

Bachelorarbeit

# Alternative knickminimale orthogonale Repräsentationen schnell berechnen

Tran Duy-Khang

Abgabedatum: 21. Juli 2023  
Betreuer: Prof. Dr. Alexander Wolff  
Johannes Zink, M. Sc.



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

# Zusammenfassung

Wir betrachten die Fragestellung, bei einer bereits gegebenen optimalen Lösung eine alternative Lösung zu finden. Wir beschränken uns hierbei auf das Problem, für einen planaren Graphen mit Grad höchstens 4 unter Vorgabe einer bereits optimalen Lösung eine alternative knickminimale orthogonale Repräsentation zu berechnen. Wir reduzieren hierfür das Problem auf das Finden von kürzesten Kreisen in einem äquivalenten Flussnetzwerk. Die Laufzeit des Algorithmus, der in dieser Arbeit vorgestellt wird, beträgt  $\mathcal{O}(n \log^2 n / \log \log n)$ . Somit geht es schneller eine zweite optimale Lösung zu finden als die erste, deren Berechnung nach Stand der Forschung  $\mathcal{O}(n^{3/2})$  Zeit benötigt [CK12].

Außerdem beschäftigen wir uns mit dem Problem, alle mögliche Lösungen mit polynomieller Verzögerung auszugeben, und erstellen hierfür einen Aufzählungsalgorithmus, der jede Lösung genau einmal aufzählt.

## Abstract

We analyze the question of finding an alternative solution for a problem for which an optimal solution is already given. We limit the scope of the question to the problem of computing an alternative orthogonal representation with a minimum number of bends for a planar graph with a degree of at most 4, if we are already given an optimal solution. We reduce the problem of finding an alternative optimal solution to that of finding shortest cycles in an equivalent flow network. The running time of our algorithm is  $\mathcal{O}(n \log^2 n / \log \log n)$ . It is therefore faster to find a second optimal solution than the first, which, according to the current state of research, requires  $\mathcal{O}(n^{3/2})$  time to be calculated [CK12].

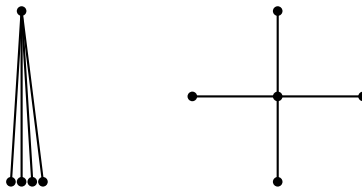
We also consider the problem of enumerating all possible solutions. We present an enumeration algorithm that has polynomial delay and goes through each solution exactly once.

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>4</b>
1.1	Verwandte Arbeiten . . . . .	4
1.2	Eigener Beitrag . . . . .	5
<b>2</b>	<b>Grundlagen</b>	<b>7</b>
2.1	Grundlagen zu knickminimalen orthogonalen Repräsentationen . . . . .	7
2.2	Flussnetzwerke zum Finden einer knickminimalen orthogonalen Repräsentation . . . . .	8
<b>3</b>	<b>Knickminimale orthogonale Repräsentationen von Graphen mit nur kleinen Innenfacetten</b>	<b>13</b>
3.1	Reduziertes Flussnetzwerk für Graphen mit nur kleinen Innenfacetten . . . . .	13
3.2	Aufzählungsalgorithmus für alle knickminimalen orthogonalen Repräsentationen . . . . .	14
<b>4</b>	<b>Berechnung von einer Alternativlösung in beliebigen Graphen</b>	<b>19</b>
4.1	Erweiterung des Flussnetzwerks durch eine vorgegebene orthogonale Repräsentation . . . . .	19
4.2	Finden alternativer Flüsse mithilfe von Kreismengen im erweiterten Flussgraph . . . . .	20
4.3	Anzahl an Knicken in alternativen Repräsentationen . . . . .	22
4.4	Alternativlösungen bei Kantenrestriktionen . . . . .	24
4.5	Alternativlösungen ohne Kantenrestriktion . . . . .	26
<b>5</b>	<b>Aufzählung aller Lösungen</b>	<b>28</b>
5.1	Funktionsweise des Aufzählungsalgorithmus . . . . .	28
5.2	Korrektheit und Laufzeit des Aufzählungsalgorithmus . . . . .	29
<b>6</b>	<b>Fazit und offene Probleme</b>	<b>31</b>
	<b>Literaturverzeichnis</b>	<b>32</b>

# 1 Einleitung

Heutzutage werden viele verschiedene Strukturen mithilfe von Graphen modelliert, dazu gehören beispielsweise Organisationsdiagramme, Schaltpläne und Concept-Maps. Diese Graphen lassen sich dann durch Zeichnungen darstellen. Ziel der Zeichnungen ist es häufig, die Graphen zu veranschaulichen. Somit liegt ein besonderer Fokus darauf, die Zeichnungen übersichtlich zu gestalten und das Lesen der Zeichnungen zu vereinfachen. Um die Übersichtlichkeit und Lesbarkeit eines Graphen zu verbessern, eignen sich vor allem gerade Liniensegmente und große Winkel. Letzteres rührt daher, dass individuelle Kanten dann einfacher zu unterscheiden sind (siehe Abbildung 1.1). In orthogonalen Zeichnungen, welche alle Kanten in einem orthogonalen Gitter anordnen, wird dies besonders deutlich, da in solchen Zeichnungen alle Kanten nur im rechten Winkel, welches der größtmögliche Winkel für Graphen mit Grad höchstens 4 ist, zueinander angeordnet werden. Wir schränken für orthogonale Zeichnungen den Grad der Knoten ein, da in orthogonalen Zeichnungen in einem Punkt aufgrund der Orthogonalität höchstens 4 Kanten anliegen können.

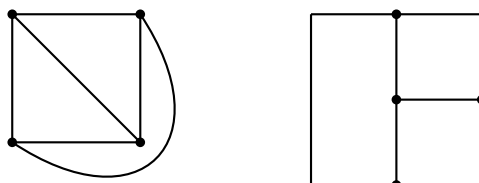


**Abb. 1.1:** Größere Winkel erleichtern es dem Betrachtern einzelne Kanten zu folgen.

In orthogonalen Zeichnungen ergibt sich jedoch das Problem, dass sich Knicke im Allgemeinen nicht vermeiden lassen (siehe Abbildung 1.2). Ähnlich wie das Verwenden von großen Winkel verbessert auch eine Knickminimierung normalerweise die Lesbarkeit eines Graphen und hat zudem auch eine ästhetische Bedeutung. Die Knickminimierung einer orthogonalen Zeichnung ist eine Anforderung, aus der die in dieser Arbeit behandelte Fragestellung resultiert. Wir werden uns damit beschäftigen, aus bereits knickminimalen orthogonalen Zeichnungen beziehungsweise ihren kombinatorischen Beschreibungen weitere knickminimale Repräsentationen zu finden, wobei der Fokus besonders darauf liegt, die Laufzeit im Vergleich zum Finden einer Erstlösung zu verringern.

## 1.1 Verwandte Arbeiten

Als Grundlage für das Finden von knickminimalen Zeichnungen dienen zu orthogonalen Einbettungen äquivalente Flussnetzwerke, wie sie Tamassia im Jahr 1987 in seiner Ar-



**Abb. 1.2:** Es gibt keine knickfreie orthogonale Zeichnung des  $K_4$ .

beit [Tam87] beschrieben hat. Cornelsen und Karrenbauer [CK12] veröffentlichten 2011 einen Algorithmus, der eine knickminimale orthogonale Repräsentation in  $\mathcal{O}(n^{3/2})$  findet, wobei  $n$  die Anzahl an Knoten beschreibt. Diese asymptotische Laufzeitschranke wollen wir bei der Suche nach einer Alternativlösung unterbieten. Der Algorithmus basiert dabei auf der Äquivalenz von Flüssen mit minimalen Kosten und der minimalen Anzahl von Knicken für ein zur orthogonalen Einbettung äquivalentes Flussnetzwerk. Die Suche nach einem Fluss mit minimalen Kosten in einem planaren Graphen erfolgt dann in der gegebenen Laufzeit von  $\mathcal{O}(n^{3/2})$ .

Als Inspiration für diese Fragestellung dient ein Artikel von Deligkas et al. [DMSZ20], in dem unter Angabe eines Hamiltonkreises in einem kubischen hamiltonschen Graphen ein zweiter Hamiltonkreis berechnet werden kann, wobei die Laufzeit im Vergleich zum Finden des ersten Hamiltonkreises leicht verbessert wird.

In dieser Arbeit werden, wie im Algorithmus von Tamassia, Flussnetzwerke verwendet, um die Alternativlösungen zu finden. Wir benutzen dafür die Eigenschaft, dass jedes dieser speziellen Flussnetzwerke ein planarer Graph ist, sodass wir schnelle Algorithmen zum Finden von kürzesten Pfaden [MWN10] und Kreisen [LL17] anwenden können.

## 1.2 Eigener Beitrag

Generell geht es um die Fragestellung, mithilfe einer bereits optimalen oder ausreichend guten Lösung eines Problems eine weitere nicht allzu ähnliche, aber dennoch genauso gute Lösung zu berechnen. Entscheidend ist hierbei, wie sehr einem eine Erstlösung für das Berechnen einer Alternativlösung hilft.

Wir werden in Kapitel 3 knickminimale orthogonale Repräsentationen von Graphen mit Innenfacetten von Grad höchstens 4 betrachten. Eine erste Erkenntnis ist hierbei, dass das Finden von Lösungen in solchen Graphen besonders einfach ist, und dass wir sogar einen Aufzählungsalgorithmus erstellen können, der uns ohne doppeltes Aufzählen jede Lösung mit linearer Verzögerung aufzählen kann.

Ein weiterer Beitrag ist in Kapitel 4 dann die allgemeine Berechnung von Alternativlösungen in beliebigen 4-planaren Graphen, also planare Graphen mit gegebener Einbettung, in denen jeder Knoten einen Grad von höchstens 4 hat. Wir zeigen hier, dass eine Lösung sowohl unter einer bestimmten Einschränkung als auch ohne Einschränkung schneller, nämlich in  $\mathcal{O}(n \log^2 n / \log \log n)$  Zeit, berechnet werden kann, wobei  $n$  die Anzahl der Knoten ist.

Im letzten Kapitel werden wir auch für beliebige 4-planare Graphen einen Aufzäh-

lungsalgorithmus vorstellen, der mit einer Verzögerung von  $\mathcal{O}(n^2 \log^2 n / \log \log n)$  jede Lösung aufzählen kann.

## 2 Grundlagen

Wir betrachten in diesem Kapitel die Grundlagen der Arbeit, die auf der Arbeit von Tamassia [Tam87] basiert. Die Notation ist an Di Battista et al. [BETT94] angelehnt. Dafür werden zunächst orthogonale Repräsentationen erläutert und danach Flussnetzwerke zum Finden von knickminimalen Lösungen eingeführt.

### 2.1 Grundlagen zu knickminimalen orthogonalen Repräsentationen

Eine orthogonale Repräsentation ist eine kombinatorische Beschreibung einer orthogonalen Zeichnung und ist dadurch gekennzeichnet, dass alle Kanten horizontal oder vertikal verlaufen. Wichtig für diese Arbeit ist hierbei, dass eine Kante auch Knicke besitzen darf, das heißt, dass sie nicht nur aus einer einzigen geraden Strecke bestehen muss. Weiterhin dürfen zwei paarweise verschiedene Kanten nicht übereinanderliegen und sich auch nicht schneiden. Sie müssen also bis auf Startpunkt und Endpunkt disjunkt sein, was zur Folge hat, dass jeder Knoten einen Grad von maximal vier besitzt. Im Gegensatz zu orthogonalen Zeichnungen wird den Kanten in orthogonalen Repräsentationen jedoch keine Länge zugeordnet. Dies führt uns zu folgenden Definitionen für 4-planare Graphen, also planare Graphen, in denen jeder Knoten einen Grad von höchstens 4 hat.

**Definition 2.1** (Kantenbeschreibung, Repräsentation einer Facette und orthogonale Repräsentation). *Sei  $G = (V, E)$  ein 4-planarer Graph mit Facettenmenge  $F$  und äußerer Facette  $f_0$ .*

- *Sei  $e$  eine Kante mit Facette  $f$  zu ihrer Rechten. Eine Kantenbeschreibung von  $e$  in Bezug auf  $f$  ist dann ein Tripel  $(e, \delta, \alpha)$ , in dem
  - $\delta$  eine beliebig lange Folge von Nullen und Einsen ist, also eine Folge  $\{0, 1\}^*$ , wobei 0 für einen rechten Knick und 1 für einen linken Knick steht,
  - und  $\alpha$  den Winkel zwischen  $e$  und der nächsten Kante  $e'$  beschreibt, wobei  $\alpha$  aufgrund der Orthogonalität nur einen Wert in  $\{\pi/2, \pi, 3\pi/2, 2\pi\}$  annehmen kann.*
- *Eine Repräsentation einer Facette  $H(f)$  von  $f$  ist eine im Uhrzeigersinn geordnete Folge an Kantenbeschreibungen  $(e, \delta, \alpha)$  für jede Kante  $e$  von  $f$ .*
- *Eine orthogonale Repräsentation  $H(G)$  von  $G$  ist definiert als*

$$H(G) = \{H(f) \mid f \in F\}.$$

Damit eine orthogonale Repräsentation valide ist, müssen folgende Bedingungen gelten:

- $H(G)$  beschreibt alle Kanten und Facetten.
- Für eine Kante  $\{u, v\}$ , die zwischen zwei Facetten  $f$  und  $g$  mit  $((u, v), \delta_1, \alpha_1) \in H(f)$  und  $((v, u), \delta_2, \alpha_2) \in H(g)$  verläuft, ist die Folge  $\delta_1$  die umgekehrte und invertierte Sequenz von  $\delta_2$ .
- Seien  $|\delta|_0$  und  $|\delta|_1$  die Anzahl an Nullen und Einsen in  $\delta$  und sei  $r = (e, \delta, \alpha)$  eine Kantenbeschreibung. Dann gilt für  $C(r) := |\delta|_0 - |\delta|_1 + 2 - 2\alpha/\pi$ , dass:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{wenn } f = f_0 \\ +4 & \text{sonst.} \end{cases}$$

- Für einen Knoten  $v$  gilt, dass für die Menge an Kantenbeschreibungen  $\{(e, \delta, \alpha) \mid e = (v', v) \in E\}$  die Summe ihrer Winkel  $\alpha$  gleich  $2\pi$  ist.

## 2.2 Flussnetzwerke zum Finden einer knickminimalen orthogonalen Repräsentation

Um knickminimale orthogonale Repräsentationen zu finden, eignet sich die Verwendung von Flussnetzwerken. Im Gegensatz zu s-t-Flussnetzwerken, welche eine feste Senke und Quelle besitzen und bei welchen das Ziel ist, den Wert des Flusses zu maximieren, kann in Flussnetzwerken im allgemeineren Fall jeder Knoten einen Bedarf besitzen, der ihn als Senke oder Quelle beschreibt. Ziel bei diesen Flussnetzwerken ist auch nicht die Flussmaximierung, sondern die Kostenminimierung. In unserem Fall beschreibt der Fluss dann den Winkel, den ein Knoten an eine Facette abgibt, oder einen Knick. Ein allgemeines Flussnetzwerk wird wie folgt definiert:

**Definition 2.2** (Flussnetzwerk). *Ein Flussnetzwerk ist ein Quadrupel  $D = ((V, A); l; u; b)$  mit*

- einem gerichteten Graphen  $D = (V, A)$  mit Knotenmenge  $V$  und Kantenmenge  $A$ ,
- einer unteren Grenze für den Fluss  $l: A \rightarrow \mathbb{R}_0^+$  für jede Kante,
- eine Kantenkapazität  $u: A \rightarrow \mathbb{R}_0^+$  für jede Kante
- und einem Bedarf  $b: V \rightarrow \mathbb{R}$  mit  $\sum_{i \in V} b(i) = 0$ .

Wir definieren nun auch, wann ein valider Fluss vorliegt:

**Definition 2.3** (valider Fluss). *Eine Funktion  $X: A \rightarrow \mathbb{R}_0^+$  wird ein valider Fluss genannt, wenn*

$$\begin{aligned} l(i, j) &\leq X(i, j) \leq u(i, j) && \forall (i, j) \in A \\ \sum_{(i, j) \in A} X(i, j) - \sum_{(j, i) \in A} X(j, i) &= b(i) && \forall i \in V \end{aligned}$$



Im Allgemeinen existieren unterschiedliche Funktionen  $X$ , die einen validen Fluss darstellen. Wichtig ist, dass der Gesamtfluss an jedem Knoten  $v$  aufgrund von  $b(v)$  erhalten bleibt. Falls also die unteren Grenzen und die Kapazitäten eingehalten werden, müssen wir einen weiteren validen Fluss nur auf die Einhaltung des Gesamtflusses an jedem Knoten testen, was wir im weiteren Verlauf der Arbeit mehrfach benutzen werden. Da ein solches Flussnetzwerk keine Aussage über die Bewertung eines Flusses macht, wird zusätzlich eine Kostenfunktion  $\text{cost}: A \rightarrow \mathbb{R}_0^+$  mit  $\text{cost}(x) := \sum_{(i,j) \in A} \text{cost}(i,j) \cdot X(i,j)$  hinzugefügt. Ziel ist es nun, einen validen Fluss  $X: A \rightarrow \mathbb{R}_0^+$  zu finden, der zudem die Kostenfunktion minimiert.

## Aufbau des Flussnetzwerks

Wir definieren nun zu einem Graphen  $G = (V, E)$  das dazugehörige Flussnetzwerk  $N(G) = ((V \cup F, A); l; u; b; \text{cost})$ :

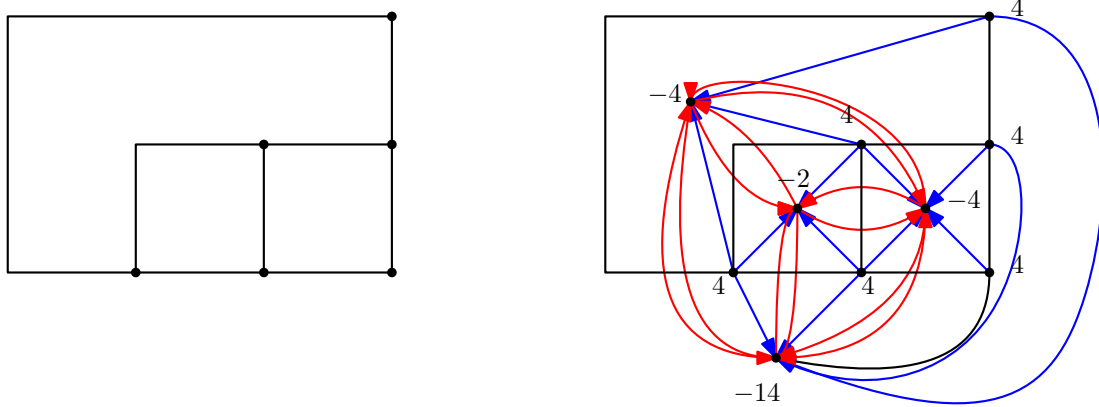
**Definition 2.4** (Flussnetzwerk). *Sei  $G = (V, E)$  ein eingebetteter 4-planarer Graph mit Facettenmenge  $F$  und äußerer Facette  $f_0$ , dann erfüllt das dazugehörige Flussnetzwerk  $N(G) = ((V \cup F, A); l; u; b; \text{cost})$  folgende Eigenschaften:*

- $A = \{(v, f)_{ee'} \in V \times F \mid v \text{ zwischen Kanten } e, e' \text{ von } f\} \cup \{(f, g)_e \in F \times F \mid f, g \text{ haben eine gemeinsame Kante } e\}$
- $b(v) = 4 \quad \forall v \in V$
- $b(f) = -2 \deg_G(f) + \begin{cases} -4 & \text{falls } f = f_0 \\ +4 & \text{sonst} \end{cases}$
- $\forall (v, f) \in A, v \in V, f \in F: l(v, f) := 1, u(v, f) := 4$
- $\forall (v, f) \in A, v \in V, f \in F: \text{cost}(v, f) = 0$
- $\forall (f, g) \in A, f, g \in F: l(f, g) := 0, u(f, g) := \infty$
- $\forall (f, g) \in A, f, g \in F: \text{cost}(f, g) = 1$

Wir werden in den folgenden Kapiteln hauptsächlich mit modifizierten Versionen dieses Flussnetzwerks arbeiten, da die Kantenmenge des Flussnetzwerks schnell unübersichtlich wird, was in Abbildung 2.1 verdeutlicht wird. Jedoch ist dieses Flussnetzwerk Grundlage für das Verständnis des reduzierten und erweiterten Flussnetzwerks, welche im weiteren Verlauf eingeführt werden.

## Aufbau des reduzierten Flussnetzwerk

Für das Arbeiten mit eingebetteten Graphen mit Innenfacetten, die einen maximalen Grad von vier besitzen, eignet sich ein reduziertes Flussnetzwerk, welches wir wie folgt definieren.



**Abb. 2.1:** Das Flussnetzwerk des Graphen besitzt eine unübersichtliche Kantenmenge.  
Links: Originalgraph; Rechts: Flussnetzwerk mit Bedarf und Kanten mit Kosten 1 (rot) und kostenfreien Kanten (blau)

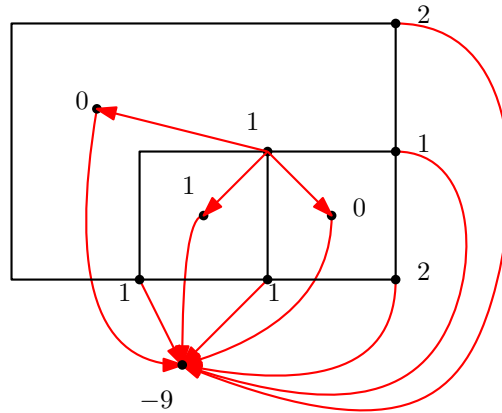
**Definition 2.5** (Reduziertes Flussnetzwerk). Sei  $G = (V, E)$  ein eingebetteter 4-planarer Graph, dann erfüllt das dazugehörige reduzierte Flussnetzwerk  $N^-(G) = ((V \cup F, A); l; u; b; \text{cost})$  folgende Eigenschaften:

- $A = \{e \in E(N(G)) \mid e \text{ ist Teil eines kürzesten Weges zu einem Knoten mit negativem Bedarf}\}$
- $b(v) = 4 - \deg_G(v) \quad \forall v \in V$
- $b(f) = -\deg_G(f) + \begin{cases} -4 & \text{falls } f = f_0 \\ +4 & \text{sonst} \end{cases}$
- $\forall (v, f) \in A, v \in V, f \in F: l(v, f) := 0, u(v, f) := 3$
- $\forall (v, f) \in A, v \in V, f \in F: \text{cost}(v, f) = 0$
- $\forall (f, g) \in A, f, g \in F: l(f, g) := 0, u(f, g) := \infty$
- $\forall (f, g) \in A, f, g \in F: \text{cost}(f, g) = 1$

Die Kantenmenge  $A$  kann beispielsweise mithilfe von Breitensuche berechnet werden. Die Breitensuche wird dann in inverser Kantenrichtung für die Außenfacette  $f_0$  ausgeführt und zusätzlich für jede Facette  $f$  mit  $b(f) < 0$ , also alle Facetten mit einem Grad größer als 4. Kanten, die Teil eines kürzesten Weges zu  $f_0$  oder  $f$  sind, werden dann zu  $A$  hinzugefügt. Die Knoten und Facetten mit positivem Bedarf und Facetten, die mehrere kürzeste Wege zu  $f_0$  besitzen, werden dann in der Reihenfolge nummeriert, wie sie in der Breitensuche für  $f_0$  besucht werden.

Das reduzierte Flussnetzwerk besitzt aufgrund der Tatsache, dass nur bestimmte Kanten hinzugefügt werden, eine niedrigere Kantendichte als das normale Flussnetzwerk (siehe Abbildung 2.2). Jedoch ist die Breitensuche für jeden negativen Bedarf nur dann sinnvoll, wenn wenige Facetten mit negativem Bedarf im Flussnetzwerk enthalten sind,

da diese Facetten die Senken beschreiben. Wir benutzen in dieser Arbeit das reduzierte Flussnetzwerk daher nur für Graphen mit einem Knoten, der einen negativen Bedarf hat, also einer Senke.



**Abb. 2.2:** Die Kantenmenge des reduzierten Flussnetzwerks beinhaltet nur relevante Kanten (rot) und angepasste Bedarfe.

Facetten mit mehreren kürzesten Wegen zur Außenfacette, die im reduzierten Flussnetzwerk mindestens eine eingehende Kante besitzen, bezeichnen wir von nun an als innerer Knoten mit Verzweigung.

Es bleibt noch zu zeigen, dass im reduzierten Flussnetzwerk äquivalent zum normalen Flussnetzwerk dieselben knickminimalen orthogonalen Repräsentationen gefunden werden können.

**Lemma 2.6.** *Eine Knickfolge von einer Facette  $f$  zu einer anderen Facette  $f'$  ist genau dann minimal, wenn sie im Flussnetzwerk einem kürzesten Weg von  $f$  zu  $f'$  entspricht.*

*Beweis.* Die Aussage folgt direkt aus der Äquivalenz von Kosten und Knicken und aus der Äquivalenz eines validen Flusses zu einer orthogonalen Repräsentation.  $\square$

Dieses Lemma erlaubt uns nun die beiden Flussnetzwerke für knickminimale orthogonale Repräsentationen gleichzusetzen.

**Korollar 2.7.** *Jeder valide Fluss mit minimalen Kosten im Flussnetzwerk ist äquivalent zu einem validen Fluss mit minimalen Kosten im reduzierten Flussnetzwerk.*

*Beweis.* Wir zeigen zunächst, dass die Bedarfsänderung keine Auswirkung auf den Fluss hat. Der Bedarf von jedem Knoten  $v$  in  $G$  ist um den Grad des Knotens reduziert, dies entspricht gleichzeitig dem Fluss, welcher im normalen Flussnetzwerk aufgrund von  $l(v, f) \geq 1$  mindestens an jede anliegende Facette abgegeben werden muss. Da der minimal nötige Fluss eines Knoten im reduzierten Flussnetzwerk dementsprechend angepasst wurde und auch der Bedarf für jede Facette für jeden erzwungenen Fluss verringert wurde, behandelt das reduzierte Flussnetzwerk quasi nur den frei wählbaren Fluss im Flussnetzwerk.

Ein valider Fluss im Flussnetzwerk mit minimalen Kosten ist äquivalent zu einer knickminimalen orthogonalen Repräsentation. Nach Lemma 2.6 sucht sich ein Bedarf damit den kürzesten Weg zu einer Senke. Da im reduzierten Flussnetzwerk genau die kürzesten Wege verzeichnet sind, ist der Fluss - ausgenommen der fixen Flüsse von Knoten zu Facette – auch ein valider Fluss im reduzierten Flussnetzwerk.  $\square$

# 3 Knickminimale orthogonale Repräsentationen von Graphen mit nur kleinen Innenfacetten

Wir werden uns in diesem Kapitel mit Graphen beschäftigen, die nur Innenfacetten mit maximal vier Kanten besitzen, wobei wir diese Facetten als *kleine Facetten* bezeichnen. Diese Eigenschaft erlaubt es uns, das Finden von knickminimalen orthogonalen Repräsentation in solchen Graphen zu vereinfachen. Zu Beginn werden wir untersuchen, wie sich das Fehlen von großen Innenfacetten im reduzierten Flussnetzwerk auswirkt, und wie wir die resultierenden Besonderheiten nutzen können, um knickminimale orthogonale Repräsentationen zu finden. Daraufhin erstellen wir einen Aufzählungsalgorithmus, der jede knickminimale orthogonale Repräsentation genau einmal aufzählt und bei dem die nächste Lösung jeweils in linearer Laufzeit gefunden wird.

## 3.1 Reduziertes Flussnetzwerk für Graphen mit nur kleinen Innenfacetten

Wir betrachten nun die Auswirkungen auf das reduzierte Flussnetzwerk, die das Fehlen von großen Innenfacetten im Graphen hat. Wir benutzen dafür die Eigenschaft, dass der Bedarf einer Facette mit ihrem Grad immer niedriger wird. Bei kleinen Facetten ist der Bedarf im reduzierten Flussnetzwerk somit besonders hoch.

**Lemma 3.1.** *Sei  $G = (V, E)$  ein 4-planarer Graph mit nur kleinen Innenfacetten  $F$  und einer Außenfacette  $f_0$ , dann existiert im reduzierten Flussnetzwerk  $N^-(G)$  genau ein negativer Bedarf, und zwar an der Außenfacette.*

*Beweis.* Die Aussage folgt direkt aus dem Aufbau des reduzierten Flussnetzwerks. Jede Innenfacette  $f$  hat einen Bedarf von  $4 - \deg(f)$ , wobei  $\deg(f)$  die Anzahl der Knoten am Rand der Facette ist. Da für kleine Facetten per Definition  $\deg(f) \leq 4$  gilt, folgt somit, dass der Bedarf mindestens 0 ist. Zudem hat jeder Knoten nach der Bedarfszuweisung einen Bedarf von 0 bis 4 und ist somit immer positiv. Für die Außenfacette beträgt der Bedarf hingegen  $-4 - \deg(f)$  und kann somit negativ sein. Da alle anderen Bedarfe positiv sind, besitzt das reduzierte Flussnetzwerk also genau einen negativen Bedarf an der Außenfacette.  $\square$

Das reduzierte Flussnetzwerk besitzt für Graphen mit nur kleinen Facetten somit nur eine Senke, in die ein Fluss münden kann. Da beim Aufbau des reduzierten Flussnetzwerks nur Kanten hinzugefügt werden, die Teil eines kürzesten Weges zu einer Senke

sind, sind alle Kanten Teil eines kürzesten Weges zur selben Facette, nämlich der Außenfacette. Das reduzierte Flussnetzwerk vereinfacht sich somit bei einer einzelnen Senke auf einen azyklischen gerichteten Multigraphen, wobei die Außenfacette von jedem anderen Knoten erreichbar ist. Dies ist für das reduzierte Flussnetzwerk von Graphen im Allgemeinen aufgrund der Existenz von unterschiedlichen Senken nicht der Fall und erlaubt uns, für Graphen mit nur kleinen Facetten das reduzierte Flussnetzwerk effizient für das Finden von knickminimalen orthogonalen Repräsentationen zu verwenden.

Wir wollen nun die Eigenschaft aus Lemma 3.1 verwenden, um knickminimale orthogonale Repräsentation einfach zu charakterisieren. Damit ein Fluss gültig ist, muss jeder positive Bedarf einen Weg zu einem negativen Bedarf finden. Da im reduzierten Flussnetzwerk für Graphen mit nur kleinen Innenfacetten genau *ein* negativer Bedarf existiert, nämlich an der Außenfacette, muss daher jeder Fluss auch in der Außenfacette münden. Aufgrund des Aufbaus des reduzierten Flussnetzwerks sind die Wege zur Senke dann auch die kürzesten Wege zur Außenfacette. Mit Lemma 2.6 lassen sich knickminimale orthogonale Repräsentationen somit wie folgt durch kürzeste Wege darstellen.

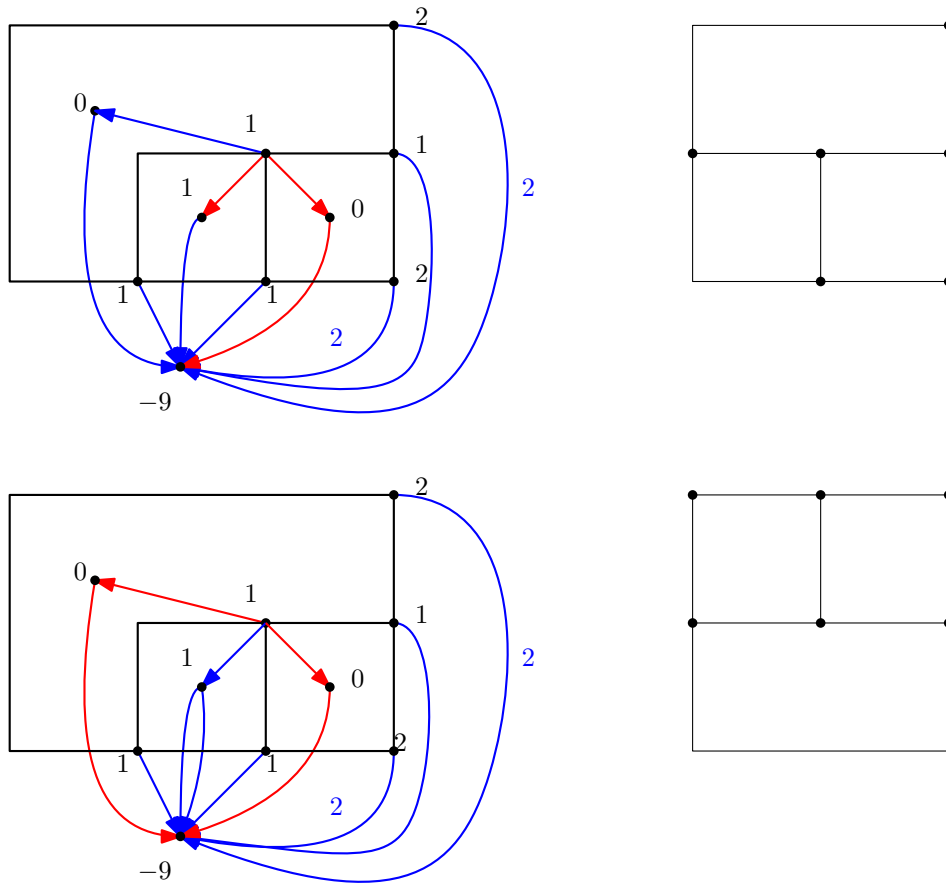
**Lemma 3.2.** *Sei  $G = (V, E)$  ein 4-planarer Graph mit nur kleinen Innenfacetten. Im reduzierten Flussnetzwerk ist dann jede Kombination von kürzesten Wegen von den positiven Bedarfen zur Außenfacette äquivalent zu einer knickminimalen orthogonalen Repräsentation. Dabei müssen von jedem Knoten aus so viele Wege gewählt werden, wie der Bedarf angibt.*

*Beweis.* Die Aussage folgt direkt aus Lemma 2.6 und Lemma 3.1. □

Dieses Lemma erlaubt es uns, das Finden von knickminimalen orthogonalen Repräsentationen auf das Finden von Kombinationen aus kürzesten Wegen zu reduzieren. Da unser reduziertes Flussnetzwerk nur die kürzesten Wege zur Außenfacette enthält, ist jede beliebige Kombination aus kürzesten Wegen äquivalent zu einer knickminimalen orthogonalen Repräsentation (siehe Abbildung 3.1). Alternativlösungen können somit schnell ermittelt werden.

## 3.2 Aufzählungsalgorithmus für alle knickminimalen orthogonalen Repräsentationen

Wir werden in den folgenden Abschnitten einen Aufzählungsalgorithmus erstellen, welcher mit polynomialer Verzögerung eine weitere Lösung findet, die zuvor noch nicht ausgegeben wurde. Nach Lemma 3.2 ist zwar jede Kombination aus kürzesten Wegen äquivalent zu einer knickminimalen orthogonalen Repräsentation, jedoch müssen diese nicht zwangsweise verschieden sein. Wir wollen uns deshalb mit dem Problem beschäftigen, alle Lösungen aufzuzählen, ohne redundante Lösungen zu erhalten, woraufhin wir die Korrektheit und Laufzeit des Aufzählungsalgorithmus zeigen.

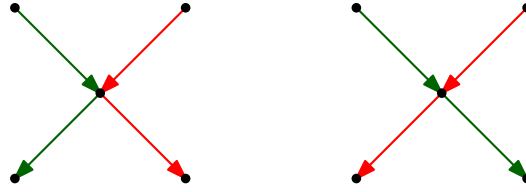


**Abb. 3.1:** Beliebige Kombinationen von kürzesten Wegen (blau) im reduzierten Flussnetzwerk führen zu knickminimalen Repräsentationen.

### Redundante Lösungen

Um den Aufzählungsalgorithmus aufzustellen, überlegen wir uns zuerst, unter welchen Bedingungen sich Lösungen unterscheiden. Nach Lemma 2.7 muss sich der Fluss im reduzierten Flussnetzwerk unterscheiden, damit auch der entsprechende Fluss im normalen Flussnetzwerk sich unterscheidet. Da jeder Fluss im Flussnetzwerk genau zu einer orthogonalen Repräsentation äquivalent ist [Tam87], sind zwei orthogonale Repräsentationen genau dann unterscheidbar, wenn sich der Fluss im reduzierten Flussnetzwerk unterscheidet. Somit muss unser Aufzählungsalgorithmus alle möglichen Flüsse im reduzierten Flussnetzwerk genau einmal aufzählen.

Da jede knickminimale orthogonale Repräsentation als Kombination von kürzesten Wegen dargestellt werden kann, zählt ein Aufzählungsalgorithmus, der jede Wegkombination genau einmal ausgibt, auch alle Lösungen auf. Jedoch ist es dann möglich, redundante knickminimale orthogonale Repräsentationen zu erhalten, wie man in Abbildung 3.2 sehen kann. Wir wollen also untersuchen, unter welchen Bedingungen redundante Lösungen möglich sind. Wir zeigen für den Beweis davor noch folgenden Hilfssatz.



**Abb. 3.2:** Zwei unterschiedliche Wegkombinationen mit gleicher orthogonalen Repräsentation.

**Lemma 3.3.** *Redundante Lösungen für Wegkombinationen sind nur möglich, falls Wege existieren, die mindestens einen inneren Knoten mit Verzweigung außerhalb des Startknotens und Endknotens enthalten. Die Wegkombinationen der Lösungen unterscheiden sich nur bei Wegen, die über mindestens einen inneren Knoten mit Verzweigung verlaufen.*

*Beweis.* Angenommen, es gäbe zwei redundante Lösungen äquivalent zu verschiedenen Wegkombinationen  $(W_1, \dots, W_m)$  und  $(W'_1, \dots, W'_m)$ , deren Wegkombinationen sich bei einem Weg  $W'_a \in (W'_1, \dots, W'_m)$  unterscheiden, der nicht über eine Verzweigung außerhalb des Startknotens und Endknotens verläuft. Da sich der Fluss bei redundanten Lösungen nicht geändert hat, sich jedoch die Wegkombination geändert hat, können sich keine Wege geändert haben, die nur über Knoten ohne Verzweigungen verlaufen. Somit müssen es Wege sein, die durch einen Knoten mit Verzweigung verlaufen. Da  $W'_a$  nach Annahme keinen inneren Knoten mit Verzweigung besitzt, der nicht der Startknoten oder Endknoten ist, müssen diese Knoten die Anfangsknoten von  $W'_a$  sein. Da die Wege somit disjunkt sind oder einander enthalten und der Fluss sich in redundanten Lösungen nicht unterscheidet, folgt daraus, dass  $W'_a$  in  $(W_1, \dots, W_m)$  enthalten ist. Dies ist ein Widerspruch zur Annahme, dass sich die Wegkombinationen bei  $W'_a$  unterscheiden. Also folgt die Aussage.  $\square$

Redundante Lösungen sind also nur in Verbindung innerer Knoten mit Verzweigungen möglich. Unser Aufzählungsalgorithmus muss daher gesondert bei Innenfacetten, die Verzweigungen besitzen, redundante Lösungen vermeiden können.

### Funktionsweise des Aufzählungsalgorithmus

Wir erstellen nun mithilfe der vorangegangenen Ergebnisse einen Aufzählungsalgorithmus, der jede Lösung genau einmal aufzählt. Sei  $G = (V, E)$  ein 4-planarer Graph und  $X$  eine vorgegebene Lösung.

Im ersten Schritt erstellen wir mithilfe der vorgegebenen Lösung das reduzierte Flussnetzwerk  $N^-(G)$ . Da die Kantenanzahl linear zur Knotenanzahl ist, erfolgt dies in linearer Laufzeit.

Im zweiten Schritt wird jeder innere Knoten mit Verzweigung und jeder Knoten mit positiven Bedarf im reduzierten Flussnetzwerk nummeriert. Dabei ist die Nummerierung wie beim Aufbau des reduzierten Flussnetzwerks gewählt.



Im dritten Schritt berechnet der Aufzählungsalgorithmus zunächst für jeden inneren Knoten, der im reduzierten Flussnetzwerk eine Verzweigung besitzt, und für jeden Knoten mit positivem Bedarf eine geordnete Liste an kürzesten Wegen zu erreichbaren inneren Knoten mit Verzweigungen und zur Außenfacette, falls diese direkt erreichbar sind.

Im vierten Schritt berechnen wir die erste Lösung, die vom Aufzählungsalgorithmus ausgegeben wird, indem wir zunächst von jedem nicht nummerierten Knoten für den Bedarf den kürzesten Weg wählen. Danach wählen wir vom höchsten nummerierten Knoten aus für jeden Knoten den ersten Weg in seiner zugehörigen Wegliste für seinen kompletten Bedarf.

Sei nun also bereits eine Lösung gegeben, wir berechnen nun die nächste Lösung. In Schritt 5 wird vom niedrigsten Knoten aus überprüft, ob eine Änderung des Weges an diesem Knoten vonstatten gehen soll. Sei nun  $b(v)$  der Bedarf an diesem Knoten und  $(W_1, \dots, W_b)$  die Wegkombination, die an diesem Knoten gewählt wurde. Falls am Knoten jeder Weg  $W_i$  gleich dem letzten Weg in der Wegliste ist, so wird für Schritt 5 der nächste nummerierte Knoten betrachtet. Falls am Knoten hingegen nicht jeder Weg gleich dem letzten Weg in der Wegliste des Knotens ist, werden die Wege wie folgt geändert. Sei nun  $j \leq b$  der höchste Index, für den  $W_j$  ungleich dem letzten Weg in der Wegliste des Knotens ist, und sei  $W$  der Weg in der Wegliste, der gleich  $W_j$  ist. Dann wird in Schritt 6 jeder Weg  $W_k$  mit  $j \leq k \leq b$  auf den Weg gesetzt der nach  $W$  in der Wegliste kommt. Außerdem wird in Schritt 7 an jedem vorausgehenden Knoten jeder Weg auf den ersten seiner Wegliste gesetzt. Hierbei ist zu beachten, dass sich bei vorangehenden inneren Knoten mit Verzweigungen sich der Bedarf geändert haben kann. Die nach Schritt 7 vorliegende Wegkombination ist dann die nächste Lösung. Um alle Lösungen zu finden, werden Schritt 5 bis 7 nun solange wiederholt, bis alle Bedarfe auf den letzten Weg der jeweiligen Wegliste gesetzt sind.

Wir haben somit einen Aufzählungsalgorithmus erstellt, der uns unterschiedliche knickminimale Lösungen ausgibt. Es bleibt noch zu zeigen, dass der Algorithmus die gewünschte Eigenschaft besitzt, jede Lösung genau einmal aufzuzählen.

### Korrektheit und Laufzeit des Aufzählungsalgorithmus

Wir wollen in diesem Abschnitt die Korrektheit und Laufzeit des Aufzählungsalgorithmus zeigen. Für die Korrektheit zeigen wir im folgenden Satz, dass jede Lösung genau einmal aufgezählt wird.

**Satz 3.4.** *Durch den Aufzählungsalgorithmus wird jede Lösung genau einmal aufgezählt.*

*Beweis.* Wir zeigen zunächst, dass jede knickminimale orthogonale Repräsentation vom Aufzählungsalgorithmus aufgezählt wird. Nach Lemma 3.2 gilt, dass jede knickminimale orthogonale Repräsentation äquivalent zu einer Kombination aus kürzesten Wegen zur Außenfacette ist. So kann jeder Weg, der durch innere Knoten mit Verzweigungen verläuft, als Menge von Wegen dargestellt werden, bei der jeweils jeder Weg bis zum nächsten inneren Knoten mit Verzweigung beziehungsweise zur Außenfacette geht, ohne

dass sich die zugehörige orthogonale Repräsentation ändert. Wir zeigen nun, dass der Aufzählungsalgorithmus alle Wegkombinationen mit dieser Eigenschaft aufzählt.

Sei  $(W_1, \dots, W_m)$  eine beliebige Wegkombination mit den gewünschten Eigenschaften. Durch Schritt 6 wird jede Wegkombination für  $v$  in mindestens einer Lösung ausgegeben. Sei nun  $v'$  ein weiterer Knoten. Wir nehmen an, dass eine Lösung ausgegeben werden kann, die für jeden höher nummerierten Knoten die richtige Wegkombination besitzt. Aufgrund von Schritt 7 ist zunächst jeder andere Bedarf auf den vordersten Weg in der jeweiligen Wegliste gesetzt. Aufgrund von Schritt 6 wird dann auch eine Lösung ausgegeben, die für  $v'$  und für alle Knoten höher als  $v'$  die richtige Wegkombination besitzt, da alle Wege an Knoten höher als  $v'$  nicht verändert werden. Da der Aufzählungsalgorithmus somit alle Wegkombinationen über innere Knoten mit Verzweigungen aufzählt, zählt der Aufzählungsalgorithmus auch die ursprüngliche Wegkombination zur Außenfacette auf, womit auch die knickminimale orthogonale Repräsentation aufgezählt wird.

Es kann nun auch leicht gezeigt werden, dass jede orthogonale Repräsentation nur einmal aufgezählt wird. Da im Aufzählungsalgorithmus kein Weg auftritt, der einen inneren Knoten mit Verzweigung außerhalb ihres Startknotens oder Endknotens besitzt, folgt mit Lemma 3.3, dass keine redundante Lösung durch den Aufzählungsalgorithmus erzeugt werden kann.  $\square$

Der Aufzählungsalgorithmus ist somit korrekt. Wir wollen nun die Laufzeit des Algorithmus untersuchen, insbesondere beschränken wir uns darauf, die Laufzeit zu untersuchen, mit der die nächste Lösung gefunden wird.

**Satz 3.5.** *Der Aufzählungsalgorithmus findet in linearer Laufzeit die nächste Lösung.*

*Beweis.* Wir betrachten den aufwendigsten Fall, dass für die nächste Lösung bis zum letzten nummerierten Knoten getestet werden muss. Schritt 5 wird in diesen Fall so oft wie die Anzahl an nummerierten Knoten im reduzierten Flussnetzwerk ausgeführt, da für diesen Schritt nur der erste Weg in der Wegkombination betrachtet werden muss, erfolgt dies in linearer Laufzeit. Nach Tamassia [Tam87] und James A. Storer [Sto80] ist die maximale Anzahl an Knicken gleich  $3n+2$ . Da die Laufzeit des sechsten Schrittes von der Anzahl an Wegen in der Wegkombination des Knotens abhängt, die geprüft werden müssen, folgt, dass die Laufzeit in  $\mathcal{O}(3n+2)$  liegt und somit auch linear ist. Da Schritt 7 auch in linearer Zeit durchgeführt werden kann, folgt die Aussage.  $\square$

## 4 Berechnung von einer Alternativlösung in beliebigen Graphen

Im weiteren Verlauf der Arbeit wollen wir uns mit Graphen beschäftigen, die auch Innenfacetten mit mehr als vier Kanten besitzen können. Für das Flussnetzwerk bedeutet dies, dass nun mehrere Senken existieren, in die ein Fluss münden kann, wodurch die Komplexität beim Finden von knickminimalen orthogonalen Repräsentationen deutlich erhöht wird. Wir schauen uns zunächst das Finden von alternativen orthogonalen Repräsentationen unter Vorlage einer beliebigen bereits gegebenen orthogonalen Repräsentation an und werden dann mithilfe einer knickminimalen Lösung eine Alternativlösung berechnen, wobei die Laufzeit der Berechnung geringer sein soll als die direkte Berechnung einer Lösung ohne Vorlage einer bereits optimalen Lösung.

### 4.1 Erweiterung des Flussnetzwerks durch eine vorgegebene orthogonale Repräsentation

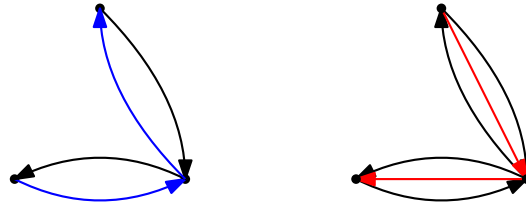
Um aus einer vorgegebenen orthogonalen Repräsentation eines Graphen  $G = (V, E)$  eine alternative Repräsentation zu berechnen, führen wir folgende Definition des erweiterten Flussgraphen ein:

**Definition 4.1** (Erweiterter Flussgraph). *Sei  $G = (V, E)$  ein 4-planarer Graph mit Facettenmenge  $F$ ,  $N(G)$  das dazugehörige Flussnetzwerk und  $X$  ein valider Fluss in  $N(G)$ , dann erfüllt der mit dem Fluss  $X$  erweiterte Flussgraph  $N_X^+(G) = ((V \cup F, A))$  folgende Eigenschaften:*

- $A' = \{(w_i, v_i) \mid v, w \in F(G), (v, w) \in E(N(G)), 1 \leq i \leq X(v, w)\} \cup \{(w_i, v_i) \mid v \in V(G), w \in F(G), (v, w) \in E(N(G)), 1 \leq i \leq X(v, w) - 1\}$
- $A = E(N(G)) \cup A'$

Es werden also für jede Kante, durch die ein Fluss fließt, entgegengesetzt gerichtete Kanten eingefügt, wobei die Anzahl der hinzugefügten Kanten gleich des Flusses ist. Insbesondere wird bei Knoten-Facetten-Kanten eine Kante weniger eingefügt. Da für den weiteren Verlauf der Bedarf eines Knotens keine Rolle spielt, fassen wir den erweiterten Flussgraph  $N_X^+(G)$  als einen gewöhnlichen gewichteten und gerichteten Graphen auf, wobei Kanten in  $A'$  mit der Gewichtung  $-1$  versehen werden und Kanten aus  $N(G)$  mit  $1$ . Ein Beispiel für einen erweiterten Flussgraph ist in Abbildung 4.1 gegeben. Der erweiterte Flussgraph fügt also für jede Kante, durch die ein Fluss verläuft, eine entgegengesetzt gerichtete Kante ein, die negative Kosten enthält. Wichtig für den erweiterten Flussgraph

ist zudem, dass für jede Knoten-Facetten-Kante eine entgegengesetzt gerichtete Kante ausgelassen wird. Dies hat den Grund, dass dadurch der Mindestfluss einer Knoten-Facetten-Kante gewährleistet ist.



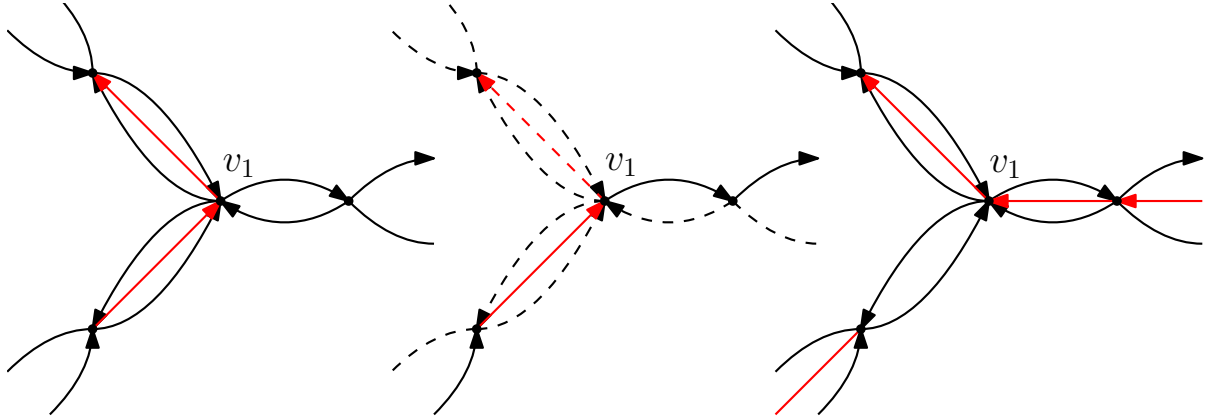
**Abb. 4.1:** Links: Flussnetzwerk mit vorgegebenen Fluss (blau);  
Rechts: Erweiterter Flussgraph, wobei mit rot eine negative Gewichtung gekennzeichnet ist

## 4.2 Finden alternativer Flüsse mithilfe von Kreismengen im erweiterten Flussgraph

Wir wissen bereits, dass jede orthogonale Repräsentation eines Graphen  $G = (V, E)$  einem bestimmten Fluss im Flussnetzwerk  $N(G)$  entspricht. Um also eine alternative Repräsentation zu erhalten, müssen wir den Fluss im Flussnetzwerk ändern. Zusätzlich ist nun jede orthogonale Repräsentation auch äquivalent zu einem erweiterten Flussgraph  $N_X^+(G)$ . Das heißt, dass zu jedem alternativen validen Fluss  $X'$  ein neuer erweiterter Flussgraph  $N_{X'}^+(G)$  aufgebaut werden kann. Damit jedoch der Fluss bei einer Änderung valide bleibt, also weiterhin jeden Bedarf abdeckt, darf sich der Gesamtfluss an jedem Knoten gegenüber  $X$  nicht geändert haben. Es stellt sich nun also die Frage, unter welchen Voraussetzungen ein neuer Fluss  $X'$  bestimmt werden kann, beziehungsweise wann  $X'$  ein valider Fluss ist. In den folgenden Lemmata erarbeiten wir uns eine notwendige und hinreichende Bedingung für die Existenz eines alternativen validen Fluss  $X'$ . In den folgenden Beweisen wird eine mehrmals besuchte Kante als mehrere distinkte Kanten aufgefasst.

**Lemma 4.2.** *Sei  $G = (V, E)$  ein 4-planarer Graph,  $X$  ein gegebener Fluss in  $N(G)$  und seien  $C_1, \dots, C_k$  Kreise mit einer Länge größer 3 in  $N_X^+(G)$ . Sei  $N_{X'}^+(G)$  der Graph, der entsteht, wenn jede Kante mit negativen Kosten in  $\bigcup_{i=1}^k C_i$  entfernt wird und für jede Kante mit positiven Kosten in  $C_i$  mit  $i = 1, \dots, k$  eine entgegengesetzt gerichtete Kante mit Kosten  $-1$  hinzugefügt wird (siehe Abbildung 4.2). Der Graph  $N_{X'}^+(G)$  ist dann gleich einem erweiterten Flussgraphen, der durch einen validen Fluss  $X'$  in  $N(G)$  entsteht.*

*Beweis.* Wir untersuchen jeden Knoten auf eine Änderung des anliegenden Gesamtflusses. Falls an jedem Knoten der Gesamtfluss erhalten bleibt, liegt nach Definition eine valide Flussumleitung vor. Sei  $v$  ein beliebiger Knoten in  $N_X^+(G)$  und seien  $E_{\text{out}}(v)$  und  $E_{\text{in}}(v)$  jeweils die Menge an ausgehenden und eingehenden Kanten von  $v$ , die



**Abb. 4.2:** Eine valide Flussumleitung an einem Knoten mit einem Gesamtfluss von 0.  
 1: Auszug eines erweiterten Flussgraphen  $N_X^+(G)$  mit negativen Kanten (rot); 2: In  $\bigcup_{i=1}^k C_i$  enthaltene Kanten am Knoten  $v_1$  durchgezogen dargestellt;  
 3: Resultierender erweiterter Flussgraph  $N_{X'}^+(G)$

in  $C_1, \dots, C_k$  liegen. Seien dementsprechend  $E_{\text{out}}^-(v)$  und  $E_{\text{in}}^-(v)$  die Menge an ausgehenden beziehungsweise eingehenden Kanten an  $v$  in  $C_1, \dots, C_k$  mit negativer Gewichtung. Für  $X(v)$  bezüglich  $N(G)$  gilt dann:

$$X(v) = |E_{\text{out}}^-(v)| - |E_{\text{in}}^-(v)| + x \quad (4.1)$$

wobei  $x$  der Gesamtfluss über die Kanten an  $v$  bezeichnet, die nicht in  $\bigcup_{i=1}^k C_i$  liegen.

Da in  $N_{X'}^+(G)$  für jede Kante mit positiven Gewicht in  $C_i$  mit  $i = 1, \dots, k$  eine entgegengesetzt gerichtete Kante mit Kosten  $-1$  hinzugefügt wurde, gilt für die Anzahl an neuen Kanten mit negativer Gewichtung  $|E'_{\text{out}}{}^-(v)|$  und  $|E'_{\text{in}}{}^-(v)|$ :

$$|E'_{\text{in}}{}^-(v)| = |E_{\text{out}}(v)| - |E_{\text{out}}^-(v)| \quad \text{und} \quad |E'_{\text{out}}{}^-(v)| = |E_{\text{in}}(v)| - |E_{\text{in}}^-(v)| \quad (4.2)$$

Da jede negativ gewichtete Kante in  $\bigcup_{i=1}^k C_i$  entfernt wurde, gilt für den neuen Gesamtfluss  $X'(v)$  bezüglich  $N_{X'}^+(G)$ :

$$\begin{aligned} X'(v) &= |E'_{\text{out}}{}^-(v)| - |E'_{\text{in}}{}^-(v)| + x \\ &= |E_{\text{in}}(v)| - |E_{\text{in}}^-(v)| - (|E_{\text{out}}(v)| - |E_{\text{out}}^-(v)|) + x \\ &= X(v) + |E_{\text{in}}(v)| - |E_{\text{out}}(v)| \end{aligned} \quad (4.3)$$

Da es sich bei  $C_1, \dots, C_k$  um Kreise handelt, folgt, dass die Anzahl an ausgehenden und eingehenden Kanten im Kreis gleich sind, also  $|E_{\text{out}}| = |E_{\text{in}}|$  gilt und somit auch  $X'(v) = X(v)$ . Also existiert eine Gesamtflussänderung an keinem Knoten  $v$ . Da zudem bei Knoten-Facetten-Kanten ein Fluss von mindestens 1 weiterhin anliegt, handelt sich um einen validen Fluss  $X'$ .  $\square$

**Lemma 4.3.** *Seien  $X$  und  $X'$  unterschiedliche valide Flüsse. Jede Änderung von  $X'$  bezüglich des vorgegebenen Flusses  $X$  lässt sich durch eine Menge an Kreisen  $\{C_1, \dots, C_k\}$  in  $N_X^+(G)$  beschreiben.*

*Beweis.* Angenommen, die Aussage gilt nicht, dann existiert ein Knoten  $v$  in  $N_X^+(G)$ , bei dem die Anzahl  $|E_{\text{in}}(v)|$  der eingehenden Kanten, deren Fluss geändert wird, ungleich der Anzahl  $|E_{\text{out}}(v)|$  der ausgehenden Kanten ist, deren Fluss geändert wird. Seien nun  $|E_{\text{in}}^-(v)|$  und  $|E_{\text{out}}^-(v)|$  die Anzahl von diesen Kanten mit negativen Gewicht, dann gelten für die Gesamtflüsse  $X(v)$  und  $X'(v)$  am Knoten  $v$  analog zu Lemma 4.2 die Gleichungen (4.1), (4.2) und (4.3). Da der Gesamtfluss an  $v$  bei jedem Fluss gleich bleibt, also  $X'(v) = X(v)$  gilt, folgt  $|E_{\text{out}}(v)| = |E_{\text{in}}(v)|$ , was jedoch im Widerspruch zur Annahme steht, dass  $|E_{\text{out}}(v)| \neq |E_{\text{in}}(v)|$  gilt. Also folgt die Behauptung.  $\square$

Eine Menge an Kreisen im erweiterten Flussgraph entspricht also zusammen mit einem Fluss  $X$  einem validen Fluss  $X'$ . Analog dazu gilt aber auch umgekehrt, dass ein valider Fluss  $X'$  einer Menge an Kreisen in  $N_X^+(G)$  entspricht, es handelt sich somit um eine Äquivalenz.

Somit können wir also weitere orthogonale Repräsentationen finden, indem wir Kreisfamilien im erweiterten Flussgraphen  $N_X^+(G)$  betrachten. Diese orthogonalen Repräsentationen besitzen jedoch noch keine Einschränkungen bezüglich ihrer Knicke.

### 4.3 Anzahl an Knicken in alternativen Repräsentationen

Wir wollen nun Änderungen am Fluss bezüglich ihrer assoziierten Knickänderungen in der entsprechenden orthogonalen Repräsentation einordnen. Dafür werden wir zeigen, dass die Kosten der Kreise in direktem Zusammenhang zur Anzahl der Knicke stehen.

**Lemma 4.4.** *Sei  $G = (V, E)$  ein 4-planarer Graph und sei eine Menge an Kreisen  $\{C_1, \dots, C_k\}$  gegeben. Dann erhöht sich die Anzahl der Knicke in der resultierenden orthogonalen Repräsentation um die Gesamtkosten von  $\{C_1, \dots, C_k\}$ .*

*Beweis.* Seien  $c = \text{cost}(\{C_1, \dots, C_k\})$  die Gesamtkosten der Kreismenge  $\{C_1, \dots, C_k\}$  und  $E^-(v), E^+(v)$  die Menge an negativ beziehungsweise positiv gewichteten Kanten an einem Knoten  $v$ , die in den Kreisen durchlaufen werden, dann folgt:

$$c = |E^+(v)| - |E^-(v)|$$

Dies bewirkt für die neue Anzahl an negativen Kanten  $|E'^-|$  in  $N_{X'}^+(G)$ :

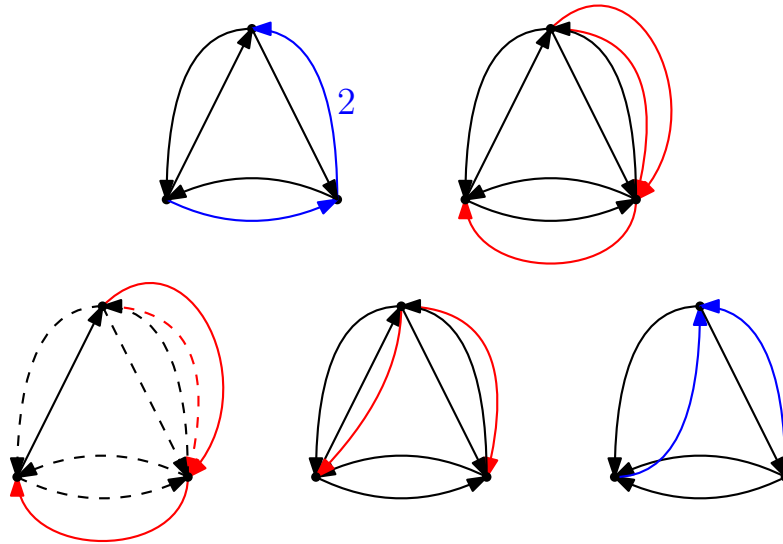
$$|E'^-(v)| = |E^-(v)| + c$$

Da Knoten-Facetten-Kanten nur Abflüsse vom Knoten aus enthalten können und der Gesamtfluss an jedem Knoten gleich bleibt, folgt, dass für Knoten-Facetten-Kanten die Anzahl an negativ und positiv gewichteten Kanten gleich ist. Die Anzahl an Knoten-Facetten-Kanten mit negativer Gewichtung bleibt somit erhalten. Es folgt somit, dass die Änderung der Anzahl an negativen Kanten aufgrund von Facetten-Facetten-Kanten vonstatten geht. Da aber jede negative Facetten-Facetten-Kante einem Knick in der orthogonalen Repräsentation entspricht, besitzt die neue orthogonale Repräsentation genau in der Anzahl  $c$  entsprechend mehr Knicke.  $\square$

Das Problem zum Finden von alternativen validen Flüssen reduziert sich somit auf das Problem, Kreise bestimmter Kosten zu bestimmen. Falls  $N(G)$  mit einer kostenminimalen Fluss  $X$  erweitert wurde, also mit einer knickminimalen orthogonalen Repräsentation, folgen außerdem mehrere Eigenschaften über die Art an Kreisen, die im erweiterten Flussgraph  $N_X^+(G)$  enthalten sein können.

**Korollar 4.5** ([Tam87]). *Sei  $G = (V, E)$  ein 4-planarer Graph, dann existiert in  $N_X^+(G)$  kein Kreis mit negativen Kosten.*

*Beweis.* Das Korollar folgt direkt aus Lemma 4.4. Denn angenommen, es gäbe einen Kreis mit negativen Kosten, dann gibt es nach Lemma 4.4 eine orthogonale Repräsentation, deren Anzahl der Knicke um den Betrag der Kosten geringer ist als die Anzahl der Knicke in der knickminimalen Lösung, wie in Abbildung 4.3 veranschaulicht wird. Da die Lösung knickminimal ist, ist dies ein Widerspruch, weshalb ein Kreis mit negativen Gesamtkosten in  $N_X^+(G)$  nicht existieren kann.  $\square$



**Abb. 4.3:** Existenz eines negativen Kreises führt zu einer Repräsentation mit weniger Knicke.  
 1: Flussgraph mit eingezeichnetem Fluss (blau); 2: Erweiterter Flussgraph  $N_X^+(G)$ ;  
 3: Markierte Kanten bilden einen Kreis mit negativen Kosten;  
 4: Resultierender erweiterter Flussgraph  $N_X^+(G)$ ; 5: Flussgraph mit neuem Fluss

Ähnlich wie in Korollar 4.5 können wir auch folgern, dass die Art der existierenden Kreise in  $N_X^+(G)$  Aufschluss über die alternativen knickminimalen orthogonalen Repräsentationen gibt.

**Korollar 4.6.** *Sei  $G = (V, E)$  ein 4-planarer Graph. Es existiert genau dann eine alternative knickminimale Lösung, falls in  $N_X^+(G)$  ein nicht-trivialer Kreis mit Kosten null existiert.*

*Beweis.* Auch dieses Korollar folgt aus Lemma 4.4.

$\Rightarrow$ : Angenommen, es existiert eine alternative knickminimale Lösung, dann existiert auch eine Flussumleitung, die nach Lemma 4.4 einer Menge an Kreisen  $\{C_1, \dots, C_k\}$  in  $N_X^+(G)$  mit Gesamtkosten null entspricht. Da nach Korollar 4.5 kein Kreis mit negativen Kosten existiert, besitzt jeder Kreis  $C_i$  in  $\{C_1, \dots, C_k\}$  nicht-negative Kosten. Also folgt für alle  $C_i$ , dass ihre Gesamtkosten null betragen.

$\Leftarrow$ : Angenommen, es existiert ein nicht trivialer Kreis  $C$  mit Kosten null in  $N_X^+(G)$ , dann ist  $C$  nach Lemma 4.4 äquivalent zu einer Flussumleitung, die die Anzahl der Knicke nicht erhöht. Damit ist also auch eine alternative knickminimale Lösung gegeben.

Aufgrund der gegenseitigen Implikation folgt die Äquivalenz.  $\square$

Wir erhalten also eine alternative Lösung, wenn wir einen Kreis in  $N_X^+(G)$  mit den Kosten 0 finden. Zusätzlich kann die Knickminimalität getestet werden, indem die Existenz eines Kreises mit negativen Kosten geprüft wird.

## 4.4 Alternativlösungen bei Kantenrestriktionen

Wir werden zunächst alternative Lösungen betrachten, die einer bestimmten Einschränkung unterliegen, nämlich einer sogenannten Kantenrestriktion. Kantenrestriktionen sind in der Praxis sehr nützlich, da der Wunsch nach einer Alternativlösung häufig dadurch geprägt ist, eine bestimmte Kante knickfrei zu lassen, oder eine maximale Anzahl an Knicken pro Kante nicht zu überschreiten.

**Definition 4.7** (Kantenrestriktion). *Sei  $G = (V, E)$  ein 4-planarer Graph,  $N(G)$  das dazugehörige Flussnetzwerk und  $X$  ein valider Fluss in  $N(G)$ . Sei nun  $N_X^+(G)$  das mit  $X$  erweiterte Flussnetzwerk von  $N(G)$  und sei  $e$  eine beliebige Kante aus  $N_X^+(G)$ . Falls  $e$  negative Kosten besitzt, verlangt eine Kantenrestriktion von  $e$ , dass  $N_X^+(G)$  die Kante  $e$  nicht enthält. Falls  $e$  positive Kosten besitzt, verlangt eine Kantenrestriktion von  $e$ , dass in  $N_X^+(G)$  eine entgegengesetzt gerichtete Kante  $e'$  mit negativen Kosten hinzugefügt werden muss.*

Diese Einschränkung erlaubt es uns das Problem, alternative knickminimale orthogonale Repräsentation zu finden, weiter auf das Problem zu reduzieren, einen kürzesten Weg im erweiterten Flussnetzwerk zu finden. Dieses Problem ist für planare Graphen nach Mozes und Wulff-Nilsen [MWN10] schneller lösbar als eine neue Repräsentation zu berechnen.

**Korollar 4.8.** *Sei  $G = (V, E)$  ein 4-planarer Graph und sei  $e$  die Kante mit Kantenrestriktion, dann gilt:*

1. *Falls  $e$  die Kosten  $-1$  besitzt, existiert genau dann eine alternative knickminimale Lösung, wenn vom Endpunkt zum Startpunkt von  $e$  ein nicht-trivialer Weg mit Kosten eins existiert.*



2. Falls  $e$  die Kosten 1 besitzt, existiert genau dann eine alternative knickminimale Lösung, wenn vom Endpunkt zum Startpunkt von  $e$  ein nicht-trivialer Weg mit Kosten  $-1$  existiert.

*Beweis.* 1. Sei  $e$  eine Kante mit negativen Kosten:

$\Rightarrow$ : Angenommen, es gibt eine alternative Lösung, für die  $N_X^+(G)$  die Kante  $e$  nicht enthält. Dann existiert nach Korollar 4.6 in  $N_X^+(G)$  ein nicht-trivialer Kreis  $C$  mit Kosten null, der  $e$  enthält. Da  $e$  negative Kosten besitzt, existiert somit auch ein nicht-trivialer Weg mit Kosten eins vom Endpunkt zum Startpunkt von  $e$ .

$\Leftarrow$ : Angenommen, es gibt vom Endpunkt zum Startpunkt von  $e$  einen nicht-trivialen Weg mit Kosten eins, dann lässt sich dieser mit  $e$  zu einem Kreis mit Kosten null erweitern. Nach Lemma 4.4 existiert also eine alternative Lösung, für die  $N_X^+(G)$  die Kante  $e$  nicht enthält.

Da beide Implikationen gelten, folgt Äquivalenz.

2. Sei  $e$  nun eine Kante mit positiven Kosten:

$\Rightarrow$ : Angenommen, es gibt eine alternative Lösung, für die  $N_X^+(G)$  die Kante  $e'$  hinzufügt. Dann existiert nach Korollar 4.6 in  $N_X^+(G)$  ein nicht-trivialer Kreis  $C$  mit Kosten null, der  $e$  enthält. Da  $e$  positive Kosten besitzt, existiert somit auch ein nicht-trivialer Weg mit Kosten  $-1$  vom Endpunkt zum Startpunkt von  $e$ .

$\Leftarrow$ : Angenommen, es gibt vom Endpunkt zum Startpunkt von  $e$  einen nicht-trivialen Weg mit Kosten  $-1$ , dann lässt sich dieser mit  $e$  zu einem Kreis mit Kosten null erweitern. Nach Lemma 4.4 existiert also eine alternative Lösung, für die  $N_X^+(G)$  die Kante  $e'$  hinzufügt.

Aufgrund der gegenseitigen Implikation gilt Äquivalenz. □

Somit müssen wir, falls eine Kantenrestriktion existiert, für das Finden einer alternativen knickminimalen orthogonalen Repräsentation einen Weg bestimmter Länge ermitteln, was uns zu folgenden Satz führt:

**Satz 4.9.** *Sei  $G = (V, E)$  ein 4-planarer Graph. Bei einer Kantenrestriktion lässt sich eine alternative knickminimale orthogonale Repräsentation in  $\mathcal{O}(n \log^2 n / \log \log n)$  Zeit finden, wobei  $n$  die Anzahl an Knoten in  $G$  ist.*

*Beweis.* Nach Mozes und Wulff-Nilsen [MWN10] lässt sich für einen gerichteten planaren Graphen mit reellen Kantengewichten und ohne negative Kreise ein kürzester Weg in  $\mathcal{O}(n \log^2 n / \log \log n)$  Zeit finden, wobei  $n$  die Anzahl an Knoten im Graphen ist. Wir müssen also zeigen, dass  $N_X^+(G)$  die nötigen Vorgaben erfüllt, und dass die Anzahl an Knoten in  $N_X^+(G)$  linear zu  $n$  ist.

Offensichtlich ist  $N_X^+(G)$  ein gerichteter Graph mit reellen Kantengewichten. Aufgrund der Planarität von  $N(G)$  ist auch  $N_X^+(G)$  planar. Nach Korollar 4.5 besitzt  $N_X^+(G)$  auch

keine negativen Kreise. Sei nun  $n'$  die Anzahl an Knoten in  $N_X^+(G)$ . Da  $N_X^+(G)$  alle nötigen Eigenschaften erfüllt, kann in  $\mathcal{O}(n' \log^2 n' / \log \log n')$  Zeit ein wie in Korollar 4.8 beschriebener Weg gefunden werden.

Da in  $N(G)$  und somit auch in  $N_X^+(G)$  zusätzlich zu den Knoten von  $G$  auch die Facetten von  $G$  Knoten bilden, gilt  $n' = n + |F|$ . Da jedoch jeder Knoten aufgrund der 4-Planarität nur an maximal vier Facetten anliegen kann gilt:  $|F| \leq 4n$ . Also gilt insgesamt:  $n' = n + |F| \leq n + 4n = 5n$ . Also ist die Anzahl an Knoten in  $N_X^+(G)$  linear zu  $n$ , woraus folgt:

$$\mathcal{O}\left(n' \frac{\log^2(n')}{\log \log(n')}\right) = \mathcal{O}\left(n \frac{\log^2(n)}{\log \log(n)}\right)$$

Somit gilt der Satz. □

Wir können also bei einer Kantenrestriktion eine Alternativlösung schneller berechnen, als eine komplett neue Lösung zu berechnen. Überraschenderweise können wir im nächsten Kapitel zeigen, dass jedoch auch ohne Kantenrestriktion die Laufzeit erhalten bleibt.

## 4.5 Alternativlösungen ohne Kantenrestriktion

Alternativlösungen ohne Nebenbedingungen zu finden kann bei vielen Problemen schwieriger sein, als eine Alternativlösung mit Nebenbedingung zu finden. In unserem Fall kann bei Existenz einer Alternativlösung mit gewählter Kantenrestriktion auch eine Lösung ohne Kantenrestriktion in derselben Laufzeit gefunden werden. Jedoch ist die Existenz einer Alternativlösung mit gewählter Kantenrestriktion nicht garantiert. Ein Algorithmus, der auf Lösungen mit Kantenrestriktionen basiert, müsste im schlechtesten Fall alle möglichen negativen Kantenrestriktionen testen, was die Laufzeit erheblich erhöht. Wir versuchen also direkt eine Alternativlösung zu berechnen, wobei nach Korollar 4.6 die Alternativlösung äquivalent zu einem Kreis im erweiterten Flussnetzwerk ist.

**Satz 4.10.** *Sei  $G = (V, E)$  ein 4-planarer Graph. Es lässt sich eine alternative knick-minimale orthogonale Repräsentation in  $\mathcal{O}(n \log^2 n / \log \log n)$  Zeit finden, wobei  $n$  die Anzahl an Knoten in  $G$  ist.*

*Beweis.* Nach Liang und Lu [LL17] lassen sich Kreise in gerichteten planaren Graphen mit  $n$  Knoten und nicht-negativen Kantengewichtungen in  $\mathcal{O}(n \log n \log \log n)$  Zeit finden. Im Beweis von Satz 4.9 haben wir bereits alle Voraussetzungen bis auf die nicht-negativen Kantengewichtungen geprüft. Da unser erweiterter Flussgraph jedoch auch negative Kantengewichte enthält, muss zunächst eine Umverteilung der Kantengewichtungen vonstatten gehen.

Nach Liang und Lu [LL17] können wir das Problem einen kostenminimalen Kreis in einem Graphen mit negativen Kantengewichten mithilfe kürzester Wege auf das Problem mit nicht-negativen Kantengewichten reduzieren. Da wir die Länge der kürzesten Wege zu jedem Knoten in Graphen nach Mozes und Wulff-Nilsen [MWN10] in  $\mathcal{O}(n \log^2 n / \log \log n)$  Zeit finden können, folgt für die Gesamtlaufzeit, dass sie in  $\mathcal{O}(n \log^2 n / \log \log n + n \log n \log \log n) = \mathcal{O}(n \log^2 n / \log \log n)$  Zeit liegt. □

Wir können somit auch Alternativlösungen ohne Nebenbedingungen in einer schnelleren Laufzeit finden, als eine Erstlösung zu finden, was das Hauptresultat dieser Arbeit ist. Wir wollen uns jedoch noch im nächsten Kapitel mit einem weiteren Problem beschäftigen, das bereits in Kapitel 3 für kleine Innenfacetten untersucht wurde.

## 5 Aufzählung aller Lösungen

Wir wollen uns nun zuletzt noch mit dem Problem beschäftigen, einen Aufzählungsalgorithmus für alle knickminimalen orthogonalen Repräsentationen zu finden, der jede Lösung genau einmal aufzählt und beim Finden der nächsten Lösung die Verzögerung polynomiell ist. Dazu führen wir zunächst eine weitere mögliche Einschränkung für das Finden von alternativen knickminimalen orthogonalen Repräsentationen ein.

**Definition 5.1** (Kantenfixierung). *Sei  $G = (V, E)$  ein 4-planarer Graph,  $N(G)$  das dazugehörige Flussnetzwerk und  $X$  ein valider Fluss in  $N(G)$ . Sei nun  $N_X^+(G)$  der mit  $X$  erweiterte Flussgraph von  $N(G)$  und seien  $\{e_1, \dots, e_i\}$  beliebige Kanten aus  $N_X^+(v)$  mit negativen Kosten. Dann verlangt eine Kantenfixierung von  $\{e_1, \dots, e_i\}$ , dass  $N_{X'}^+(G)$  die Kanten in  $\{e_1, \dots, e_i\}$  weiterhin enthält.*

Wir können eine alternative Lösung mit Kantenfixierung erhalten, indem wir die entsprechende Kante beim Anwenden des Algorithmus aus dem erweiterten Flussgraph entfernen. Die nötigen Eigenschaften für den Algorithmus von Liang und Lu bleiben dadurch weiterhin erhalten.

### 5.1 Funktionsweise des Aufzählungsalgorithmus

Der Algorithmus berechnet im ersten Schritt zum Graphen  $G$  das Flussnetzwerk  $N(G)$ , wobei durch jede Kante in  $N(G)$  durchnummeriert wird. Im zweiten Schritt wird aus der vorgegebenen Lösung der erweiterte Flussgraph  $N_X^+(G)$  berechnet. Um die erste auszugebene Lösung zu berechnen, wird in Schritt 3 für die erste Kante eine alternative Repräsentation mit Kantenrestriktion berechnet, bis keine weitere mehr möglich ist, also der Fluss an der Kante maximal ist. In Schritt 4 wird für die nächste Kante so lange eine Lösung mit Kantenrestriktion berechnet, bis der Fluss maximal ist, wobei für die vorherigen Kanten alle entgegengesetzt gerichteten Kanten fixiert werden. Schritt 4 wird dann für alle Kanten der Reihe nach wiederholt. Die resultierende knickminimale orthogonale Repräsentation ist nun unsere erste Lösung, wobei von der ersten Kante aus der Fluss auf jeder Kante abhängig von den vorausgegangenen Kanten maximiert ist.

Für die nächste Lösung wird in Schritt 5 von der niedrigsten Kante aus versucht, eine Lösung mit Kantenrestriktion zu finden, wobei Kanten, die entgegengesetzt zu Kanten mit niedrigerer Nummerierung sind, fixiert werden. Falls bei einer Kante mit der Nummer  $n'$  eine Alternativlösung möglich ist, wird für diese Alternativlösung von der Kante mit der Nummer  $n' + 1$  aus Schritt 4 angewendet. Der Fluss auf den Kanten wird somit wieder maximiert. Schritt 5 wird nun solange wiederholt, bis jede Lösung ausgegeben wurde.

## 5.2 Korrektheit und Laufzeit des Aufzählungsalgorithmus

Wir zeigen in diesem Abschnitt die Korrektheit und Laufzeit des Aufzählungsalgorithmus. Für die Korrektheit müssen wir zeigen, dass der Aufzählungsalgorithmus jede Lösung genau einmal ausgibt. Wir beweisen davor jedoch noch ein simples Lemma.

**Lemma 5.2.** *Sei  $G$  ein 4-planarer Graph und  $N(G)$  das dazugehörige Flussnetzwerk. Sei nun  $x_{\min}$  der minimale Fluss, der für einen validen Fluss durch eine Kante  $e$  fließen muss, und  $x_{\max}$  der maximale Fluss der durch diese Kante fließen kann, dann gibt es für jede Zahl  $x$  mit  $x_{\min} \leq x \leq x_{\max}$  einen validen Fluss  $X$  mit  $X(e) = x$ .*

*Beweis.* Seien  $X_{\min}$  und  $X_{\max}$  Flüsse mit minimalen Fluss durch  $e$  und maximalen Fluss durch  $e$ . Dann gibt es nach Lemma 4.3 eine Menge an Kreisen, die der Änderung von  $X_{\min}$  zu  $X_{\max}$  entsprechen. Sei  $m = X_{\max} - X_{\min}$ . Ohne Beschränkung der Allgemeinheit seien nun die Kreise  $C_1, \dots, C_m$  die Kreise, die jeweils einmal die Kante  $e$  enthalten. Dann folgt mit Lemma 4.2, dass durch die Kreismenge  $\{C_1, \dots, C_{m'}\}$  mit  $m' \leq m$  eine Lösung beschrieben wird, bei der die Kante  $e$  einen Fluss von  $X_{\min} + m'$  besitzt.  $\square$

Das Lemma ist zudem auch gültig, wenn beliebige Kanten fixiert sind, wobei der maximale und minimale Fluss jedoch dann abhängig von der vorliegenden Fixierung ist. Mithilfe dieses Lemmas können wir direkt die Korrektheit des Algorithmus beweisen.

**Satz 5.3.** *Der Aufzählungsalgorithmus gibt jede Lösung genau einmal aus.*

*Beweis.* Der Aufzählungsalgorithmus zählt offensichtlich keine Lösung zweimal auf. Somit müssen wir nur zeigen, dass jede Lösung tatsächlich auch ausgegeben wird. Sei  $X$  beliebige Lösung. Wir zeigen nun induktiv, dass der Aufzählungsalgorithmus eine Lösung  $X'$  ausgibt, welcher gleich der Lösung  $X$  ist, also bei der durch jede Kante der nötige Fluss fließt. Für  $X(e_1) = x_1$  gilt:  $X_{1,\min} \leq x_1 \leq X_{1,\max}$ . Falls  $x_1 = X_{1,\max}$  gilt, gibt es eine Lösung, die vom Aufzählungsalgorithmus ausgegeben wird, und den Fluss  $x_1$  an der Kante  $e_1$  besitzt. Wir setzen nun als Induktionsvoraussetzung voraus, dass für jeden Fluss größer als  $x_1$  vom Aufzählungsalgorithmus eine Lösung aufgezählt wird, falls eine solche Lösung existiert. Nach Lemma 5.2 werden alle Flüsse  $x'$  mit  $x_1 \leq x' \leq X_{1,\max}$  bei einer Lösung angenommen und nach Induktionsvoraussetzung wird dann auch für jedes  $x'$  eine Lösung vom Aufzählungsalgorithmus ausgegeben. Da beim Aufzählen, falls eine Lösung mit einem Fluss von  $x' - 1$  an  $e_1$  existiert, diese auch aufgezählt wird, wird aufgrund der Induktionsvoraussetzung auch eine Lösung mit dem Fluss  $x_1$  an  $e_1$  aufgezählt.

Wir zeigen nun mit einer zweiten Induktion, dass dies für alle Kanten gilt. Sei nun Induktionsvoraussetzung, dass eine Lösung ausgegeben wird, bei der an jeder Kante  $e_{m'}$  mit  $m' \leq m$  der passende Fluss fließt, dann können wir die erste Induktion unter Fixierung der ersten  $m - 1$  Kanten für  $e_m$  durchführen, wodurch insgesamt eine Lösung  $X'$  ausgegeben wird, bei der jede Kante  $e_{m'}$  mit  $m' \leq m$  den passenden Fluss besitzt. Mit dem Prinzip der vollständigen Induktion gilt dies dann für alle Kanten in  $N(G)$ .  $\square$

Der Aufzählungsalgorithmus ist somit korrekt. Es fehlt noch zu zeigen, dass die nächste Lösung in polynomieller Zeit gefunden wird.

**Satz 5.4.** *Der Aufzählungsalgorithmus zählt die nächste Lösung in  $\mathcal{O}(n^2 \log^2 n / \log \log n)$  auf.*

*Beweis.* Wir betrachten für die Laufzeit wieder den schlechtesten Fall, wobei  $n$  für die Anzahl an Knoten in  $G$  steht. Im fünften Schritt wird der Algorithmus zum Finden einer Alternativlösung maximal so oft wie die Anzahl an Kanten in  $N(G)$  angewendet. Da die Anzahl an Kanten linear zur Knotenanzahl in  $G$  ist, folgt für den Schritt somit, dass er in  $\mathcal{O}(n(n \log^2 n / \log \log n))$  Zeit durchgeführt werden kann.

Für den vierten Schritt, welcher die Maximierung beinhaltet, wird der Algorithmus für das Finden einer Alternativlösung maximal so oft wie die Anzahl an Knicken und die Anzahl an Kanten angewendet. Dies ist der Fall, da in der Maximierung der Algorithmus für jede Kante höchstens einmal mehr als ihr schlussendlicher Fluss in der Lösung angewendet wird. Da die maximale Anzahl nach Tamassia [Tam87] und Storer [Sto80]  $3n + 2$  beträgt, liegt die Laufzeit des zweiten Schrittes somit auch in  $\mathcal{O}(n(n \log^2 n / \log \log n))$ . Folglich gilt für den gesamten Algorithmus, dass die nächste Lösung in  $\mathcal{O}(n^2 \log^2 n / \log \log n)$  Zeit ausgegeben wird.  $\square$

## 6 Fazit und offene Probleme

Wir haben in der Arbeit gezeigt, dass das Finden von alternativen knickminimalen orthogonalen Repräsentationen in einer Laufzeit von  $\mathcal{O}(n \log^2 n / \log \log n)$  möglich ist, was schneller ist als die Laufzeitschranke von  $\mathcal{O}(n^{3/2})$  zum Finden einer initialen Lösung. Außerdem konnten wir zeigen, dass alle knickminimalen Lösungen mit polynomieller Verzögerung aufgezählt werden können. Ausschlaggebend ist hierbei die Verwendung von Flussnetzwerken und die Äquivalenz von Flüssen und Lösungen. Mithilfe von Kantenrestriktionen und Kantenfixierungen haben wir auch gezeigt, dass wir Repräsentationen mit bestimmten Eigenschaften ohne Verschlechterung der Laufzeit finden können.

Da die Algorithmen in dieser Arbeit nur auf Flussnetzwerke basieren und andere Eigenschaften von orthogonalen Repräsentationen vernachlässigen, sind die Algorithmen möglicherweise bezüglich der Laufzeit nicht optimal. Weitere Eigenschaften, die ein effizienterer Algorithmus berücksichtigen könnte, werden beispielsweise von Tamassia [Tam87] beschrieben. So kann man mithilfe von geschlossenen und gerichteten Kurven die Minimalität einer orthogonalen Zeichnung zeigen und bei der Existenz von bestimmten Kurven eine knickminimale Repräsentation ableiten.

Wir haben uns außerdem nur auf Alternativlösungen für orthogonale Repräsentationen beschränkt. Relevant sind oft aber auch die entsprechenden orthogonalen Zeichnungen. Welche Lösungen eine Zeichnung mit minimaler Fläche implizieren, also welche Lösungen tatsächlich relevant sind, ist eine weitere offene Frage dieser Arbeit.

In unserer Arbeit impliziert die Orthogonalität zudem die Anordnung der Kanten und Knoten auf einer Vierecksparkettierung. Die Resultate sind hier aber auch für alle drei platonischen Parkettierungen anwendbar, indem die Bedarfe im Flussnetzwerk angepasst werden, jedoch gelten die Resultate dieser Arbeit nicht für archimedische und semireguläre Parkettierungen. Auch für das Finden einer knickminimalen Lösung in diesen Anordnungen ist noch kein Resultat veröffentlicht worden.

# Literaturverzeichnis

- [BETT94] Giuseppe Di Battista, Peter Eades, Roberto Tamassia und Ioannis G Tollis: Algorithms for drawing graphs: An annotated bibliography. *Computational Geometry*, 4(5):235–282, 1994, 10.1016/0925-7721(94)00014-X, ISSN 0925-7721.
- [CK12] Sabine Cornelsen und Andreas Karrenbauer: Accelerated Bend Minimization. *Journal of Graph Algorithms and Applications*, 16(3):635–650, 2012, 10.7155/jgaa.00265.
- [DMSZ20] Argyrios Deligkas, George B. Mertzios, Paul G. Spirakis und Viktor Zamaraev: Exact and Approximate Algorithms for Computing a Second Hamiltonian Cycle. In: *45th International Symposium on Mathematical Foundations of Computer Science (MFCS 2020)*, Band 170 der Reihe *Leibniz International Proceedings in Informatics (LIPIcs)*, Seiten 27:1–27:13. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2020, 10.4230/LIPIcs.MFCS.2020.27.
- [LL17] Hung Chun Liang und Hsueh I Lu: Minimum Cuts and Shortest Cycles in Directed Planar Graphs via Noncrossing Shortest Paths. *SIAM Journal on Discrete Mathematics*, 31(1):454–476, 2017, 10.1137/16M1057152.
- [MWN10] Shay Mozes und Christian Wulff-Nilsen: Shortest Paths in Planar Graphs with Real Lengths in  $\mathcal{O}(n \log^2 n / \log \log n)$  Time. In: *Proceedings of the 18th Annual European Conference on Algorithms (ESA)*, Seite 206–217. Springer-Verlag, 2010, 10.5555/1882123.1882147.
- [Sto80] James A. Storer: The Node Cost Measure for Embedding Graphs on the Planar Grid (Extended Abstract). In: *Proceedings of the Twelfth Annual ACM Symposium on Theory of Computing*, Seite 201–210, 1980, 10.1145/800141.804667.
- [Tam87] Roberto Tamassia: On Embedding a Graph in the Grid with the Minimum Number of Bends. *SIAM Journal on Computing*, 16(3):421–444, 1987, 10.1137/0216030.