTOP-SP Planar Topological StoryPlan problem. xi, xiii, xv, 2, 4, 5, 8, 11, 16, 17, 34, 46, 49, 61, 63, 65, 69–79, 87

Bibliography

[1] Saugata Basu, Richard Pollack, and Marie-Françoise Roy. Existential theory of the reals. In *Algorithms in Real Algebraic Geometry*, chapter 13, pages 505–532. Springer Berlin Heidelberg, 2006.

[2] Carla Binucci, Emilio Di Giacomo, William J. Lenhart, Giuseppe Liotta, Fabrizio Montecchiani, Martin Nöllenburg, and Antonios Symvonis. On the complexity of the storyplan problem. *Journal of Computer and System Sciences*, 139:103466, 2024.

[3] Hans L. Bodlaender and Ton Kloks. Efficient and constructive algorithms for the pathwidth and treewidth of graphs. *Journal of Algorithms*, 21(2):358–402, 1996.

[4] Manuel Borrazzo, Giordano Da Lozzo, Fabrizio Frati, and Maurizio Patrignani. Graph stories in small area. In Daniel Archambault and Csaba D. Tóth, editors, *Graph Drawing and Network Visualization*, volume 11904 of *Lecture Notes in Computer Science*, pages 545–558, Cham, 2019. Springer International Publishing.

[5] Marek Chrobak and Thomas H. Payne. A linear-time algorithm for drawing a planar graph on a grid. *Information Processing Letters*, 54(4):241–246, 1995.

[6] Giuseppe Di Battista, Walter Didimo, Luca Grilli, Fabrizio Grosso, Giacomo Ortali, Maurizio Patrignani, and Alessandra Tappini. Small point-sets supporting graph stories. In Patrizio Angelini and Reinhard von Hanxleden, editors, *Graph Drawing and Network Visualization*, volume 13764 of *Lecture Notes in Computer Science*, pages 289–303, Cham, 2022. Springer International Publishing.

[7] Jiří Fiala, Oksana Firman, Giuseppe Liotta, Alexander Wolff, and Johannes Zink. Outerplanar and forest storyplans. In Henning Fernau, Serge Gaspers, and Ralf Klasing, editors, *SOFSEM 2024: Theory and Practice of Computer Science*, volume 14519 of *Lecture Notes in Computer Science*, pages 211–225, Cham, 2024. Springer Nature Switzerland.

[8] Simon D. Fink, Matthias Pfretzschner, and Ignaz Rutter. Parameterized complexity of simultaneous planarity. In Michael A. Bekos and Markus Chimani, editors, *Graph Drawing and Network Visualization*, volume 14466 of *Lecture Notes in Computer Science*, pages 82–96, Cham, 2023. Springer Nature Switzerland.

[9] István Fáry. On straight-line representation of planar graphs. *Acta Sci. Math. (Szeged)*, 11:229–233, 1948.

[10] John E. Hopcroft and Robert E. Tarjan. Efficient planarity testing. *Journal of the ACM (JACM)*, 21(4):549–568, October 1974.

[11] Burkhard Monien and Ivan Hal Sudborough. Min cut is NP-complete for edge weighted trees. *Theoretical Computer Science*, 58(1):209–229, 1988.

[12] Marcus Schaefer. Toward a theory of planarity: Hanani-tutte and planarity variants. *J. Graph Algorithms Appl.*, 17(4):367–440, 2013.