

Bachelorarbeit

# Aufwärtsplanare Zeichnungen von Außenpfaden unter Verwendung von drei Steigungen

Joshua Geis

Abgabedatum: 15. September 2022  
Betreuer: Prof. Dr. Alexander Wolff  
Johannes Zink, M.Sc.



Julius-Maximilians-Universität Würzburg  
Lehrstuhl für Informatik I  
Algorithmen und Komplexität

# Zusammenfassung

Die Steigungszahl eines Graphen gibt an, wie viele unterschiedliche Steigungen beim Erstellen einer geradlinigen Zeichnung mindestens benötigt werden. Sie kann als ein Parameter für die visuelle Komplexität einer Zeichnung genutzt werden. Für bestimmte Graphenklassen existieren bereits Grenzen für die Steigungszahl.

Diese Arbeit beschäftigt sich mit dem Erstellen von aufwärtsplanaren Zeichnungen von *Außenpfaden*, welche nur geradlinige Kanten sowie drei verschiedene Steigungen verwenden. Eine Zeichnung eines gerichteten Graphen wird *aufwärtsplanar* genannt, wenn sie planar ist und jede Kante aufwärtsgerichtet ist.

Für *maximale* Außenpfade präsentieren wir einen Algorithmus, welcher eine solche Zeichnung in Polynomialzeit erstellt. Anschließend stellen wir eine Anpassung vor, wodurch wir den Algorithmus auch auf *maximale außenplanare* Graphen anwenden können.

Wir stellen die bereits existierende NP-Schwere-Reduktion für außenplanare Graphen vor und zeigen anschließend, dass man einen Außenpfad mit ähnlichen Eigenschaften konstruieren kann, sodass die Reduktion auch bei Außenpfaden hält und wir somit die NP-Schwere für das Problem nachweisen können.

## Abstract

The slope number of a graph indicates the minimum number of different slopes required for creating a straight-lined drawing. It can be used as a parameter for the visual complexity of a drawing. Limits for the slope number already exist for certain graph classes.

In this thesis we discuss the creation of upward planar drawings of *outerpaths* using only straight-line edges and three different slopes. A drawing of a directed graph is called upward planar if it is planar and all edges point upwards.

For *maximal* outerpaths we present an algorithm that creates such a drawing in polynomial time. We then present an adaptation that allows us to apply the algorithm to *maximal outerplanar* graphs.

We modify an existing NP-hardness reduction for outerplanar graphs to construct an outerpath with similar properties so that the reduction also holds for outerpaths and we can thus prove the NP-hardness of the problem.

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>4</b>
1.1	Verwandte Arbeiten . . . . .	4
1.2	Beitrag . . . . .	5
<b>2</b>	<b>Grundlagen</b>	<b>7</b>
2.1	Definitionen und Begriffe . . . . .	7
2.2	NP-Schwere für außenplanare Graphen nach Klawitter und Zink . . . . .	8
<b>3</b>	<b>Maximale Außenpfade</b>	<b>14</b>
3.1	Algorithmus . . . . .	14
3.2	Korrektheit . . . . .	16
3.3	Laufzeit . . . . .	17
3.4	Maximale außenplanare Graphen . . . . .	19
<b>4</b>	<b>Allgemeine Außenpfade</b>	<b>20</b>
4.1	Anpassung des NP-Schwere-Beweises . . . . .	20
4.2	Allgemeine Außenpfade bei variabler Einbettung . . . . .	25
<b>5</b>	<b>Fazit</b>	<b>26</b>
	<b>Literaturverzeichnis</b>	<b>27</b>

# 1 Einleitung

Ein Ziel beim Graphzeichnen ist es, *gute Zeichnungen* von Graphen zu erstellen. Ein wichtiges Maß hierbei ist die *visuelle Komplexität* [Sch15], welche die Anzahl an geometrischen Objekten in einer Zeichnung beschreibt. Hierfür wurden bereits verschiedene Parameter identifiziert. Die *Kreisbogenzahl* nach Schulz [Sch15] gibt die minimale Anzahl der Kreisbögen einer Zeichnung an. Die *Geradenüberdeckungsanzahl* oder *Ebenenüberdeckungsanzahl* nach Chaplick et al. [CFL<sup>+</sup>20] hingegen bezeichnet die minimale Anzahl der Geraden bzw. Ebenen, welche eine kreuzungsfreie geradlinige Zeichnung eines Graphen im 2- bzw. im 3-dimensionalen überdecken. Kryven et al. [KRW19] wiederum haben die *Kugelüberdeckungsanzahl* eingeführt, welche die minimale Anzahl an Kreisen im 2-dimensionalen und Kugeln im 3-dimensionalen beschreibt, die eine kreuzungsfreie Kreisbogenzeichnung überdecken. Unter der *Streckenzahl* nach Dujmović et al. [DESW07] versteht man die minimale Anzahl an benötigten Strecken einer Zeichnung. Darüber hinaus kann die *Steigungszahl* nach Wade und Chu [WC94] auch als Parameter für die visuelle Komplexität herangezogen werden, welche angibt, wie viele verschiedene Steigungen mindestens für die Zeichnung eines Graphen benötigt werden. Wir widmen uns in dieser Arbeit der Erstellung von aufwärtsplanaren Zeichnungen mit geradlinigen Kanten und der Steigungszahl 3 für Außenpfade, eine Unterklasse der außenplanaren Graphen.

## 1.1 Verwandte Arbeiten

Eine wichtige Erkenntnis zur Erstellung von aufwärtsplanaren Zeichnungen haben Di Battista und Tamassia [DT88] geliefert. Sie haben gezeigt, dass jeder aufwärtsplanare Graph eine Zeichnung mit nur geradlinigen Kanten besitzt. Ein Aufwärtsplanaritätstest für einen gerichteten Graphen ist im Allgemeinen jedoch NP-vollständig, wie Garg und Tamassia bewiesen haben [GT01]. Allerdings existieren für diverse Graphenklassen, wie außenplanare Graphen [Pap95], serien-parallele Graphen [DGL10] oder dreifach zusammenhängende Graphen [BDLM94], Algorithmen, welche diesen Test in Polynomialzeit durchführen können.

Zu Zeichnungen mit geraden Kanten und einer Begrenzung der verwendeten Steigungen existieren bereits zahlreiche Arbeiten. Die *Steigungszahl* eines Graphen nach Wade und Chu [WC94] gibt an, wie viele verschiedene Steigungen mindestens für eine Zeichnung benötigt werden. Im Fall von planaren Zeichnungen nennt man sie *planare Steigungszahl*. Hierfür hat Hoffmann [Hof17] gezeigt, dass die Bestimmung schwer in der existentiellen Theorie der reellen Zahlen ( $\exists\mathbb{R}$ ) ist. Mit der planaren Steigungszahl haben sich unter anderem Jelínek et al. [JJK<sup>+</sup>13], Knauer et al. [KMW14] oder Lenhart et al. [LLMN13] beschäftigt. Mukkamala und Szegedy [MS09], Di Giacomo et al. [DLM18]

oder Mukkamala et al. [MP12] hingegen haben die Steigungszahl von kubischen Graphen betrachtet.

Der Grundstein zum Zeichnen von aufwärtsplanaren Graphen mit einer begrenzten Zahl an Steigungen wurde durch Klawitter und Mchedlidze [KM22] gelegt. Sie haben sich hierbei auf zwei Steigungen beschränkt. Dabei haben sie gezeigt, dass man bei gegebener Einbettung in Polynomialzeit entscheiden kann, ob ein Graph eine solche Zeichnung besitzt und diese auch in derselben Zeit konstruieren kann. Für das Szenario einer variablen Einbettung haben sie einen parametrisierten Algorithmus zur Lösung des Problems angegeben.

Auf diesen Ergebnissen aufbauend, haben Klawitter und Zink [KZ21] anschließend aufwärtsplanare Zeichnungen mit drei und mehr Steigungen betrachtet. Dabei haben sie sich mit den Graphenklassen Baum, Kaktus, außenplanar und planar beschäftigt und gezeigt, dass das Entscheidungsproblem, ob eine solche Zeichnung existiert, bei Bäumen und Kaktusgraphen in Polynomialzeit entscheidbar ist, bei außenplanaren sowie planaren Graphen jedoch NP-schwer ist. Außerdem ist die Komplexität bei diesen Klassen unabhängig davon, ob eine Einbettung gegeben ist.

Des Weiteren hat Quapil in [Qua21] einen Algorithmus für die aufwärtsplanare Zeichnung von serien-parallelen Graphen mit zwei oder drei Steigungen präsentiert und zusätzlich noch gezeigt, dass das Entscheidungsproblem, ob die Zeichnung eines Graphen mehr Steigungen benötigt als sein maximaler Eingangs- und Ausgangsgrad,  $\exists\mathbb{R}$ -vollständig ist.

## 1.2 Beitrag

Eine offene Frage von Klawitter und Zink [KZ21] war, wo der Übergang von NP-Schwere zu Polynomialzeit liegt. Davon haben wir uns inspirieren lassen und liefern zwei Beiträge zu diesem Thema.

Die erste Erkenntnis in Kapitel 3 behandelt maximale Außenpfade, eine Unterklasse der Außenpfade, welche wiederum eine Unterklasse der außenplanaren Graphen sind. Für diese Klasse geben wir einen Algorithmus an, der in Polynomialzeit eine aufwärtsplanare Zeichnung mit drei Steigungen für einen gegebenen Graphen generiert, sofern eine derartige Zeichnung existiert. Außerdem zeigen wir in Abschnitt 3.4 auch, wie man diesen Algorithmus auf maximale außenplanare Graphen ausweiten kann.

Für den zweiten Beitrag lösen wir uns in Kapitel 4 von dieser Unterklasse und betrachten allgemeine Außenpfade. Hier zeigen wir, dass das Entscheidungsproblem, ob eine aufwärtsplanare Zeichnung mit drei Steigungen existiert, NP-schwer ist. Hierfür betrachten wir die NP-Schwere-Reduktion für außenplanare Graphen in [KZ21] und geben eine Konstruktion vor, mit der diese Reduktion auch auf Außenpfade anwendbar ist.

Eine Übersicht der Ergebnisse zur Konstruktion von aufwärtsplanaren Zeichnungen mit drei Steigungen ist in Tabelle 1.1 zu finden.

**Tab. 1.1:** Komplexität der Erstellung von aufwärtsplanaren Zeichnungen mit drei Steigungen.  
P steht für Polynomialzeit lösbar, NPs für NP-schwer.

Graphenklasse	feste Einbettung		freie Einbettung	
Baum	P	[KZ21]	P	[KZ21]
Serien-parallel	P	[Qua21]	P	[Qua21]
Kaktus	P	[KZ21]	P	[KZ21]
Maximaler Außenpfad	P	[Korollar 3.7]	P	[Korollar 3.8]
Außenpfad	NPs	[Satz 4.2]	NPs	[Korollar 4.3]
Maximal außenplanar	P	[Korollar 3.9]	P	[Korollar 3.9]
Außenplanar	NPs	[KZ21]	NPs	[KZ21]
Planar	NPs	[KZ21]	NPs	[KZ21]

## 2 Grundlagen

In diesem Kapitel werden wir die Grundlagen der nachfolgenden Arbeit betrachten. Dafür werden wir zuerst wichtige Definitionen und Begriffe kennenlernen und anschließend den Beweis zur NP-Schwere von außenplanaren Graphen betrachten.

### 2.1 Definitionen und Begriffe

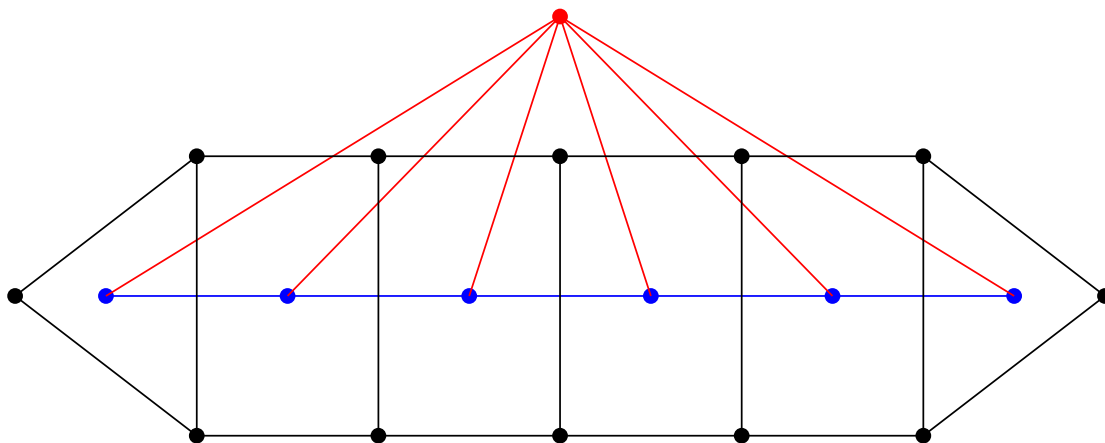
Eine *Zeichnung* (oder *Einbettung*) eines Graphen  $G = (V, E)$  ordnet jedem Knoten  $v \in V$  einen Punkt und jeder Kante  $e \in E$  eine Kurve in der *Zeichenfläche* zu. Die hierbei in der Zeichenfläche durch Kanten abgegrenzten Bereiche nennt man *Facetten*. Diese lassen sich in zwei Kategorien unterteilen. *Innenfacetten* sind Facetten, welche komplett durch Kanten des Graphen eingegrenzt werden, die *Außenfacette* hingegen umspannt die gesamte Zeichnung des Graphen.

Falls für einen gegebenen Graphen  $G$  eine Zeichnung  $\Gamma$  existiert, bei der sich keine Kanten kreuzen, nennen wir  $G$  *planar* und  $\Gamma$  *planare Zeichnung*. Darüber hinaus heißt  $G$  *außenplanar*, wenn zusätzlich jeder Knoten an der Außenfacette liegt. Außerdem nennen wir einen (außen)planaren Graphen *maximal*, falls keine Kante zwischen zwei Knoten mehr hinzugefügt werden kann, ohne dabei die (Außen)planarität zu verletzen.

Wir nennen eine planare Zeichnung eines gerichteten Graphen *aufwärtsplanar*, wenn jede Kante beginnend bei ihrem Startknoten streng monoton steigt. Wir beschränken uns in dieser Arbeit auf geradlinige Zeichnungen bei denen jede Kante eine Strecke ist und somit einen Winkel zwischen  $0^\circ$  und  $180^\circ$  zur Horizontalen besitzt. Damit ein gegebener Graph eine solche Zeichnung besitzt, muss er kreisfrei sein.

Für eine planare Zeichnung  $\Gamma$  eines Graphen  $G$  können wir darüber hinaus auch noch den zugehörigen *Dualgraphen*  $D$  definieren. Wir erhalten  $D$ , indem wir jede Facette von  $\Gamma$  als einen Knoten auffassen und anschließend zwei Knoten über eine Kante verbinden, wenn die entsprechenden Facetten in  $\Gamma$  eine Kante teilen. Den *schwachen Dualgraphen*  $D'$  erhalten wir, indem wir den Knoten aus  $D$  streichen, welcher die Außenfacette repräsentiert, sowie alle mit ihm verbundenen Kanten. In Abbildung 2.1 befindet sich eine planare Zeichnung eines Graphen sowie ihr zugehöriger Dualgraph und ihr schwacher Dualgraph.

Wir werden uns hauptsächlich mit einer Teilklasse der außenplanaren Graphen beschäftigen, den *Außenpfaden*. Eine außenplanare Zeichnung  $\Gamma$  heißt *Außenpfad*, wenn der zugehörige schwache Dualgraph  $D'$  einen Pfad bildet. Ferner bezeichnen wir  $\Gamma$  als *maximal*, wenn  $\Gamma$  zweifach knotenzusammenhängend ist und wir keine Kanten mehr hinzufügen können, ohne dass  $\Gamma$  danach kein Außenpfad mehr ist. Ein Graph  $G$  heißt (*maximaler*) *Außenpfad*, wenn er eine Zeichnung besitzt, die ein (*maximaler*) *Außenpfad* ist.



**Abb. 2.1:** Zeichnung eines Außenpfades sowie des zugehörigen Dualgraphen (rot und blau) und des schwachen Dualgraphen (blau).

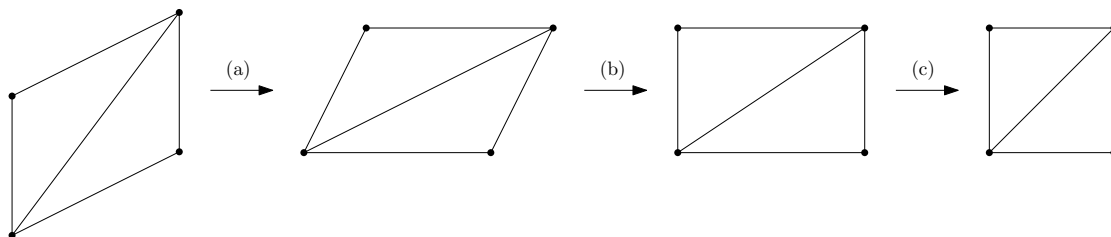
Für einen maximalen Außenpfad  $G$  mit  $n$  Knoten können wir die *Stacking Order*  $O$  definieren. Diese ist eine Sortierung der Knotenmenge, wobei der erste Knoten  $v_1$  vom Knotengrad 2 ist und der zweite Knoten  $v_2$  über eine Kante mit  $v_1$  verbunden ist. Für  $i = 3, \dots, n$  gilt, dass  $v_i$  über jeweils eine Kante mit genau zwei Knoten aus  $\{v_1, \dots, v_{i-1}\}$  verbunden ist. Da ein maximaler Außenpfad genau zwei Knoten vom Grad 2 besitzt und für den zweiten Knoten jeweils zwei Knoten zur Auswahl stehen, ist die Stacking Order nicht eindeutig.

Das Ziel der Arbeit ist es, aufwärtsplanare Zeichnungen von Außenpfaden zu erstellen, welche nur geradlinige Kanten mit drei verschiedenen Steigungen verwenden. Dies bedeutet offensichtlich, dass alle Knoten maximal Eingangs- und Ausgangsgrad 3 besitzen dürfen. Wie in Abbildung 2.2 zu sehen ist, können wir durch die affinen Transformationen Drehen, Scheren und Strecken jede beliebige Wahl der Steigungen in die Steigungen  $\{45^\circ, 90^\circ, 135^\circ\} = \{\nearrow, \uparrow, \nwarrow\}$  übersetzen. Dies bietet den Vorteil, dass alle Knoten rationale Koordinaten besitzen. Daher werden wir, sofern nicht weiter angegeben, diese Steigungen verwenden und in Abschnitt 2.2 und Kapitel 4 alle Zeichnungen um  $45^\circ$  drehen, sodass wir hier die Steigungen  $\{\uparrow, \nearrow, \rightarrow\}$  erhalten. Des Weiteren werden wir, sofern nicht explizit angegeben, immer von gerichteten Graphen sprechen.

## 2.2 NP-Schwere für außenplanare Graphen nach Klawitter und Zink

Wie bereits erwähnt, haben Klawitter und Zink [KZ21] für außenplanare Graphen gezeigt, dass das Entscheidungsproblem, ob zu einem gegebenen Graphen eine aufwärtsplanare Zeichnung mit drei Steigungen existiert, NP-schwer ist. Dafür haben sie eine Konstruktion erstellt, mit deren Hilfe sie eine Variante von 3-SAT, das *planare monotone 3-SAT*, welches NP-schwer ist [DK12], auf das Problem, ob für außenplanare Graphen eine aufwärtsplanare Zeichnung mit drei Steigungen existiert, reduziert haben. Bei



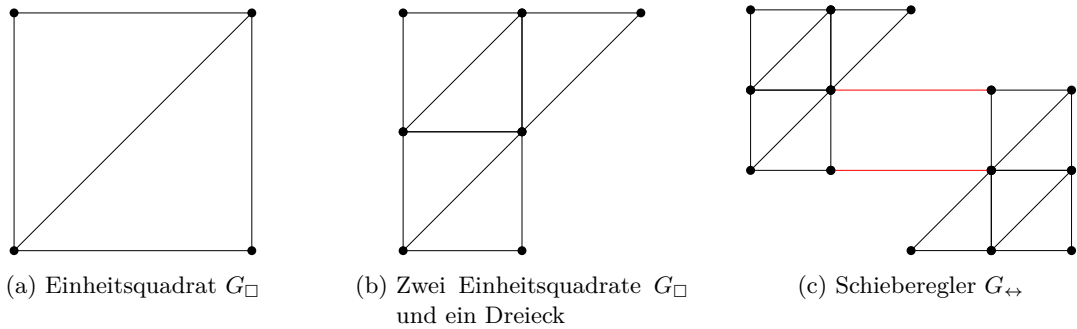


**Abb. 2.2:** Jede gegebene Menge von drei verschiedenen Steigungen lässt sich durch Drehen (a), Scheren (b) und Strecken (c) zu  $\{\searrow, \uparrow, \nearrow\}$  transformieren (hier zu veranschaulichenden Zwecken um  $45^\circ$  gedreht zu  $\{\uparrow, \nearrow, \rightarrow\}$ ).

dieser Abwandlung muss jede Klausel monoton sein, sie darf also nur aus entweder unnegierten oder negierten Variablen bestehen. Klauseln, welche nur unnegierte Variablen beinhalten, nennen wir *positive Klauseln* und dementsprechend sind *negative Klauseln* Klauseln, welche nur aus negierten Variablen bestehen. Der *Inzidenzgraph* einer 3-SAT Formel besitzt für jede Variable und jede Klausel einen Knoten und eine Kante zwischen Klausel- und Variabelknoten, wenn die entsprechende Variable in der Klausel auftaucht. Der Inzidenzgraph einer Instanz des Problems muss planar sein und alle positiven Klauseln oberhalb sowie alle negativen Klauseln unterhalb der Variablen gezeichnet werden [DK12]. In diesem Abschnitt lernen wir diese Konstruktion kennen.

Für eine rechteckige Zeichnung des Inzidenzgraphen einer gegebenen planaren monotonen 3-SAT Formel  $F$  haben Klawitter und Zink eine Vorschrift angegeben, mit der man einen außenplanaren Graphen  $G_F$  konstruieren kann, welcher genau dann aufwärtsplanar zeichnenbar ist und dabei nur geradlinige Kanten mit drei verschiedenen Steigungen verwendet, wenn  $F$  erfüllbar ist. Ein solches  $G_F$  setzt sich aus drei Elementen zusammen. Das Erste ist das *Einheitsquadrat*  $G_\square$ , zu sehen in Abbildung 2.3a, dessen Zeichnung bis auf Spiegelung an der Diagonalen eindeutig ist. Dadurch müssen zwei aneinanderhängende  $G_\square$  gleich groß gezeichnet werden. Diese  $G_\square$  stellen den größten Teil der Konstruktion dar. Das zweite Element ist ein simples *Dreieck*, welches dadurch einem halben Einheitsquadrat  $G_\square$  entspricht, wie in Abbildung 2.3b zu sehen ist. Dieses Dreieck wird immer im Zusammenhang mit dem letzten Element, einem *Schieberegler*  $G_{\leftrightarrow}$ , verwendet, damit gewährleistet wird, dass der Mittelteil eines  $G_{\leftrightarrow}$  die Form eines Rechtecks annimmt, indem an zwei gegenüberliegenden Knoten der maximale Eingangs- bzw. Ausgangsgrad erreicht wird. Ein solches  $G_{\leftrightarrow}$  ist in Abbildung 2.3c abgebildet. Die beiden roten Kanten besitzen im Gegensatz zu allen anderen Kanten keine feste Länge, sondern können nach Bedarf ein- und ausgefahren werden. Mithilfe dieser Schieberegler werden die Belegungen der Variablen festgelegt und an die Klauseln übertragen und die Klauseln so ausgerichtet, dass der Graph bei einer gültigen Belegung zeichnenbar ist.

Variablen stellen sie durch *Variabel-Gadgets* dar, Klauseln werden mithilfe von *Klausel-Gadgets* repräsentiert und diese werden durch *Kanten-Gadgets* miteinander verbunden. In Abbildung 2.4a befindet sich eine Darstellung eines Inzidenzgraphen einer planaren monotonen 3-SAT Formel  $F$ , bei der positive Klauseln grün, negative Klauseln rot, Variablen blau und Kanten gelb gefärbt wurden. Eine außenplanare Zeichnung nach der



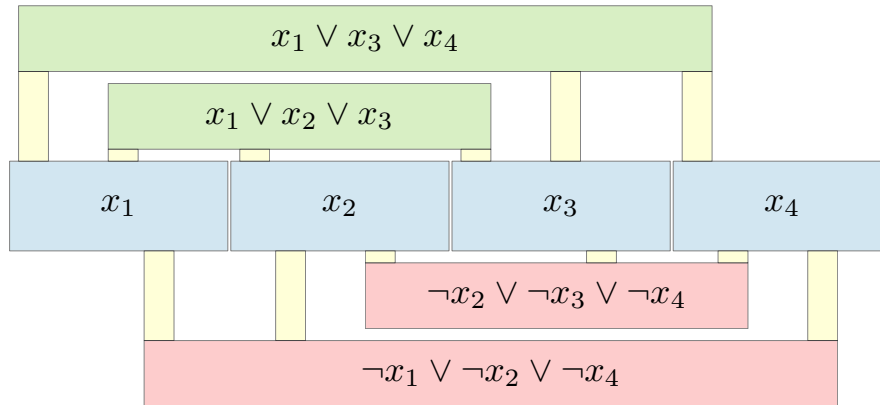
**Abb. 2.3:** Die drei Bausteine eines außenplanaren Graphen  $G_F$  [KZ21].

Konstruktion von Klawitter und Zink ist in Abbildung 2.4b zu sehen und die entsprechenden Gadgets wurden auch hier mit den gleichen Farben markiert. Den lila gefärbte Teil des Graphen bezeichnen sie als *Rahmen*, welcher nur aus Einheitsquadraten  $G_{\square}$  besteht und somit starr ist, wodurch er die gesamte Konstruktion festhält und die einzelnen Elemente nur innerhalb davon bewegt werden können.

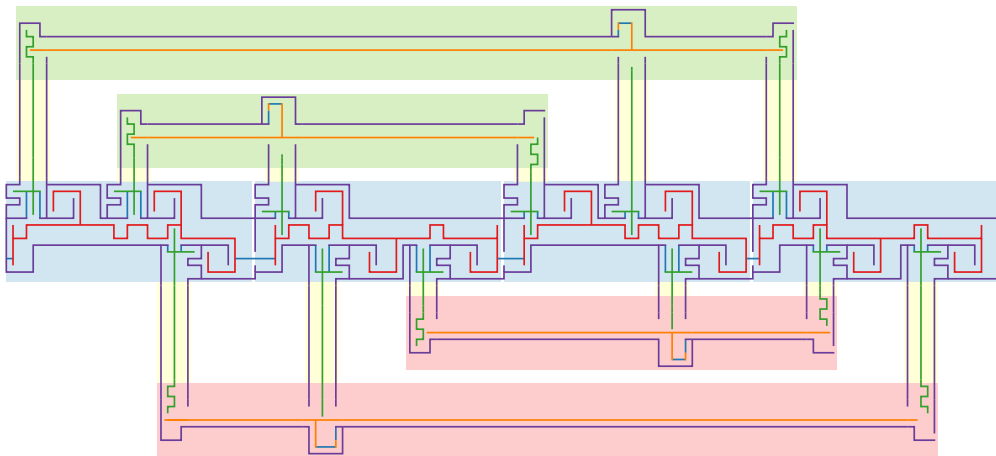
Betrachten wir zunächst ein Variabel-Gadget. Wie in Abbildung 2.5 zu sehen ist, besteht ein solches Gadget aus mehreren Elementen. Mithilfe der rot gekennzeichneten *Bürste* wird der Wahrheitswert der zugehörigen Variable festgelegt. Diese Bürste kann nach links oder rechts verschoben werden, wird dabei allerdings vom Rahmen beschränkt. Ist die Variable auf *wahr* gesetzt, wird die Bürste am rechten Rand des Rahmens positioniert, und entsprechend bei *falsch* am linken Rand. Die grünen Abschnitte, welche oben oder unten am Gadget platziert sind, bezeichnen sie als *Bolzen*. Für diese Bolzen gibt es Einkerbungen im Rahmen, in denen sie sich einhaken können. Diese Kerben sind auch die einzigen Positionen, an denen genug Platz für einen Bolzen ist. Darüber hinaus gibt es noch eine weitere Einkerbung in der Bürste. Abhängig von der Konfiguration der Variable ist diese Kerbe für den Bolzen zugänglich und er kann eingefahren werden, sodass dieser Bolzen den Zustand der Variable übertragen kann. Die blau markierten Elemente, welche die Bolzen mit dem Rahmen verbinden, sind die zuvor beschriebenen Schieberegler  $G_{\leftrightarrow}$ .

Betrachten wir als nächstes ein Kanten-Gadget. Dieses besteht nur aus drei vertikalen Teilgraphen, welche Ketten von Einheitsquadraten sind. Sie dienen dazu, ein Variabel-Gadget mit einem Klausel-Gadget zu verbinden und die Konfiguration der Variable an die Klausel zu übermitteln.

Abschließend betrachten wir auch noch ein Klausel-Gadget. Wie in Abbildung 2.6 zu sehen, besitzt ein Klausel-Gadget sieben zulässige Konfigurationen, lediglich die Konfiguration *falsch-falsch-falsch* bei einer positiven Klausel bzw. *wahr-wahr-wahr* bei einer negativen Klausel führt zu einem Bruch mit der Planarität der Zeichnung. Dies entspricht auch den gültigen Konfigurationen von  $F$ . Dabei besteht ein Klausel-Gadget aus den Enden von drei Bolzen, zwei senkrechten, orange gefärbten *Seilen*, welche über einen Schieberegler miteinander und über einen weiteren Schieberegler mit dem Rahmen verbunden sind, und einer waagrechten, orange gefärbten *Leiste*, welche nur mit einem Seil



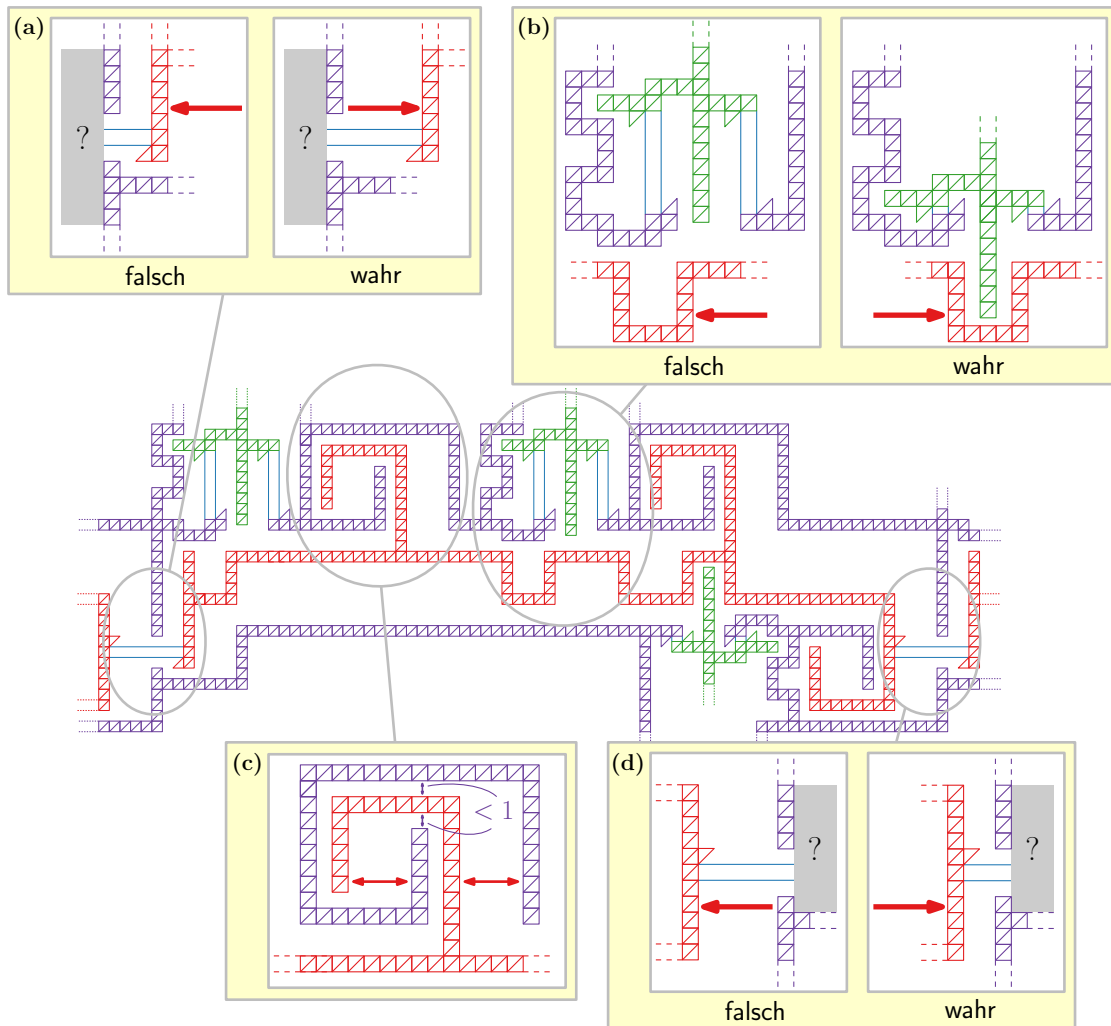
(a) Rechteckige Darstellung des Inzidenzgraphen einer planaren monotonen 3-SAT Formel  $F$



(b) Außenplanare Zeichnung des Graphen  $G_F$ . Hier sind die Variablen  $x_1$  und  $x_4$  negativ,  $x_2$  und  $x_3$  positiv eingestellt.

**Abb. 2.4:** Schematische Darstellung der NP-Schwere-Reduktion nach Klawitter und Zink [KZ21].

verbunden ist. Diese Leiste verhindert, dass eine ungültige Konfiguration gezeichnet werden kann. Ihre Position wird mithilfe von Schieberegler an den Seilen reguliert, wobei der Schieberegler zwischen den Seilen die Leiste horizontal verschieben kann und der Schieberegler zwischen Seil und Rahmen für vertikale Verschiebungen genutzt wird.



**Abb. 2.5:** Beispiel für ein auf falsch gesetztes Variabel-Gadget [KZ21].

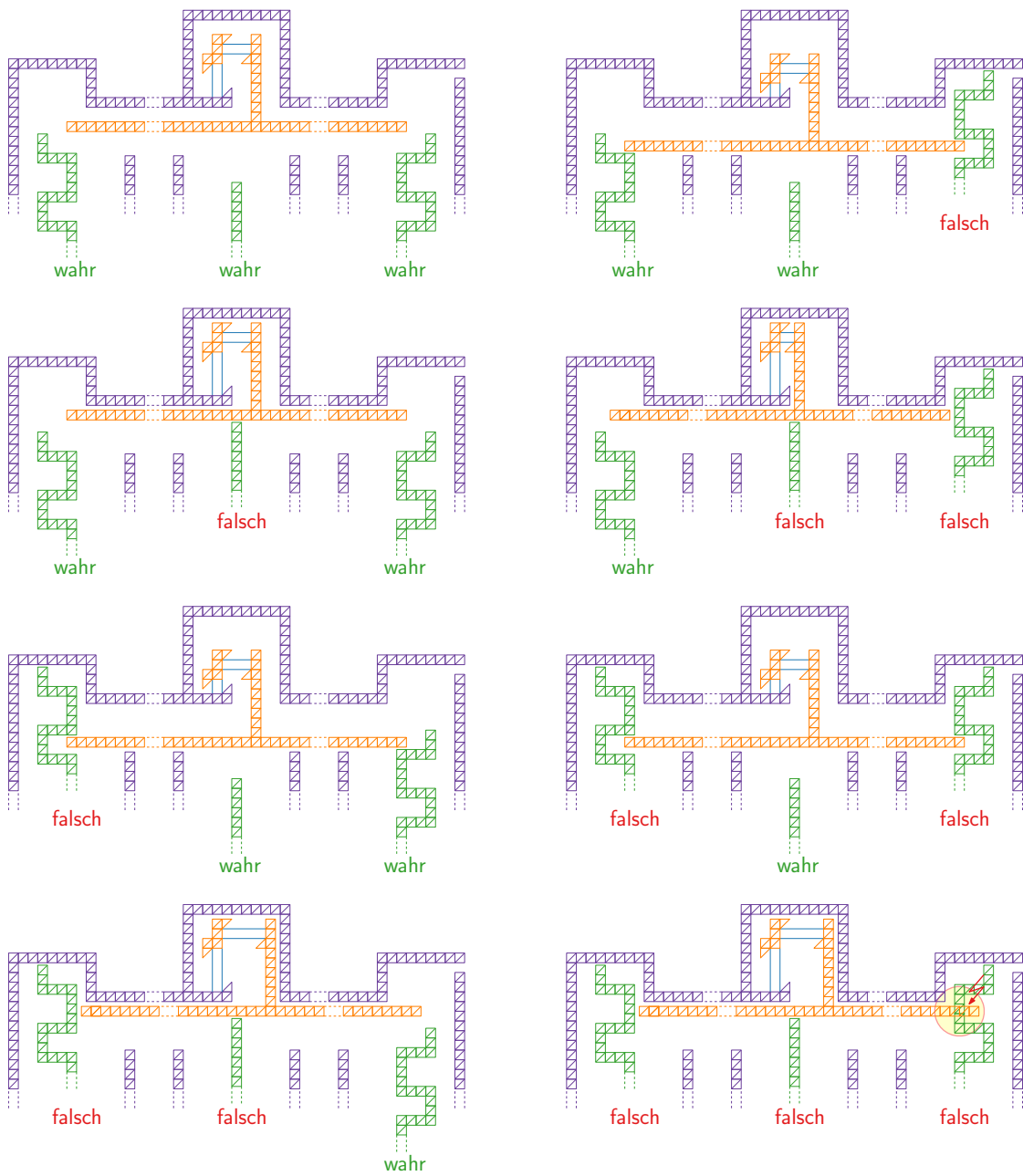


Abb. 2.6: Ein positives Klausel-Gadget in allen acht Konfigurationsmöglichkeiten [KZ21].

## 3 Maximale Außenpfade

Bevor wir uns mit allgemeinen Außenpfaden beschäftigen, betrachten wir zuerst den Spezialfall der maximalen Außenpfade.

Zur Erinnerung, ein Außenpfad  $G$  ist maximal, wenn er zweifach zusammenhängend ist und keine weiteren Kanten mehr hinzugefügt werden können, ohne dabei die Außenpfad-Eigenschaft zu verletzen. Dies bedeutet, dass alle Facetten einer außenplanaren Zeichnung von  $G$  von genau drei Kanten begrenzt werden. Da sich keine Kanten kreuzen dürfen, muss also jede Facette ein Dreieck bilden.

Damit erhalten wir zwei wichtige Eigenschaften für maximale Außenpfade.

**Beobachtung 3.1.** *Ein maximaler Außenpfad mit  $n \geq 2$  Knoten besitzt  $2n - 3$  Kanten.*

**Beobachtung 3.2.** *Jede außenplanare Zeichnung eines (un)gerichteten maximalen Außenpfads besteht aus aneinandergereihten Dreiecken.*

Wie in Abbildung 3.1 zu sehen ist, gilt Beobachtung 3.2 bei planaren Zeichnungen im Allgemeinen nicht. Begrenzt man allerdings die Anzahl der Steigungen auf drei, so lässt sich dies auch auf planare Zeichnungen übertragen.

**Beobachtung 3.3.** *Jede planare Zeichnung eines (un)gerichteten maximalen Außenpfads, welche nur drei Steigungen verwendet, besteht aus aneinandergereihten Dreiecken.*

Zusätzlich können wir die Form eines Dreiecks genau vorgeben, wenn wir die Steigungen der drei Kanten kennen. Wissen wir darüber hinaus auch die Länge einer dieser Kanten, ist auch die Größe des Dreiecks eindeutig.

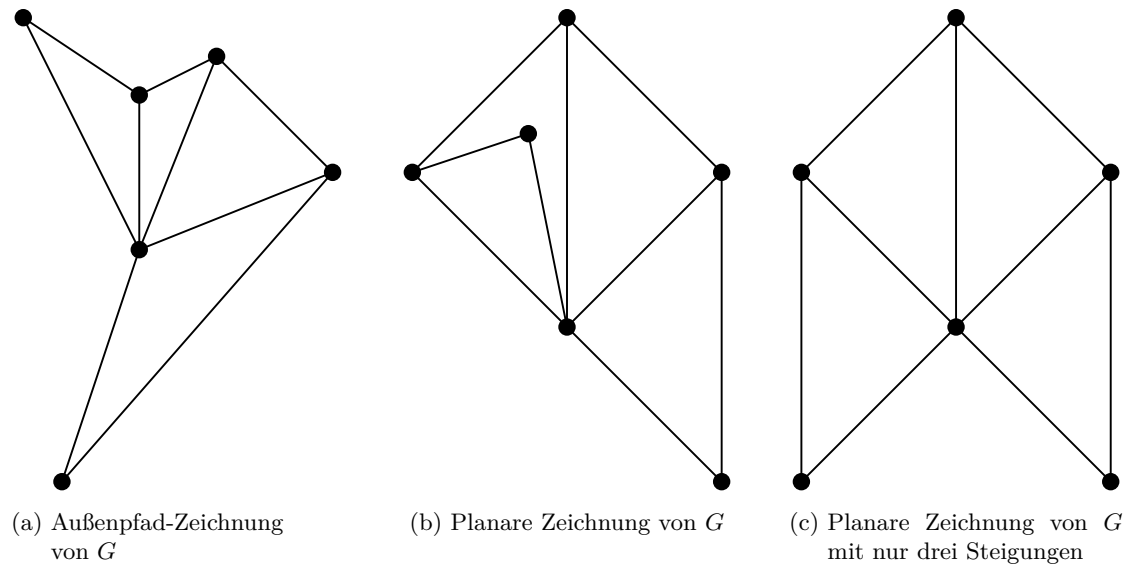
**Beobachtung 3.4.** *Bei einer aufwärtsplanaren Zeichnung mit drei Steigungen des Graphen  $G = (\{u, v, w, x\}, \{uv, uw, vw, vx, wx\})$  besitzen Kanten mit derselben Steigung dieselbe Länge.*

Ein  $k$ -*d-Baum* ist ein balancierter Suchbaum zum Speichern von Punkten aus  $\mathbb{R}^2$ , welcher in  $\mathcal{O}(n \log n)$  Zeit erstellt werden kann. Wir verwenden einen solchen Suchbaum, um die Planarität der entstandenen Zeichnung zu überprüfen.

Mithilfe dieser Eigenschaften können wir nun einen effizienten Algorithmus für maximale Außenpfade präsentieren.

### 3.1 Algorithmus

Unser Algorithmus lässt sich in drei Abschnitte unterteilen, Vorverarbeitung, Zeichnung und Planaritätstest.



**Abb. 3.1:** Zeichnungen eines maximalen Außenpfads  $G$

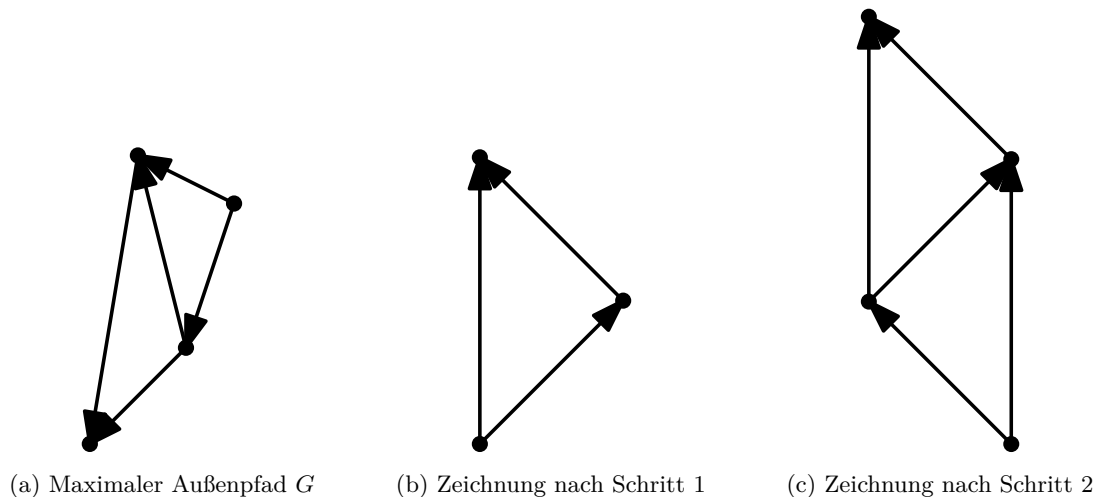
**Vorverarbeitung.** Für einen gegebenen Außenpfad  $G = (V, E)$  erzeugen wir bei der Vorverarbeitung eine Stacking Order. Dafür wählen wir zuerst einen Knoten vom Grad 2, welcher als Startknoten fungiert. Als nächstes bestimmen wir die beiden Knoten, welche mit unserem Startknoten verbunden sind. Anschließend erhalten wir die Reihenfolge der restlichen Knoten, indem wir als nächstes den Knoten wählen, welcher mit zwei Knoten aus unserer Stacking Order über je eine Kante verbunden ist.

**Zeichnung.** Wir beginnen im ersten Schritt mit einer Zeichnung der ersten drei Knoten unserer Stacking Order und den Kanten zwischen ihnen. Wie wir in Abschnitt 2.1 besprochen haben, haben wir uns für die Steigungen  $\{45^\circ, 90^\circ, 135^\circ\} = \{\nearrow, \uparrow, \searrow\}$  entschieden. Die Größe kann beliebig gewählt werden und gibt die Größe für alle Facetten vor.

Im  $i$ -ten Schritt wählen wir den nächsten zu zeichnenden Knoten nach der Stacking Order sowie die beiden Kanten, welche ihn mit der im vorherigen Schritt erhaltenen Zeichnung verbindet. Damit wir den Kanten die korrekte Steigung zuweisen, wählen wir die Steigungen so, dass ein Knoten der geteilten Kante in beiden Facetten unterschiedliche Steigungen verwendet. Nach Beobachtung 3.4 besitzen in zwei benachbarten Facetten Kanten mit derselben Steigung, dieselbe Länge. Durch Zeichnen der beiden Kanten erhalten wir die Position für den neuen Knoten und können auch diesen einzeichnen.

Diesen Schritt wiederholen wir solange, bis alle Knoten gezeichnet wurden. In Abbildung 3.2 befindet sich eine Veranschaulichung der Zeichnung eines maximalen Außenpfads.

**Planaritätstest.** Abschließend wird noch die Planarität der entstandenen Zeichnung geprüft. Da wir bereits wissen, dass es sich bei der Zeichnung um eine Aneinanderrei-



**Abb. 3.2:** Veranschaulichung des Algorithmus für  $G$

hung von kongruenten Dreiecken handelt, kann die Planarität nur verletzt werden, wenn zwei Knoten aufeinander liegen, also sich dieselben Koordinaten teilen. Daher verwenden wir einen k-d-Baum, bei dessen Erstellung wir in jedem Schritt bei allen Knoten, welche auf der Medianachse liegen, prüfen, ob sie dieselben Koordinaten besitzen wie der Medianknoten.

## 3.2 Korrektheit

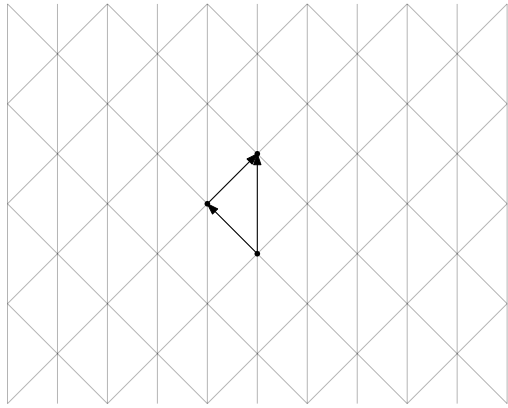
Nach Konstruktion liefert der Algorithmus eine Zeichnung des Graphen  $G$ , welche genau drei streng monotone Steigungen für die Kanten verwendet.

Die Planarität kann durch drei verschiedene Situationen verletzt werden: zwei Kanten kreuzen sich, ein Knoten liegt auf einer Kante oder ein Knoten liegt auf einem anderen Knoten. Erzeugt man nach der Wahl des ersten Dreiecks ein Gitter über der Zeichenfläche wie in Abbildung 3.3 zu sehen, erhält man alle möglichen Positionen der kongruenten Dreiecke, also alle möglichen Positionen für Knoten und Kanten. Dadurch wird ersichtlich, dass die Planarität nur gebrochen wird, wenn zwei Knoten dieselben Koordinaten besitzen. Daher ist es hinreichend nur diesen Fall zu überprüfen.

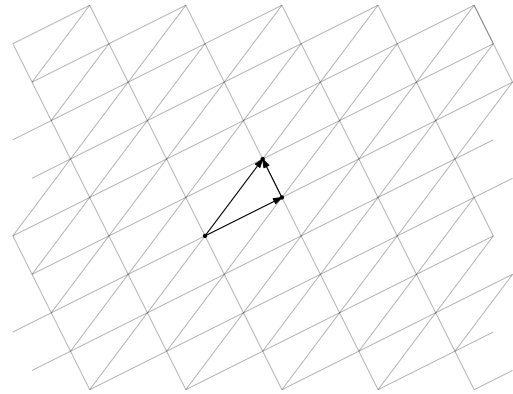
**Satz 3.5.** *Wenn ein maximaler Außenpfad  $G$  eine aufwärtsplanare Zeichnung mit drei Steigungen besitzt, liefert der angegebene Algorithmus eine solche Zeichnung.*

*Beweis.* Da  $G$  ein maximaler Außenpfad ist, wissen wir aus Beobachtung 3.3, dass jede Facette einer aufwärtsplanaren Zeichnung ein Dreieck darstellt, wenn sie nur drei Steigungen verwendet. Da sich jedes Dreieck eine Kante mit einem benachbarten Dreieck teilt und somit eine Kante und die zu verwendenden Steigungen vorgegeben werden, müssen alle Facetten sogar kongruente Dreiecke bilden. Dadurch ist unsere Zeichnung bis auf Spiegelung an der vertikalen Achse eindeutig. Da unser Algorithmus eine Zeich-





(a) Gittermuster für die Steigungen  $45^\circ$ ,  $90^\circ$  und  $135^\circ$



(b) Gittermuster für die Steigungen  $26,57^\circ$ ,  $59,04^\circ$  und  $116,57^\circ$

**Abb. 3.3:** Dreiecksgitter für Positionen der Knoten auf der Zeichenfläche

nung mit drei streng monotonen Steigungen liefert, liefert er somit eine aufwärtsplanare Zeichnung mit drei Steigungen, sofern diese existiert.  $\square$

### 3.3 Laufzeit

**Satz 3.6.** *Der in Abschnitt 3.1 angegebene Algorithmus hat eine Gesamtlaufzeit von  $\mathcal{O}(n \log^2 n)$ .*

*Beweis.* Die Laufzeit setzt sich aus den drei Abschnitten Vorverarbeitung, Zeichnung sowie Planaritätstest zusammen.

**Vorverarbeitung.** Zum Erstellen der Stacking Order eines gegebenen Graphens  $G = (V, E)$  greifen wir auf einen Rot-Schwarz-Baum zurück, der neben dem Knoten  $v$ , auch den Knotengrad  $g$  und alle Knoten  $z_1, \dots, z_6$ , welche mit  $v$  verbunden sind, in seinen Einträge speichert. Bei der Erstellung des Baums durchlaufen wir  $E$  einmal komplett und führen für jede Kante  $e \in E$  zwei Suchoperationen und für jeden Knoten  $v \in V$  eine Insert-Operation aus. Damit erhalten wir hier bereits eine Laufzeit von  $\mathcal{O}(n \log n)$ . Anschließend durchlaufen wir den Baum solange, bis wir auf einen Eintrag mit Knotengrad  $g = 2$  stoßen und wählen diesen als Startknoten  $v_1$ . Damit führen wir nun Algorithmus 1 aus. Bei diesem Pseudocode haben wir alle Arrays mit 1 indiziert. Für die Suchoperationen benötigen wir wieder jeweils  $\mathcal{O}(\log n)$  Laufzeit. Da die Knotengrade auf  $g = 6$  beschränkt sind, durchlaufen wir die Schleifen in den Zeilen 10 und 11 sowie 17 und 18 im schlechtesten Fall je 36 mal, wobei die if-Abfrage nur genau einmal wahr wird, weshalb trotzdem maximal eine Suchoperation ausgeführt wird. Die äußere For-Schleife besitzt  $n - 3$  Durchläufe und führt somit im schlechtesten Fall zwei Suchoperationen pro Durchlauf aus, wodurch sich hier eine Laufzeit von  $\mathcal{O}(n \log n)$  ergibt. Führt man

---

**Algorithmus 1:** erzeugeStackingOrder

---

**Eingabe:** Rot-Schwarz-Baum  $T$ , Länge  $n$ , Startknoten  $v_1$

**Ausgabe:** Stacking Order  $O$  von  $V$

```
1  $t_0 = search(T, v_1)$ 
2  $O[1] = v_1$ 
3  $O[2] = t_0[3]$ 
4  $O[3] = t_0[4]$ 
5  $t_0 = search(T, O[2])$ 
6  $t_1 = t_0$ 
7  $t_h = t_0$ 
8  $changed = falsch$ 
9 for  $i \leftarrow 4$  to  $n$  do
10    $t_1 = t_0$ 
11    $t_0 = search(T, O[i - 1])$ 
12   for  $j \leftarrow 3$  to  $t_0[2] + 2$  do
13     for  $k \leftarrow 3$  to  $t_1[2] + 2$  do
14       if  $t_0[j] == t_1[k]$  then
15          $O[i] = t_0[j]$ 
16          $t_h = search(T, t_0[j])$ 
17          $changed = wahr$ 
18   if  $changed == falsch$  then
19     for  $j \leftarrow 3$  to  $t_0[2] + 2$  do
20       for  $k \leftarrow 3$  to  $t_h[2] + 2$  do
21         if  $t_0[j] == t_h[k]$  then
22            $O[i] = t_0[j]$ 
23    $changed = falsch$ 
24 return  $O$ 
```

---

dies nun mit der Laufzeit zur Erstellung des Rot-Schwarz-Baumes zusammen, bleibt die Laufzeit zur Erstellung der Stacking Order bei  $\mathcal{O}(n \log n)$ .

**Zeichnung.** Nach Beobachtung 3.1 besitzt jeder maximale Außenpfad  $2n - 3$  Kanten. Das Zeichnen der Knoten und Kanten ist in konstanter Zeit möglich. Damit erhalten wir hier eine Laufzeit von  $\mathcal{O}(n)$ .

**Planaritätstest.** Für das Erstellen des k-d-Baums benötigen wir  $\mathcal{O}(n \log n)$  Zeit. Da wir in jedem Schritt noch die Koordinaten der Knoten auf der Medianachse mit denen des Medianknotens abgleichen, erhalten wir zusätzlich noch eine Laufzeit von  $\mathcal{O}(\log n)$ , da im schlechtesten Fall die Hälfte aller Knoten des Schnitts auf der Achse liegen. Daher benötigen wir für diesen Teil eine Laufzeit von  $\mathcal{O}(n \log^2 n)$ .

Damit ergibt sich die Gesamtlaufzeit von  $\mathcal{O}(n \log^2 n)$ . □

Da wir mit dem angegebenen Algorithmus zuerst eine Zeichnung mit drei streng monotonen Steigung erhalten und anschließend überprüfen, ob diese aufwärtsplanar ist, können wir folgendes Korollar ableiten.

**Korollar 3.7.** *Das Entscheidungsproblem, ob ein maximaler Außenpfad eine aufwärtsplanare Zeichnung mit drei Steigungen besitzt, kann in Polynomialzeit entschieden werden.*

Darüber hinaus ist die Zeichnung bis auf Spiegelung an der vertikalen Achse eindeutig, weshalb unsere Behauptung auch im Szenario einer variablen Einbettung hält und wir können folgendes Korollar ableiten.

**Korollar 3.8.** *Das Entscheidungsproblem, ob ein maximaler Außenpfad eine aufwärtsplanare Zeichnung mit drei Steigungen besitzt, kann im Szenario einer variablen Einbettung in Polynomialzeit entschieden werden.*

### 3.4 Maximale außenplanare Graphen

Nachdem wir einen geeigneten Algorithmus für maximale Außenpfade präsentiert haben, werden wir nun noch skizzieren, dass man diesen Algorithmus durch eine Anpassung auch bei maximalen außenplanaren Graphen verwenden kann.

Im Gegensatz zu Außenpfaden können wir bei außenplanaren Graphen keine Stacking Order mehr festlegen, da im Allgemeinen Facetten mehr als zwei benachbarte Facetten haben können. Da allerdings auch maximale außenplanare Graphen bei  $n$  Knoten genau  $2n - 3$  Kanten haben und jede Facette einer außenplanaren Zeichnung ein Dreieck bildet, zerlegen wir den Graph zuerst so, dass wir maximale Außenpfade erhalten. Nun wenden wir für jeden dieser Teilgraphen den zuvor vorgestellten Algorithmus an, mit der Einschränkung, dass wir nach dem ersten Durchlauf, Form und Größe des Dreiecks für alle anderen Teilgraphen vorgeben. Die entstehenden Zeichnungen setzen wir anschließend wieder zu einer Zeichnung zusammen und überprüfen abschließend noch, ob keine Knoten aufeinander liegen.

**Korollar 3.9.** *Das Entscheidungsproblem, ob ein maximaler außenplanarer Graph eine aufwärtsplanare Zeichnung mit drei Steigungen besitzt, ist in Polynomialzeit entscheidbar.*

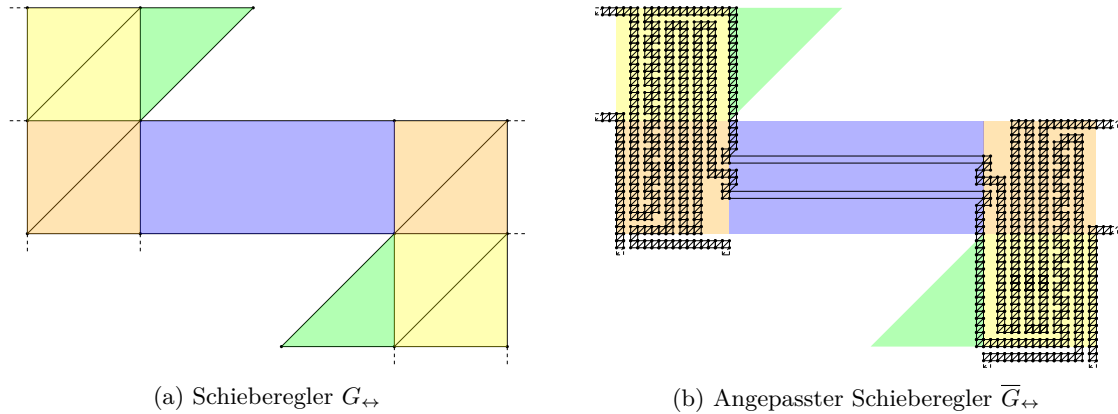
## 4 Allgemeine Außenpfade

Wenn wir uns nun von der Unterklasse der maximalen Außenpfade lösen und allgemeine Außenpfade betrachten, können wir den gerade vorgestellten Algorithmus nicht verwenden. Wie bereits besprochen sind Außenpfade eine Unterklasse der außenplanaren Graphen und wir wissen dank Klawitter und Zink bereits, dass das Entscheidungsproblem, ob zu einem gegebenen außenplanaren Graphen eine aufwärtsplanare Zeichnung mit drei Steigungen existiert, NP-schwer ist. In diesem Abschnitt werden wir eine Vorschrift präsentieren, welche aus der von ihnen verwendeten und in Abschnitt 2.2 zusammengefassten Konstruktion einen Außenpfad erzeugt, der die gleichen relevanten Eigenschaften wie das Original besitzt. Damit können wir die NP-Schwere-Reduktion für außenplanare Graphen auch auf Außenpfade übertragen.

### 4.1 Anpassung des NP-Schwere-Beweises

Wir verfolgen die Idee, den Rand des außenplanaren Graphen mit einem Außenpfad nachzubauen. Dafür verwenden wir Einheitsquadrate  $\overline{G}_\square$ , welche dieselbe Form wie ein  $G_\square$  aus der ursprünglichen Konstruktion besitzen. Wir werden jedoch in unserer Konstruktion mehr Einheitsquadrate benötigen und somit ein Verhältnis von 1 : 16 der Seitenlänge erhalten, wenn man beide Konstruktionen übereinander legt. Wir bezeichnen die Seitenlänge der Einheitsquadrate  $\overline{G}_\square$  als *Einheitslänge 1*. Im Folgenden werden wir eine Vorschrift präsentieren, wie wir die in Abbildung 2.3 dargestellten Bausteine nachbauen, und eine Verfeinerung an den Variabel-Gadgets aus Abbildung 2.5 vornehmen.

Zuerst betrachten wir die Schieberegler  $G_{\leftrightarrow}$ . Diese Elemente bestehen ursprünglich aus vier Einheitsquadraten  $G_\square$ , zwei Dreiecken und einem Verbindungsstück, wobei der Aufbau links und rechts (bzw. oberhalb und unterhalb) des Verbindungsstücks analog ist. In Abbildung 4.1a sind die entsprechenden Einheitsquadrate  $G_\square$  gelb und orange, die Dreiecke grün und das Verbindungsstück blau eingefärbt. Da die Dreiecke nur dazu dienen, an einem Schieberegler die Form für das Verbindungsstück vorzugeben, indem an zwei Knoten der maximale Knotengrad erreicht wird, werden wir die Anpassung daran einfach als Teil des Schiebereglers betrachten. Außerdem werden wir die beiden aneinanderhängenden Einheitsquadrate als ein größeres Element verwenden. Das blaue Verbindungsstück ersetzen wir durch zwei Verbindungsstücke, welche mittig mit dem minimalen Abstand von vier Einheitslängen plaziert werden. Diesen Abstand benötigen wir, da an jeder Kante ein kleines Dreieck angebracht werden muss, damit die waagrechten (bzw. senkrechten) Kanten mit jeweils einem Knoten vom Ausgangsgrad 3 verbunden sind, wodurch sie zwingend diese Steigungen verwenden müssen. Damit beide Verbindungsstücke bis auf einen Versatz von  $2\varepsilon$  mit  $\varepsilon < 1$  synchron sind, werden die beiden



**Abb. 4.1:** Anpassungen an Schieberreglern und Dreiecken

Einheitsquadrate auf jeder Seite zu einem *Schloss* umgeformt, welches verhindert, dass ein Verbindungsstück auf der entsprechenden Seite weiter als  $\varepsilon$  ein- oder ausgefahren werden kann. Dadurch ergibt sich der angegebene Versatz von  $2\varepsilon$ . Um zu verhindern, dass ein Schieberregler um mehr als 16 Einheitslängen ausgefahren wird und sich somit aus dem Schloss „aushängt“, wurden die Verbindungsstücke mittig platziert und besitzen somit einen Abstand von fünf Einheitslängen nach Außen zur Position des ursprünglichen Schieberreglers. Dadurch schneidet ein zu weit ausgefahrenes Verbindungsstück selbst mit dem maximalen Versatz von  $2\varepsilon$  den Teil des Schlosses, der mit dem anderen Verbindungsstück verbunden ist. Einheitsquadrate  $G_{\square}$ , welche den Graph senkrecht (bzw. waagrecht) an einem der Schlösser fortsetzen, müssen so fortgesetzt werden, dass die Funktion des entsprechenden Schlosses unverändert bleibt. Da die Aufgabe der grün gefärbten Dreiecke nun in die Verbindungsstücke verlegt wurde, werden wir diese nur durch eine einfache Kette von sechzehn neuen Einheitsquadraten  $\overline{G}_{\square}$  ersetzen. In Abbildung 4.1b befindet sich eine Veranschaulichung dieser Konstruktion mit senkrechten oder waagrechten Fortsetzungen an allen ursprünglichen Einheitsquadraten.

**Lemma 4.1.** *Sei  $\varepsilon < 1$ . In jeder aufwärtsplanaren Zeichnung von  $\overline{G}_{\leftrightarrow}$*

- *sind die Kanten der Verbindungsstücke parallel,*
- *sind alle Kanten wie in Abbildung 4.1b bis auf Spiegelung an einer Diagonalen ausgerichtet,*
- *sind alle vertikalen und horizontalen Kanten mit Ausnahme der Verbindungsstücke gleich lang, genauso wie alle diagonalen Kanten, und*
- *beträgt der Längenunterschied der beiden Verbindungsstücke höchstens  $2\varepsilon$ .*

*Beweis.* Durch die Anpassung wird das Verbindungsstücke eines  $G_{\leftrightarrow}$  durch zwei parallele  $G_{\leftrightarrow}$  ersetzt, wobei in unserer Veranschaulichung die Größe der Einheitsquadrate nun  $1/16$  der ursprünglichen Größe beträgt. Für  $G_{\leftrightarrow}$  haben Klawitter und Zink die ersten

drei Eigenschaften bereits bewiesen [KZ21]. Daher gelten diese Eigenschaften auch für  $\overline{G}_{\leftrightarrow}$ .

Durch die Konstruktion der Schlösser kann ein Verbindungsstück maximal um  $\varepsilon < 1$  nach links oder rechts (bzw. oben oder unten) ein- oder ausgefahren werden. Ohne Beschränkung der Allgemeinheit wird das obere Verbindungsstück in Abbildung 4.1b sowohl nach links als auch nach rechts ausgefahren. Dadurch erhalten wir eine gesamte Längendifferenz von  $2\varepsilon$ .  $\square$

Damit besitzt  $\overline{G}_{\leftrightarrow}$  bis auf den Versatz  $2\varepsilon$  zwischen den Verbindungsstücken die gleichen Eigenschaften wie  $G_{\leftrightarrow}$ .

Zusätzlich zu den größeren Änderungen an den Schieberegeln  $G_{\leftrightarrow}$  benötigen wir noch eine weitere besondere Anpassung. Wie in Abbildung 2.5 zu sehen ist, besitzen die Einkerbungen im Rahmen für die Bolzen eine Höhe von zwei  $G_{\square}$  und der Bolzen zum Rahmen den minimalen Abstand. Dies würde bei der Außenpfad-Konstruktion aber zu insgesamt  $16 \overline{G}_{\square}$  Spielraum nach oben und unten bzw. nach links und rechts führen, wodurch sich ein Bolzen in einer falschen Kerbe verhaken könnte. Daher verkleinern wir die Einkerbungen auf eine Höhe von  $17 \overline{G}_{\square}$  und verlängern den Bolzen soweit, dass er auch nach unserer Konstruktion minimalen Abstand zum Rahmen besitzt. In Abbildung 4.2 befindet sich eine Darstellung dieser Konstruktion, wobei Abschnitte, welche nicht von dieser Anpassung betroffen sind, mit der Konstruktion nach Klawitter und Zink gezeichnet wurden.

Für alle übrigen Teilgraphen setzen wir zuerst ein Einheitsquadrat  $\overline{G}_{\square}$  in die Ecke jedes ursprünglichen Quadrats und verbinden anschließend zwei Einheitsquadrate, wenn ihre zugehörige Kante der ursprünglichen Zeichnung an der Außenfacette liegt. Die Einheitsquadrate in den Ecken benötigen wir, damit der Außenpfad nicht unterbrochen wird, wenn ein  $G_{\square}$  sowohl eine vertikale als auch eine horizontale Fortsetzung besitzt. In Abbildung 4.3 sind zwei Beispiele hierfür skizziert.

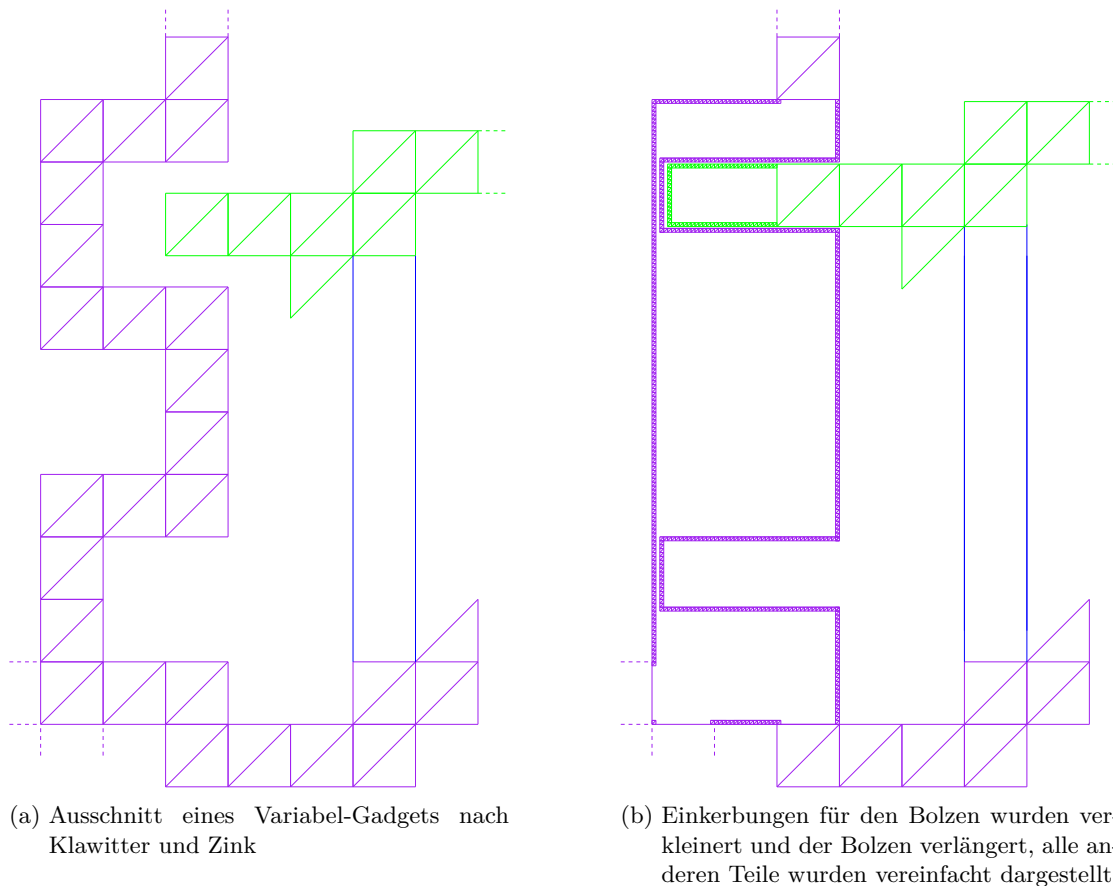
Wie in Abbildung 4.4 zu sehen ist, haben wir in der bisherigen Konstruktion noch Probleme an den Ecken, wenn ein  $\overline{G}_{\square}$  eine senkrechte Fortsetzung an der unteren und der rechten oder analog an der oberen und der linken Kante hat. Daher entfernen wir noch die entsprechenden Knoten und zugehörigen Kanten an diesen Stellen.

Durch diese Konstruktion erhalten wir bisher einen Kreis. Diesen öffnen wir an einer Stelle im Rahmen, indem wir ein  $\overline{G}_{\square}$  aus dem Rahmen wählen und die drei Kanten entfernen, welche die beiden angrenzenden Einheitsquadrate  $\overline{G}_{\square}$  miteinander verbinden. Dadurch erhalten wir nun einen Außenpfad.

**Satz 4.2.** *Das Entscheidungsproblem, ob ein aufwärtsgerichteter Außenpfad eine aufwärtsplanare Zeichnung mit drei Steigungen besitzt, ist NP-schwer.*

*Beweis.* Um die NP-Schwere zu beweisen, verweisen wir an dieser Stelle auf den Beweis von Klawitter und Zink in [KZ21]. Im Folgenden werden wir die kritischen Stellen davon betrachten und zeigen, dass die Behauptung auch mit unserer Konstruktion hält.

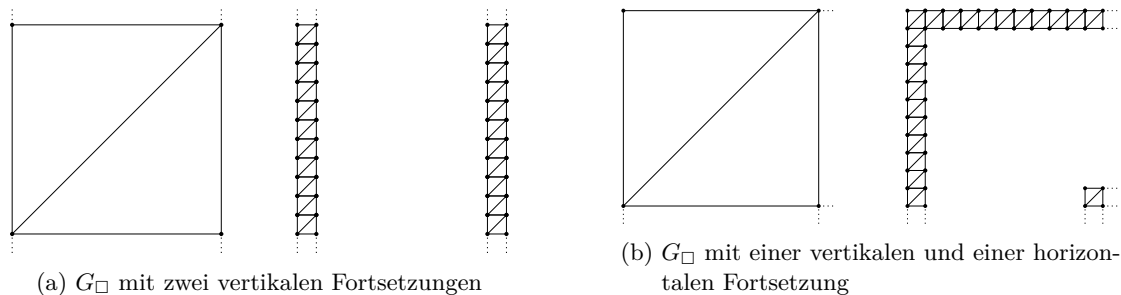
Wir haben zwei kritische Stellen im Beweis identifiziert. Zum einen liefert die ursprüngliche Konstruktion nur eine aufwärtsplanare Zeichnung mit drei Steigungen eines



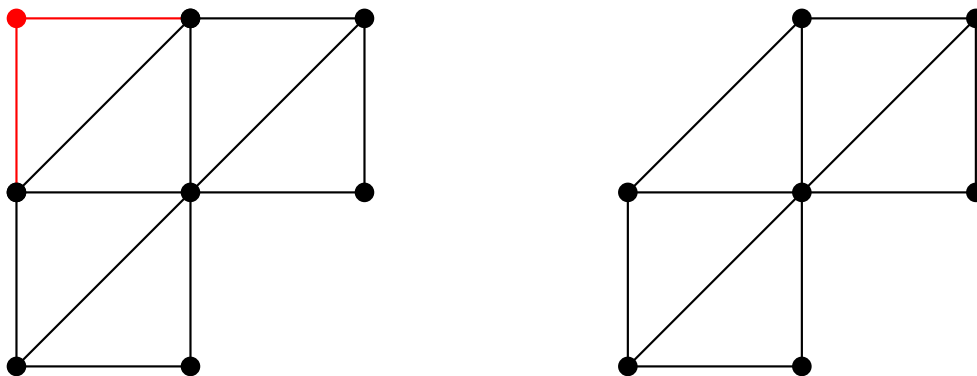
**Abb. 4.2:** Anpassungen an Rahmen und Bolzen

außenplanaren Graphen, zum anderen haben wir einen zusätzlichen vertikalen und horizontalen Versatz von  $2\varepsilon$  mit  $\varepsilon < 1$ , wodurch die Argumentation für die Konfiguration falsch-falsch-falsch (bzw. wahr-wahr-wahr) nicht mehr hält.

Klawitter und Zink haben bereits gezeigt, dass eine Zeichnung ihre Konstruktion entsprechend aufwärtsplanar ist und nur drei Steigungen verwendet. Zusätzlich ist sie auch noch außenplanar [KZ21]. Da unsere Anpassung an dieser Konstruktion nur Teile der ursprünglichen Konstruktion mit  $1/16$  der ursprünglichen Größe verwendet, bleibt eine entsprechende Zeichnung davon aufwärtsplanar und verwendet nur drei Steigungen. Da alle Facetten bis auf die beiden Facetten an der Öffnung, so konstruiert wurden, dass an jede Facette genau zwei weitere Facetten angrenzen, erhalten wir für den schwachen Dualgraphen einer solchen Zeichnung einen Pfad. Betrachtet man außerdem den Graph, den man vor der Öffnung des Kreises erhält, so liegt jede Facette, welche nicht mit der Außenfacette benachbart ist, an einer großen Innenfacette. Durch die Öffnung allerdings wird diese Innenfacette ein Teil der Außenfacette und somit grenzt jede Facette nun an die Außenfacette an. Damit ist der entstehende Graph außenplanar und somit ein Außenpfad.



**Abb. 4.3:** Beispiele für das Ersetzen von Einheitsquadraten  $G_{\square}$



**Abb. 4.4:** Beispiel für ein Problem mit Ecken in unserer Konstruktion. Dies wird durch Entfernen des roten Knotens und der roten Kanten behoben

Wie in Abbildung 2.6 zu sehen, müssen wir gewährleisten, dass bei einer falsch-falsch-falsch-Konfiguration (bzw. wahr-wahr-wahr-Konfiguration) innerhalb eines Klausel-Gadgets verhindert wird, dass der Graph ohne Überschneidung gezeichnet werden kann. Im ursprünglichen Beweis haben Klawitter und Zink argumentiert, dass eine Differenz von drei mal der Länge eines  $G_{\square}$  nötig ist, damit die horizontale Leiste trotz dieser Konfiguration gezeichnet werden kann, die Konstruktion aber maximal einen vertikalen Versatz von  $2 - \tilde{\varepsilon}$  durch Verschiebung des Rahmens und  $1 - \tilde{\varepsilon}$  durch den Spielraum eines Bolzens besitzt, wenn dieser am unteren Rand der Einkerbung platziert wird, wobei  $\tilde{\varepsilon} > 0$ . Da wir die Größe der Konstruktion nicht ändern, benötigen wir dementsprechend einen Versatz von  $3 \cdot 16 = 48$  mal der Länge unserer Einheitsquadrate  $\overline{G}_{\square}$ . Der vertikale Unterschied von  $2 - \tilde{\varepsilon}$  überträgt sich einfach und wird dementsprechend zu  $32 - \tilde{\varepsilon}$ , wodurch uns noch ein Spielraum von  $16 + \tilde{\varepsilon}$  zur Verfügung steht. Da wir allerdings durch die Schieberegler noch einen zusätzlichen Versatz von  $2\varepsilon < 2$  erhalten, müssen wir den Spielraum der Position des Bolzens eingrenzen. Durch die Anpassung an den Kerben für den Bolzen haben wir dies aber erreicht und haben hier nur noch einen Spielraum von  $1 - \tilde{\varepsilon}$ . Damit erfüllt unser vertikaler Versatz diese Vorgabe und die horizontale Leiste kann bei dieser Konfiguration nicht gezeichnet werden.  $\square$



## 4.2 Allgemeine Außenpfade bei variabler Einbettung

Bisher haben wir nur den Fall einer festen Einbettung betrachtet. Da unsere Konstruktion nur aus aneinanderhängenden Graphen vom Typ  $\overline{G}_{\square}$  und  $\overline{G}_{\leftrightarrow}$  besteht, ist eine entsprechende Zeichnung bis auf Spiegelung an der Diagonalen eindeutig. Damit hält unsere Behauptung auch im Falle einer variablen Einbettung und wir können folgendes Korollar ableiten.

**Korollar 4.3.** *Das Entscheidungsproblem, ob eine aufwärtsplanare Zeichnung mit drei Steigungen zu einem aufwärtsplanaren Außenpfad existiert, ist im Fall einer variablen Einbettung NP-schwer.*

## 5 Fazit

Das Ziel dieser Arbeit war es, die Komplexität des Erstellens von aufwärtsplanaren Zeichnungen von Außenpfaden mit drei Steigungen zu betrachten.

Als Erstes haben wir uns mit der Teilklasse der maximalen Außenpfade beschäftigt. Hierfür haben wir einen Algorithmus entworfen, welcher für einen gegebenen Graphen eine aufwärtsplanare Zeichnung liefert und dabei nur drei verschiedene Steigungen verwendet, sofern eine solche Zeichnung existiert. Darüber hinaus haben wir auch eine Erweiterung für unseren Algorithmus vorgestellt, wodurch wir das Ergebnis auch auf maximale außenplanare Graphen erweitern konnten.

Danach haben wir uns allgemeinen Außenpfaden gewidmet. Für diese Graphenklasse haben wir NP-Schwere nachgewiesen, indem wir uns auf ein Ergebnis von Klawitter und Zink [KZ21] gestützt haben. Dabei haben wir auf ihrer Konstruktion basierend einen Außenpfad konstruiert. Anschließend haben wir gezeigt, dass dieser ähnliche Eigenschaften besitzt, und die NP-Schwere-Reduktion daher auch darauf anwendbar ist.

Unsere Arbeit lässt aber noch einige Fragen offen. Wir haben zwar für drei Steigungen bei Außenpfaden im Allgemeinen NP-Schwere nachweisen und im Sonderfall von maximalen Außenpfaden sogar einen Polynomialzeit-Algorithmus angeben können. Allerdings wissen wir nicht, ob diese beiden Komplexitätsklassen auch noch bei mehr als drei Steigungen gelten. Im Falle der maximalen Außenpfade bleibt außerdem noch zu klären, ob sich der angegebene Algorithmus auch auf maximale planare Graphen anwenden lässt und welche Anpassungen dafür notwendig wären.

# Literaturverzeichnis

- [BDLM94] Paola Bertolazzi, Giuseppe Di Battista, Giuseppe Liotta und Carlo Mannino: Upward drawings of triconnected digraphs. *Algorithmica*, 12(6):476–497, 1994, 10.1007/BF01188716.
- [CFL<sup>+</sup>20] Steven Chaplick, Krzysztof Fleszar, Fabian Lipp, Alexander Ravsky, Oleg Verbitsky und Alexander Wolff: Drawing Graphs on Few Lines and Few Planes. *Journal of Computational Geometry*, 11(1):433–475, 2020, 10.20382/jocg.v11i1a17.
- [DESW07] Vida Dujmović, David Eppstein, Matthew Suderman und David R. Wood: Drawings of planar graphs with few slopes and segments. *Computational Geometry*, 38(3):194–212, 2007, 10.1016/j.comgeo.2006.09.002.
- [DGL10] Walter Didimo, Francesco Giordano und Giuseppe Liotta: Upward Spirality and Upward Planarity Testing. *SIAM Journal on Discrete Mathematics*, 23(4):1842–1899, 2010, 10.1137/070696854.
- [DK12] Mark De Berg und Amirali Khosravi: Optimal binary space partitions for segments in the plane. *International Journal of Computational Geometry & Applications*, 22(03):187–205, 2012, 10.1142/S0218195912500045.
- [DLM18] Emilio Di Giacomo, Giuseppe Liotta und Fabrizio Montecchiani: Drawing subcubic planar graphs with four slopes and optimal angular resolution. *Theoretical Computer Science*, 714:51–73, 2018, 10.1016/j.tcs.2017.12.004.
- [DT88] Giuseppe Di Battista und Roberto Tamassia: Algorithms for plane representations of acyclic digraphs. *Theoretical Computer Science*, 61(2):175–198, 1988, 10.1016/0304-3975(88)90123-5.
- [GT01] Ashim Garg und Roberto Tamassia: On the Computational Complexity of Upward and Rectilinear Planarity Testing. *SIAM Journal on Computing*, 31(2):601–625, 2001, 10.1137/S0097539794277123.
- [Hof17] Udo Hoffmann: On the Complexity of the Planar Slope Number Problem. *Journal of Graph Algorithms and Applications*, 21(2):183–193, 2017, 10.7155/jgaa.00411.
- [JJK<sup>+</sup>13] Vít Jelínek, Eva Jelínková, Jan Kratochvíl, Bernard Lidický, Marek Tešsař und Tomáš Vyskočil: The Planar Slope Number of Planar Partial 3-Trees of Bounded Degree. *Graphs and Combinatorics*, 29(4):981–1005, 2013, 10.1007/s00373-012-1157-z.

- [KM22] Jonathan Klawitter und Tamara Mchedlidze: Upward planar drawings with two slopes. *Journal of Graph Algorithms and Applications*, 26(1):171–198, 2022, 10.7155/jgaa.00587.
- [KMW14] Kolja Knauer, Piotr Micek und Bartosz Walczak: Outerplanar graph drawings with few slopes. *Computational Geometry*, 47(5):614–624, 2014, 10.1016/j.comgeo.2014.01.003.
- [KRW19] Myroslav Kryven, Alexander Ravsky und Alexander Wolff: Drawing Graphs on Few Circles and Few Spheres. *Journal of Graph Algorithms and Applications*, 23(2):371–391, 2019, 10.7155/jgaa.00495.
- [KZ21] Jonathan Klawitter und Johannes Zink: Upward Planar Drawings with Three and More Slopes. In: Helen C. Purchase und Ignaz Rutter (Herausgeber): *International Symposium on Graph Drawing and Network Visualization*, Seiten 149–165. Springer, 2021, 10.1007/978-3-030-92931-2\_11.
- [LLMN13] William Lenhart, Giuseppe Liotta, Debajyoti Mondal und Rahnuma Islam Nishat: Planar and Plane Slope Number of Partial 2-Trees. In: Stephen Wismath und Alexander Wolff (Herausgeber): *International Symposium on Graph Drawing*, Seiten 412–423. Springer, 2013, 10.1007/978-3-319-03841-4\_36.
- [MP12] Padmini Mukkamala und Dömötör Pálvölgyi: Drawing Cubic Graphs with the Four Basic Slopes. In: Marc van Kreveld und Bettina Speckmann (Herausgeber): *International Symposium on Graph Drawing*, Seiten 254–265. Springer, 2012, 10.1007/978-3-642-25878-7\_25.
- [MS09] Padmini Mukkamala und Mario Szegedy: Geometric representation of cubic graphs with four directions. *Computational Geometry*, 42(9):842–851, 2009, 10.1016/j.comgeo.2009.01.005.
- [Pap95] Achilleas Papakostas: Upward planarity testing of outerplanar dags (extended abstract). In: Roberto Tamassia und Ioannis G. Tollis (Herausgeber): *International Symposium on Graph Drawing*, Seiten 298–306. Springer, 1995, 10.1007/3-540-58950-3\_385.
- [Qua21] Valentin Andreas Quapil: *Upward and Upward-Planar Drawings with Limited Slopes*. Bachelor’s Thesis, Karlsruhe Institute of Technology, 2021. [https://i11www.itl.kit.edu/\\_media/teaching/theses/ba-quapil-21.pdf](https://i11www.itl.kit.edu/_media/teaching/theses/ba-quapil-21.pdf).
- [Sch15] André Schulz: Drawing Graphs with Few Arcs. *Journal of Graph Algorithms and Applications*, 19(1):393–412, 2015, 10.7155/jgaa.00366.
- [WC94] Greg A. Wade und Jiang-Hsing Chu: Drawability of Complete Graphs Using a Minimal Slope Set. *The Computer Journal*, 37(2):139–142, 1994, 10.1093/comjnl/37.2.139.

# Erklärung

Hiermit versichere ich die vorliegende Abschlussarbeit selbstständig verfasst zu haben, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt zu haben, und die Arbeit bisher oder gleichzeitig keiner anderen Prüfungsbehörde unter Erlangung eines akademischen Grades vorgelegt zu haben.

Würzburg, den 15. September 2022

.....  
Joshua Geis