



Grapheneinbettungen und Optimierung

Autor:
Jonas Neukamm

Betreuer und Erstgutachter:
Prof. Dr. Stefan Felsner

Matrikelnummer:
324283

Zweitgutachter:
Dr. Frank Lutz

*Masterarbeit zur Prüfung zum Master of Science
an der*

Technische Universität Berlin
Institut für Mathematik

17.7.2019

Eidesstattliche Erklärung

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

Berlin, den

.....

Jonas Neukamm

Inhaltsverzeichnis

| | | |
|----------|---|-----------|
| 1 | Einleitung | 1 |
| 2 | Grundlagen | 3 |
| 2.1 | Geradlinige Dreiecks-Darstellungen (SLTRs) | 3 |
| 2.2 | Schnyder Woods | 6 |
| 2.2.1 | Einbettungen via Schnyder Woods | 9 |
| 2.3 | α -Orientierungen | 10 |
| 2.4 | Gerichtete-Multi-Fluss-Probleme | 13 |
| 3 | Bedingungen für die Existenz von SLTRs | 15 |
| 3.1 | SLTRs durch harmonische Funktionen | 16 |
| 3.1.1 | Harmonische Funktionen auf planaren Graphen | 20 |
| 3.1.2 | Das resultierende Gleichungssystem | 21 |
| 3.2 | Ecken kompatible Paare | 22 |
| 4 | Algorithmen zur Erkennung von SLTRs | 45 |
| 4.1 | SLTRs als Lösungen eines Zwei-Fluss-Problems | 45 |
| 4.1.1 | Schnyder-Wood-Fluss | 46 |
| 4.1.2 | FAA-Fluss | 48 |
| 4.1.3 | Ein Zwei-Fluss Netzwerk zur Erkennung von SLTRs | 50 |
| 4.2 | Nicht-ganzzahlige Flüsse auf \mathcal{N}_G | 58 |
| 4.2.1 | Minimale Schnitte in \mathcal{N}_G | 62 |
| 4.2.2 | Alternierende Zykel auf \mathcal{N}_G | 66 |
| 5 | Das Programm | 73 |
| 5.1 | Visualisierung | 74 |
| 5.1.1 | Möglichkeiten bei der Wahl von λ | 75 |
| 5.2 | Experimentelle Rechnungen | 77 |
| 5.3 | Dokumentation | 80 |
| 6 | Fazit | 83 |
| | Literaturverzeichnis | 85 |

1 Einleitung

Wir werden uns in dieser Arbeit mit Zeichnungen von einfachen planaren Graphen in der Ebene beschäftigen. Planare Graphen haben durch die Existenz kreuzungsfreier Einbettungen in gewissem Sinne besonders „schöne“ Zeichnungen. Aus diesem Grund ist eine der Fragen, mit der sich schon viele Mathematiker*innen auseinander gesetzt haben: „*How to draw a Graph?*“ [Tut63]. Mit dieser Frage werden wir uns auch im nun Folgenden beschäftigen.

Beginnen wir mit Varianten von Einbettungen, die wenige Einschränkungen haben. Bei einer topologischen Zeichnung eines planaren Graphen werden die Kanten als Kurven dargestellt. Hierbei dürfen sich die Kanten nur in den Knoten treffen. Erste mathematische Arbeiten hierzu zeigten, dass man diese Kurven auch als Geraden zeichnen kann. So wurde in den fünfziger Jahren des letzten Jahrhunderts unter anderem von István Fáry gezeigt, dass für jeden planaren Graphen und für jede Wahl eines äusseren Gebietes eine geradlinige und kreuzungsfreie Einbettung existiert [Fár48].

In den siebziger Jahren des letzten Jahrhunderts betrachtete William Thomas Tutte die Unterklasse der 3-zusammenhängenden planaren Graphen und zeigte, dass für diese nicht nur geradlinige, sondern *konvexe* Zeichnungen existieren [Tut63]. Bei einer konvexen Einbettung bilden die Kantenfolgen, die ein Gebiet einschließen, die Randkurven von konvexen Polygonen.

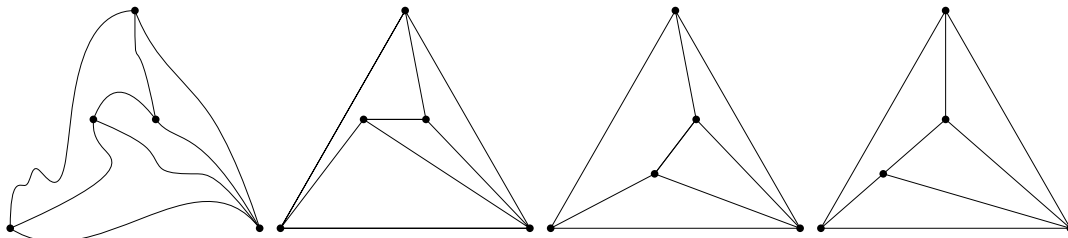


Abbildung 1.1 – Der gleiche planare Graph mit einer topologischen, einer geradlinigen und einer konvexen Darstellung und unten rechts einer geradlinigen Dreiecks-Darstellung.

Im Weiteren folgt eine Auseinandersetzung mit einer spezifischen Form dieser Zeichnungen nach Niek Aerts und Stefan Felsner. Wir fordern, dass alle Gebiete, inklusive dem Äußeren, nicht degenerierte Dreiecke einschließen. Wir nennen so eine Darstellung eine *geradlinige Dreiecks-Darstellung*. Nicht alle planaren Graphen haben solche Zeichnungen, wie wir im Verlauf der Arbeit sehen werden. Man kann die Klasse der Graphen, die eine geradlinige Dreiecks-Darstellung zulassen, jedoch eingrenzen und

Aerts und Felsner haben dies in einigen Publikationen getan [AF13a, AF13b, AF15].

Der Beitrag dieser Arbeit zum Thema liegt in der Implementierung eines von Aerts und Felsner entwickelten Algorithmus. Die Existenz einer geradlinigen Dreiecks-Darstellung wird in diesem als ganzzahlige Lösung eines Gerichteten-Multi-Fluss-Problems charakterisiert. Die Lösung eines solchen Problems ist im Allgemeinen NP-schwer. Experimentelle Rechenergebnisse legen aber den Schluss nahe, dass sich geradlinige Dreiecks-Darstellungen in polynomineller Zeit berechnen lassen. So kann man die Vermutung aufstellen, dass nicht-ganzzahlige Lösungen zu Berechnung von geradlinigen Dreiecks-Darstellungen genügen. Auch wenn wir den Beweis dieser Vermutung noch nicht gefunden haben, hilft sie bei der Beschleunigung des Algorithmus von Aerts und Felsner. Wir werden Indizien für die Korrektheit der Vermutung und mögliche Beweisansätze besprechen und zum Abschluss die Darstellung von errechneten geradlinigen Dreiecks-Darstellungen betrachten.

Es folgt ein kurzer Überblick der Struktur der Arbeit. In Kapitel 2 wiederholen wir zunächst zum Verständnis der Arbeit wichtige Resultate aus der Graphentheorie und Optimierung. Kapitel 3 befasst sich mit notwendigen und hinreichenden Bedingungen für die Existenz von geradlinigen Dreiecks-Darstellungen. In Kapitel 4 wird aufbauend auf diesen Ergebnissen ein Algorithmus entwickelt, um eine geradlinige Dreiecks-Darstellung eines planaren Graphen zu berechnen, falls diese existiert. Abschließend wird in Kapitel 5 ein Überblick über die Ergebnisse einer Implementierung des erarbeiteten Algorithmen gegeben.

2 Grundlagen

Wir werden uns in dieser Arbeit hauptsächlich mit einfachen planaren Graphen beschäftigen, also solchen, die keine Mehrfachkanten und Schleifen besitzen und für die kreuzungsfreie Zeichnungen, beziehungsweise Einbettungen, in der Ebene existieren. Sei $G = (V, E)$ ein Graph bestehend aus der Menge der Knoten V und der Menge der Kanten $E \subseteq (V \times V)$. Eine Kante $e = (u, v)$ verbindet die beiden Knoten u und v . Einen Graphen G , definiert als Kombination der Mengen von Knoten und Kanten, nennen wir auch einen *kombinatorischen* Graphen. Ein Pfad von u nach v ist eine Folge von Kanten $\{(u, w_1), (w_1, w_2), \dots, (w_k, v)\} \subset E$ die u und v verbindet. Mit dem Grad $\deg(v)$ eines Knoten meinen wir die Anzahl der adjazenten Kanten, der Kanten, die v als einen Endpunkt haben.

Eine Einbettung (oder Zeichnung) eines Graphen in der Ebene ist eine Abbildung $m : G \rightarrow \mathbb{R}^2$, die die Kanten und Knoten kreuzungsfrei in die Ebene zeichnet. Einen planaren Graphen zusammen mit einer möglichen kreuzungsfreien Einbettung in der Ebene bezeichnen wir als *ebenen Graphen*. Für einen ebenen Graphen können wir, zusätzlich zu den Knoten und Kanten, auch die Menge der Gebiete (engl. faces) F betrachten, das heißt, die durch die Kanten und Knoten begrenzten Regionen in der Ebene. Wir bezeichnen das unbeschränkte Gebiet als das *äußere* Gebiet. Wir können den Begriff der Einbettung von G auch etwas weiter fassen und mit diesem nur die Definition der Menge der Gebiete F von G und eines äußeren Gebietes $f_{aus} \in F$ meinen, sodass mindestens eine planare Zeichnung mit diesen Gebieten existiert. Wir nennen dies auch eine *kombinatorische* Einbettung. Die kombinatorische Einbettung bildet eine Äquivalenzklasse aller möglichen Zeichnungen von G . In ihr enthalten sind alle Zeichnungen von G , deren Gebiete (inklusive dem Äußeren) denen der kombinatorischen Einbettung entsprechen. Wir werden im Verlauf der Arbeit, wenn wir von Einbettung reden, manchmal eine spezifische Zeichnung und manchmal ihre Äquivalenzklasse meinen, die Bedeutung sollte sich jedoch aus dem Kontext ergeben.

2.1 Geradlinige Dreiecks-Darstellungen (SLTRs)

Wir werden uns in dieser Arbeit mit Möglichkeiten auseinandersetzen, geradlinige Dreiecks-Darstellungen für planare Graphen zu finden. Die englische Bezeichnung der geradlinigen Dreiecks-Darstellungen lautet straight line triangle representation, kurz SLTR. Die nächste Definition formalisiert diese Darstellung.

Definition 2.1 (SLTR). Eine Zeichnung eines ebenen Graphen G wird *SLTR* genannt, falls gilt:

- S1 Alle Kanten sind Segmente von Geraden.

S2 Alle Gebiete, inklusive dem Äußeren, sind nicht-degenerierte Dreiecke.

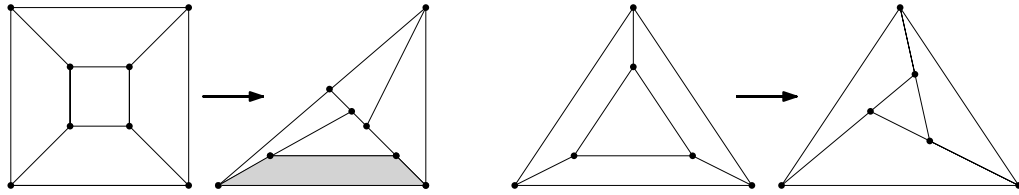


Abbildung 2.1 – Links einer der beiden 3-zusammenhängenden Graphen mit acht Knoten ohne SLTR und rechts ein Graph mit einer möglichen SLTR.

Für die weiteren Betrachtungen ist es nützlich die drei Knoten $\{a_1, a_2, a_3\}$, die das äußere Gebiet berühren, gesondert zu betrachten. Wir nennen sie die *Aufhängungen* von G . Die Knoten a_1, a_2 und a_3 sind dann die designierten Ecken des äußeren Gebietes einer möglichen SLTR. Einen Graphen zusammen mit einem äußeren Gebiet und festen Aufhängungen als Paar zu behandeln ist sinnvoll, wie in Beispiel 2.2 zu sehen sein wird.

Beispiel 2.2. Es existieren planare Graphen, von denen manche Einbettungen SLTRs zulassen, andere jedoch nicht. Betrachten wir den planaren Graphen mit zehn Knoten aus Abbildung 2.2. Mit rot und grün sind die beiden Gebiete markiert, die jeweils einmal als das äußere Gebiet festgelegt wurden. Die kombinatorische Einbettung auf der rechten Seite lässt zu dieser Wahl des äußeren Gebietes keine SLTR zu. Das nicht dreieckige Gebiet ist grau eingefärbt. Zur Auswahl auf der linken Seite existiert hingegen eine. Dies ist der kleinste 3-zusammenhängende kombinatorische Graph, der diese Eigenschaft hat.

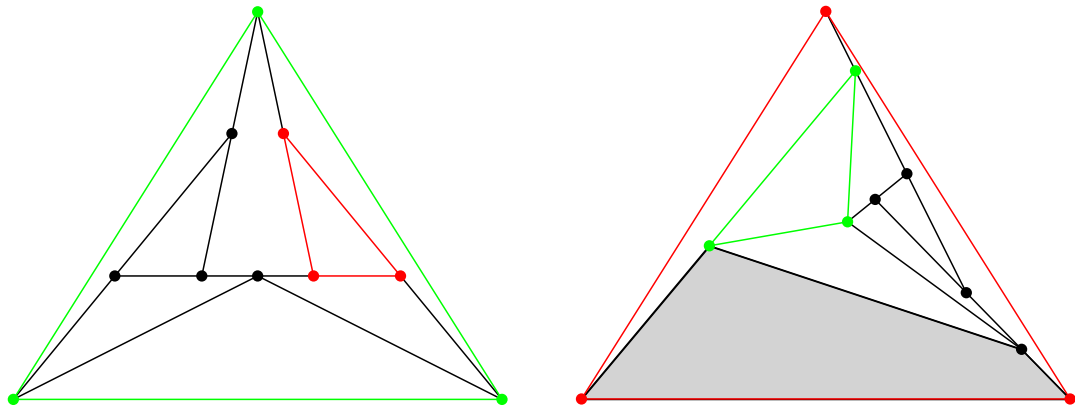


Abbildung 2.2 – Zwei topologische Einbettungen des (kombinatorisch) gleichen planaren Graphen, wobei die linke keine SLTR zulässt. Die jeweils gewählten äußeren Gebiete sind rot und grün umrandet.

Bevor wir zur ersten Proposition kommen werden wir die Klasse der planaren

Graphen, die wir betrachten wollen, etwas weiter einschränken. Dabei hilft uns die nächste Definition.

Definition 2.3 (intern- k -zusammenhängend). Ein Graph G ist *zusammenhängend*, falls für alle Knoten u, v ein Pfad von u nach v existiert. G ist *k -zusammenhängend*, falls er nach der Entfernung von $k - 1$ beliebigen Knoten weiterhin zusammenhängend ist. Sei G eben mit den Aufhängungen $\{a_1, a_2, a_3\}$, weiter sei a_∞ ein zusätzlicher Knoten eingefügt im äußeren Gebiet. Dann ist G *intern k -zusammenhängend*, falls $G + a_\infty := (V \cup \{a_\infty\}, E \cup \{(a_1, a_\infty), (a_2, a_\infty), (a_3, a_\infty)\})$ k -zusammenhängend ist.

Die nächste Präposition enthält eine erste notwendige Bedingung für die Existenz von geradlinigen Dreiecksdarstellungen (SLTRs).

Proposition 2.4. [AF13b, Proposition 1.2] Sei G ein ebener Graph mit den Aufhängungen $\{a_1, a_2, a_3\}$ als äußere Ecken einer SLTR. Weiter gelte für alle Knoten $v \in V \setminus \{a_1, a_2, a_3\}$ $\deg(v) \geq 3$. Dann ist G intern-3-zusammenhängend.

Beweis: Sei Δ die SLTR von $G = (V, E)$. Angenommen es existiert eine Menge $U \subseteq V$ in G mit $|U| = 2$, deren Entnahme G in nicht zusammenhängende Komponenten trennt. Wir werden zeigen, dass jeder Teil von $G \setminus U$ eine der Aufhängungen enthält und somit $G + a_\infty$ nicht von U getrennt wird. Da G intern-3-zusammenhängend ist, können nur die Aufhängungen a_i Grad 2 haben. Sei K eine der Komponenten von $G \setminus U$. Betrachte $K \cup U$ als induzierten Teilgraphen von G . Falls $K \cup U$ ein Pfad ist, also nur Knoten von Grad 1 oder 2 enthält, dann kann nur $K = \{a_i\}$ gelten.

Falls $K \cup U$ kein Pfad ist, betrachte die konvexe Hülle von $U \cup K$ in Δ . Mindestens drei der Ecken von $U \cup K$ haben Außenwinkel grösser als π (an diesen Ecken befinden sich Knoten aus $U \cup K$). Zwei dieser Winkel können an den Knoten aus U liegen, aber der dritte muss ein Winkel sein, der schon in Δ existiert. Es handelt sich somit um eine Aufhängung. \square

Bemerkung. Für innere Knoten von Grad 2 in einer SLTR müssen die beiden angrenzenden Winkel gerade sein. Somit kann man diese Knoten durch eine gerade Kante zwischen ihren Nachbarn ersetzen und den resultierenden Graphen betrachten. Knoten von Grad 1 können nicht existieren, da sie nicht aus dem Rand eines Dreiecks liegen können. Wir werden somit von nun an nur intern-3-zusammenhängende Graphen mit Aufhängungen betrachten, da alle anderen Graphen, die eine SLTR zulassen, auf diese reduziert werden können.

Für 3-zusammenhängende planare Graphen mit mehr als drei Knoten ist die kombinatorische Einbettung nach Hassler Whitney bis auf Wahl des äußeren Gebietes eindeutig [Whi33]. Zusammen mit der nächsten Proposition definiert somit die Wahl von Aufhängungen $\{a_1, a_2, a_3\}$, die in mindestens einem gemeinsamen Gebiet liegen, ein eindeutiges äußeres Gebiet f_{aus} .

Proposition 2.5. Sei G ein planarer 3-zusammenhängender Graph und $\{a_1, a_2, a_3\} \subset V$. Dann existiert höchstens ein Gebiet f , das adjazent zu a_1, a_2, a_3 ist.

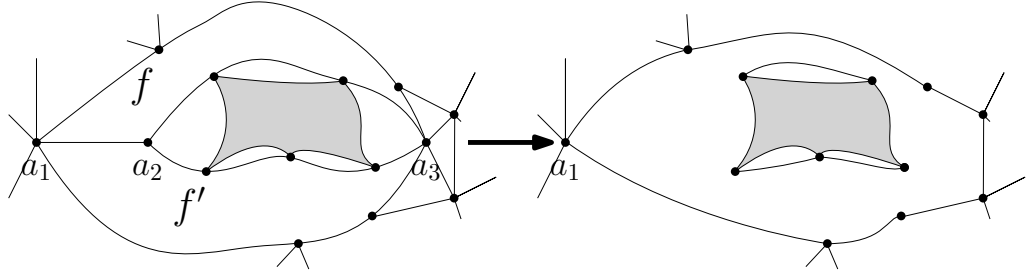


Abbildung 2.3 – Die Entnahme der Knoten a_2 und a_3 trennt G , was zu einem Widerspruch zum 3-Zusammenhang führt.

Beweis: Angenommen es existieren zwei Gebiete f und f' , die a_1, a_2 und a_3 enthalten. Dann berühren sich diese Gebiete an mehr als einer Kante (vergleiche Abbildung 2.3 links). Im grauen Gebiet müssen Knoten liegen, da G 3-zusammenhängend ist. Wenn wir aber nun a_2 und a_3 entfernen, dann teilen wir G in zwei nicht zusammenhängende Gebiete. Es kann somit nur ein Gebiet f geben, in welchem die a_1, a_2 und a_3 liegen. \square

Bemerkung. Wir könnten uns somit für 3-zusammenhängende planare Graphen auf die Wahl von Aufhängungen $\{a_1, a_2, a_3\}$ beschränken. Für eine kombinatorische Einbettung folgt dann ein eindeutiges äußeres Gebiet f_{aus} . Für nur intern-3-zusammenhängende planare Graphen gilt dies nicht. Wir betrachten somit im Folgenden ebene Graphen mit Aufhängungen $\{a_1, a_2, a_3\}$ im äußeren Gebiet.

Mit den Fragen, welche notwendigen und hinreichenden Bedingungen für die Existenz von SLTRs gelten und welche algorithmischen Ansätze man bei der Suche nach einer spezifischen Darstellung verfolgen kann, werden wir uns in den nächsten beiden Kapiteln auseinandersetzen. Zuvor werden in diesem Kapitel noch einige Konzepte eingeführt, die notwendig sind, um der Argumentation zu folgen.

2.2 Schnyder Woods

Betrachten wir einen ebenen Graphen mit Aufhängungen a_1, a_2, a_3 . Anschaulich handelt es sich bei einem Schnyder Wald um drei aufspannende Bäume T_1, T_2, T_3 , sodass jeder der Bäume T_i zu seiner Wurzel a_i hin gerichtet ist. Jede der Kanten wird mindestens von einem der Bäume genutzt und Kanten können von zwei der drei Bäume gleichzeitig genutzt werden.

Schnyder Wälder, im weiteren nach der englischen Bezeichnung *Schnyder Woods*, wurden zuerst von Walter Schnyder eingeführt [Sch89]. Es handelt sich um eine Färbung und Orientierung der inneren Kanten einer Triangulierung. Sie dienen der Betrachtung der von planaren Graphen induzierten Ordnungen. In einer weiteren Arbeit wurden mit ihrer Hilfe geradlinige und konvexe Einbettungen Triangulierungen auf einem $(|V| - 2) \times (|V| - 2)$ Gitter erzeugt [Sch90].

Wir wollen hier die Verallgemeinerung auf 3-zusammenhängende planare Graphen durch Felsner und die zu ihnen in Bijektion stehenden Schnyder Labelings einführen [Fel01]. Wir orientierten uns dabei an einem Lehrbuch von Felsner [Fel12]. Für den Rest dieses Kapitels sei G , wenn nicht weiter spezifiziert, ein 3-zusammenhängender ebener Graph mit Aufhängungen $\{a_1, a_2, a_3\}$.

Definition 2.6 (Schnyder Woods). Ein Schnyder Wood ist eine Orientierung und Beschriftung der Kanten von G mit den Labeln 1, 2 und 3¹ (alternativ wird hier auch oft rot, grün und blau genutzt), unter Berücksichtigung der folgenden Regeln:

- W1 Jede Kante ist entweder in eine oder zwei Richtungen orientiert. Falls sie bigerichtet ist, haben beide Richtungen unterschiedliche Label.
- W2 An jeder Aufhängung a_i existiert eine nach außen gerichtete Halbkante² mit Label i .
- W3 Jeder Knoten v hat Ausgangsgrad 1 in jedem Label. Um v existieren im Uhrzeigersinn eine Auskante mit Label 1, null oder mehr eingehende Kanten mit Label 3, eine Auskante mit Label 2, null oder mehr eingehende Kanten mit Label 1, eine Auskante mit Label 2 und null oder mehr eingehende Kanten mit Label 2.
- W4 Es existiert kein inneres Gebiet mit einem gerichteten Zykel in einer Farbe als Rand.

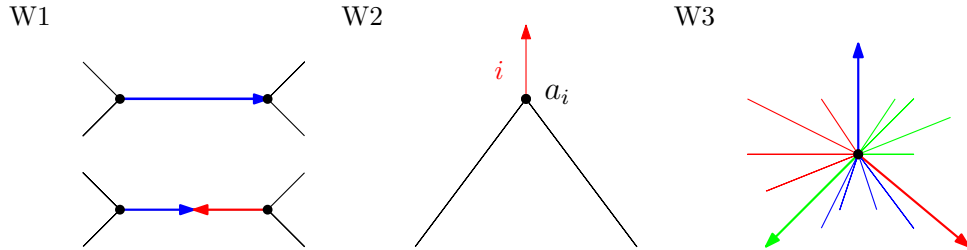


Abbildung 2.4 – Illustration der ersten drei Bedingungen für Schnyder Woods.

Analog zu den Schnyder Woods kann man Schnyder Labelings definieren, die zu diesen in Bijektion stehen. Hier betrachten wir nicht zuerst die Kanten eines planaren Graphen, sondern die Winkel an den Knoten.

Definition 2.7 (Schnyder Labeling). Ein Schnyder Labeling ist eine Beschriftung der Winkel von G mit den Labeln 1, 2 und 3 (oder rot, grün und blau) unter Berücksichtigung der folgenden Regeln:

- L1 Um jedes innere Gebiet bilden die Label im Uhrzeigersinn nichtleere Intervalle von $1en$, $2en$ und $3en$. Am äußeren Gebiet gilt dies gegen den Uhrzeigersinn.

¹Es wird davon ausgegangen, dass die Label zyklisch sortiert sind, sodass $i + 1$ und $i - 1$ immer definiert sind.

²Eine Halbkante ist eine Kante mit nur einem Endpunkt.

- L2 Um jeden inneren Knoten bilden die Label im Uhrzeigersinn nichtleere Intervalle von 1en, 2en und 3en.
- L3 An den Aufhängungen a_i haben die äußeren Winkel die Label $i-1$ und $i+1$ im Uhrzeigersinn mit der Halbkante dazwischen. Die inneren Winkel haben das Label i .

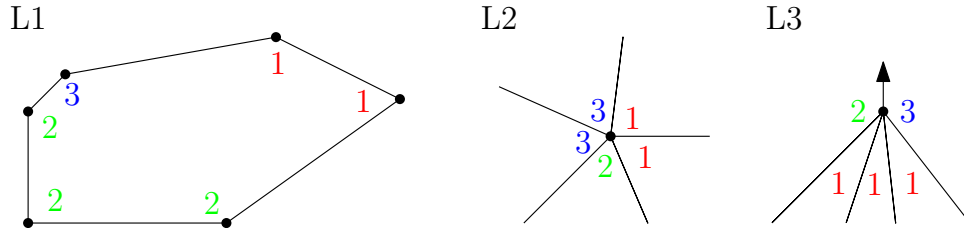


Abbildung 2.5 – Illustration der drei Bedingungen für Schnyder Labelings.

In Abbildung 2.6 wird eine Verbindung zwischen Schnyder Woods und Schnyder Labelings illustriert. Das nächste Lemma folgt aus den Bedingungen L1 und L2.

Lemma 2.8. *Sei G ein ebener, intern-3-zusammenhängender Graph mit den Aufhängungen a_1, a_2, a_3 und einem Schnyder Labeling. Dann beinhalten die vier Winkel entgegen dem Uhrzeigersinn an jeder Kante die Label 1, 2 und 3. Somit hat jede Kante einen der beiden Typen in Abbildung 2.6.*

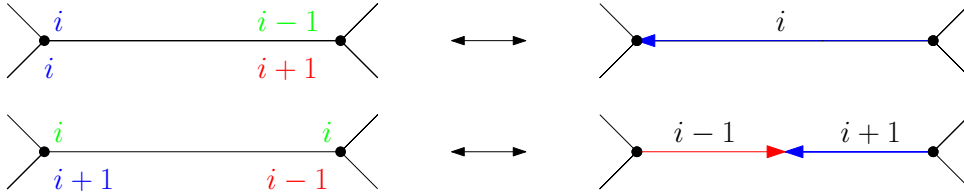


Abbildung 2.6 – Bijektion zwischen Schnyder Wood auf der rechten und Schnyder Labeling auf der linken Seite.

Theorem 2.9. *Sei G ein ebener, intern-3-zusammenhängender Graph mit den Aufhängungen a_1, a_2, a_3 . Der in Abbildung 2.6 dargestellte Zusammenhang erzeugt eine Bijektion zwischen Schnyder Woods und Schnyder Labelings auf G .*

Dies ermöglicht es zwischen den beiden Strukturen hin und her zu wechseln, wenn es für die Darstellung und für das Verständnis sinnvoll ist. So kann es im Verlauf dieser Arbeit vorkommen, dass wir vom einen schreiben, aber implizit Eigenschaften des anderen meinen. Für die von uns im Folgenden betrachteten Graphen existiert mindestens ein Schnyder Wood. Dies belegt das nächste Theorem nach Ezra Miller [Mil02, Theorem A].

Theorem 2.10. *Sei G ein ebener Graph mit Aufhängungen $\{a_1, a_2, a_3\}$. G ist genau dann intern-3-zusammenhängend, wenn ein Schnyder Wood auf G mit den Ecken $\{a_1, a_2, a_3\}$ existiert.*

2.2.1 Einbettungen via Schnyder Woods

Es existieren einige Anwendungen von Schnyder Woods in Bezug auf Einbettungen. Wie schon erwähnt, bezieht sich eines der ersten Resultate auf die konvexe Einbettung auf einem Gitter. Das im Folgenden skizzierte *face-counting* nach Felsner (deutsch: Gebiete zählen) erzeugt eine Einbettung auf einem kleineren Gitter als nach Schnyder [Fel01]. Betrachte G mit einem Schnyder Wood T_1, T_2, T_3 . Es handelt es sich bei den Bäumen T_i um gerichtete Bäume mit Wurzeln in a_i [Fel12, Korollar 2.5]. Von jedem Knoten v aus existieren also eindeutige Pfade $P_i(v)$ zu den Aufhängungen a_i . Jeweils zwei der Pfade von v zu den Aufhängungen haben v als einzigen gemeinsamen Knoten [Fel12, Lemma 2.4]. Wir können somit zu jedem Knoten v drei Regionen R_i definieren, die jeweils von den Pfaden $P_{i-1}(v)$ und $P_{i+1}(v)$ und dem äußeren Gebiet eingegrenzt werden³. In jeder dieser Regionen können wir nun die eingeschlossenen Gebiete von G zählen. Durch das Zählen der Gebiete in den Regionen zu v lässt sich eine konvexe Zeichnung von G erzeugen.

Hierzu ordnet man jedem Knoten v seinen *Gebietsvektor* (v_1, v_2, v_3) zu, wobei v_i die Anzahl der inneren Gebiete in $R_i(v)$ beschreibt. In Abbildung 2.7 sind auf der rechten Seite die Regionen von v eingefärbt. Seien $\alpha_1 = (0, 1)$, $\alpha_2 = (1, 0)$ und $\alpha_3 = (0, 0)$ die äußeren Ecken unserer Zeichnung. Sie entsprechen ebenfalls den Bildern der Aufhängungen von G . Die Position der inneren Knoten ergibt sich nun durch die Funktion

$$\mu : V \rightarrow \mathbb{R}^2, v \mapsto v_1\alpha_1 + v_2\alpha_2 + v_3\alpha_3.$$

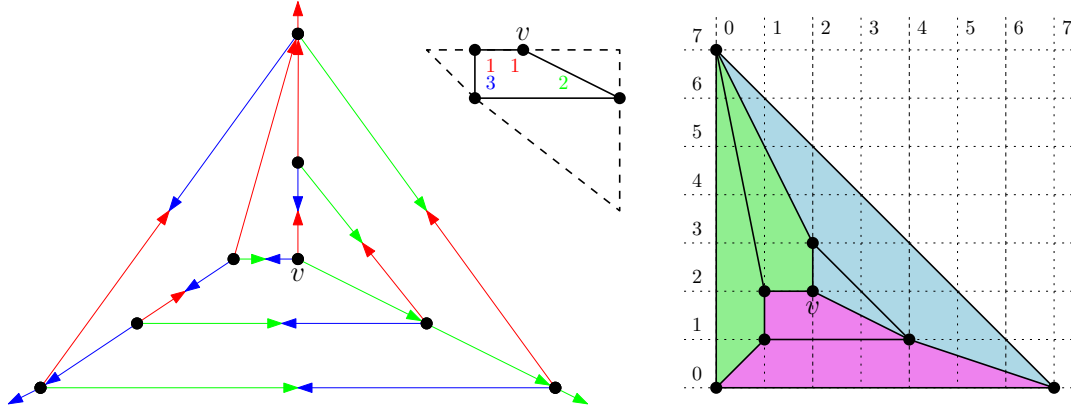


Abbildung 2.7 – Ein Schnyder Wood auf G und die durch face-counting erhaltene Einbettung von G . Die eingefärbten Gebiete sind die Regionen, die den Gebietsvektor (v_1, v_2, v_3) ergeben. In der Mitte ist W5 illustriert.

³Ein Beispiel für diese Regionen findet sich in Abbildung 2.7 auf der rechten Seite.

Theorem 2.11 (Theorem 2.7 [Fel12]). *Sei G ein ebener Graph mit einem Schnyder Wood σ und den Aufhängungen a_1, a_2, a_3 und (v_1, v_2, v_3) der unter Berücksichtigung von σ berechnete Gebietsvektor von $v \in V$, mit $v_i = \text{Anzahl der Gebiete in } R_i(v)$, dann ist die Zeichnung $\mu(G)$ konvex.*

Die mit diesen Koordinaten erzeugte Einbettung von G ist planar, konvex und passt, falls G 3-zusammenhängend ist, auf ein $(|F| - 1) \times (|F| - 1)$ -Gitter [Fel12, Korollar 2.8]. Sie hat noch eine weitere Eigenschaft, die später von Nutzen sein wird und die in der nächsten Proposition festgehalten ist. Dies ist in Abbildung 2.7 in der Mitte skizziert.

Proposition 2.12. *Sei G ein ebener Graph mit einem Schnyder Wood σ und den Aufhängungen a_1, a_2, a_3 . Für $\mu(G)$, die durch face-counting erzeugte Einbettung von G , gilt:*

W5 Die Knoten eines inneren Gebietes werden auf die Seiten eines Dreiecks mit den Seiten $c_i(\alpha_{i-1} - \alpha_{i+1})$ mit passenden Konstanten c_i abgebildet. Im Inneren dieses Dreiecks befinden sich keine Knoten. Die Winkel im Inneren des Gebietes an den Knoten auf der Seite $c_i(\alpha_{i-1} - \alpha_{i+1})$ haben Label i im Schnyder Labeling.

2.3 α -Orientierungen

Für den Algorithmus in Kapitel 4 führen wir nun eine weitere zu Schnyder-Woods und Labelings in Bijektion stehende Struktur auf Graphen ein und folgen dabei Felsner [Fel04].

Definition 2.13 (α -Orientierung). Sei $G = (V, E)$ ein ungerichteter Graph und $\alpha : V \mapsto \mathbb{N}$ eine Funktion auf G . Eine α -Orientierung ist eine Orientierung der Kanten von G , sodass der Ausgangsgrad⁴ eines jeden Knoten $\alpha(v)$ entspricht, dass heißt,

$$\text{outdeg}(v) = \alpha(v).$$

Um von α -Orientierungen zu Schnyder Woods zu gelangen, müssen wir Primal-Dual Graphen betrachten, die mit den nächsten beiden Definitionen eingeführt werden.

Definition 2.14 (schwacher dualer Graph). Sei G ein ebener Graph. Wir definieren G^* , den *schwachen dualen Graphen* von G . G^* hat einen (Gebiets-)Knoten für jedes innere Gebiet von G . Für jede innere Kante in G fügen wir eine Kante zwischen den beiden (Gebiets-)Knoten f, f' in G^* ein, die adjazent zu dieser Kante in G sind.

Definition 2.15 (Primal-Dual Graph). Betrachte einen ebenen Graphen G und seinen schwachen dualen Graphen G^* . Der *Primal-Dual Graph* $G + G^*$ ist eine Vereinigung der Graphen G und G^* . Wenn wir G und G^* übereinander legen, dann kreuzen sich jeweils eine Kante aus G und eine aus G^* . An so einer Kantenkreuzung fügen wir

⁴Der Ausgangsgrad eines Knoten v entspricht der Anzahl der adjazenten Kanten von v , die von v weg gerichtet sind.

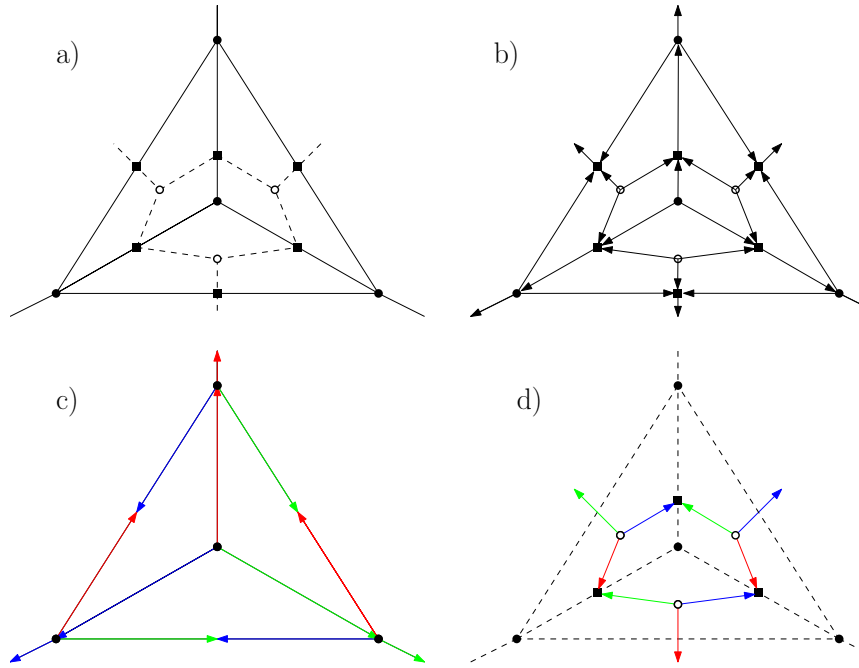


Abbildung 2.8 – a) Der Primal-Duale Graph $K_4 + K_4^*$. b) Mit einer α_s -Orientierung und c) den zugehörigen Schnyder Woods auf K_4 d) und K_4^* .

in $G + G^*$ einen Knoten ein und verbinden ihn mit den adjazenten Knoten aus G und G^* . Die Menge der Knoten von $G + G^*$ besteht aus Knoten-Knoten (Knoten in G), Kanten-Knoten (an den Kreuzungen) und Gebiets-Knoten (Knoten in G^*). Kanten in $G + G^*$ existieren, sowohl zwischen inzidenten Kanten und Knoten, als auch Kanten und Gebieten in G . Hinzu kommen Halbkanten von den Kanten-Knoten und Knoten-Knoten am äußeren Gebiet von G . Wenn wir einen Knoten f_∞ für das äußere Gebiet hinzufügen und die Halbkanten zu diesem verlängern, spricht man vom *Abschluss* von $G + G^*$. Ein Beispiel ist in Abbildung 2.8 a) zu sehen.

Wir trennen bei der Erzeugung jede Kante in zwei Teile. Die Kanten-Knoten auf einer Seite und die Knoten-Knoten und Gebiets-Knoten auf der anderen Seite bilden eine Bipartition. Somit sind $G + G^*$ und sein Abschluss bipartit. Das folgende Theorem liefert eine Bijektion zwischen den Schnyder Woods auf G und einer bestimmten α -Orientierung auf dem Abschluss von $G + G^*$, die wir α_s nennen [Fel04, Propositionen 3 und 4].

Theorem 2.16. *Sei G ein ebener Graph mit Aufhängungen $\{a_1, a_2, a_3\}$, dann stehen die folgenden Strukturen in Bijektion:*

A1 Die Schnyder Woods auf G .

A2 Die Schnyder Woods auf dem (schwachen) dualen Graphen G^* .

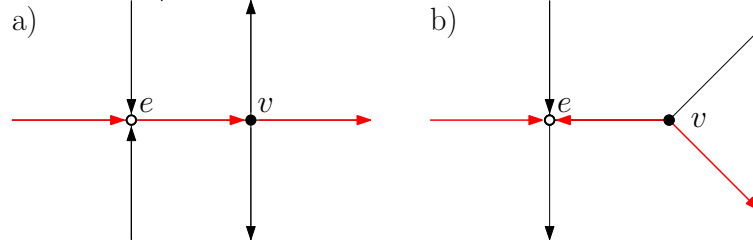


Abbildung 2.9 – Die zwei Fälle der Gerader Pfad Regel beim Betreten eines Knoten f aus G oder G^* . In rot der gewählte und somit eingefärbte Pfad.

A3 Die α_s -Orientierungen des Abschlusses von $G + G^$ mit $\alpha_s(v) = \alpha_s(f) = 3$ für jeden Knoten-Knoten v und Gebiets-Knoten f , $\alpha_s(e) = 1$ für jeden Kanten-Knoten e und $\alpha_s(f_\infty) = 0$.*

Wir erklären kurz die erhaltene Bijektion von A1 und A2 zu A3. Angenommen wir haben ein Paar in Bijektion stehender Schnyder Woods σ, σ^* auf G und G^* . Die von σ und σ^* induzierte Orientierung der Kanten von $G + G^*$ ist eine α_s -Orientierung.

Angenommen wir haben eine α_s -Orientierung auf dem Abschluss von $G + G^*$. Sie induziert die Orientierung der Kanten der resultierenden Schnyder Woods σ und σ^* . Wir müssen jedoch noch die Label bestimmen. Dies erfolgt mit Hilfe der Gerader Pfad Regel.

Gerader Pfad Regel: Beginnen wir auf einer beliebigen Kante von $G + G^*$. Wenn wir einen Kanten-Knoten e betreten, dann verlassen wir diesen auf der gegenüberliegenden Seite (wir folgen der zugrunde liegenden Kante in G bzw. G^*). Wenn wir einen Knoten-Knoten oder Gebiets-Knoten v betreten und über eine Kante kommen, die zu v orientiert ist, dann laufen wir auf der gegenüberliegenden, von v weg orientierten Kante weiter⁵ (vergleiche Abbildung 2.9 a). Falls wir v über eine von v weg orientierte Kante betreten, dann hängt unsere Wahl der Auskante von der Orientierung am letzten Kanten-Knoten e ⁶ ab. Falls dort die Auskante auf der rechten Seite liegt, wählen wir jetzt die rechte⁷, von v weg orientierte Kante, sonst die linke (vergleiche Abbildung 2.9 b). Falls wir a_∞ erreichen, laufen wir nicht weiter.

Betrachten wir einen Pfad, der mit einer gerichteten Kante $e \in G + G^*$ beginnt, und mit der Gerader Pfad Regel fortfährt. Dann endet er in a_∞ [Fel01, Lemma 15]. Der letzte passierte Knoten muss nun entweder eine der Aufhängungen von G oder G^* sein. Wir färben den Pfad in der Farbe dieser Aufhängung. Die so erhaltene Orientierung und Färbung auf den Kanten von G und G^* entspricht zwei in Bijektion stehenden Schnyder Woods σ und σ^* [Fel04].

⁵Es existieren genau drei von v weg orientierte Kanten und wir wählen die von uns aus gesehen Mittlere.

⁶Wir betreten Kanten-Knoten nur über zu diesen hin orientierte Kanten.

⁷Falls wir v durch eine von v weg orientierte Kante betreten, dann bleiben noch zwei andere übrig.

2.4 Gerichtete-Multi-Fluss-Probleme

Wir werden in Kapitel 4 einen gerichteten Graphen \mathcal{N} auf Basis von G konstruieren, sodass ein maximaler Fluss φ einer SLTR von G entspricht. Es gibt viele unterschiedliche Arten von Flussproblemen. So kann man zum Beispiel Graphen mit nur einem Paar von Quellen und Senken oder mit jeweils mehreren Paaren betrachten und die Kanten können gerichtet oder ungerichtet sein. Im Fall von mehreren Quellen und Senken werden diese normalerweise als Paare s_i, t_i gehandhabt und es wird gefordert, dass insgesamt Fluss φ_i mit Stärke $d_i \in \mathbb{R}_+$ von s_i zu t_i fließt. Als zusätzliche Einschränkung haben die Kanten e Kapazitäten $c(e) \in \mathbb{R}_+$, die nicht überschritten werden dürfen. Für jede Kante muss also gelten $\varphi(e) \leq c(e)$. Wir werden uns in Kapitel 4 mit einem Fluss der folgenden Form befassen.

Definition 2.17 (Gerichtetes-Multi-Fluss-Problem). Sei $\mathcal{N} = (V, E)$ ein gerichteter Graph, im Weiteren auch Netzwerk genannt, mit den Kapazitäten $c : E \mapsto \mathbb{R}_+$, Paaren von ausgezeichneten Knoten $\{(s_1, t_1), \dots, (s_n, t_n)\} \subset V \times V$ und positiven Bedarfen $\{d_1, \dots, d_n\} \in \mathbb{R}_+^n$, dann bilden die Funktionen $\varphi_i : E \rightarrow \mathbb{R}_+^{|E|}$ einen zulässigen Fluss $\varphi = (\varphi_1, \dots, \varphi_n)$ auf \mathcal{N} , falls

$$\text{M1 } \forall (u, v) \in E : \sum_{i=1}^n \varphi_i(u, v) \leq c(u, v)$$

$$\text{M2 } \forall u \neq s_i, t_i : \sum_{w \in V} \varphi_i(u, w) - \sum_{w \in V} \varphi_i(w, u)$$

$$\text{M3 } \forall s_i : \sum_{w \in V} \varphi_i(s_i, w) - \sum_{w \in V} \varphi_i(w, s_i) = d_i$$

$$\text{M4 } \forall t_i : \sum_{w \in V} \varphi_i(w, t_i) - \sum_{w \in V} \varphi_i(t_i, w) = d_i$$

Mit $|\varphi_i|$ bezeichnen wir die Menge des Flusses von s_i nach t_i .

Definition 2.18. Wir nennen die Kantenmenge $S \subseteq \mathcal{N}_G(E)$ einen *Schnitt* in \mathcal{N}_G , falls seine Entnahme alle Paare $\{s_i, t_i\}$ trennt. Die Kantenmenge S ist ein *minimaler Schnitt*, falls für alle anderen Schnitte \tilde{S} gilt: $c(S) \leq c(\tilde{S})$.

Es folgen zwei bekannte Resultate für den Fall $n = 1$. Das erste Theorem stammt von Ford und Fulkerson [FF56].

Theorem 2.19 (Max-Flow Min-Cut). *Der maximale zulässige Fluss auf einem Netzwerk \mathcal{N} mit einer Quelle und Senke entspricht der Kapazität eines minimalen Schnittes.*

Ein maximaler Fluss lässt sich zum Beispiel mit dem Edmonds-Karp-Algorithmus in polynomineller Zeit bestimmen [KV18, Theorem 8.15]. Ein direktes Resultat folgt nach Danzig und Fulkerson [FF56]

Theorem 2.20 (Ganzzahliger Fluss). *Sei \mathcal{N} ein Netzwerk mit einer Quelle und einer Senke und ganzzahligen Kapazitäten $c : E \mapsto \mathbb{N}$. Sei $\tilde{\varphi}$ ein nicht ganzzahliger Fluss auf \mathcal{N} . Dann existiert auch ein Fluss φ , mit $|\tilde{\varphi}| = |\varphi|$, sodass der Fluss φ auf allen Kanten ganzzahlig ist.*

Nach dem Max-Flow Min-Cut Theorem impliziert die Kapazität eines minimalen Schnittes im eindimensionalen Fall die Stärke eines maximalen Flusses. Für $n = 2$ existieren für ungerichtete Graphen analoge Aussagen nach T. Chiang Hu [Hu63]. Für diese Arbeit wäre im Folgenden jedoch der Fall $n = 2$ für gerichtete Graphen interessant. Es existieren jedoch keine analogen Aussagen zum Max-Flow Min-Cut Theorem, welche für Gerichtete-Multi-Fluss-Probleme mit mindestens zwei Quellen und Senken aussagekräftig wären, sondern nur Schranken und Annäherungen [LR99].

Im eindimensionalen Fall mit ganzzahligen Kapazitäten $c : E \mapsto \mathbb{N}$ impliziert die Existenz eines zulässigen Flusses nach Theorem 2.20 die Existenz einer ganzzahligen Lösung, sowohl für gerichtete als auch für ungerichtete Graphen. Die Unterscheidung zwischen ganzzahligen und nicht ganzzahligen Flüssen ist relevant, da das Entscheidungsproblem, ob ein solcher Fluss existiert, im mehrdimensionalen Fall in unterschiedlichen Komplexitätsklassen liegt. Die Berechnung eines nicht (zwangsläufig) ganzzahligen zulässigen Flusses φ auf \mathcal{N} ist für den ein- und mehrdimensionalen Fall über eine LP-Formulierung in polynomieller Zeit möglich [KV18, Theorem 4.18]. Für ganzzahlige Lösungen gilt dies selbst in einfachen Fällen im Allgemeinen nicht und somit kann auch keine analoge Formulierung zu Theorem 2.20 existieren.

Theorem 2.21 ([EIS75]). *Die Berechnung einer ganzzahligen Lösung eines Gerichtetes-Multi-Fluss-Problems auf einem Netzwerk \mathcal{N} mit zwei Paaren von Quellen und Senken $\{(s_1, t_1), (s_2, t_2)\}$ und Kapazitäten $c : E \mapsto \mathbb{N}$ ist sogar dann NP-schwer, wenn \mathcal{N} keine gerichteten Zykel enthält.*

NP-schwere Entscheidungsprobleme lassen sich nicht in polynomieller Zeit lösen. Somit existiert kein im Allgemeinen gültiger deterministischer polynomieller Algorithmus zur Berechnung eines zulässigen ganzzahligen Flusses für ein Gerichtetes-Multi-Fluss-Problem für den Fall $n \geq 2$.

3 Bedingungen für die Existenz von SLTRs

Wir werden in diesem Kapitel weitere notwendige und zwei hinreichende Bedingungen für die Existenz von SLTRs erarbeiten. Als ersten Schritt wollen wir jedoch eine notwendige Eigenschaft festhalten.

Sei G ein ebener intern 3-zusammenhängender Graph mit Aufhängungen $\{a_1, a_2, a_3\}$. Nehmen wir für einen Moment an, dass wir schon eine SLTR für G gefunden haben. Für die Zeichnung müssen die folgenden Bedingungen gelten:

- Jeder Knoten $v \in V$ hat in maximal einem inzidenten Gebiet f einen flachen Winkel, also Winkel π .
- Jedes Gebiet $f \in F$ (inklusive dem Äußeren) hat genau drei Ecken. Es muss also in jedem Gebiet $|f| - 3$ flache Winkel an Knoten geben¹.

Diese Bedingungen liefern eine notwendige Bedingung für die Existenz einer SLTR. Sie ist in der nächsten Definition festgehalten.

Definition 3.1 (FAA). Sei $G = (V, E, F)$ ein ebener Graph. Eine *flache Winkel Zuweisung* ϕ , kurz *FAA* (nach dem englischen *flat angle assignment*), ist eine partielle Zuordnung der Knoten zu den Gebieten. Es muss gelten:

F1 Jedem Gebiet f sind genau $|f| - 3$ Knoten zugewiesen.

F2 Jeder Knoten v ist höchstens einem Gebiet zugewiesen.

Für den Fall, dass Aufhängungen gegeben sind, fordern wir zusätzlich:

F3 Die inzidenten Knoten des äußeren Gebietes, die keine Aufhängungen sind, müssen dem äußeren Gebiet zugewiesen sein.

Beispiel 3.2. Links und rechts in Abbildung 3.6 sehen wir denselben planaren Graphen G mit zwei unterschiedlichen Zeichnungen. Das äußere Gebiet ist bei beiden dasselbe. Die Pfeile stellen hier die Zuweisung der Knoten zu den inneren Gebieten dar (also das FAA). Auf der rechten Seite sehen wir eine SLTR von G mit dem passenden FAA. Auf der linken Seite ist ein FAA eingezeichnet, das keine SLTR induziert.

Somit induziert jede SLTR genau ein FAA. Ein ebener Graph kann also nur dann eine SLTR besitzen, wenn mindestens ein FAA existiert. Wie in Beispiel 3.2 gezeigt, induziert jedoch nicht jedes FAA eine SLTR. Unser Ziel ist die Erarbeitung eines Algorithmus zu Erkennung von SLTRs und für diesen werden hinreichende

¹ $|f|$ sei die Menge der adjazenten Knoten an f .

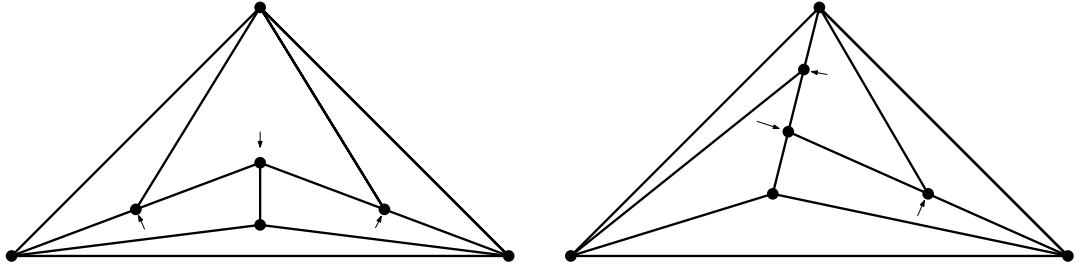


Abbildung 3.1 – Der selbe planare Graph mit zwei FAA, von denen nur das rechte eine SLTR induziert.

Bedingungen für ihre Existenz benötigt. Deswegen werden sich die nächsten beiden Abschnitten mit zwei Arbeiten von Aerts und Felsner beschäftigen. Das Hauptresultat der ersten liefert ein Gleichungssystem, welches harmonische Funktionen beinhaltet. Eine Lösung dieses Systems liefert eine SLTR [AF13b]. In Teilen darauf basierend stellt die zweite Arbeit einen Zusammenhang zwischen Schnyder Woods und FAAs her, und die Existenz passender Paare impliziert wieder die Existenz von SLTRs [AF15].

3.1 SLTRs durch harmonische Funktionen

Zum Einstieg eine weitere Definition, die es ermöglicht eine Beobachtung zu SLTRs festzuhalten. Die Beweise zu den in diesem Abschnitt aufgestellten Propositionen und Theoremen werden ausgelassen. Sie befinden sich, wenn nicht anders angegeben, in der entsprechenden Arbeit von Aerts und Felsner [AF13b].

Definition 3.3 (Begrenzende Zykel und kombinatorisch konvexe Ecken). Sei G ein ebener Graph mit Aufhängungen $\{a_1, a_2, a_3\}$ und einem FAA ϕ von G . Sei H ein zusammenhängender Teilgraph von G und $\gamma = \gamma(H)$ der H umrandende Weg in G . Somit ist γ eine Folge aus Kanten und Knoten die H in G begrenzt. Knoten und Kanten können in γ mehrfach vorkommen (vergleiche Abbildung 3.2). Wir werden so erhaltene γ als *begrenzende Zykel* bezeichnen. Die Menge der Knoten, Kanten und Gebiete aus G , die im Inneren von γ oder auf γ liegen, bezeichnen wir mit $\text{int}(\gamma)$. Einen Knoten v aus γ bezeichnen wir als *kombinatorisch konvexe Ecke* von γ im Bezug auf ϕ , falls gilt:

- E1 v ist eine Aufhängung, oder
- E2 v ist nicht durch ϕ zugeordnet und es existiert eine Kante $e = (v, w)$ mit $e \notin \text{int}(\gamma)$, oder
- E3 v ist einem Gebiet f zugeordnet, $f \notin \text{int}(\gamma)$ und es existiert eine Kante $e = (v, w)$, sodass $e \notin \text{int}(\gamma)$.

Bemerkung. Angenommen wir haben einen ebenen Graphen G mit Aufhängungen $\{a_1, a_2, a_3\}$ und ein FAA auf G gegeben. Um aus diesem FAA eine SLTR zu erzeugen,

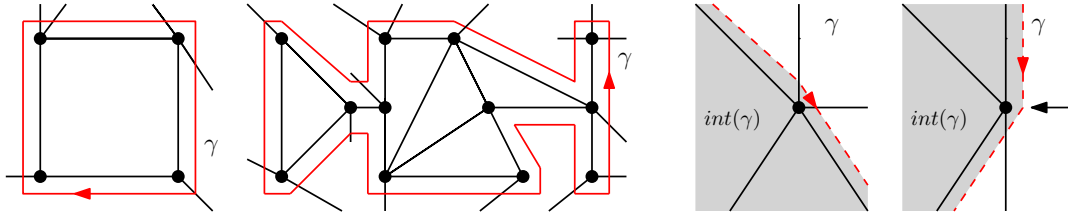


Abbildung 3.2 – Auf der linken Seite zwei Beispiele für *begrenzende Zyklen* und rechts für *kombinatorisch konvexe Ecken* mit und ohne zugewiesenem Knoten.

müssen wir die Gebiete in die Form von Dreiecken bringen. Stellen wir uns vor, dass alle Knoten auf einem Haufen liegen. Als Erstes „ziehen“ wir die Aufhängungen auseinander, um uns dann Stück für Stück nach innen vorzuarbeiten. Sind wir so bei einem Restgraphen H angekommen, dann brauchen wir mindestens drei Knoten auf Rand, also $\gamma(H)$, an denen wir „ziehen“ können, um H in eine konvexe Form zu bringen, ohne dabei einen flachen Winkel des FAA zu verletzen. Kombinatorisch konvexe Ecken entsprechen anschaulich gesehen diesen Knoten.

Beispiel 3.4. Betrachten wir eine SLTR und einen begrenzenden Zykel γ , der nicht von einem Pfad induziert wird. In Abbildung 3.3 ist so eine SLTR mit dem begrenzenden Zykel in rot abgebildet. Sei K die konvexe Hülle von γ in der SLTR. Sie ist hier grau unterlegt. Dann muss jede Ecke von K einen Außenwinkel haben, welcher grösser als π ist. Und da K nicht von einem Pfad induziert ist, müssen mindestens drei Ecken existieren (die drei roten Kreise). Es handelt sich um geometrisch konvexe Ecken in der SLTR. Nun ist der Knoten v an dieser Ecke entweder eine Aufhängung (wie unten links) oder es muss eine Kante geben, die K verlässt, denn in einer SLTR kann kein Winkel (ausser an den Aufhängungen) grösser sein als π . Wenn der Knoten v an der Ecke keine Aufhängung ist, dann existiert also mindestens eine Kante $(v, w) \notin K$. Es handelt sich also bei v auch um eine kombinatorisch konvexe Ecke von γ . Somit hat γ mindestens drei kombinatorisch konvexe Ecken in der SLTR.

Die Beobachtung aus dem Beispiel gilt allgemein für SLTRs und begrenzende Zyklen. Die folgende Präposition nach [AF13b, Prop 2.2, Prop 2.4] hält sie fest:

Proposition 3.5. *Sei G ein ebener Graph, der eine SLTR zulässt. Sei weiter ϕ das von der SLTR induzierte FAA. Sei H ein zusammenhängender Teilgraph von G (kein Pfad) und $\gamma = \gamma(H)$ sein begrenzender Zykel. Falls v eine geometrisch konvexe Ecke von γ in der SLTR ist, dann ist v auch eine kombinatorisch konvexe Ecke von γ hinsichtlich ϕ . Somit gilt:*

E4 Jeder begrenzende Zykel γ , der nicht von einem Pfad induziert wird, hat hinsichtlich ϕ mindestens drei kombinatorisch konvexe Ecken.

Proposition 3.5 liefert also eine notwendige Bedingung, damit ein FAA von einer SLTR induziert sein kann. Dies ist sogar eine hinreichende Bedingung, wie im Verlauf des Kapitels in Theorem 3.13 gezeigt wird.

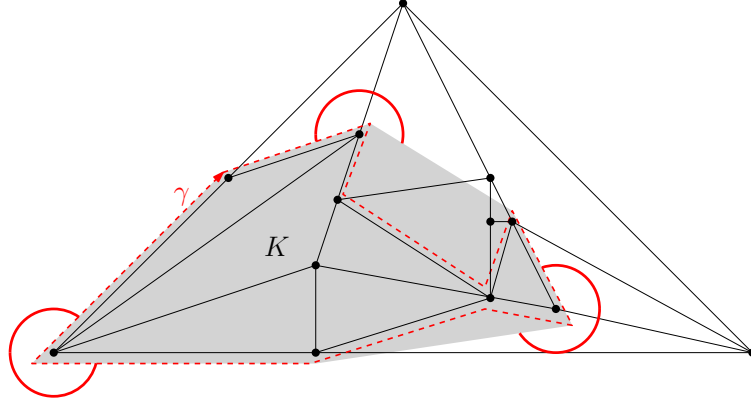


Abbildung 3.3 – Eine SLTR eines Graphen mit einem begrenzenden Zykel γ in rot. Die roten Kreise stellen drei geometrisch und kombinatorisch konvexe Ecken von γ dar.

Definition 3.6 (Gutes-FAA). Wir nennen ein FAA, das E4 aus Proposition 3.5 erfüllt, im Weiteren *Gutes-FAA* oder kurz *GFAA*.

Aerts und Felsner zeigen, dass ein Gutes-FAA eine *Kontaktfamilie von Pseudosegmenten* induziert, die sich somit geradlinig darstellen lässt. Wir beginnen mit der Definition.

Definition 3.7 (Kontaktfamilie von Pseudosegmenten). Eine *Kontaktfamilie von Pseudosegmenten* ist eine Familie $\Sigma = \{c_i\}_{i \in I}$ von einfachen Kurven

$$c_i : [0, 1] \rightarrow \mathbb{R}^2, \text{ mit } c_i(0) \neq c_i(1),$$

sodass alle Kurven c_i, c_j mit $i \neq j$ maximal einen gemeinsamen Punkt haben. Dieser Punkt muss dann ein Endpunkt von mindestens einer der Kurven sein.

Ein GFAA ϕ liefert eine Relation ρ auf den Kanten von G . Zwei Kanten (v, w) und (v, u) , beide adjazent zu f , stehen genau dann in Relation, wenn $\phi(v) = f$. (v, w) und (v, u) müssen also auf der selben Seite des Dreiecks f in der SLTR liegen. Der transitive Abschluss dieser Relation liefert eine Äquivalenzrelation ρ . Die Äquivalenzklassen von ρ bilden eine Kontaktfamilie von Pseudosegmenten.

Nennen wir die Äquivalenzklassen von ρ Kurven, dann gilt nach F2, dass jeder Knoten nur einem Gebiet zugeordnet werden kann und somit auch nur im Inneren von einer Kurve liegt. Die Kurven können sich also nicht kreuzen, sondern es kann nur eine an einer anderen enden. Weiter hat jede Kurve unterschiedliche Anfangs- und Endpunkte und kann sich nicht selbst berühren. Dies kann man so begründen, dass sonst der resultierende begrenzende Zykel γ nur eine, beziehungsweise zwei kombinatorisch konvexe Ecken hätte. Das wäre ein Widerspruch zu E4. Analog können zwei Kurven nicht ihre Anfangs- und Endpunkte teilen, da sonst wieder ein Zykel mit zu wenigen Ecken entstehen würde. Für eine von einem FAA ϕ induzierte Kontaktfamilie schreiben wir auch Σ_ϕ .

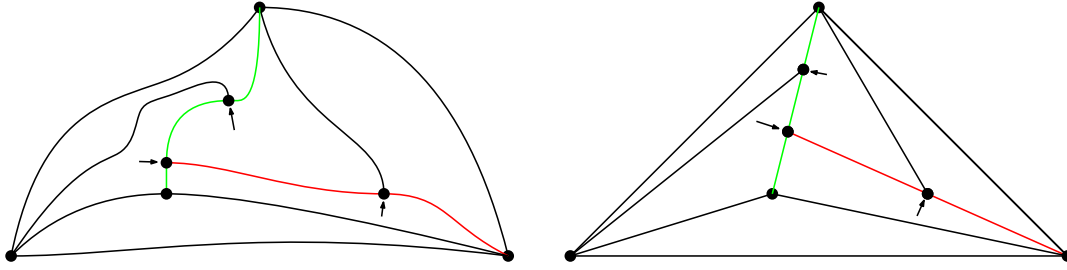


Abbildung 3.4 – Die Kanten von G als Kontaktfamilie von Pseudosegmenten induziert durch die Äquivalenzrelation. In rot und grün die beiden Äquivalenzklassen bzw. Kurven, die mehr als eine Kante beinhalten.

Definition 3.8. Sei Σ ein Kontaktfamilie von Pseudosegmenten und $S \subseteq \Sigma$. Wir nennen einen Punkt $p \in S$ einen *freien Punkt*, falls er die folgenden Bedingungen erfüllt.

- p ist ein Endpunkt eines Pseudosegmentes aus S .
- p liegt nicht im Inneren eines Pseudosegmentes aus S .
- p liegt am äußeren Rand von S .
- p ist entweder eine Aufhängung von G oder berührt ein Pseudosegment, welches nicht zu S gehört.

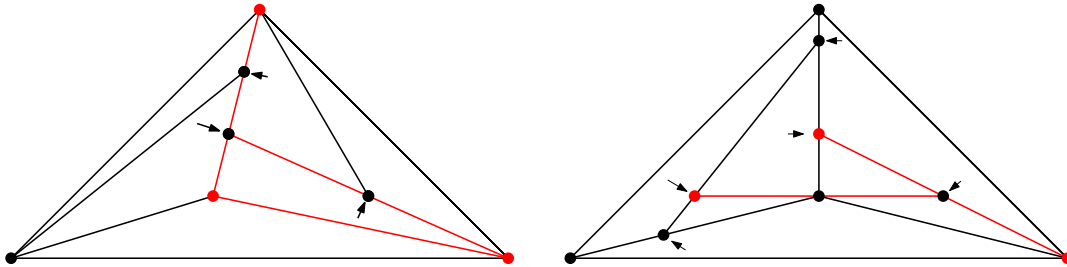


Abbildung 3.5 – Zwei SLTRs mit Teilfamilien von Pseudosegmenten in rot und den freien Punkten dieser Familien ebenfalls in rot.

Lemma 3.9. [AF13b, Lemma 2.8] Sei ϕ ein Gutes-FAA auf einem ebenen und intern 3-zusammenhängenden Graphen, dann gilt:

E5 Jede Teilmenge $S \subseteq \Sigma_\phi$ mit $|S| \geq 2$ hat mindestens 3 freie Punkte.

Betrachte einen ebenen, intern 3-zusammenhängenden Graphen G mit Aufhängungen $\{a_1, a_2, a_3\}$ und einem GFAA ϕ . Wenn die von ϕ induzierte Kontaktfamilie Σ_ϕ mit geradlinigen Segmenten darstellbar ist, dann ist diese Darstellung eine zu ϕ passende SLTR für G . Für den Fall, dass eine solche Darstellung $f : G \rightarrow \mathbb{R}^2$

existiert, können für die Koordinaten der Segmente oder genauer für die Knoten v von G auf den Segmenten im Folgenden Gleichungen aufgestellt werden. Die Positionen $f(v)$ der Knoten v in der Einbettung $f(G)$ müssen diese Gleichungen erfüllen. Das resultierende Gleichungssystem beinhaltet harmonische Funktionen. Bevor wir die Gleichungen, die unser Problem charakterisieren, in Abschnitt 3.1.2 angeben, werden wir zu harmonischen Funktionen einen kurzen Überblick geben.

3.1.1 Harmonische Funktionen auf planaren Graphen

Die Theorie zu (diskreten) harmonischen Funktionen auf planaren Graphen und ihre Anwendung wird in [LV99] ausführlich behandelt. Es handelt sich um eine Diskretisierung von allgemeinen harmonischen Funktionen, also glatten Funktionen $f : G \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$, mit $\Delta f = 0$, wobei Δ den Laplace Operator beschreibt. Für diese Funktionen gilt, dass der Funktionswert an einem Punkt x dem Durchschnitt der Funktionswerte auf einem Ball um x entspricht.

Dies führt zu der folgenden Definition im diskreten Fall.

Definition 3.10 (Harmonische Funktionen). Sei $G = (V, E)$ ein planarer zusammenhängender Graph und $S \subseteq V$. Eine Funktion $g : V \rightarrow \mathbb{R}$ nennen wir am Knoten $v \in V$ *harmonisch*, falls gilt:

$$\text{H1} \quad \sum_{u \in N(v)} (g(u) - g(v)) = 0$$

Wir können H1 durch das Hinzufügen einer nichtnegativen Gewichtsfunktion $\lambda : E \rightarrow \mathbb{R}_+$ verallgemeinern. Es gilt $\lambda(v, w) = \lambda_{vw}$.

$$\text{H2} \quad \sum_{u \in N(v)} \lambda_{uv} (g(u) - g(v)) = 0$$

Einen Knoten, für den g nicht harmonisch ist, nennen wir *Pol*.

Theorem 3.11. [LV99, Theorem 3.1.2] Für jede nichtleere Teilmenge $S \subseteq V$ und jede Funktion $g_S : S \rightarrow \mathbb{R}$ existiert genau eine Funktion $g : V \rightarrow \mathbb{R}$, die g_S auf V fortsetzt, sodass g in jedem Knoten $v \in V \setminus S$ harmonisch ist. Wir nennen sie die *harmonische Fortsetzung von g_S auf V* .

Ein bekanntes Resultat, das sich in Form harmonischer Funktionen darstellen lässt, ist Tuttés *rubber-band-representation*, die konvexe Zeichnungen für planare Graphen liefert [Tut63]. Man stelle sich einen planaren Graphen vor, bei dem jede Kante durch ein idealisiertes Gummiband² ersetzt wird. Fixiere für den Moment alle Knoten in einem beliebigen Punkt. Wähle nun ein äußeres Gebiet f_{aus} . Die Menge $S \subseteq V$ seien die zu f_{aus} adjazenten Knoten. Nun fixieren wir diese Knoten in zyklischer Reihenfolge und in gleichen Abständen auf einem Kreis in der Ebene. Dies definiert $f_x : S \rightarrow \mathbb{R}$

²Die Gummibänder müssen das Hook'sche Gesetz erfüllen, sodass eine Streckung auf Länge l genau Kraft l benötigt.

und $f_y : S \rightarrow \mathbb{R}$. Wenn wir nun die restlichen Knoten loslassen, dann werden sie von den Gummibändern in eine neue Position gezogen. Das resultierende Gleichgewicht, das genau dann entsteht, wenn H1 für beide Komponenten von f erfüllt ist, entspricht der (komponentenweisen) harmonischen Fortsetzung von $f = (f_x, f_y)$ auf V , wobei $f(v)$ genau der Position von v in der resultierenden Einbettung entspricht und S die Menge der Pole von f ist. Wir können die Kanten zusätzlich noch mit nicht negativen Gewichten λ_{vw} , versehen, um die Einbettung zu verändern. Das folgende Theorem ist das Hauptresultat dieser Vorgehensweise [Tut63].

Theorem 3.12. *Sei G ein planarer Graph, dann ist eine Gummiband-Representation (rubber-band-representation) von G eine planare Einbettung in der Ebene.*

3.1.2 Das resultierende Gleichungssystem

Die Theorie zu harmonischen Funktionen lässt sich auf SLTRs anwenden. Sei G ein planarer Graph und ϕ ein FAA. Nehme für den Moment an, es existiert eine geradlinige Darstellung der Pseudosegmente Σ_ϕ . Wir haben also eine geradlinige Einbettung $f(G)$ der von ϕ induzierten Segmente.

Es gilt für jeden Knoten v im Inneren eines Segmentes, also für jeden zugewiesenen Knoten, dass er auf einer Gerade zwischen seinen beiden benachbarten Knoten u, w auf dem Segment liegen muss. Diese Eigenschaft liefert die komponentenweisen Bedingungen

$$\begin{aligned} f_x(v) &= \lambda_v f_x(u) + (1 - \lambda_v) f_x(w), \text{ mit } \lambda_v \in (0, 1) \\ f_y(v) &= \lambda_v f_y(u) + (1 - \lambda_v) f_y(w), \text{ mit } \lambda_v \in (0, 1). \end{aligned} \quad (3.1)$$

Für die nicht zugewiesenen Knoten aus G muss in einer SLTR gelten, dass sie sich in der konvexen Hülle ihrer Nachbarn befinden. Wir bilden einen (gewichteten) Schwerpunkt und erhalten komponentenweise

$$\begin{aligned} f_x(v) &= \sum_{u \in N(v)} \lambda_{uv} f_x(u), \text{ mit } \sum_{u \in N(v)} \lambda_{uv} = 1 \text{ und } \lambda_{uv} \geq 0. \\ f_y(v) &= \sum_{u \in N(v)} \lambda_{uv} f_y(u), \text{ mit } \sum_{u \in N(v)} \lambda_{uv} = 1 \text{ und } \lambda_{uv} \geq 0. \end{aligned} \quad (3.2)$$

Somit erfüllt die so gegebene Funktion $f : V \rightarrow \mathbb{R}^2$ mit $f = (f_x, f_y)$ und passend gewählten λ wegen (3.1) und (3.2) in beiden Komponenten H2. Es handelt sich somit bei f_x und f_y um harmonische Funktionen mit den Polen $\{a_1, a_2, a_3\}$. Nach Theorem 3.11 existiert für jede, den Beschränkungen entsprechende Wahl von λ , somit genau eine Funktion $f = (f_x, f_y)$, welche die Gleichungen 3.1 und 3.2 erfüllt.

Dies führt uns zum Hauptresultat aus [AF13b]:

Theorem 3.13. *Sei G ein intern-3-zusammenhängender, ebener Graph und Σ eine Familie von Pseudosegmenten, induziert von einem FAA, sodass jede Teilfamilie $S \subset \Sigma$ entweder mindestens drei freie Punkte hat oder maximal ein Element ent-*

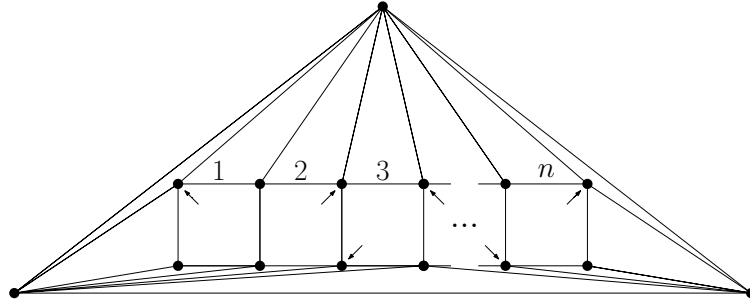


Abbildung 3.6 – Ein Graph mit exponentiell vielen FAAs.

hält. Dann induziert die eindeutige Lösung $f : V \rightarrow \mathbb{R}^2$ des aus Σ resultierenden Gleichungssystems eine SLTR.

Bemerkung. Dies bedeutet, dass die notwendigen Bedingungen E4 und E5, die in Lemma 3.9 und in Proposition 3.5 festgehaltenen wurden, auch hinreichende Bedingungen sind. Falls wir schon ein Gutes-FAA gefunden haben, dann können wir mit Hilfe des obigen Ansatzes auch eine Einbettung in der Ebene erhalten. Es gilt jedoch für viele Graphen, dass sie, wie in Beispiel 3.14, exponentiell vielen FAAs besitzen. Selbst wenn wir die hinreichende Bedingung E4 in polynomineller Zeit überprüfen könnten, erreichen wir auf diesem Weg keinen schnellen Algorithmus.

Aerts und Felsner werfen am Ende von [AF13b] die Frage nach einer guten Wahl von λ auf und wie dies die resultierenden Einbettungen beeinflussen kann. Kapitel 5 wird einer möglichen Antwort dieser Frage nachgehen.

Beispiel 3.14. Betrachten wir eine zusammenhängende Kette von n Quadraten in der Ebene und verbinden wir ihre Eckpunkte mit drei Aufhängungen a_1, a_2, a_3 , welche ein Dreieck bilden. Der so erzeugte Graph G hat $2n + 5$ Knoten und ist in Abbildung 3.6 zu sehen. Wir wollen nun ein FAA für diesen Graphen erstellen (welches nicht zwangsläufig eine SLTR zulässt). Wir müssen für ein FAA jedem der inneren Quadrate einen Winkel zuordnen. Wenn wir von links beginnen und nach rechts laufen, können wir in jedem der n inneren Quadrate mindestens aus drei Winkeln auswählen, um ein FAA zu erstellen. Es wurde höchstens einer der beiden Knoten auf der linken Seite von Quadrat i im Schritt zuvor zugewiesen. Somit existieren mehr als 3^n FAAs für G .

3.2 Ecken kompatible Paare

In diesem Abschnitt werden wir uns mit einer zweiten Charakterisierung von SLTRs auf planaren Graphen nach [AF15] beschäftigen, die eine Verbindung zwischen Schnyder Woods und FAAs herstellt und so zu einer hinreichenden Bedingung für SLTRs führt. Zum Einstieg folgt die Definition dieses Zusammenhangs.

Definition 3.15 (Ecken Kompatibilität). Ein Paar (σ, ϕ) aus einem Schnyder Labeling σ und einem FAA ϕ nennen wir *Ecken kompatibel*, falls:

K1 Das Schnyder Labeling σ und das FAA ϕ die selben Aufhängungen nutzen, und

K2 die drei Ecken aus ϕ in jedem inneren Gebiet drei unterschiedliche Label in σ haben.

Beispiel 3.16. Wir greifen vor und nehmen an, dass ein Graph genau dann eine SLTR zulässt, wenn ein Ecken kompatibles Paar existiert. Wir müssen für die Beantwortung, ob ein Graph eine SLTR zulässt unter Umständen alle Schnyder Woods von G überprüfen. Dies sehen wir anhand von Abbildung 3.7. Hier ist derselbe Graph mit zwei unterschiedlichen Schnyder Labelings abgebildet. Zu dem Labeling auf der linken Seite existiert ein Ecken kompatibles FAA. Auf der rechten Seite kann kein solches FAA existieren. Wie wir auf der linken Seite sehen, muss jeder innere Knoten zugewiesen werden. Insbesondere also der Knoten v . Dann fehlt jedoch im zugewiesenen Gebiet eine Ecke mit diesem Label, da die Label an v jeweils nur einmal in den Gebieten auftreten.

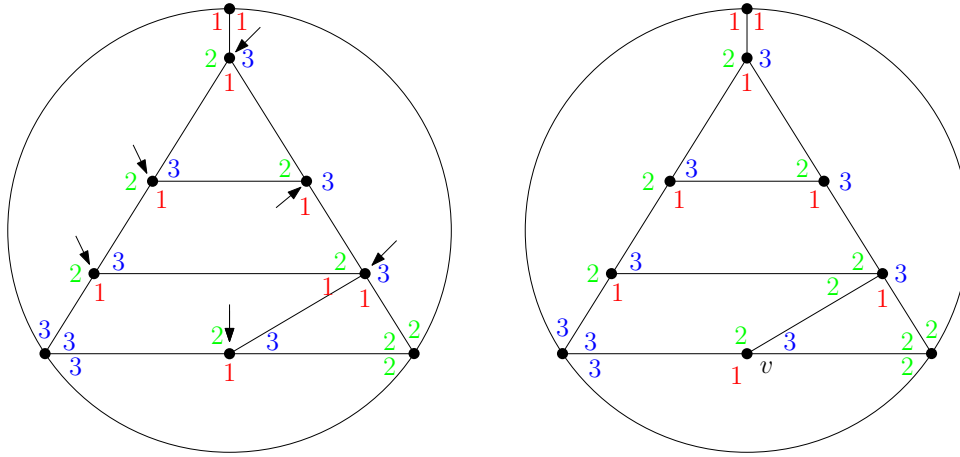


Abbildung 3.7 – Nicht jedes Schnyder Labeling lässt ein Ecken kompatibles Paar zu, selbst wenn der Graph eine SLTR hat.

Der Rest dieses Kapitels wird sich mit dem Beweis beschäftigen, dass zu jedem SLTR (mindestens) ein Ecken kompatibles Paar existiert, und dass andersherum jedes Ecken kompatible Paar eine SLTR induziert.

Theorem 3.17. *Sei G ein ebener, intern-3-zusammenhängender Graph mit den Aufhängungen $\{a_1, a_2, a_3\}$. G besitzt genau dann eine SLTR, wenn ein Ecken kompatibles Paar (σ, ϕ) aus einem Schnyder Labeling σ und einem FAA ϕ existiert.*

Wir beweisen zuerst die (deutlich einfachere) Rückrichtung des Theorems. Hier können wir die durch das in Abschnitt 2.7 erklärte *face-counting* erhaltene Einbettung nutzen, um zu zeigen, dass jeder begrenzende Zykel γ mindestens drei kombinatorisch konvexe Ecken besitzt.

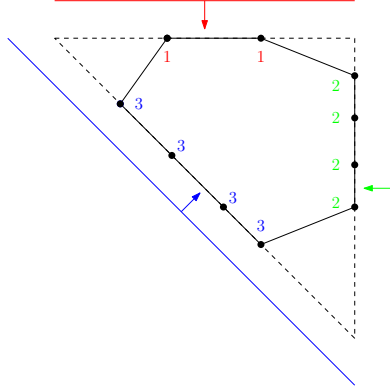


Abbildung 3.8 – Die drei Geraden die wir von den Aufhängungen aus über den Graphen schieben.

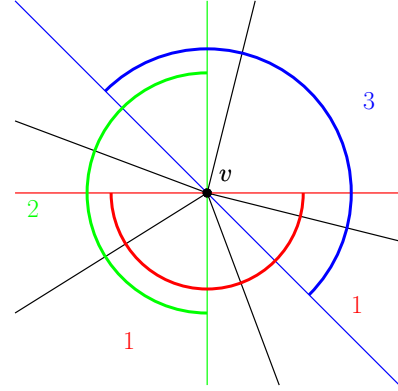


Abbildung 3.9 – Winkel die komplett gegenüber der Geraden i liegen haben Label i .

Lemma 3.18. *Sei G ein ebener, intern-3-zusammenhängender Graph mit den Aufhängungen $\{a_1, a_2, a_3\}$. Falls ein Paar (σ, ϕ) aus einem Schnyder Labeling σ und einem FAA ϕ Ecken kompatibel ist, dann hat jeder begrenzende Zykel γ , der nicht von einem Pfad induziert wird, mindestens drei kombinatorisch konvexe Ecken im Bezug auf ϕ .*

Beweis: Sei γ ein begrenzender Zykel und F_{in} die Menge der inneren Gebiete von G . Seien $\alpha_1 = (0, 1)$, $\alpha_2 = (1, 0)$ und $\alpha_3 = (0, 0)$ die Bilder der Aufhängungen und D die durch *face counting* erhaltende Zeichnung von G mit den Ecken $\alpha_1, \alpha_2, \alpha_3$. Ein Beispiel einer solchen Zeichnung findet sich in Abbildung 2.7. Betrachte nun den begrenzenden Zykel γ in D . Wir schieben nun, wie in Abbildung 3.8 illustriert, ausgehend von α_i die Geraden $(\alpha_{i+1}, \alpha_{i-1})$ über den Graphen. Sei M_i die Menge der zuerst von $(\alpha_{i+1}, \alpha_{i-1})$ getroffenen Knoten von γ für $i \in (1, 2, 3)$.

Beobachtung 1 *Alle Knoten um ein inneres Gebiet f mit Label i in f werden von der Gerade $(\alpha_{i+1}, \alpha_{i-1})$ zum gleichen Zeitpunkt getroffen. Dies folgt direkt aus Proposition 2.12 W5, weil alle Knoten mit dem selben Label in der Zeichnung auf $c_i(\alpha_{i+1}, \alpha_{i-1})$ platziert werden.*

Beobachtung 2 *Sei $v \in M_i$. Alle Winkel an v im Inneren von γ haben das Label i . Die Geraden $(\alpha_{i+1}, \alpha_{i-1})$ teilen die Winkel um einen Knoten in Intervalle (siehe Abbildung 3.9). Die Winkel an v , die von a_i aus gesehen vollständig auf der anderen Seite der Gerade $(\alpha_{i+1}, \alpha_{i-1})$ liegen, haben Label i .*

Nach Beobachtung 2 sind die Mengen M_1, M_2 und M_3 disjunkt. Wir suchen nun nach drei kombinatorisch konvexen Ecken von γ . Das FAA und das Schnyder Labeling sind Ecken kompatibel und somit hat jedes Gebiet $f \in F_{in}$ einen Winkel mit Label i . Also liegt in jeder Menge M_i ein Knoten v_i , der vom FAA nicht einem Gebiet innerhalb von γ zugewiesen wird. Nehmen wir an $a_i \notin M_i$, denn sonst hätten wir nach E1 eine Ecke gefunden. Da D eine konvexe Zeichnung ist, muss v_i einen Nachbarn

außerhalb von γ besitzen. Somit liegt v_i auf γ , ist nicht in γ zugewiesen und hat einen Nachbarn außerhalb von γ . Der Knoten v_i erfüllt also E2. Somit hat jeder begrenzende Zykel drei kombinatorisch konvexe Ecken (jeweils eine aus jedem M_i). \square

Zusammen mit Theorem 3.13 folgt, dass es sich bei dem FAA um ein Gutes-FAA handelt. Somit induziert das Ecken kompatible Paar ein SLTR von G .

Machen wir uns an den Beweis der Hinrichtung. Zu jedem SLTR können wir ein eindeutiges FAA erstellen, indem wir die flachen Winkel der SLTR im FAA zuweisen. Wir müssen also zeigen, dass zu jeder SLTR ein Schnyder Labeling existiert, das zusammen mit dem induzierten FAA ein Ecken kompatibles Paar bildet. Sei G ein ebener, intern-3-zusammenhängender Graph mit den Aufhängungen $\{a_1, a_2, a_3\}$, der (mindestens) eine SLTR besitzt. Sei Δ eine SLTR von G und sei ϕ das von Δ induzierte FAA.

Vor dem nächsten Lemma müssen wir zwei geometrische Objekte einführen. Beispiele finden sich in Abbildung 3.10.

Definition 3.19 (Unterteilendes Dreieck). Ein *unterteilendes Dreieck* ist ein Dreieck in der Zeichnung einer SLTR von G , sodass gilt:

- Jeder Knoten auf dem Rand des Dreiecks (der keine Ecke des Dreiecks ist) ist entweder außerhalb oder innerhalb des Dreiecks zugewiesen
- Es existiert ein Knoten (der keine Ecke ist), der keine Nachbarn außerhalb des Dreiecks hat und es existiert ein Knoten (der keine Ecke ist), der keine Nachbarn im Inneren des Dreiecks hat.

Dieses Dreieck kann Teile des Randes der Zeichnung beinhalten.

Definition 3.20 (teilendes Segment). Ein *teilendes Segment* eines SLTR von G ist eine Menge von Kanten $\{e_1, \dots, e_m\}$, die alle auf einer Gerade liegen, sodass gilt:

- Die Vereinigung der Kanten trennt die Zeichnung in zwei nichtleere Teile.
- Jeder innere Knoten v auf dem Segment ist einem Gebiet zugeordnet, dass zwei Kanten beinhaltet die auf dem Segment liegen. Diese beiden Kanten haben v als Endpunkt.

Um zu zeigen, dass wir zu jeder SLTR ein passendes Ecken kompatibles Paar (σ, ϕ) finden, führen wir einen Widerspruchsbeweis durch. Sei G ein kleinstmögliches Gegenbeispiel, zu dem kein Ecken kompatibles Paar existiert. Damit seien hier zuerst die minimale Anzahl an Knoten und darauf folgend die kleinste Anzahl von Kanten gemeint. Sei Δ eine SLTR von G , ϕ das induzierte FAA und a_1, a_2 und a_3 die Aufhängungen von Δ .

Wir zeigen zuerst zwei Eigenschaften von Δ .

Lemma 3.21. *Ein minimales Gegenbeispiel Δ hat kein unterteilendes Dreieck.*

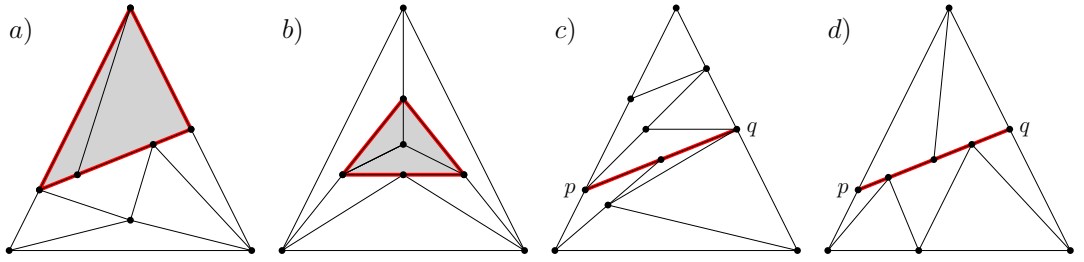


Abbildung 3.10 – Beispiele von unterteilenden Dreiecken in a) und b) und teilenden Segmenten in c) und d) jeweils in rot.

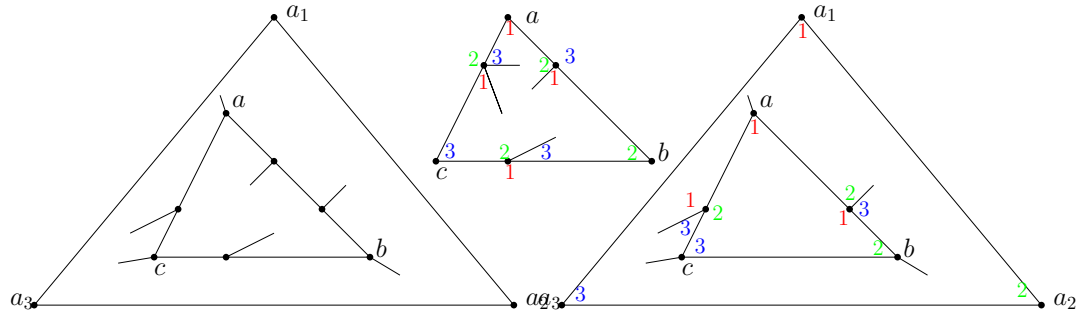


Abbildung 3.11 – Wir teilen Δ in zwei Graphen und erhalten zwei Ecken kompatible Paare, die wir zusammenfügen können.

Beweis: Nehmen wir an, dass Δ ein unterteilendes Dreieck mit den Ecken (a, b, c) beinhaltet. Dies ist in Abbildung 3.11 dargestellt. Seien Δ_1 und Δ_2 die Teile von Δ , die alles außerhalb (Δ_1) und innerhalb (Δ_2) des Dreiecks beinhalten. Der Rand des Dreiecks (a, b, c) liegt in beiden Teilen.

Wir ersetzen Knoten auf dem Rand des Dreiecks, die Grad 2 in Δ_i haben, mit einer Kante zwischen ihren Nachbarn. Somit sind Δ_1 und Δ_2 SLTRs mit weniger Knoten als Δ . Da sie weniger Knoten haben als Δ , können sie keine Gegenbeispiele sein und es existieren zu den SLTRs Δ_i Ecken kompatible Paare (σ_i, ϕ_i) , wobei die ϕ_i die induzierten FAAs von Δ_i sind. Wenn wir die Paare nun zusammensetzen, dann stoßen wir auf einen Widerspruch. Die Ecken a, b, c sind die Aufhängungen von Δ_2 . Wir wählen ihre Label so, dass sie mit den inneren Labeln des (jetzt) leeren Dreiecks in Δ_1 übereinstimmen³. Die auf diese Weise kombinierten Schnyder Labelings σ_1 und σ_2 ergeben ein Schnyder Labeling auf G . Die FAAs ϕ_1 und ϕ_2 ergeben zusammen, wenn wir die Zuweisungen an den äußeren Knoten von Δ_2 und den am leeren Dreieck liegenden Knoten von Δ_1 anpassen, ein FAA ϕ für G . Somit folgt die Ecken Kompatibilität aus der Tatsache, dass (σ_1, ϕ_1) und (σ_2, ϕ_2) Ecken kompatibel sind. Die SLTR Δ induziert somit ein Ecken kompatibles Paar und kann kein Gegenbeispiel sein. Δ kann kein unterteilendes Dreieck haben. \square

³Wir können die Label beliebig umbenennen, ohne das Schnyder Labeling zu verändern.

Lemma 3.22. *Ein minimales Gegenbeispiel Δ hat kein teilendes Segment.*

Bemerkung. Als Vorgriff auf den Beweis bedeutet dies insbesondere, dass in Δ für jede Aufhängung $\deg(a_i) \geq 3$ gelten muss.

Beweis: Angenommen Δ hat ein teilendes Segment mit den Endpunkten p und q . Falls auf beiden Seiten des teilenden Segmentes eine Aufhängung mit Grad größer als 2 liegt, dann wird ein unterteilendes Dreieck in Δ induziert. In diesem Fall existieren sowohl Knoten (auf dem Inneren des Segmentes), die keine Nachbarn links des teilenden Segmentes besitzen, als auch Knoten, die keine Nachbarn rechts des teilenden Segmentes besitzen. Man kann dies anhand von Abbildung 3.10 d) nachvollziehen. Angenommen es handelt sich bei p oder q um eine Aufhängung und auf der einen Seite des Segmentes liegt nur eine Aufhängung a_i . Dann existiert mit $\Delta' = \Delta \setminus \{a_1\}$ ein kleineres SLTR (welches dann kein Gegenbeispiel ist) und aus diesem lässt sich ein Ecken kompatibles Paar für Δ konstruieren. Es müssen also auf beiden Seiten des Segmentes mindestens zwei Knoten liegen und somit existiert wieder ein unterteilendes Dreieck. Wir können also annehmen, dass das teilende Segment zwischen p und q die Aufhängung a_1 von Grad 2 abtrennt. p, q und a_1 bilden somit ein Dreieck. Wir betrachten zwei Fälle. Entweder das teilende Segment besteht nur aus der Kante (p, q) (Fall 1) oder es existiert mindestens ein weiterer Knoten auf dem Segment (Fall 2).

Fall 1: Das teilende Segment besteht nur aus der Kante (p, q) . Angenommen q hat Grad 3 und q' sei der dritte Nachbar von q . Dann existiert eine Gerade zwischen q' und p mit keinen Kanten in das Dreieck p, q, q' . Somit existiert ein unterteilendes Dreieck mit den Ecken p, q', a_1 (siehe Abbildung 3.12 a). Dies ist ein Widerspruch zu Lemma 3.11 und es folgt $\deg(p), \deg(q) \geq 4$. Da $\deg(a_1) = 2$ gelten muss und G intern-3-zusammenhängend ist, liegt a_1 alleine auf der einen Seite des Segments und alle anderen Nachbarn von p und q auf der anderen. Wir behaupten, dass mindestens eine der Kanten (a_1, p) und (a_1, q) kontrahierbar ist, sodass der resultierende Graph eine SLTR besitzt. Die Zuweisungen des FAA bleiben, bis auf bei p und q , gleich (siehe Abbildung 3.12 b). Dieser Schritt ist nicht trivial. Wir nutzen als Kriterium die begrenzenden Zykel aus Definition 3.15. Wir wollen zeigen, dass für eine der Kontraktionen wieder gilt, dass jeder begrenzende Zykel drei kombinatorisch konvexe Ecken bezüglich des FAA hat. Damit dies für keine der beiden Kontraktionen gilt, müssten zwei begrenzende Zykel γ_v, γ_w mit genau drei kombinatorisch konvexen Ecken, p, q und v respektive w , existieren (siehe Abbildung 3.12 c). Nur dann induziert die Kontraktion von (a_1, p) und (a_1, q) jeweils einen Zykel mit nur zwei kombinatorisch konvexen Ecken. Und somit hätte der resultierende Graph keine SLTR.

Dieser Fall kann aber nicht auftreten. Seien v, w die Ecken dieser Zykel. Dann existieren Pfade P_{pw} und P_{qv} von p nach w und von q nach v . Diese Pfade sind Teil von γ_v , beziehungsweise γ_w , und enthalten somit keine kombinatorisch konvexen Ecken mit Ausnahme der Endknoten. Die Winkel an diesen Pfaden im Inneren der Zykel sind somit $\geq \pi$. Sei z der Knoten an dem sich P_{pw} und P_{qv} kreuzen (siehe Abbildung 3.12 c). Da z keine Ecke sein kann, muss z auf beiden Zykeln γ_v, γ_w zugewiesen sein. Somit

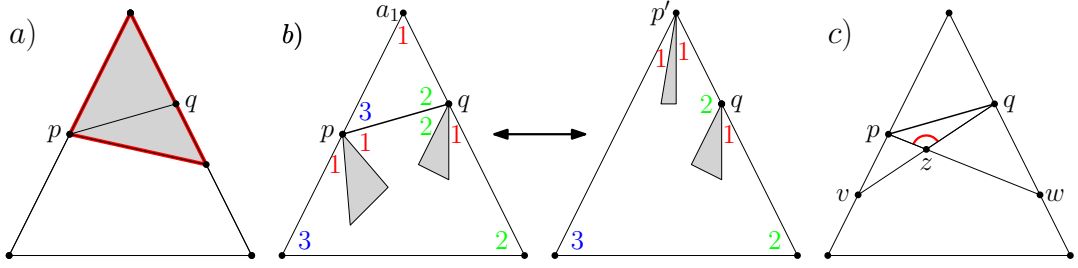


Abbildung 3.12 – a) Unterteilendes Dreieck bei Grad 3. b) Kantenkontraktion der Kante (a_1, p) . c) Die Pfade P_{pw} und P_{qv} , die bei der Kontraktion von (a_1, p) und (a_1, q) Degeneriertheit induzieren.

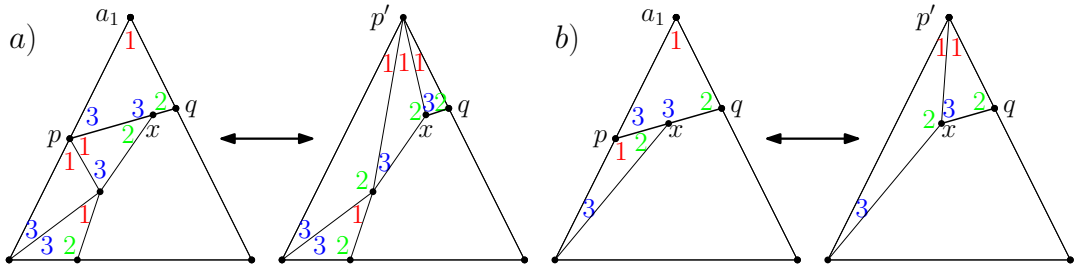


Abbildung 3.13 – Zwei Beispiele zur Kontraktion von (a_1, p) mit passendem Schnyder Labeling im zweiten Fall.

müsste der Winkel im inneren der Zykel, der an z von P_{pw} und P_{qv} eingeschlossen wird, mindesten π sein. Dies ist ein Widerspruch zur Annahme das Δ eine SLTR ist (siehe Abbildung 3.12 c).

Es kann also mindestens eine der Kanten kontrahiert werden. Sei (a_1, p) diese Kante und G' der Graph der durch Kontraktion von (a_1, p) und das Löschen von (p, q) entsteht. Wir erhalten das FAA ϕ' durch Löschen der Zuweisung von p aus ϕ . Der Knoten q ist weiterhin dem äußeren Gebiet zugewiesen. Da G' weniger Knoten als G hat, ist G' kein Gegenbeispiel und wir erhalten einen Schnyder Labeling σ' , das zusammen mit ϕ' ein Ecken kompatibles Paar bildet. Wir können ϕ' zu einem Labeling von G erweitern, indem wir, beginnend bei a_1 , im Uhrzeigersinn die Label 1, 2 und 3 im Gebiet a_1, q, p einfügen. Wir erhalten ein Ecken kompatibles Paar.

Fall 2: Sei x der erste Nachbar von p auf dem teilenden Segment. Wir kontrahieren wieder die Kante (a_1, p) , um G' zu erhalten und zeigen, dass G' eine SLTR besitzt. Bei der Kontraktion müssen keine weiteren Kanten gelöscht werden, wie in Abbildung 3.13 a) zu sehen ist. Wieder erhalten wir das FAA ϕ' auf G' , indem wir die Zuweisung von p aus ϕ löschen. Zu jedem begrenzenden Zykel γ in G existiert ein begrenzender Zykel γ' in G' . Falls x eine kombinatorisch konvexe Ecke von γ ist, dann ist x dies auch für γ' , weil keine Kante an x gelöscht wurde.

Nun ist G' kein Gegenbeispiel und es existiert ein zu ϕ' kompatibles Schnyder Labeling σ' . σ' ist erweiterbar zu einem Labeling σ für G . Füge Label 1 bei a_1 und 3

bei p ein. Δ kann also kein teilendes Segment haben.

□

Im nächsten Lemma wird eine Eigenschaft von Ecken kompatiblen Paaren festgehalten, die für den Beweis von Theorem 3.17 nützlich sein wird.

Lemma 3.23. *Sei G ein ebener, intern-3-zusammenhängender Graph mit den Aufhängungen $\{a_1, a_2, a_3\}$ und einem Ecken kompatiblen Paar (σ, ϕ) aus einem Schnyder Labeling und einem FAA. Sei v ein Nachbar einer Aufhängung a_i . Falls v von ϕ einem Gebiet f zugewiesen ist, das a_i beinhaltet, folgt, dass das Label von v in f nur einmal an v vorkommt. Alle anderen Winkel an v haben ein anderes Label.*

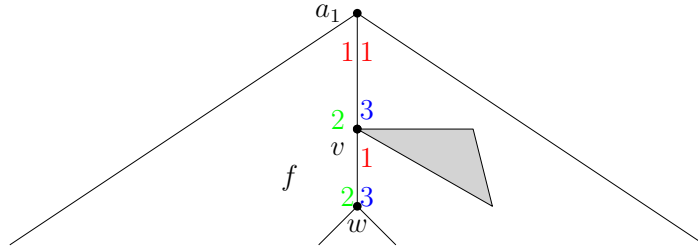


Abbildung 3.14 – Das Label 2 kann an v nur einmal vorkommen.

Beweis: Nehme ohne Beschränkung der Allgemeinheit an, dass v ein Nachbar von a_1 ist und v dem Gebiet f auf der linken Seite der Kante (a_1, v) zugeordnet ist (siehe Abbildung 3.14). Sei w der andere Nachbar von v in f . Die Kante (a_1, v) hat zweimal Label 1, jeweils links und rechts von a_1 , und Label 2 am zugewiesenen Winkel von v . Nach Definition 2.7 muss an jeder Kante jedes Label einmal vorkommen. Somit ist das letzte Label an (a_1, v) von Typ 3. Da (σ, ϕ) ein kompatibles Paar ist, muss der Winkel von w in f ebenfalls Label 2 haben, da sonst keine Ecke mit Label 2 existiert (vergleiche Definition 2.6). Um die Kante (v, w) müssen ebenfalls alle Label vorkommen und somit müssen wir 1 und 3 (wie in der Abbildung 3.14) einfügen. Um v existieren nach L2 im Uhrzeigersinn drei nichtleere Intervalle mit Label 1, 2 und 3. Folglich können die verbliebenen unbekannten Label an v nur von Typ 1 oder 3 sein. □

Lemma 3.24. *In einem minimalen Gegenbeispiel Δ kann kein Nachbar x_i einer der Aufhängungen a_i einem Gebiet zugeordnet sein, zu dessen Rand die Kante (a_i, x_i) gehört.*

Bemerkung. Lemma 3.24 impliziert insbesondere, dass die drei Aufhängungen a_1, a_2 und a_3 von Δ ein Dreieck bilden. Es liegen also keine weiteren Knoten auf dem äußeren Rand von Δ .

Beweis: Angenommen es existiert ein Nachbar x von a_1 , der dem Gebiet f zugewiesen ist und a_1 liegt auf dem Rand von f . Somit tut dies auch die Kante (a_1, x) . Wir werden wieder einen kleineren Graphen G' erstellen und zeigen, dass G' eine SLTR zulässt. Das zu diesem SLTR korrespondierende Schnyder Labeling ϕ' werden wir zu einem Schnyder Labeling von G erweitern. Im besten Fall kontrahieren wir die Kante (a_1, x) , um G' zu erhalten, doch dies ist nicht immer möglich. Ein Beispiel ist in Abbildung 3.15 a) zu sehen. Wenn z nur Grad 3 hat, dann wäre G' nach der Kontraktion nicht mehr intern-3-zusammenhängend.

Sei z die dritte Ecke des Gebietes f mit den Ecken a_1 und x in der Zeichnung Δ . Es kann nur ein solches z geben, da sonst (wie in Abbildung 3.15 a) ein unterteilendes Dreieck in Δ existieren müsste. Wir haben aber in Lemma 3.11 gezeigt, dass dies nicht der Fall sein kann.

Wir können annehmen, dass z und a_1 benachbart sind. Falls nicht, muss ein Knoten x' zwischen z und a_1 liegen. Sei z' die dritte Ecke im Gebiet mit Ecken a_1, x', z' . Der Knoten x' muss in diesem Gebiet eine Ecke sein, da er im zuvor betrachteten Gebiet zugewiesen ist (siehe Abbildung 3.15 b). Nun ist entweder z' ein Nachbar von a_1 , oder wir führen diesen Schritt noch einmal durch und finden x'' . Da a_1 endlich viele Nachbarn hat, können wir den Schritt nur endlich oft durchführen. Es existiert also ein Gebiet mit a_1, x und z als Ecken und x und z sind Nachbarn von a_1 . Es sei angemerkt, dass x am äusseren Gebiet von G liegen kann.

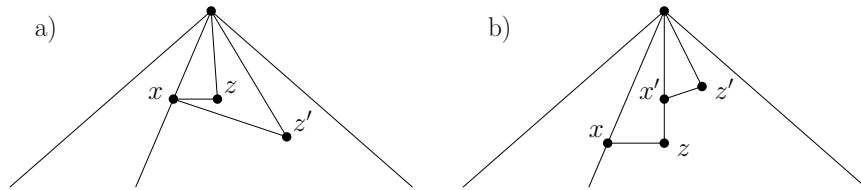


Abbildung 3.15 – a) Es können keine zwei Knoten mit geradlinigen Pfaden zu x und a_1 existieren. b) Falls a_1 kein Nachbar von z ist, dann müssen die Knoten x' und z' existieren.

Wir werden je nach Fall unterschiedliche Ansätze wählen, um G' zu erzeugen. Es folgt eine Übersicht der Fälle, die in Abbildung 3.16 skizziert sind.

1. Wenn x und z Nachbarn sind:
 - a) Und z dem Gebiet auf der anderen Seite von (x, z) zugewiesen ist, dann kontrahieren wir (x, z) .
 - b) Falls sonst $\deg(z)=3$ gilt, löschen wir z und fügen eine passende Kante ein.
 - c) Falls sonst (a_1, x) kontrahierbar ist, wird sie kontrahiert.
 - d) Sonst wird die Kante (a_1, z) gelöscht.
2. Wenn x und z keine Nachbarn sind, dann kontrahieren wir die Kante (a_1, x) .

Für jeden dieser Fälle entsteht ein Graph G' , der kleiner ist als G . Wir werden zeigen, dass er eine SLTR hat, und da er kein Gegenbeispiel sein kann, besitzt er ein

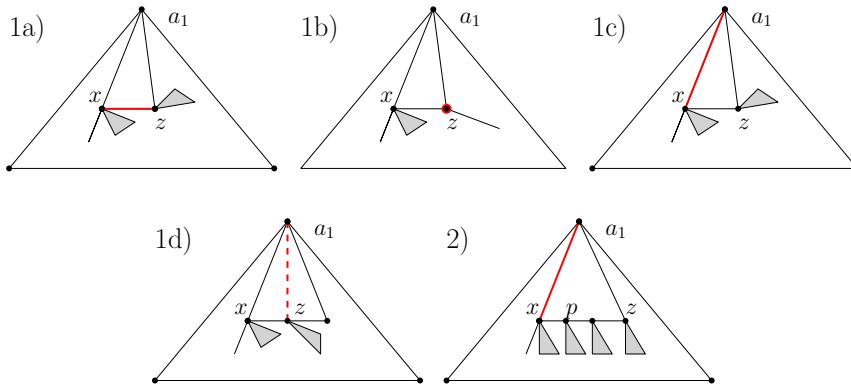


Abbildung 3.16 – Die möglichen Fälle bei der Erzeugung von G' . Die durchgezogenen roten Kanten werden kontrahiert. Der rote Knoten und die gestrichelte rote Kante werden gelöscht.

Ecken kompatibles Paar. Wir werden das Schnyder Labeling von G' zu einem von G erweitern und zeigen, dass dieses Ecken kompatibel mit dem von Δ induzierten FAA ist. Wir wenden uns nun den einzelnen Fällen im Genauen zu.

Im Folgenden seien x und z benachbart. Das Dreieck f_{a_1zx} in Δ hat somit keine weiteren Knoten auf den Rändern.

Fall 1a: Wir erzeugen G' , indem wir die Kante (a_1, z) löschen und (x, z) kontrahieren. Wir bezeichnen den neuen Knoten mit x' (siehe Abbildung 3.17 1b). Wir erhalten das FAA ϕ' von G' , indem wir die Zuweisung von x aus ϕ löschen. Sei γ' ein begrenzender Zykel in G' . Betrachte den induzierten Zykel γ in G . Dieser hat mindestens drei kombinatorisch konvexe Ecken. Diejenigen Ecken, die nicht x und z sind, müssen auch Ecken von γ' sein. Falls entweder x oder z eine kombinatorisch konvexe Ecke von γ ist, dann ist x' eine kombinatorisch konvexe Ecke von γ' . Angenommen x und z sind beide kombinatorisch konvexe Ecken von γ . Merke, dass es sich bei γ nicht nur um das Dreieck f_{a_1zx} handeln kann, da γ' nicht von einem Pfad induziert wird. Somit muss entweder an x oder z mindestens ein weiterer Winkel im Inneren von γ liegen. Angenommen ein Winkel an z , aber nicht der im Dreieck f_{a_1zx} , liegt im Inneren von γ . Dann zeigt dieser Winkel nicht in Richtung von x . Somit muss der zugewiesene Winkel an z im Inneren von γ liegen (wie in Abbildung 3.17 a). Es folgt, dass γ mindestens vier kombinatorisch konvexe Ecken in G hat. Falls γ einen weiteren Winkel an x in seinem Inneren hat, dann kann man analog zeigen, dass γ wieder mindestens vier kombinatorisch konvexe Ecken in G hat. Somit hat G' eine SLTR.

Da G' weniger Knoten als G hat, kann es kein Gegenbeispiel sein. Sei σ' ein zu ϕ' Ecken kompatibles Schnyder Labeling. Nach Lemma 3.14 kommt das Label des zugewiesenen Winkels an x' nur einmal vor. Wir kehren nun die Kontraktion von (x, z) um (vergleiche Abbildung 3.17 b). Die Winkel von a_1, x und z im Dreieck f_{a_1zx} haben Label 1, 2 und 3 im Uhrzeigersinn bei a_1 beginnend. Wir geben dem zugewiesenen Winkel an z Label 1. Der Winkel an z , dem wir ein neues Label gegeben haben, ist keine Ecke und sonst hat sich im Bezug auf ϕ' nichts verändert. Somit

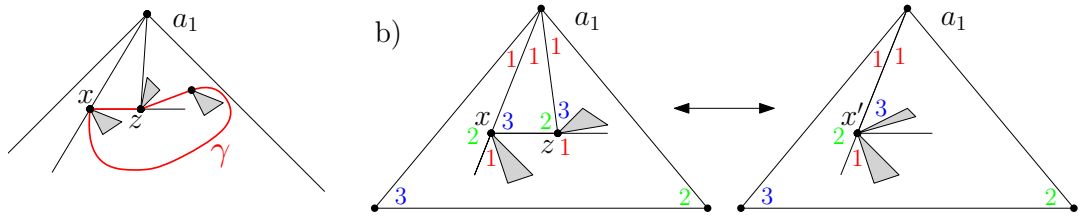


Abbildung 3.17 – a) Es existiert ein weiterer Winkel im Inneren von γ . b) Wir erhalten G' durch Kontraktion der Kante (x, z) .

ist das konstruierte Schnyder Labeling σ Ecken kompatibel zu ϕ . G kann also kein Gegenbeispiel sein.

Fall 1b: Der Knoten z habe Grad 3. Falls der Knoten z von ϕ zugeordnet ist, dann kann er keinem zu (x, z) adjazenten Gebiet zugeordnet sein (wegen Fall 1a). Angenommen z ist einem Gebiet zugeordnet, zu dem, wie in Abbildung 3.18, a_1 gehört. Dann existiert ein unterteilendes Dreieck. Die gestrichelte Gerade muss in Δ existieren, weil z Grad 3 hat. Somit ist G kein Gegenbeispiel. Der Knoten z kann also nicht zugewiesen sein.

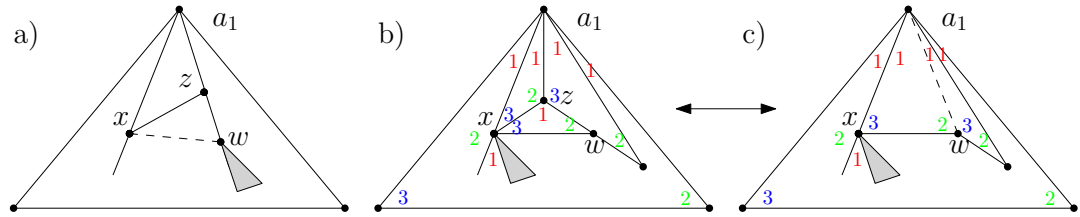


Abbildung 3.18 – a) Falls w dem Gebiet mit a_1 zugewiesen ist, existiert ein unterteilendes Dreieck. b),c) Das Löschen von z , falls w dem Gebiet mit a_1 zugewiesen ist oder wenn w Grad 3 hat.

Sei w der dritte Nachbar von z . Dieser muss zugewiesen sein, da sonst wieder ein unterteilendes Dreieck mit z in seinem Inneren existiert (man stelle sich in Abbildung 3.18 b) eine Gerade zwischen w und a_1 vor). Wir erhalten nun G' , indem wir z löschen und eine noch zu definierende Kante einfügen. Falls w dem Gebiet mit der Kante (a_1, z) zugewiesen ist (vergleiche Abbildung 3.18 b und c) oder wenn w Grad 3 hat (vergleiche Abbildung 3.19), fahren wir fort wie folgt. Nachdem wir z gelöscht haben, fügen wir, in Abhängigkeit der Zuweisung von z , einer der beiden Kanten (x, w) und (a_1, w) ein (vergleiche Abbildungen 3.18 c) und 3.19 c). Wir erhalten das FAA ϕ' durch Löschung der Zuweisung von w aus ϕ . Wir können die Kante in der Zeichnung löschen und die neue Kante einfügen, ohne etwas anderes an Δ zu verändern. Somit hat G' eine SLTR. Sei σ' das zu ϕ' Ecken kompatible Schnyder Labeling von G' . Die jeweils resultierenden Labelings sind in Abbildung 3.18 c) und Abbildung 3.19 c) zu sehen. Merke, dass die Labelings an w die gleichen sind, falls a_1 und w keine Nachbarn in G sind, also noch Knoten auf der Geraden zwischen ihnen liegen.

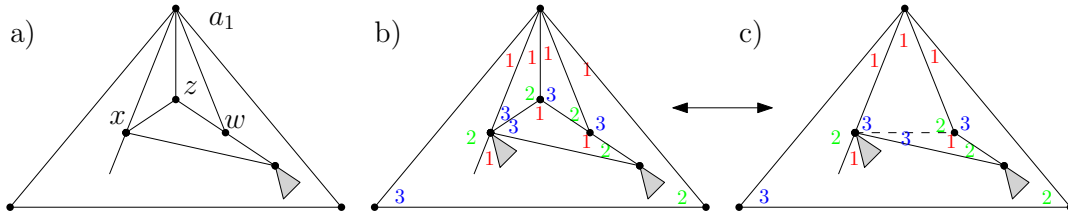


Abbildung 3.19 – Das Löschen von z , falls w dem Gebiet mit z und x zugewiesen ist oder wenn w Grad 3 hat.

Wir müssen nun z wieder einfügen und zeigen, dass wir ein Ecken kompatibles Paar erhalten. Um z wieder einzufügen, unterteilen wir die an w neu eingefügte Kante, setzen z auf ihr ein und verbinden z mit dem verbliebenen Nachbarn in G (x bzw. a_1). Die Label an den Enden der unterteilten Kante ändern sich nicht. An x bzw. a_1 übernehmen wir das Label, dass der jetzt geteilte Winkel in G' hatte und fügen es links und rechts der neuen Kante ein. Die Label um z sind eindeutig, weil z Grad 3 hat und Nachbar einer Aufhängung ist. Wir fügen nun die Zuweisung von w wieder ein, um ϕ zu erhalten. Der Winkel an z neben der Zuweisung von w erhält das gleiche Label wie diese. Somit haben die drei neuen Gebiete jeweils alle Label als Ecken und das erhaltene Schnyder Labeling σ ist Ecken kompatibel zu ϕ . Wieder kann G kein Gegenbeispiel sein.

Nehmen wir nun an, dass w dem Gebiet mit der Kante (x, z) zugewiesen ist und mindestens Grad 4 hat. In diesem Fall muss der Vorgang von eben nicht zwangsläufig möglich sein. Betrachte Abbildung 3.19 c) und nehme an, dass w mindestens Grad 4 hat. Dann können wir die Label nicht mit Sicherheit vorhersagen. Zum Beispiel könnten hier die Label um das Gebiet f_{xwp} entgegen dem Uhrzeigersinn um eine Position verschoben sein und es würde trotzdem ein Schnyder Labeling vorliegen. Dann hätte aber w als einziger Knoten das Label 2 und nach der Zuweisung von w würde es keine Ecke mit Label 2 geben. Es läge also kein Ecken kompatibles Paar vor. Somit müssen wir unser Vorgehen, wie in Abbildung 3.20 illustriert, anpassen.

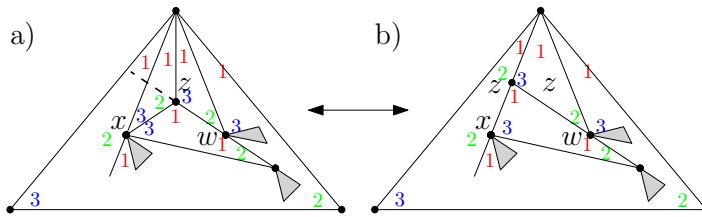


Abbildung 3.20 – Das Löschen von z , falls w mindestens Grad 4 hat. Wir verschieben den Knoten z entlang der gestrichelten Geraden auf die Kante (a_1, x) und erhalten so wieder eine SLTR.

Wir werden z nicht löschen, sondern auf der gestrichelten Geraden verschieben, bis wir die Kante (a_1, x) erreichen und dann diese Kante löschen. Wir projizieren also z in Δ auf die Gerade (a_1, x) . Dies ist möglich, da sich z in Δ in der konvexen

Hülle von x, w und a_1 befindet und keine weiteren Knoten im Inneren dieser Hülle liegen. Wir erhalten somit den Graphen G' . Nach Konstruktion hat G' eine SLTR und wir erhalten das FAA ϕ' durch Hinzufügen der Zuweisung von z zu dem Gebiet, dem auch x zugewiesen ist. G' hat nun eine Kante weniger als G und ist somit kein Gegenbeispiel. Sei σ' das zu ϕ' Ecken kompatible Schnyder Labeling auf G' . Da wir in ϕ' eine Zuweisung hinzugefügt haben, erhalten wir ein Ecken kompatibles Paar auf G , da die einzig neuen Label im Gebiet $f_{a_1, z, x}$ eingefügt werden, das ein Dreieck ist. G kann also kein Gegenbeispiel sein.

Dies beendet den Beweis für den Fall, dass z Grad 3 hat.

Fall 1c: Als Folgerungen der Fälle 1a) und 1b) können wir annehmen, dass z mindestens Grad 4 hat und keinem Gebiet zugeordnet ist, welches an (x, z) grenzt. Somit können wir die Kante (a_1, x) kontrahieren und die Kante (x, z) löschen und erhalten G' (siehe Abbildung 3.21). Der Knoten z hat nun in G' mindestens Grad 3, weil er zuvor mindestens Grad 4 hatte. Wir erhalten das FAA ϕ' auf G' , indem wir die Zuweisung von x aus ϕ löschen.

Sei γ' ein begrenzender Zykel in G' und γ der korrespondierende begrenzende Zykel in G . Wir müssen zeigen, dass γ' drei kombinatorisch konvexe Ecken hat. Falls x eine kombinatorisch konvexe Ecke von γ ist, dann muss a_1 eine kombinatorisch konvexe Ecke von γ' sein. Nehmen wir an, dass z eine kombinatorisch konvexe Ecke von γ und keine kombinatorisch konvexe Ecke von γ' ist. Dann liegen alle Nachbarn von z , mit Ausnahme von x , auf dem Rand oder im Inneren von γ . Nehmen wir an, dass γ' nur zwei kombinatorisch konvexe Ecken hat. Betrachte Abbildung 3.22 a). Sei p der entgegen dem Uhrzeigersinn nächste Nachbar von z nach x . Auf dem Pfad von a_1 zu p muss mindestens ein Nachbar liegen, da z mindestens Grad 4 hat. Falls alle Nachbarn von z zwischen p und a_1 auf dem Rand von γ liegen und Gebieten außerhalb von γ zugewiesen sind, existiert ein unterteilendes Dreieck in G (siehe Abbildung 3.22 a). Falls der Pfad, wie in Abbildung 3.22 b) konkav ist, folgt dies nicht. Wir nennen (a_1, x) nicht kontrahierbar und verweisen auf Fall 1d. Falls der Pfad anders herum konvex ist, dann liegt mindestens eine kombinatorisch konvexe Ecke x' von γ zwischen p und a_1 . Der Zykel γ' hat somit die drei kombinatorisch konvexe Ecken a_1, p und x' .

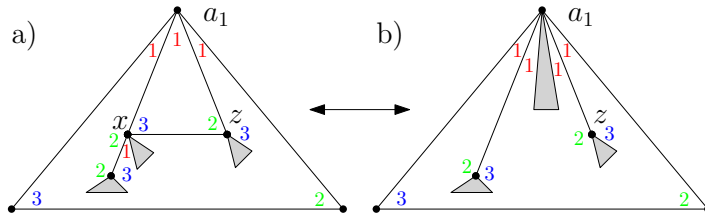


Abbildung 3.21 – Falls x dem Gebiet an (a_1, x) zugewiesen ist, (x, z) eine Kante ist und $\deg(z) \geq 4$, kontrahieren wir die Kante (a_1, x) und löschen (x, z) .

Betrachten wir den Fall, dass kein begrenzender Zykel γ in G existiert, der z als kombinatorisch konvexe Ecke hat. Weiter sei z keine kombinatorisch konvexe Ecke von γ' . Somit ist (a_1, x) kontrahierbar, jeder begrenzende Zykel in G' hat mindestens drei

kombinatorisch konvexe Ecken und G' hat eine SLTR. Somit existiert ein Schnyder Labeling σ' , das Ecken kompatibel zu ϕ' ist (wir haben ϕ' durch Löschen der Zuweisung von x erhalten). Wir können ein Schnyder Labeling σ' zu σ erweitern indem wir die Kontraktion einer Kante, die eine Aufhängung beinhaltet, umkehren (siehe Abbildung 3.22). Die resultierenden Winkel Label sind eindeutig. Wir begründen die Ecken-Kompatibilität. Gebiet $f_{a_1,z,x}$ ist ein Dreieck und wir setzen beginnend bei a_1 die Label 1, 2 und 3 ein. Hinzu kommt, dass der von ϕ zugewiesene Winkel von x nicht als einziger Label 2 hat. G kann also kein Gegenbeispiel sein.

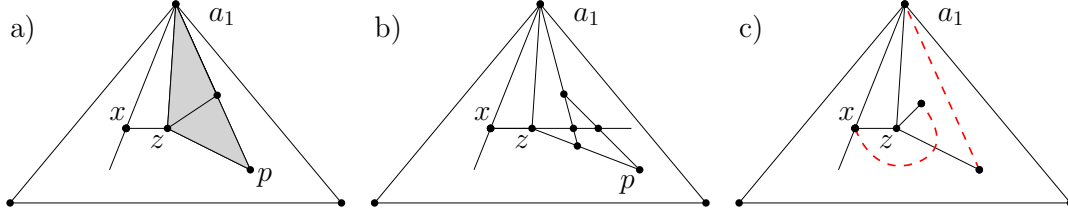


Abbildung 3.22 – a) Falls der Pfad zwischen a_1 und p entlang einer Geraden verläuft, dann existiert ein unterteilendes Dreieck. b) Wenn der Pfad konkav ist, dann können wir (a_1, x) nicht kontrahieren. c) Nun muss jedoch der Nachbar von z , der im Uhrzeigersinn vor x liegt, a_1 entlang der gestrichelten Gerade „sehen“ können, ohne von einem anderen Nachbarn von z behindert zu werden. Somit kann dies für den im Uhrzeigersinn auf a_1 folgenden Nachbarn von z und x nicht gelten, wie die gestrichelte Kurve illustriert.

Fall 1d: Aufgrund der bearbeiteten Fälle können wir annehmen, dass z mindestens Grad 4 hat, keinem Gebiet zugeordnet ist, welches an (x, z) grenzt und (s_1, x) nicht kontrahierbar ist. Weil (s_1, x) nicht kontrahierbar ist, muss ein begrenzender Pfad γ in G existieren, sodass z auf dem Rand von γ liegt, alle Nachbarn von z ausser x auf oder innerhalb von γ liegen und γ genau drei kombinatorisch konvexe Ecken hat, zu denen z gehört. Betrachte Abbildung 3.22 b). Es liegen keine kombinatorisch konvexen Ecken von γ auf dem Pfad zwischen a_1 und p . Somit wird die in Abbildung 3.22 c) eingezeichnete Gerade zwischen p und a_1 nicht von der Nachbarschaft von z unterbrochen. Der Knoten p kann s_1 entlang dieser Gerade „sehen“ (an der Nachbarschaft vorbei). Da z keinem Gebiet zugeordnet ist, welches an (x, z) grenzt und keine kombinatorisch konvexe Ecke von γ ist (die alle Nachbarn von z ausser x beinhaltet), kann z nicht zugewiesen sein.

Wir erstellen G' durch Löschen der Kante (a_1, z) aus G . Das FAA ϕ' folgt, indem wir die Zuweisung von z zum Gebiet mit a_1, x und z zu ϕ hinzufügen (siehe Abbildung 3.23 b). Angenommen es existiert in G' eine begrenzender Zykel γ' , der nur zwei kombinatorisch konvexe Ecke bezüglich ϕ' hat. Dann muss z auf dem Rand von γ' liegen und z muss eine kombinatorisch konvexe Ecke des korrespondierenden Zyklus γ in G sein. Falls nun γ' alle Nachbarn von z in G' enthält, dann hat γ mindestens vier kombinatorisch konvexe Ecken oder z ist keine Ecke von γ , wie wir begründen werden. Wie in Abbildung 3.22 c) illustriert, kann der im Uhrzeigersinn erste Nachbar

von z nach a_1 den Knoten x nicht „sehen“, weil ein Nachbar vor x im Weg ist. Somit muss zwischen diesen beiden Nachbarn eine kombinatorisch konvexe Ecke von γ liegen. Nehmen wir an, dass γ' den zugewiesenen Winkel an z in G' einschließt. Weiter sei z eine kombinatorisch konvexe Ecke des korrespondierenden Zyklus γ . Dann muss γ mindestens vier kombinatorisch konvexe Ecken in G haben. Die beiden Ränder von γ zwischen a_1 und z müssen strikt konvex in der Zeichnung von G sein und somit jeweils eine weitere kombinatorisch konvexe Ecke von γ enthalten (vergleiche Abbildung 3.23 a).

Somit haben wir gezeigt, dass G' eine SLTR hat. Erneut kann G' kein Gegenbeispiel sein und wir erhalten ein Schnyder Labeling σ' , das Ecken kompatibel zu ϕ' ist. Wir können dies, wie in Abbildung 3.23 b) zu sehen ist, zu einem Schnyder Labeling σ von G erweitern, das Ecken kompatibel zu ϕ ist, da wir kein Label an einer Zuweisung aus ϕ ändern. Somit kann G kein Gegenbeispiel sein.

Dies beendet den Beweis für den Fall, dass x und z Nachbarn sind.

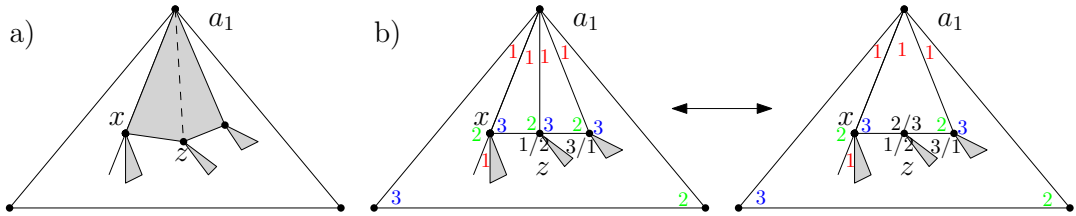


Abbildung 3.23 – a) Falls z eine kombinatorisch konvexe Ecke des begrenzenden Zyklus γ (um den grauen Teilgraphen) ist, dann hat γ mindestens vier kombinatorisch konvexe Ecken, da die Pfade von a_1 zu z auf beiden strikt Seiten konvex sind. b) Löschen der Kante (a_1, z) .

Fall 2: Seien nun x und z nicht benachbart, dann muss mindestens ein Knoten auf der Gerade zwischen ihnen liegen. Wir nennen diesen Knoten p . Der Knoten p muss dem Gebiet mit den Ecken a_1, z, x zugewiesen sein (vergleiche Abbildung 3.24). Wir wählen p , sodass die Kante (x, p) existiert. Wir erhalten G' durch die Kontraktion der Kante (a_1, x) . Das FAA ϕ' von G' folgt durch Löschen der Zuweisungen von x und p aus ϕ .

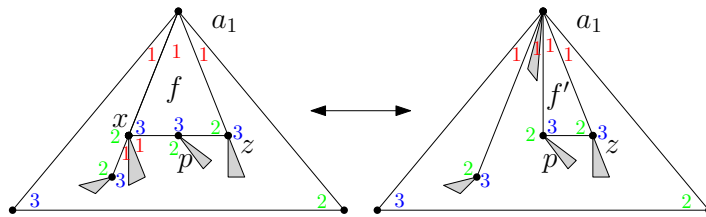


Abbildung 3.24 – Kontraktion der Kante (a_1, x) , wenn x einem Gebiet an (a_1, x) zugeordnet ist und x und z nicht benachbart sind.

Wir betrachten einen begrenzende Zykel γ in G' und seinen korrespondierenden

begrenzenden Zykel γ in G . Falls x eine kombinatorisch konvexe Ecke von γ ist, dann sind p oder a_1 die korrespondierende kombinatorisch konvexe Ecke von γ' . Beachte, dass wir die Zuweisung von p geändert haben. Falls p eine kombinatorisch konvexe Ecke eines korrespondierenden Zyklus γ in G ist, dann muss p in G einen Nachbar außerhalb von γ besitzen. Da sich der Grad von p in G' nicht ändert, muss dieser Nachbar auch außerhalb von γ' in G' liegen. Es folgt, dass p eine kombinatorisch konvexe Ecke von γ' ist, da p nicht zugewiesen ist. Somit hat jeder begrenzende Zykel in G' mindestens drei kombinatorisch konvexe Ecken und wir haben gezeigt, dass G' eine SLTR hat, die von ϕ' induziert wird.

Der Graph G' hat einen Knoten weniger als G und kann somit kein Gegenbeispiel sein. Sei nun σ' das zu ϕ' Ecken kompatible Schnyder Labeling von G' . Wir müssen nun zeigen, dass im Gebiet f mit den Ecken a_1, z und x , in dem auch p liegt, jede Ecke ein anderes Label hat. Sei f' das zu f korrespondierende Gebiet in G' (siehe Abbildung 3.24). Im Schnyder Labeling σ' hat der Winkel an p das Label 3 in f' , weil er auf der rechten Seite einer Kante zur Aufhängung mit Label 1 liegt. Der Winkel an x in f hat aus dem gleichen Grund ebenfalls Label 3 in σ . Somit haben weder p noch x in ihren von ϕ zugewiesenen Gebieten Label in σ , die in diesen Gebieten nur einmal vorkommen. Da ϕ' Ecken kompatibel zu σ' ist, bilden auch σ und ϕ ein Ecken kompatibles Paar zur SLTR Δ von G .

Wir haben somit für alle aufgelisteten Fälle einen Widerspruch zur Annahme erhalten, dass G ein minimales Gegenbeispiel ist. Dies beendet den Beweis. \square

In einem letzten Lemma werden wir nun zeigen, dass kein Gegenbeispiel G existieren kann, das eine SLTR (mit dem induzierten FAA ϕ) zulässt, auf dem aber kein Schnyder Labeling σ existiert, welches Ecken kompatibel zu ϕ ist.

Lemma 3.25. *Sei G ein ebener, intern-3-zusammenhängender Graph mit den Aufhängungen $\{a_1, a_2, a_3\}$, der eine SLTR besitzt. Sei ϕ das von dieser SLTR induzierte FAA. Dann existiert ein Schnyder Labeling σ von G , sodass σ und ϕ ein Ecken kompatibles Paar bilden.*

Beweis: Angenommen das Lemma gilt nicht. Sei G ein minimales Gegenbeispiel, wie zuvor mit der minimalen Anzahl an Knoten und unter diesen mit der minimalen Anzahl an Kanten. Sei Δ eine SLTR von G mit den Aufhängungen a_1, a_2, a_3 und ϕ das induzierte FAA. Wir wollen wieder zu einem Widerspruch gelangen, indem wir einen kleineren Graphen G' konstruieren, sodass die folgenden Punkte erfüllt sind.

- G' besitzt ein FAA ϕ' , welches von ϕ induziert wird.
- Das kompatible Paar (σ', ϕ') von G' induziert ein Schnyder Labeling von G .
- Das so erzeugte Schnyder Labeling σ ist Ecken kompatibel zu ϕ .

Diese Aussagen zusammen erzeugen einen Widerspruch zur Annahme, dass G ein Gegenbeispiel ist. Wir halten einige Beobachtungen aus den vorherigen Lemmata fest.

B1 Δ hat kein unterteilendes Dreieck.

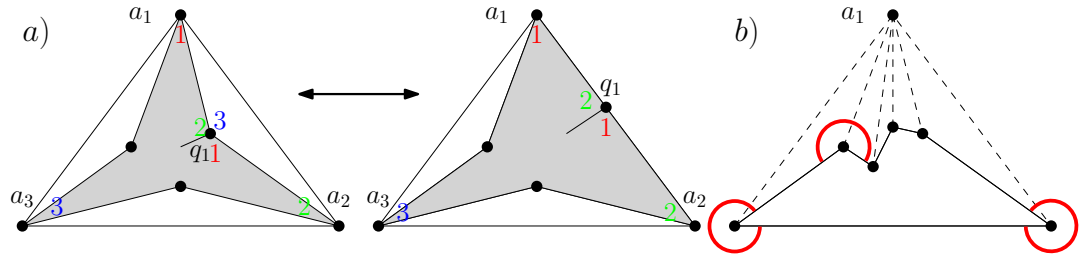


Abbildung 3.25 – a) Die Erstellung eines Graphen G' mit weniger Kanten. b) Das Löschen einer Aufhängung resultiert in einem Graphen mit mindestens drei Außenwinkeln $\geq \pi$.

B2 Δ hat kein teilendes Segment und somit keine Aufhängung von Grad 2.

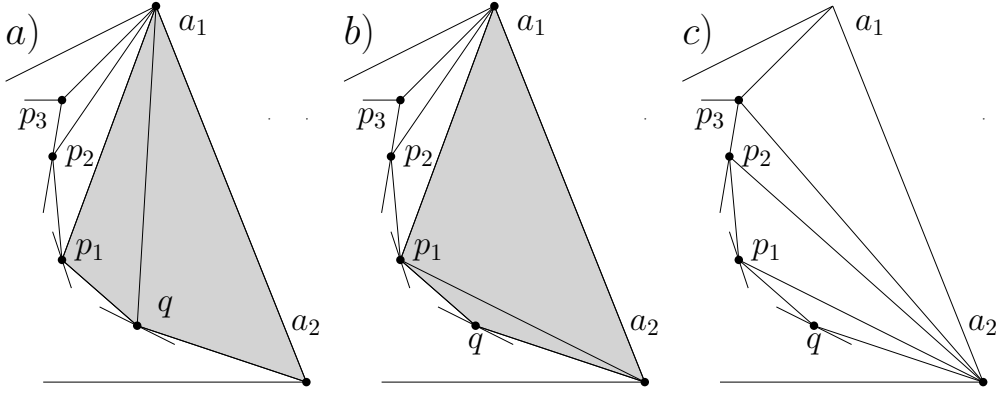
B3 Kein Nachbar x_i einer Aufhängung a_i kann einem Gebiet zugeordnet sein, zu dessen Rand die Kante (a_i, x_i) gehört. Somit bilden die Aufhängungen von Δ ein Dreieck.

Aus B3 folgt, dass sowohl das äußere Gebiet als auch seine drei benachbarten Gebiete (echte) Dreiecke⁴ sind (siehe Abbildung 3.25 a). Wir werden sehen, dass der dritte Knoten in einem der inneren Dreiecke eine wichtige Rolle spielt. Wir bezeichnen den dritten Knoten im Dreieck, das a_i und a_{i+1} enthält, mit q_i (oder einfach nur q).

Der Beweis läuft in drei Schritten. Wir zeigen zuerst, dass in einem minimalen Gegenbeispiel Δ jedes q_i einem angrenzenden Gebiet zugeordnet sein muss (und in diesem einen flachen Winkel hat). Als nächstes zeigen wir, dass wir aus G einen Graphen G' erzeugen können, der einen zugewiesenen Knoten q' auf einer Kante (a_i, q') besitzt und erhalten einen Widerspruch zu B3. Zuletzt zeigen wir, dass wir aus einem Schnyder Labeling σ' von G' ein Labeling σ von G erzeugen können, sodass aus der Ecken Kompatibilität von σ' und ϕ' auch Kompatibilität von σ und ϕ folgt.

Sei f das von a_1, a_2 und q_1 gebildete Dreieck (siehe Abbildung 3.25 a). Angenommen q_1 wird von ϕ nicht zugeordnet. Entferne die Kante (a_1, a_2) und weise q_1 in dem äußeren Gebiet zu, um G' und σ' zu erhalten. Kein begrenzender Zykel enthält den neu zugewiesenen Winkel an q in seinem Inneren. Falls a_1 und a_2 auf einem begrenzenden Zykel γ' liegen, dann ist q keine kombinatorisch konvexe Ecke des korrespondierenden Zyklus γ in G , sondern muss in seinem Inneren liegen (sonst wäre γ' von einem Pfad induziert). Somit sind die kombinatorisch konvexen Ecken von γ auch kombinatorisch konvexen Ecken von γ' . Da Zykel, die q_1 nicht enthalten, keine Ecken verlieren, hat jeder begrenzende Zykel in G' mindestens drei kombinatorisch konvexe Ecken. Somit hat G' ein SLTR Δ' und die geraden Winkel werden von ϕ' induziert. Nun ist G' ein Graph mit weniger Kanten als G und somit kein Gegenbeispiel. Wir erhalten also ein Ecken kompatibles Paar (σ', ϕ') . Wir können dies wie in Abbildung 3.25 a) zu einem Paar für G erweitern, indem wir die Zuweisung für q_1 löschen und im Dreieck q_1, a_2, a_1 die passenden Label einfügen.

⁴Sie enthalten nur jeweils ihre drei Ecken als Knoten.



Abbildungung 3.26 – Das schrittweise Flippen von Kanten.

Nehmen wir also an, dass jedes der q_i in einem Gebiet einen flachen Winkel hat, also von ϕ einem Gebiet zugeordnet wird. Jede Aufhängung a_i muss mindestens einen Nachbarn x_i haben, der nicht zugeordnet ist. Nach Lemma 3.24 kann kein Nachbar einer Aufhängung einem Gebiet an dieser Aufhängung zugewiesen sein. Somit würde ein Widerspruch zur Tatsache entstehen, dass $G \setminus \{a_1\}$ drei konkave Außenwinkel hat (siehe Abbildung 3.25 b). Der Knoten mit Winkel $\geq \pi$, der keine Aufhängung ist, kann nicht zugeordnet sein. Sei nun ohne Beschränkung der Allgemeinheit q der gemeinsame Nachbar von a_1 und a_2 . Der Knoten p_k sei der im Uhrzeigersinn von q ausgehend erste Nachbar von a_1 , der nicht zugewiesen ist. In Abbildung 3.26 ist dies p_3 .

Wir werden einen Graphen G' (mit der gleichen Anzahl an Knoten und weniger Kanten) aus G erstellen. In diesem Graphen ist p_k einem Gebiet zugeordnet, auf dessen Rand die Kante (a_1, p_k) liegt. Nach Lemma 3.24 muss somit ein Schnyder Labeling σ' existieren, dass Ecken kompatibel mit dem induzierten FAA ϕ' von G' ist. Wir können aus σ' ein Schnyder Labeling σ für G bauen, dass kompatibel zu ϕ ist.

Sei $N_{a_1} = \{q, p_1, \dots, p_k, \dots, p_l, a_3, a_2\}$ die Menge der Nachbarn von a_1 , wobei wir mit q beginnen und im Uhrzeigersinn fortfahren. Durch Löschen der Kante (a_1, q) und Einfügen der Kante (a_2, p_1) erhalten wir einen neuen Graphen G_1 . Wir nennen den durchgeführten Kantenwechsel, um G_1 zu erhalten, (nach dem Englischen) einen *Flip*. Nun lässt G_1 die SLTR Δ_1 zu. Die vier Knoten a_1, a_2, q und p bilden in Δ ein konvexes Viereck⁵ mit nur einer Diagonalen im Inneren (vergleiche Abbildung 3.26 a, b). Die durch den Flip, also das Wechseln der Diagonalen, erhaltene Zeichnung Δ_1 ist somit ebenfalls eine SLTR. Falls p_1 zugewiesen ist, wiederholen wir den Schritt (siehe Abbildung 3.26 c). Wir führen diesen Schritt nur so oft durch, bis wir bei p_k angekommen sind, dem ersten nicht zugewiesenen Nachbarn von a_1 . Wir führen einen letzten Flip im konvexen Viereck mit den Ecken a_1, a_2, p_{k-1}, p_k durch. Wir ersetzen also die Diagonale (a_1, p_{k-1}) durch (a_2, p_k) . Nach jedem Flip erhalten wir ein SLTR Δ_i und ein indiziertes Gutes-FAA ϕ_i von G_i . Somit ist ϕ_k ein Gutes-FAA.

Um nun den Graphen G' zu erhalten, löschen wir die im letzten Flip hinzugefügte

⁵ Auf der Geraden zwischen q und p_1 können sich zusätzliche Knoten befinden, doch dies spielt für unsere Argumentation keine Rolle.

Kante (a_2, p_k) (siehe Abbildung 3.27 b). G' hat somit eine Kante weniger als G . p_k ist in ϕ_k nicht zugewiesen. Wir konstruieren ϕ' durch Hinzufügen der Zuweisung von p_k zum Gebiet a_1, a_2, p_{k-1}, p_k zu ϕ (siehe Abbildung 3.27 a). Das so konstruierte ϕ' ist ein FAA von G' . Wir zeigen als Nächstes, dass es sich bei ϕ' um ein Gutes-FAA handelt.

Betrachten wir einen beliebigen begrenzenden Zykel γ' in G' und sei γ_k der korrespondierende Zykel in G_k . Wir müssen wieder zeigen, dass jeder Zykel mindestens drei kombinatorisch konvexe Ecken hat. Falls p_k nicht auf dem Rand von γ' liegt, folgt sofort, dass beide Zyklen die selben kombinatorisch konvexen Ecken haben (und somit beide mindestens drei). Betrachten wir also den Fall, dass p_k auf dem Rand von γ' liegt. Falls γ' nur einen Teil der Nachbarschaft von p_k beinhaltet und der zugewiesene Winkel an p_k nicht im Inneren von γ' liegt, dann ist p_k eine kombinatorisch konvexe Ecke von γ_k und somit auch von γ' . Sei also p_k keine kombinatorisch konvexe Ecke von γ' . Falls der zugewiesene Winkel an p_k im Inneren von γ' liegt, aber γ' nicht alle Nachbarn von p_k enthält, muss γ_k mindestens vier kombinatorisch konvexe Ecken besitzen. Die ersten drei sind a_1, a_2 und p_k und die vierte befindet sich auf dem Teil R von γ' zwischen p_k und a_2 ⁶. Somit hat γ' mindestens drei kombinatorisch konvexe Ecken. Betrachte den Fall, dass γ' alle Nachbarn von p_k beinhaltet, aber der zugewiesene Winkel nicht im Inneren von γ' liegt. In der SLTR Δ_k ist der Außenwinkel von γ_k am Knoten p_k somit nicht grösser als π und γ_k muss drei andere (kombinatorisch konvexe) Ecken haben. Diese sind auch kombinatorisch konvexe Ecken von γ' . Somit hat jeder begrenzende Zykel γ' in G' mindestens drei kombinatorisch konvexe Ecken und es folgt, dass ϕ' ein Gutes-FAA ist.

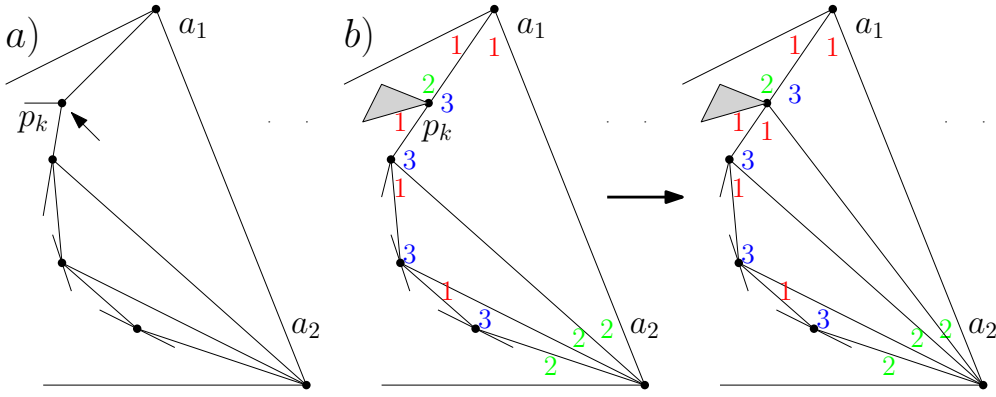
Da G' die gleiche Anzahl an Knoten und weniger Kanten hat als G , kann er kein Gegenbeispiel sein. Es existiert also ein Schnyder Labeling σ' , das Ecken kompatibel zu ϕ' ist. Es bleibt zu zeigen, dass wir aus σ' ein Labeling für G erstellen können, welches dann Ecken kompatibel zu ϕ ist. Wir stützen uns hierfür auf die folgende Eigenschaft, die aus Definition 3.15 K2 folgt.

K3 In einem Ecken kompatiblen Paar (σ, ϕ) hat jeder zugewiesene Winkel in diesem Gebiet (mindestens) einen benachbarten Winkel mit demselben Label.

Das Schnyder Labeling σ' ist eindeutig für Gebiete mit a_1 oder a_2 als Ecken und einer geflippten Kante auf dem Rand (siehe Abbildung 3.27 b). In einem ersten Schritt müssen wir (a_2, p_k) wieder einfügen und σ' zu σ_k erweitern (wie in Abbildung 3.27 b zu sehen ist).

Die Flips wurden entlang q, p_1, \dots, p_k durchgeführt und nach unserer Annahme haben die Knoten p_1, \dots, p_{k-1} von ϕ' zugewiesene Winkel in G' . Sei $p_0 = q$. Zwei Knoten p_{i-1} und p_i , mit $i \in \{1, \dots, k\}$ in dieser Folge sind nicht zwangsläufig benachbart, aber sie sind die Ecken des (dreieckigen) Gebietes a_2, p_{i-1}, p_i und alle Knoten auf dem Rand zwischen p_{i-1} und p_i sind diesem Gebiet zugewiesen (siehe Abbildung 3.28 a). Wir führen nun Schritt für Schritt, beginnend mit $i = k$, einen *Rückwärts-Flip*

⁶Diese Ecke muss existieren, da der begrenzende Zykel $\tilde{\gamma}$, der sich aus R und der Kante (a_2, p_k) bildet, mindestens drei kombinatorisch konvexe Ecken in G_k hat.

Abbildung 3.27 – Ein Schnyder Labeling von G_k , erstellt aus einem Labeling von G' .

durch. Hierfür entfernen wir die Kante (p_i, a_2) und setzen die Kante (p_i, a_1) wieder ein. Merke, dass p_i vor dem Schritt i einen zusätzlichen Nachbarn in G_i (im Verhältnis zu G) hat und somit gilt $\deg(p_i) \geq 4$, weil G intern-3-zusammenhängend ist. Wir benennen zwei der Winkel um den Knoten p_i . Sei α_i der gegen den Uhrzeigersinn auf die Kante (p_i, a_1) folgende Winkel und β_i der im Uhrzeigersinn auf die Kante zu (p_i, p_{i-1}) folgende Winkel an p_i (vergleiche Abbildung 3.28 a). Dann handelt es sich bei diesen Winkeln, wegen $\deg(p_i) \geq 4$, nicht um denselben. Bei jedem Schritt des Rückwärts-Flippens wird nun die folgende Invariante eingehalten.

Invariante Vor dem Schritt i hat der Knoten p_i die Label $2, 3, 1, 1$ im Uhrzeigersinn beginnend mit den Winkel α_i und endend mit β_i (siehe Abbildung 3.28 a). Zusätzlich bilden das Schnyder Labeling σ'_i und das FAA ϕ'_i ein Ecken kompatibles Paar auf G'_i .

Beginnen wir mit G' . Hier sind die in Abbildung 3.27 b) gewählten Label der Winkel um p_k die einzig mögliche Wahl. Genauso verhält es sich nach dem Einfügen der Kante (a_2, p_k) . Wir erhalten den Graphen G'_k , das Schnyder Labeling σ'_k und das FAA ϕ'_k , wobei σ'_k und ϕ'_k ein Ecken kompatibles Paar bilden. Somit gilt die Invariante vor dem ersten Schritt.

Die Invariante gelte vor Schritt i . In Schritt i entfernen wir die Kante (a_2, p_i) und fügen die Kante (a_1, p_{i-1}) wieder in den Graphen ein. Vor dem Rückwärts-Flip, erfüllen die Label um p_i die Invariante und σ'_i und ϕ'_i sind ein Ecken kompatibles Paar. Die folgende Argumentation basiert auf Lemma 2.8, das besagt, dass alle drei Label im Uhrzeigersinn an jeder Kante vorkommen, und der Ecken Kompatibilität von σ'_i und ϕ'_i . Die Winkel an p_{i-1} links und rechts der Kante zu p_i haben die Label 2 und 3 (siehe Abbildung 3.28 b). Der Winkel zwischen p_i und a_2 kann nur Label 3 haben, da er an einer Kante zu a_2 liegt, an deren anderen Ende somit nur Label 2 vorkommt. Der andere Winkel muss Label 2 haben, da er entweder an einer Kante mit zwei 3ern auf der anderen Seite liegt (vergleiche Abbildung 3.28 a) oder er an der Kante (p_i, p_{i-1}) liegt, welche p_i nur Label 1 hat (vergleiche Abbildung 3.28 b).

Führen wir den *Rückwärts-Flip* durch. Wir löschen also die Kante (a_2, p_i) und fügen die Kante (a_1, p_{i-1}) ein, wie in Abbildung 3.28 b). Für fast alle Winkel folgt die Wahl

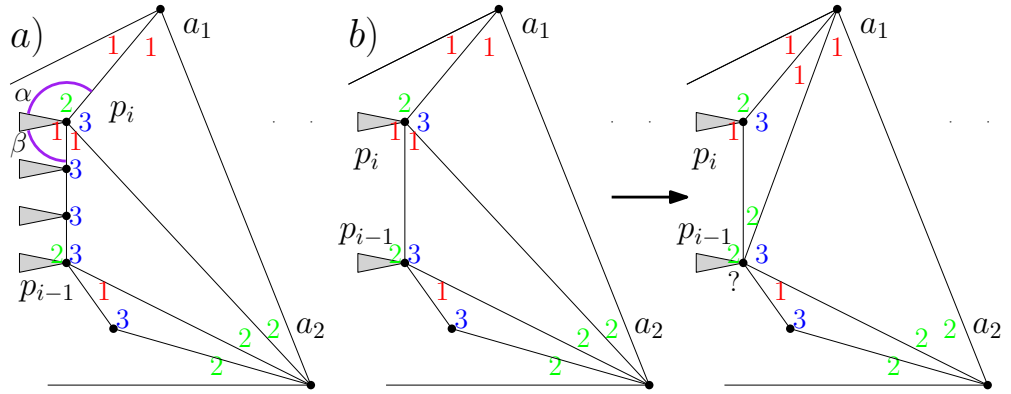


Abbildung 3.28 – a) Schnyder Labeling zwischen p_{i-1} und p_i mit den Winkeln α und β in violett. b) Änderung des Labelings beim Rückwärts-Flippen.

des Labels eindeutig, da σ'_{i-1} ein Schnyder Labeling sein muss. Um die Invariante zu erfüllen, müssen wir zeigen, dass wir dem mit einem Fragezeichen markierten Winkel in Abbildung 3.28 b) Label 1 geben können. Wenn er dieses Label schon vor dem Schritt hatte, ändert sich nichts und wir sind fertig. Angenommen, er hatte nicht Label 1. Da es sich um ein Schnyder Labeling handelt, ist die einzig andere Möglichkeit Label 2. Es muss sich bei diesem Winkel um den an p_{i-1} zugewiesenen handeln. Dies folgt aus der Ecken Kompatibilität, da ein zugewiesener Winkel und seine beiden Nachbarn um einen Knoten nicht alle die gleichen Label haben können (siehe Abbildung 3.29 a), sonst würde es zu einem Widerspruch kommen. Es existieren mindestens zwei weitere Winkel an p_{i-1} mit Label 2 (also insgesamt mindestens drei). Somit würde der Fall aus Abbildung 3.29 a) eintreten und ein angrenzendes Gebiet hätte keine Ecke mit Label 2. Der Winkel mit dem Fragezeichen muss somit zugewiesen sein.

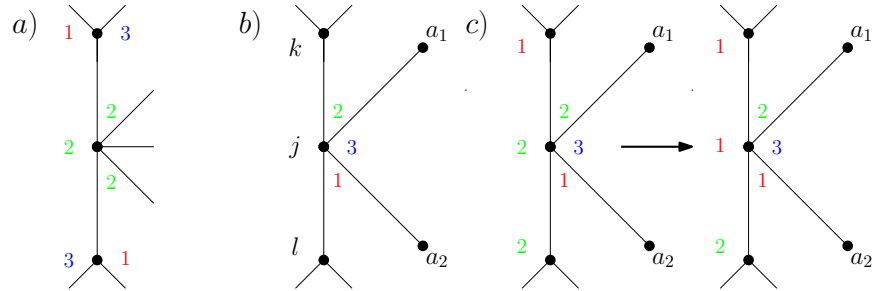


Abbildung 3.29 – a) Das Gebiet links kann keine Ecke mit Label 1 haben. b) Eine freie Wahl von Winkel in einem Schnyder Labeling. c) Nach der Änderung von Label 2 zu Label 1 am zugewiesenen Knoten liegt weiterhin ein Ecken kompatibles Paar vor.

Betrachte Abbildung 3.29 b). Wenn wir $j=2$ setzen, dann folgen sofort die Label in c). $k=1$ folgt aus der Kanten Regel, weil die Kante und p_{i-1} auf beiden Seiten Label 2 hat, und $l=2$, weil sonst das Gebiet auf der linken Seite keine Ecke mit Label 2 haben könnte. Wir können nun Label $j=1$ setzen, ohne andere Label zu verändern. Wir

erhalten ein Schnyder Labeling σ'_{i-1} (siehe Abbildung 3.29 c), das Ecken kompatibel mit dem induzierten FAA ϕ'_{i-1} ist, da wir nur das Label eines zugewiesenen Winkels verändert haben.

Wir haben also Schritt i durchgeführt und die Invariante hat Bestand. Per Induktion erhalten wir ein Ecken kompatibles Paar aus einem Schnyder Labeling σ'_0 und einem FAA ϕ'_0 von G . Somit kann G kein Gegenbeispiel sein. Es folgt die Rückrichtung des Theorems. \square

Wir haben nun alle nötigen Hilfsmittel erarbeitet, um das folgende, weiter oben schon aufgeführte Theorem zu beweisen.

Theorem 3.17. *Sei G ein ebener, intern-3-zusammenhängender Graph mit den Aufhängungen $\{a_1, a_2, a_3\}$. G besitzt eine SLTR, genau dann, wenn ein Ecken kompatibles Paar (σ, ϕ) aus einem Schnyder Labeling σ und einem FAA ϕ existiert.*

Beweis von Theorem 3.17: Sei G ein ebener, intern-3-zusammenhängender Graph mit den Aufhängungen $\{a_1, a_2, a_3\}$. Angenommen G besitzt ein Ecken kompatibles Paar (σ, ϕ) aus einem Schnyder Labeling σ und einem FAA ϕ . Nach Lemma 3.18 existiert dann eine SLTR von G .

Angenommen G besitzt eine SLTR, dann existiert nach Lemma 3.25 ein Schnyder Labeling, das Ecken kompatibel mit dem von der SLTR induzierten FAA ist. \square

Theorem 3.17 liefert noch keinen direkten Weg, um für einen Graphen die Frage zu beantworten, ob er eine SLTR zulässt. Im nächsten Kapitel werden wir uns aber mit einem Algorithmus befassen, der den Zusammenhang aus Theorem 3.17 nutzt, um für einen Graphen ein Gutes-FAA zu erkennen, falls er eine SLTR zulässt.

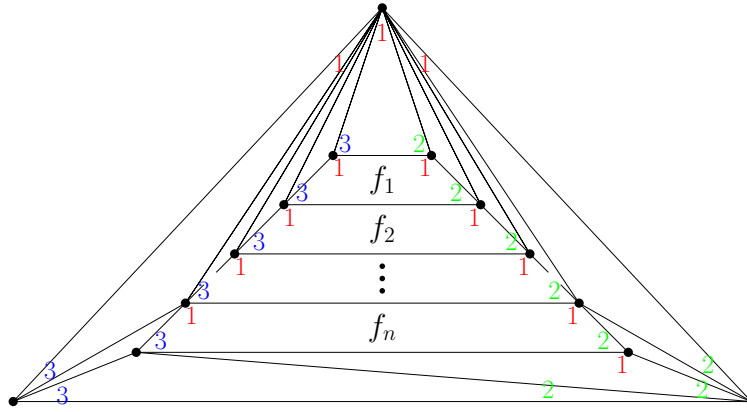


Abbildung 3.30 – Ein Graph mit exponentiell vielen SLTRs.

Beispiel 3.26. Wir können Beispiel 3.14 leicht anpassen, um mit Theorem 3.17 zeigen, dass es Graphen mit exponentiell vielen SLTRs gibt. Zu dem Schnyder Labeling σ des Graphen G auf $2n + 3$ Knoten, der in Abbildung 3.30 zu sehen ist, existieren 2^n viele FAAs, um ein Ecken kompatibles Paar zu bilden. Wir können in jedem Viereck

einen der beiden oberen Knoten zuweisen, unabhängig davon, welche Knoten in den restlichen Vierecken zugewiesen sind, um ein FAA ϕ zu konstruieren. Somit haben die inneren Gebiete jeweils eine Ecke (bezüglich ϕ) mit jedem Label aus σ .

4 Algorithmen zur Erkennung von SLTRs

Im vorherigen Kapitel wurden Kriterien für die Existenz einer SLTR für G erarbeitet. Angenommen, wir haben nun einen intern-3-zusammenhängenden ebenen Graphen G mit Aufhängungen a_1, a_2 und a_3 gegeben und wollen die Frage beantworten, ob dieser spezifische Graph eine SLTR hat.

Die beiden Hauptresultate aus Kapitel 3 geben uns jeweils die Möglichkeit für ein spezifisches FAA zu überprüfen, ob es eine SLTR induziert. Da es aber Graphen mit exponentiell vielen FAAs gibt, wie in Beispiel 3.2 gezeigt wurde, ist dies kein effizienter Weg, um die Frage zu beantworten. Im Falle der Ecken-Kompatibilität lässt sich dieses Vorgehen auch umkehren. Wir können für jeden Schnyder Wood überprüfen, ob zu diesem ein Ecken kompatibles FAA auf G existiert. Zumindest für Graphen, die nur polynominell viele Schnyder Woods haben, erhalten wir nach Aerts und Felsner so einen polynominellen Algorithmus, indem wir ein Matching-Problem lösen [AF15]. Es existieren allerdings Graphen auf n Knoten mit $3 \cdot 209^n$ vielen Schnyder Woods wie Felsner und Zinkfeld gezeigt haben [FZ08]. Es ist also nicht möglich, mit diesen Ansätzen einen Algorithmus zu konstruieren, um die Frage nach der Existenz einer SLTR effizient zu beantworten.

Um dies jedoch zu erreichen, werden wir uns in diesem Kapitel mit einem von Aerts und Felsner erarbeiteten Algorithmus beschäftigen [AF15]. Dieser Algorithmus charakterisiert die Antwort der oben gestellten Frage als ganzzahlige Lösung eines Gerichteten-Multi-Fluss-Problems. Wir stellen am Ende des Kapitels die Vermutung auf, dass man mithilfe dieses Ansatzes einen polynominellen deterministischen Algorithmus zur Bestimmung von SLTRs konstruieren kann.

4.1 SLTRs als Lösungen eines Zwei-Fluss-Problems

In einem ersten Schritt werden wir für einen gegebenen intern-3-zusammenhängenden ebenen Graphen G mit Aufhängungen a_1, a_2, a_3 , sowohl einen Schnyder Wood als auch ein FAA jeweils als ganzzahlige Lösung eines Gerichteten-Ein-Fluss-Problems charakterisieren. Die beiden so konstruierten Netzwerke werden dann zu einem Gerichteten-Zwei-Fluss-Problem kombiniert, sodass eine zulässige ganzzahlige Lösung einem Ecken kompatiblen Paar auf G entspricht. Als Resultat erhalten wir eine SLTR von G . Wir beschäftigen uns also im Verlauf des Kapitels mit verschiedenen Multi-Fluss-Problemen auf gerichteten Graphen, die wir in Definition 2.17 eingeführt haben.

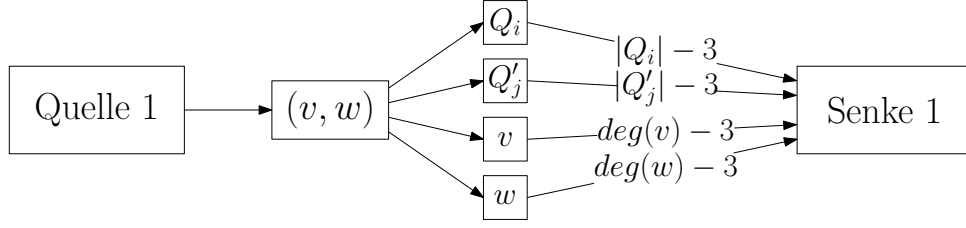


Abbildung 4.1 – Der Schnyder Wood Fluss durch eine innere Kante (v, w) . Die nicht beschrifteten Kanten haben Kapazität 1.

4.1.1 Schnyder-Wood-Fluss

Um einen Schnyder Wood als Fluss-Problem zu kodieren, kann man die in Abschnitt 2.3 eingeführten α_s -Orientierungen auf dem Abschluss von $G + G^*$ nutzen. Éric Fusy zeigt im Zuge der Untersuchung spezifischer α -Funktionen, dass sich α_s -Orientierungen von $G + G^*$ in linearer Zeit berechnen lassen[Fus07]. Daraus folgt, dass wir auch einen Schnyder Wood auf G in linearer Zeit erhalten.

Sei ein ebener intern-3-zusammenhängender Graph G mit Aufhängungen a_1, a_2, a_3 gegeben. Machen wir uns an die Konstruktion eines Netzwerks \mathcal{N}_S , sodass ein zulässiger ganzzahliger Fluss φ_s auf \mathcal{N}_S einer α_s -Orientierung von $G + G^*$ entspricht¹. Nach Theorem 4.4 würde die aus φ_s erhaltene α_s -Orientierung somit auch einen Schnyder Wood auf G induzieren. Besonderes Augenmerk ist hier auf die Möglichkeit einer späteren Kombination mit einem weiteren Netzwerk gelegt, in welchem eine ganzzahlige Lösung einem FAA auf G entspricht, um ein kombiniertes Netzwerk zu erstellen.

Wie schon erwähnt ist $G + G^*$ bipartit, Kanten-Knoten haben Grad 4, Knoten-Knoten haben Grad $\deg(v)$ und Gebiets-Knoten haben Grad $|f|$. Für eine α_s -Orientierung muss nach Definition 4.4 A3 jeder Kanten-Knoten Ausgangsgrad 1, jeder Knoten-Knoten Eingangsgrad $\deg(v) - 3$ und jeder Gebiets-Knoten Eingangsgrad $|f| - 3$ haben. Die Kanten-Knoten am äußeren Gebiet sind in $G + G^*$ immer nach außen orientiert. Somit reicht es, wenn wir nur die inneren Kanten-Kanten E_{in} betrachten und die Orientierung der äußeren Kanten voraussetzen.

Das im Folgenden konstruierte Netzwerk ist in Abbildung 4.1 skizziert und unter dem Punkt Netzwerk 1 zusammengefasst. Das Netzwerk \mathcal{N}_S hat eine Quelle s und eine Senke t und einen Knoten für jeden Knoten aus $G + G^*$ (bis auf die äußeren Kanten). Es existieren Kanten von der Quelle zu jedem Kanten-Knoten $e \in E_{in}$ mit Kapazität 1. Von den Kanten-Knoten e zu inzidenten Knoten-Knoten v und Gebiets-Knoten $f \in F_{in}$ fügen wir ebenfalls Kanten mit Kapazität 1 ein. Merke, dass der Fluss auf diesen Kanten bei einer ganzzahligen Lösung eine α_s -Orientierung auf $G + G^*$ ergeben soll. Nun fügen wir noch Kanten von $f \in F_{in}$ zur Senke mit Kapazitäten $|f| - 3$, Kanten von den (inneren) Knoten-Knoten $v \in V_{in} = V \setminus \{a_1, a_2, a_3\}$ zur Senke mit Kapazitäten $\deg(v) - 3$ und Kanten von den Aufhängungen a_i zur Senke mit Kapazitäten $\deg(a_i) - 2$. Die letzte Kapazität ergibt sich, da die Halbkante in $G + G^*$

¹Hier stehen die drei s jeweils für Schnyder Wood um spätere Verwechslungen zu vermeiden.

von a_i aus immer nach außen orientiert. Wir müssen somit nur noch zwei andere Kanten von a_i weg orientieren. Der Bedarf des Netzwerkes entspricht der Anzahl der inneren Kanten von G . Fassen wir zusammen.

Netzwerk 1 (Schnyder Wood). Bei \mathcal{N}_S handelt es sich um ein gerichtetes Netzwerk, das auf Basis eines ebenen intern-3-zusammenhängenden Graphen G mit Aufhängungen a_1, a_2, a_3 erstellt wird, um einen Schnyder Wood auf G zu finden. Ein Ausschnitt ist in Abbildung 4.1 dargestellt.

- \mathcal{N}_S hat eine Quelle s und eine Senke t
- Knoten in \mathcal{N}_S werden für jede innere Kante $e \in E_{in}$, jedes innere Gebiet $f \in F_{in}$ und jeden Knoten $v \in V$ aus G erzeugt.
- Es werden gerichtete Kanten der folgenden Typen in \mathcal{N}_S erzeugt:
 - (s, e) von der Quelle zu jeder inneren Kante mit $c(s, e) = 1$
 - $(e, v_1), (e, v_2)$ von jeder inneren Kante zu den Endknoten mit $c(e, v) = 1$
 - (e, f) von jeder inneren Kante zu adjazenten Gebieten mit $c(e, f) = 1$
 - (v, t) von den Knoten zur Senke mit $c(v, t) = 1$
 - (f, t) von den inneren Gebieten zur Senke mit $c(f, t) = |f| - 3$
 - (a_i, t) von den Aufhängungen zur Senke mit $c(f, t) = \deg(a_i) - 2$
 - (v, t) von den restlichen Knoten zur Senke mit $c(f, t) = \deg(v) - 3$
- \mathcal{N}_S hat Bedarf $d = |E_{in}|$

\Rightarrow Ein zulässiger ganzzahliger Fluss φ_s existiert genau dann, wenn ein Schnyder Wood auf G existiert.

Proposition 4.1. *Sei G ein ebener intern-3-zusammenhängender Graph mit Aufhängungen a_1, a_2, a_3 und \mathcal{N}_S wie in Netzwerk 1 konstruiert. Ein zulässiger ganzzahliger Fluss φ_s auf \mathcal{N}_S existiert genau dann, wenn ein Schnyder Wood auf G zu den Aufhängungen a_1, a_2, a_3 existiert. Insbesondere kodiert jeder Fluss einen Schnyder Wood.*

Beweis: Sei φ_s ein zulässiger ganzzahliger Fluss auf \mathcal{N}_S . Es gilt $|\varphi_s| = |E_{in}|$. Somit hat jede innere Kante-Knoten Ausgangsgrad 1. Weiter gilt

$$\sum_{v \in V} (\deg(v) - 3) + 3 + \sum_{f \in F_{in}} (|f| - 3) = |E_{in}|.$$

Somit sind die Kanten von den Kanten-Knoten und Knoten-Knoten ausgelastet. Der Fluss φ_s entlang einer Auskante von $e \in E_{in}$ in \mathcal{N}_S entspricht dann genau der Kante in $G + G^*$, die in der α_s -Orientierung von e weg orientiert wird. Durch die Knoten-Knoten v und Gebiets-Knoten f fließt Fluss auf $\deg(v) - 3$ bzw. $|f| - 3$ Kanten der von den Kanten-Knoten kommt. Somit lässt φ_s jeweils drei Kanten frei, denen wir noch keine Orientierung zugewiesen haben. Diese entsprechen somit den von v bzw. f

weg orientierten Kanten einer α_s -Orientierung auf $G + G^*$. Somit hat jedes Gebiet und jeder Knoten den passenden Ausgangsgrad und φ_s kodiert eine α_s -Orientierung auf $G + G^*$. Auf analoge Weise lässt sich aus einer α_s -Orientierung ein zulässiger ganzzahliger Fluss φ_s erstellen. Es existiert also genau dann ein Schnyder Wood auf G , wenn eine ganzzahlige Lösung φ_s für \mathcal{N}_S existiert. \square

4.1.2 FAA-Fluss

Sei wieder ein ebener intern-3-zusammenhängender Graph G mit Aufhängungen a_1, a_2, a_3 gegeben. Ein FAA ordnet nun jedem Gebiet $|f| - 3$ flache Winkel zu und jeder Knoten darf maximal einem Gebiet zugeordnet werden. Die adjazenten Knoten um das äußere Gebiet, die keine Aufhängungen sind, müssen diesem zugewiesen werden. Wir konstruieren ein Netzwerk \mathcal{N}_F , für das ein ganzzahliger zulässiger Fluss einem FAA entspricht

Das im Folgenden konstruierte Netzwerk ist in Abbildung 4.2 skizziert und unter dem Punkt Netzwerk 2 zusammengefasst. Das Netzwerk \mathcal{N}_F hat jeweils eine Quelle und Senke. Wir erstellen in \mathcal{N}_F Knoten für jeden inneren Winkel (f, v) , mit $v \in V$ und $f \in F_{in}$, Knoten für alle inneren Gebiete $f \in F_{in}$ und für die inneren Knoten $v \in V_{in}$. Von der Quelle fügen wir Kanten mit Kapazität 1 zu jedem inneren Winkel (f, v) , und von jedem inneren Winkel (f, v) jeweils eine Kante zu f und zu v (falls $v \in V_{in}$) mit Kapazität 1 ein. Merke, dass so der Fluss von den Winkeln zu den inneren Knoten einem FAA entspricht. Zuletzt erstellen wir Kanten von jedem inneren Gebiet f zur Senke mit Kapazität 3 und Kanten von jedem Knoten v zur Senke mit Kapazität 1 ein. Der Bedarf des Netzwerks ist $\sum_{f \in F_{in}} |f|$ und entspricht der Anzahl der inneren Winkel von G . Fassen wir zusammen.

Netzwerk 2 (FAA). Bei \mathcal{N}_F handelt es sich um ein gerichtetes Netzwerk, das auf Basis eines ebenen intern-3-zusammenhängenden Graphen G mit Aufhängungen a_1, a_2, a_3 erstellt wird, um ein FAA zu finden. Ein Ausschnitt ist in Abbildung 4.2 dargestellt.

- \mathcal{N}_F hat eine Quelle s und eine Senke t
- Knoten in \mathcal{N}_F werden für jeden inneren Winkel $(f, v) \in W_{in}$, jedes innere Gebiet $f \in F_{in}$ und jeden Knoten $v \in V$ aus G erzeugt.
- Es werden gerichtete Kanten der folgenden Typen in \mathcal{N}_F erzeugt:
 - $(s, (f, v))$ von der Quelle zu jedem inneren Winkel mit $c(s, (f, v)) = 1$

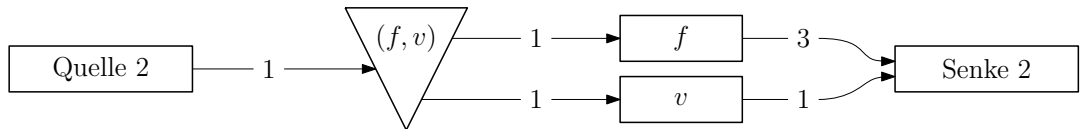


Abbildung 4.2 – Der FAA-Fluss durch einen Winkel (f, v) . Die nicht beschrifteten Kanten haben Kapazität 1.

- $((f, v), v)$ von jedem inneren Winkel zum Knoten mit $c((f, v), v) = 1$
- $((f, v), f)$ von jedem inneren Winkel zum Gebiet mit $c((f, v), f) = 1$
- (f, t) von den inneren Gebieten zur Senke mit $c(f, t) = 3$
- (v, t) von den Knoten zur Senke mit $c(f, t) = 1$

- \mathcal{N}_F hat Bedarf $d = \sum_{f \in F_{in}} |f|$

\Rightarrow Ein zulässiger ganzzahliger Fluss φ_F existiert genau dann, wenn ein FAA existiert.

Proposition 4.2. *Sei G ein ebener intern-3-zusammenhängender Graph mit Aufhängungen a_1, a_2, a_3 und \mathcal{N}_F wie in Netzwerk 2 konstruiert. Ein zulässiger ganzzahliger Fluss φ_F auf \mathcal{N}_S existiert genau dann, wenn ein FAA auf G zu den Aufhängungen a_1, a_2, a_3 existiert. Insbesondere kodiert jeder Fluss genau ein FAA.*

Beweis: Sei φ_F ein zulässiger ganzzahliger Fluss auf \mathcal{N}_F . Der Fluss auf einer Kante $((f, v), f)$ entspricht einer Ecke von f und Fluss auf $((f, v), u)$ der Zuweisung eines Knoten zu f . Zur Vereinfachung sprechen wir im Weiteren auch von Ecken- und Zuweisungs-Fluss. Somit wird jeder innere Winkel von φ_F entweder einem Gebiet zugewiesen oder als Ecke ausgezeichnet. Da die Kanten von den inneren Knoten zur Senke Kapazität eins haben, kann jeder Knoten nur einmal zugewiesen werden. F2 ist somit erfüllt. Von jedem inneren Gebiet f fließt Fluss mit Stärke $|\varphi_s(f, t)| = 3$ zur Senke. Somit muss $|f| - 3$ Fluss durch zugewiesene Knoten gehen. F1 ist somit erfüllt. φ_F respektiert also die Bedingungen aus Definition 3.1 und es existieren nur dann FAAs auf G , wenn mindestens eine ganzzahlige Lösung für \mathcal{N}_F existiert. \square

Bemerkung. Das oben eingeführte Netzwerk 2 zur Bestimmung von FAAs lässt sich auch als Zwei-Fluss-Problem konstruieren. Wir trennen für Ecken- und Zuweisungs-Fluss die Quellen und Senken. Zu jedem Winkelknoten (f, v) fügen wir eine Kopie $(f, v)'$ ein und verbinden beide durch eine Kante mit Kapazität 1 (siehe Abbildung 4.3). In den ersten Winkelknoten führt eine Kante von beiden Quellen mit Kapazität 1 und vom zweiten Knoten führen Kanten zu f und v . Der Bedarf des Ecken-Flusses ist dann $3|F_{in}|$ und der Bedarf des Zuweisungs-Flusses $\sum_{f \in F_{in}} |f| - 3$.

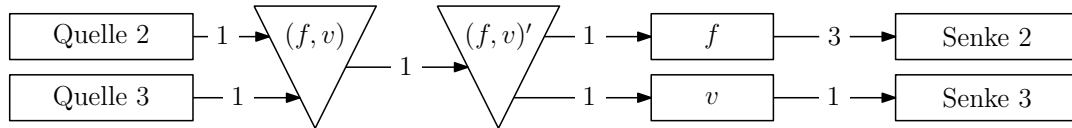


Abbildung 4.3 – Das Einfügen einer Kante und eines zweiten Winkelknotens gewährt die Trennung von Winkel- und Zuweisungs-Fluss. Die nicht beschrifteten Kanten haben Kapazität 1.

Eine zulässige ganzzahlige Lösung $\varphi_F = (\varphi_e, \varphi_z)$ entspricht dann wieder einem FAA auf G . Aus der Ganzzahligkeit folgt, dass ein Winkel entweder von φ_e (Ecke) oder φ_z (Zuweisung) genutzt wird. Dies gewährleisten die Kanten zwischen den Winkelknoten, da sie immer nur von einem der beiden Flüsse genutzt werden können.

4.1.3 Ein Zwei-Fluss Netzwerk zur Erkennung von SLTRs

Im Verlauf des Kapitels haben wir nun sowohl für Schnyder Woods als auch für FAAs Netzwerke konstruiert, für die ein ganzzahliger zulässiger Fluss einem Schnyder Wood bzw. einem FAA für einen gegebenen Graphen G liefert. Wir wollen jetzt eine Kombination aus beiden erstellen, die ein Ecken kompatibles Paar (σ, ϕ) aus einem Schnyder Labeling σ und einem FAA ϕ kodiert. Es folgt die Konstruktion eines Netzwerkes. Wir bezeichnen es mit \mathcal{N}_G , welches diese Anforderung erfüllt. Ein ganzzahliger Fluss φ kodiert dann ein Ecken kompatibles Paar und impliziert somit nach Theorem 3.17 eine SLTR für G . Dieses Netzwerk wird zwei Paare von Quellen und Senken haben. Es entspricht somit einem Gerichteten-Multi-Fluss-Problem. Wir werden nach der Konstruktion auch kurz darauf eingehen, warum sich nach diesem Ansatz kein Netzwerk mit nur einem Fluss erstellen lässt, auf welchem ein zulässiger Fluss einer SLTR entspricht.

Wie oben in Abschnitt 4.1.2 erwähnt, lässt sich ein FAA auch mit einem Zwei-Fluss kodieren und wir können Ecken- und Zuweisungs-Fluss mit den passenden Bedarfen getrennt betrachten. Wir müssen jetzt diese drei Flüsse, also Schnyder-, Ecken- und Zuweisungs-Fluss in einem Netzwerk kombinieren. Bei Aerts und Felsner ergeben Schnyder- und Ecken-Fluss zusammen Fluss von Typ 1 und der Zuweisungs-Fluss Typ 2 [AF15]. Wir wollen hier analog ein Netzwerk konstruieren, in dem wir FAA- und Schnyder-Fluss nicht trennen. Der Verständlichkeit halber werden wir Pfade, die in einer Lösung von einem der drei Flussarten genutzt werden, *Schnyder-Pfad*, *Ecken-Pfad* und *Zuweisungs-Pfad* nennen.

Bei der Kombination der beiden oben konstruierten Netzwerke \mathcal{N}_S und \mathcal{N}_F zu \mathcal{N}_G muss die Ecken-Kompatibilität von Schnyder Labeling und FAA gewährleistet werden. Bei der Konstruktion von Netzwerk 1 und Netzwerk 2 haben wir jeweils die Aufhängungen von G respektiert. Somit erfüllen wir K1 aus Definition 3.15, da das resultierende Schnyder Labeling σ und das resultierende FAA ϕ nach Konstruktion die gleichen Aufhängungen nutzen. Allerdings müssen wir für die zweite Bedingung das Netzwerk etwas komplizierter machen. Betrachten wir als Basis $\mathcal{N}_S \cup \mathcal{N}_F$ und fürs erste nur das Teilnetzwerk um ein inneres Gebiet f , dann sehen wir, dass es \mathcal{N}_S von $|f| - 3$ adjazenten Kanten (Schnyder-)Fluss aufnimmt. Es bleiben somit drei Einkanten in \mathcal{N}_S ungenutzt². Wir können auf diesen drei Kanten den Ecken-Fluss aus \mathcal{N}_F einfügen. Um K2 zu erfüllen, müssen wir gewährleisten, dass jede Ecke im Schnyder Labeling ein anderes Label hat. Betrachten wir also die von φ_s induzierte α_s -Orientierung auf dem Abschluss von $G + G^*$. Nach Theorem 4.4 erhalten wir in Bijektion stehende Schnyder Woods und nach Theorem 2.9 auch in Bijektion stehende Schnyder Labelings auf G und G^* .

Für diese gilt, wie in Abbildung 4.4 skizziert, dass das Label der Ecke eines Gebietes in G und das ihr in $G + G^*$ gegenüberliegenden Label der Ecke eines Gebiets um einen (Gebiets-)Knoten f in G^* gleich sind. Für eine zu f in G^* hin orientierte Kante folgt aus der Bijektion zwischen Schnyder Labelings und Schnyder Woods, dass die Label links und rechts am Ende dieser Kante gleich sind. Somit sind auch die Label in G

²Genau diese Kanten entsprechen in Netzwerk 1 der α_s -Orientierung.

gleich und wir können die folgende Eigenschaft festhalten (vergleiche Abbildung 4.4):

- A1 Die Label des von α_s induzierten Schnyder Labelings auf G sind zwischen zwei aufeinander folgenden, von f weg orientierten Kanten gleich.

Da es genau drei zu f orientierte Kanten gibt, müssen wir also dafür sorgen, dass für jedes Paar dieser Kanten eine Ecke zwischen ihnen liegt. So können wir sicherstellen, dass die drei Ecken unterschiedliche Label haben und K2 wäre erfüllt. Um dies zu erreichen, implementieren wir eine zyklische Struktur um jedes innere Gebiet, wie in Abbildung 4.5 skizziert.

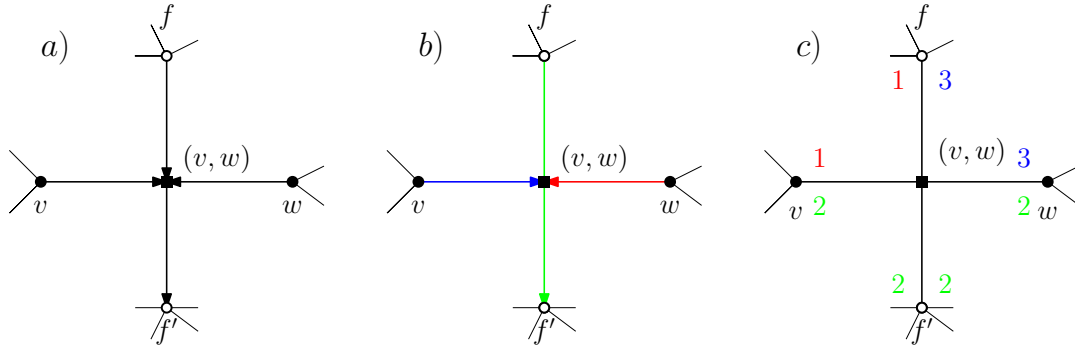


Abbildung 4.4 – a) Eine α_s -Orientierung um eine innere Kante von G . b) Teile der korrespondierenden Schnyder Woods auf G und G^* . c) Die induzierten Label, die für G und G^* gleich sind.

Betrachten wir zuerst den **Schnyder-Fluss**. Dieser wird Fluss von Typ 1, also von Quelle 1 zu Senke 1 sein. Für einen Schnyder-Pfad, der durch einen Knoten v führt, hat sich nichts geändert. Der in der Abbildung 4.5 eingezeichnete Schnyder-Pfad, der durch das innere Gebiet f führt, passiert davor einen extra Knoten, wir nennen ihn *kleines Quadrat*. Das kleine Quadrat soll gewährleisten, dass von Seite des Gebietes aus (also von jeder adjazenten Kante) entweder ein Schnyder-Pfad oder ein Ecken-Pfad in f mündet. Zuletzt fügen wir, wie oben, von jedem inneren Gebiet eine Kante mit Kapazität $|f| - 3$ zu Senke 1 ein. Somit kodiert hier eine ganzzahlige Lösung, genau wie zuvor in Netzwerk 1, einen Schnyder Wood auf G .

Kommen wir nun zum **FAA-Fluss**, also Fluss von Typ 2. Für einen Winkelknoten (f, v) aus Netzwerk 2 erstellen wir ein Winkeldreieck aus vier Knoten und drei Kanten in \mathcal{N}_G (vergleiche Abbildung 4.5). Betrachten wir zuerst den im FAA-Fluss enthaltenen **Zuweisungs-Fluss**. Von Quelle 2 geht, genau wie in Netzwerk 2, eine Kante zu jedem inneren Winkeldreieck (f, v) . Ein Zuweisungs-Pfad benutzt nur die erste Kante in einem Winkeldreieck und verlässt es über einen zusätzlich zu v eingefügten Dummy-Knoten v^* . Von jedem v^* geht eine Kante mit Kapazität 1 zu einer vorgeschobenen Dummy-Senke. Von dieser fügen wir eine Kante mit Kapazität $\sum_{f \in F_{in}} |f| - 3$ zu Senke 2 ein, wie in Abbildung 4.6 illustriert. Die Dummy-Knoten v^* sorgen dafür, dass jeder Knoten im FAA nur einmal zugewiesen werden kann, ohne in Konflikt

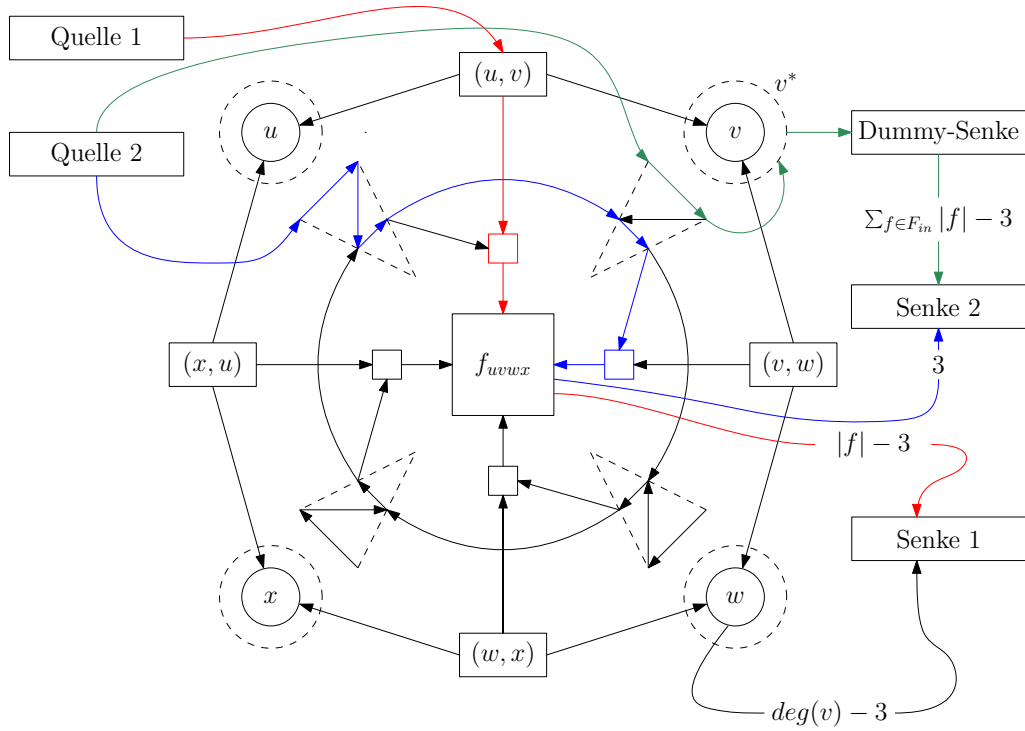


Abbildung 4.5 – Eine Skizze des kombinierten Netzwerkes auf einem inneren Gebiet mit $|f| = 4$. Beispielhaft sind ein Schnyder-Pfad (rot), ein Ecken-Pfad (blau) und ein Zuweisungs-Pfad (grün) eingezeichnet.

mit dem Schnyder-Fluss zu kommen. Die eingeschobene Dummy-Senke beschränkt die Anzahl der insgesamt zugewiesenen Knoten, genau wie im zuvor konstruierten FAA-Fluss, auf $\sum_{f \in F_{in}} |f| - 3$.

Es bleibt der **Ecken-Fluss**. Ein Ecken-Pfad betritt das Netzwerk durch ein Winkeldreieck (f, v) und muss, bevor er das Gebiet verlässt, ein vom Schnyder-Fluss ungenutztes kleines Quadrat passieren. Die zweite und dritte Kante in jedem Winkeldreieck gewährleisten, dass nicht immer das nächste kleine Quadrat genutzt werden muss. Falls dies von Schnyder-Fluss besetzt ist und der nächste Winkel zugewiesen wird, kann ein Ecken-Pfad den nächsten Winkel passieren (wie in Abbildung 4.5). Die erste Kante, die von sowohl Zuweisungs-, als auch Ecken-Pfaden genutzt werden kann, sorgt im Falle einer ganzzahligen Lösung für eine eindeutige Beschriftung (als Ecke oder Zuweisung). Wie zuvor existieren auch hier Kanten von jedem inneren Gebiet zu Senke 2 mit Kapazität drei.

Die Bedarfe der beiden Flüsse φ_s bzw. φ_F haben sich nicht verändert. Beide entsprechen jeweils den Bedarfen der oben konstruierten Netzwerke. So kann, mit den gleichen Argumenten wie oben, ein Schnyder Wood und ein FAA kodiert werden. Jedes Gebiet benötigt genau drei Ecken und $|f| - 3$ zugewiesene Knoten und je ein Schnyder-Pfad führt durch jede innere Kante, $|E_{in}|$. Hier seien wieder E_{in} die inneren

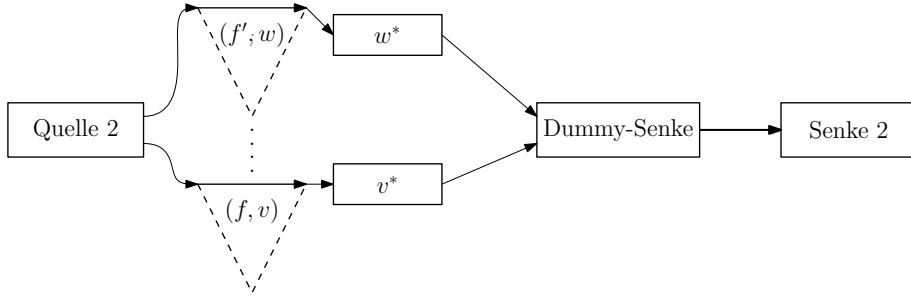


Abbildung 4.6 – Der Zuweisungsfluss durch die Winkel, Dummy-Knoten und die zusätzliche Kante vor Senke 2. Die Kante rechts hat Kapazität $\sum_{f \in F_{in}} |f| - 3$ und alle anderen Kapazität 1.

Kanten und F_{in} die inneren Gebiete von G . Fassen wir noch einmal zusammen.

Netzwerk 3 (SLTR). Bei \mathcal{N}_G handelt es sich um ein gerichtetes Netzwerk, das auf Basis eines ebenen intern-3-zusammenhängenden Graphen G mit Aufhängungen a_1, a_2, a_3 erstellt wird, um eine SLTR von G zu finden. Ein Ausschnitt um ein inneres Gebiet ist in Abbildung 4.5 dargestellt.

- \mathcal{N}_G hat zwei Quellen s_1, s_2 und zwei Senken t_1, t_2
- Es werden Knoten der folgenden Typen in \mathcal{N}_G erzeugt:
 - Knoten e für jede innere Kante $e \in E_{in}$
 - Knoten v für jeden $v \in V$ und Dummy-Knoten v^* für jeden $v \in V_{in}$
 - Knoten für jedes innere Gebiet $f \in F_{in}$.
 - $|f|$ kleine Quadrate q um jedes innere Gebiet f
 - Vier Knoten $w_i = (f, v)_i$, mit $i \in \{1, \dots, 4\}$, für jeden inneren Winkel
 - eine Dummy-Senke t_d
- Es werden gerichtete Kanten der folgenden Typen in \mathcal{N}_G erzeugt:
 - (s_1, e) von Quelle 1 zu jeder inneren Kante mit $c(s_1, e) = 1$
 - $(e, v_1), (e, v_2)$ von jeder inneren Kante zu den Endknoten mit $c(e, v_i) = 1$
 - (e, q) von inneren Kanten zu adjazenten kleinen Quadraten mit $c(e, q) = 1$
 - (q, f) einem kleinen Quadraten zu seinem inneren Gebiet mit $c(q, f) = 1$
 - (f, t_1) von den inneren Gebieten zur Senke 1 mit $c(f, t_1) = |f| - 3$
 - (a_i, t_1) von den Aufhängungen zur Senke 1 mit $c(a_i, t) = \deg(a_i) - 2$
 - (v, t_1) von den restlichen Knoten zur Senke 1 mit $c(v, t) = \deg(v) - 3$
 - (s_2, w_1) von Quelle 2 zu jedem inneren Winkel mit $c(s_2, w_1) = 1$
 - $(w_1, w_2), (w_2, w_3), (w_3, w_4)$ in jedem inneren Winkel mit $c(w_i, w_{i+1}) = 1$

- (w_2, v^*) von jedem inneren Winkel zum Dummy-Knoten mit $c(w_2, v^*) = 1$
- (w_4, q) von inneren Winkeln zum nächsten kleinen Quadrat mit $c(w_4, q) = 1$
- (w_4, w'_3) von einem inneren Winkeln zum Nächsten mit $c(w_4, w'_3) = 1$
- (v^*, t_d) von den Dummy-Knoten zur Dummy-Senke mit $c(v^*, t_d) = 1$
- (t_d, t_2) von der Dummy-Senke zu Senke 2 mit $c(t_d, t_2) = \sum_{f \in F_{in}} |f| - 3$
- \mathcal{N}_G hat Bedarfe $d_1 = |E_{in}|$ und $d_2 = \sum_{f \in F_{in}} |f|$

\Rightarrow Ein zulässiger ganzzahliger Fluss $\varphi = (\varphi_1, \varphi_2)$ existiert. \Leftrightarrow Es existiert ein SLTR auf G .

Bevor wir in Theorem 4.3 zeigen, dass eine ganzzahlige Lösung $\varphi = (\varphi_1, \varphi_2)$ auch wirklich ein Ecken kompatibles Paar kodiert, wollen wir noch ein Paar weitere Beobachtungen festhalten. Nehmen wir also an, es existiert ein ganzzahliger zulässiger Fluss φ auf \mathcal{N}_G , dann gilt für diesen:

- A2 Jede äußere Kante (w_1, w_2) in einem Winkel-Dreieck ist ausgelastet, sie wird entweder von einem Ecken- oder Zuweisungs-Pfad genutzt.
- A3 Jede Kante von einem kleinen Quadrat zu einem inneren Gebiet f ist ausgelastet, sie wird entweder von einem Schnyder- oder Ecken-Pfad genutzt.
- A4 Ein inneres Gebiet f mit $|f| = 3$ kann nicht von Zuweisungs- bzw. Schnyder-Pfaden genutzt werden.

Wir wollen diese Beobachtungen kurz begründen. Für jede mögliche ganzzahlige Lösung φ gilt

$$|\varphi| = |\varphi_1| + |\varphi_2| = |E_{in}| + \sum_{f \in F_{in}} |f|.$$

Da es genau $\sum_{f \in F_{in}} |f|$ innere Winkel gibt und der FAA-Fluss \mathcal{N}_G nur durch diese betreten kann, ergibt sich A2. Punkt A3 wird aus Gleichung 4.1 weiter unten folgen. Durch ein inneres Gebiet f müssen drei Ecken-Pfade laufen und im Fall $|f| = 3$ führt dies zu A4, da in den Winkeln kein Platz für Zuweisungs-Pfade ist und keine freien kleinen Quadrate für Schnyder-Pfade existieren.

Theorem 4.3. *Sei G ein intern-3-zusammenhängender Graph mit gegebenen Aufhängungen $\{a_1, a_2, a_3\}$, dann existiert eine SLTR von G genau dann, wenn ein ganzzahliger zulässiger Fluss $\varphi = (\varphi_1, \varphi_2)$ auf \mathcal{N}_G existiert.*

Beweis von Theorem 4.3: Sei G ein intern-3-zusammenhängender Graph mit Aufhängungen $\{a_1, a_2, a_3\}$ und $\varphi = (\varphi_1, \varphi_2)$ sei ein ganzzahliger zulässiger Fluss auf \mathcal{N}_G . Im ersten Schritt konstruieren wir einen Schnyder-Wood σ und ein FAA ϕ aus φ , um dann zu zeigen, dass sie ein Ecken kompatibles Paar bilden. Für einen zulässigen Fluss

müssen die Bedarfe erfüllt werden. Es gilt somit $|\varphi_1| = |E_{in}|$ und $|\varphi_2| = \sum_{f \in F_{in}} |f|$. In der nächsten Gleichung bezeichne f_{aus} das äußere Gebiet.

$$\begin{aligned}
 |\varphi_1| + |\varphi_2| &= \sum_{f \in F_{in}} (|f| - 3) + 3|F_{in}| + |E_{in}| \\
 &= \sum_{f \in F_{in}} (|f| - 3) + 2|E| - |V| - 1 + 2|F| - |f_{aus}| \\
 &= \sum_{f \in F_{in}} (|f| - 3) + \sum_{v \in V} (\deg(v) - 3) + 2|V| + 2|F| - 1 - |f_{aus}| \\
 &= \sum_{f \in F_{in}} (|f| - 3) + \sum_{v \in V} (\deg(v) - 3) + 2|E| + 3 - |f_{aus}| \\
 &= \underbrace{\sum_{f \in F_{in}} (|f| - 3)}_{\text{Dummy-Senke zu Senke 2}} + \underbrace{\sum_{v \in V} (\deg(v) - 3) + 3}_{\text{Kapazität Senke 2 von den Knoten.}} + \underbrace{\sum_{f \in F_{in}} (|f| - 3) + 3|F_{in}|}_{\text{Kanten von den Quadraten zu den inneren Gebieten}}
 \end{aligned} \tag{4.1}$$

Die beiden Terme in der rechten unteren Klammer entsprechen den Kapazitäten von den inneren Gebieten zu Senke 1 und Senke 2. Somit sind alle Kanten zu den Senken ausgelastet. Die Kanten von den kleinen Quadraten zu den inneren Gebieten sind ebenfalls ausgelastet. Diese sind die einzigen Kanten in \mathcal{N}_G , die sowohl von φ_1 als auch φ_2 genutzt werden können. Kapazität eins und Ganzzahligkeit von φ impliziert somit A3.

Beginnen wir mit φ_1 , um einen Schnyder Wood, oder genauer eine α_s -Orientierung, zu erhalten. Es gilt $|\varphi_1| = |E_{in}|$, und somit führt durch jede innere Kante ein Schnyder-Pfad. Dieser gibt uns die nach außen gerichtete Kante in α_s . Es bleibt zu zeigen, dass für jedes innere Gebiet und jeden Knoten die Bedingungen aus Theorem 4.4 für eine α_s -Orientierung eingehalten werden. Da alle Kanten von den Knoten zu Senke 1 ausgelastet sind, folgt, dass durch jeden inneren Knoten v genau $\deg(v) - 3$ Schnyder-Pfade führen. Somit ergeben die leeren Einkanten von v in \mathcal{N}_G die drei Auskanten für α_s . Für eine Aufhängung a_i folgt analog, dass die beiden ungenutzten Einkanten zusammen mit der Halbkante ins äußere Gebiet, die Bedingungen der α_s -Orientierung erfüllen. Es bleibt zu zeigen, dass durch jedes innere Gebiet $|f| - 3$ Schnyder-Pfade führen. Der restliche Schnyder-Fluss mit Stärke $|E_{in}| - \sum_{v \in V} (\deg(v) - 3)$ muss durch die inneren Gebiete führen und aus der ersten und letzten Zeile von Gleichung 4.1 folgt

$$|E_{in}| - \sum_{v \in V} (\deg(v) - 3) = \sum_{f \in F_{in}} (|f| - 3).$$

Somit führen $|f| - 3$ Schnyder-Pfade durch jedes innere Gebiet und wir können die α_s -Orientierung vervollständigen und erhalten einen Schnyder Wood σ auf G .

Betrachten wir nun φ_2 , also den FAA-Fluss. Nach A4 sind alle äußeren Kanten in den Winkeln ausgelastet. Falls diese nun in jedem inneren Gebiet von drei Ecken-Pfaden und $|f| - 3$ Zuweisungs-Pfaden genutzt werden, können wir ein FAA konstruieren. Da alle Kanten zu Senke 2 ausgelastet sind, führen $\sum_{f \in F_{in}} (|f| - 3)$ Pfade durch die Dummy-

Senke. Somit werden auch $\sum_{f \in F_{in}} (|f| - 3)$ Knoten inneren Gebieten zugewiesen. Indem wir diese Pfade zurückverfolgen und überprüfen, aus welchem Gebiet der Zuweisungspfad einen Dummy-Knoten betritt, erhalten wir die Zuweisungen. Es bleibt zu zeigen, dass jedem Gebiet genau $|f| - 3$ Knoten zugewiesen werden. Dies gilt genau dann, wenn durch jedes Gebiet drei Ecken-Pfade laufen. Da die Kanten von den inneren Gebieten zu Senke 2 ausgelastet sind, können wir also aus φ_2 ein FAA ϕ für G konstruieren.

Nun müssen wir zeigen, dass das Schnyder Labeling σ und das FAA ϕ ein Ecken kompatibles Paar ergeben. Die Bedingung K1, dass beide die gleichen Aufhängungen nutzen, folgt sofort aus der Konstruktion von \mathcal{N}_G . Es bleibt K2, also betrachten wir ein Teilnetzwerk (wie in Abbildung 4.5) um ein inneres Gebiet f . Die drei Ecken-Pfade können keine der $|f| - 3$ kleinen Quadrate nutzen, die schon von Schnyder-Fluss okkupiert werden. Die drei übrigen kleinen Quadrate nennen wir *verfügbar*. Ausgehend von f folgen wir den Ecken-Pfaden rückwärts zu den verfügbaren kleinen Quadraten. Wenn wir das Quadrat verlassen, gelangen wir zur dritten Kante eines Winkeldreiecks (entgegen dem Uhrzeigersinn). Nun verlassen wir das Gebiet entweder über diesen Winkel oder bewegen uns weiter (entgegen dem Uhrzeigersinn) zum nächsten Winkeldreieck. Wir zeigen in Behauptung 1, dass dies nur dann geschieht, wenn das kleine Quadrat zwischen diesen nicht verfügbar ist. Also betritt zwischen jeweils zwei verfügbaren kleinen Quadraten ein Ecken-Pfad das Gebiet und die Winkel haben nach A1 unterschiedliche Label in σ .

Behauptung 1 *Seien Q_1, Q_2 und Q_3 im Uhrzeigersinn die drei verfügbaren kleinen Quadrate um ein inneres Gebiet f . Dann existiert ein Ecken-Pfad, welcher das Netzwerk über Q_i verlässt. Dieser betritt es in einem Winkel zwischen, im Uhrzeigersinn, Q_{i-1} und Q_i .*

Angenommen dies ist nicht der Fall und nehmen wir ohne Beschränkung der Allgemeinheit an, dass der Ecken-Pfad P_e das Gebiet durch Q_3 verlässt. Der Winkel, über den P_e das Teilnetzwerk um das innere Gebiet betritt, liegt also nicht zwischen Q_2 und Q_3 (vergleiche Abbildung 4.7). Angenommen er liegt zwischen Q_1 und Q_2 . Betrachte das letzte Winkeldreieck vor Q_2 . Nach unserer Annahme ist die innere Kante dieses Dreiecks von P_e ausgelastet. Somit kann kein Ecken-Fluss zu Q_2 gelangen und wir erhalten einen Widerspruch, da alle kleinen Quadrate entweder von Ecken- oder von Schnyder-Fluss genutzt werden müssen. Mit dem gleichen Argument kann P_e das Teilnetzwerk nicht zwischen Q_3 und Q_1 betreten. Somit ist Behauptung 1 wahr.

Behauptung 2 *Alle Winkel zwischen zwei aufeinander folgenden verfügbaren kleinen Quadraten haben die selben Label im Schnyder Labeling σ .*

Diese Behauptung folgt aus A1. Die Winkel links und rechts von einem kleinen Quadrat, dass von einem Schnyder-Pfad genutzt wird, haben das gleiche Label in σ , da diese den Einkanten in α_s entsprechen. Die Auskanten entsprechen den verfügbaren kleinen Quadraten, und hier ändern sich die Label. Diese beiden Behauptungen zusammen zeigen, dass jede Ecke aus ϕ ein anderes Label in σ hat. Somit handelt es sich um ein Ecken kompatibles Paar (σ, ϕ) .

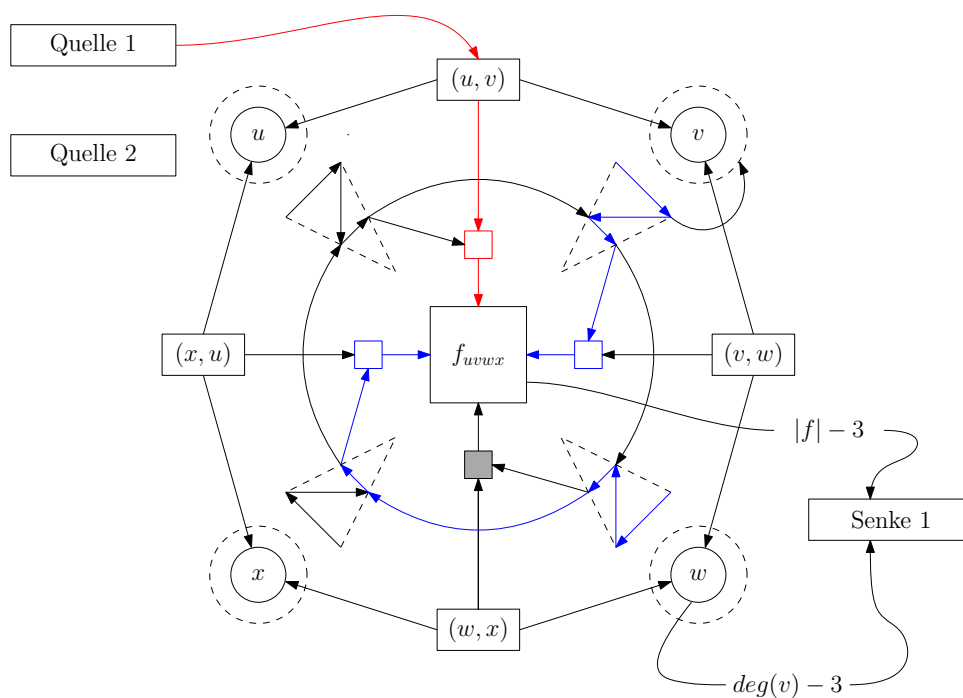


Abbildung 4.7 – Das untere *verfügbare* kleine Quadrat (grau) wird vom Ecken-Fluss (blau) übersprungen und kann nicht mehr ausgelastet werden.

Wir haben die Rückrichtung gezeigt. Nehmen wir also an, dass eine SLTR für G existiert. Wir müssen nun einen zulässigen ganzzahligen Fluss $\varphi = (\varphi_1, \varphi_2)$ auf \mathcal{N}_G konstruieren, der die SLTR kodiert. Nach Theorem 3.17 existiert ein Ecken kompatibles Paar (σ, ϕ) aus einem Schnyder Labeling σ und einem FAA ϕ , das zu diesem SLTR passt. Betrachte die zu σ gehörige α_s -Orientierung.

Wir beginnen mit einem leeren, wie oben konstruierten Netzwerk \mathcal{N}_G und werden nun Schritt für Schritt einen zulässigen Fluss φ konstruieren. Zuerst fügen wir für jeden zugewiesenen Winkel einen Pfad von Quelle 2, über die äußere Kante des Winkeldreiecks, den zugehörigen Dummy-Knoten und die Dummy-Senke hin zu Senke 2 ein. Es kommen somit $\sum_{f \in F_{in}} |f| - 3$ Einheiten Fluss hinzu und die Kante von der Dummy-Senke zu Senke 2 wird ausgelastet.

Als nächsten fügen wir den Schnyder-Fluss hinzu, der die α_s -Orientierung kodiert. Zuerst von Quelle 1 zu jedem inneren Kanten-Knoten e , dann von den inneren Kanten entweder über ein kleines Quadrat in ein angrenzendes Gebiet oder zu einem benachbarten Knoten, je nachdem, wohin die Auskante von e in α_s zeigt. Als Konsequenz müssen wir auch die Kanten von den inneren Knoten und inneren Gebieten zu Senke 1 sättigen.

Zuletzt fügen wir den Ecken-Fluss ein. Ein Ecken-Pfad P_e beginnt in Quelle 1, nutzt das zugehörige Winkeldreieck (diese sind noch frei) und verlässt das Gebiet über das im Uhrzeigersinn nächste verfügbare kleine Quadrat. Das dieser Weg frei ist folgt wieder aus Behauptung 2. Es sind somit alle Kanten zu den Senken ausgelastet, ohne die

Kapazität einer Kante zu überschreiten. Somit haben wir einen zulässigen ganzzahligen Fluss konstruiert, der eine SLTR kodiert. Damit ist der Beweis abgeschlossen. \square

Bemerkung. Wir können die Quellen und Senken von Netzwerk 3 nicht vereinen und trotzdem gewährleisten, dass eine zulässige ganzzahlige Lösung ein SLTR kodiert. In Netzwerk 3 verlieren wir so die Kontrolle über die Menge von Zuweisungs-Fluss der durch jedes innere Gebiet fließt. Wir erhalten mit einer zulässigen ganzzahligen Lösung also nicht mehr zwangsläufig ein FAA.

Wir können das Netzwerk leicht abwandeln, um sicher zu stellen, dass eine ganzzahlige Lösung immer eine FAA und einen Schnyder Wood induziert, doch dann geht uns die Kontrolle über die Ecken-Kompatibilität verloren.

4.2 Nicht-ganzzahlige Flüsse auf \mathcal{N}_G

Wir werden uns zum Abschluss des Kapitels mit der von Aerts und Felsner offen gelassenen Frage beschäftigen, ob die Erkennung von Graphen mit einer SLTR in P liegt – ob also polynominelle deterministische Algorithmen für die Entscheidung existieren, ob für einen gegebenen Graphen eine SLTR existiert. Wie in Abschnitt 2.4 erwähnt, impliziert eine nicht ganzzahlige Lösung für ein Gerichtetes-Multi-Fluss-Problem auf einem Graphen mit $n \geq 2$ Paaren von Quellen und Senken im Allgemeinen nicht die Existenz einer ganzzahligen Lösung. Und nach Theorem 2.21 ist die Berechnung einer ganzzahligen Lösung NP-schwer. Die experimentellen Ergebnisse aus Kapitel 5 lassen jedoch die Möglichkeit offen, dass wir für das betrachtete Netzwerk \mathcal{N}_G die folgende Vermutung beweisen können.

Vermutung 1 Sei $\tilde{\varphi} = (\tilde{\varphi}_1, \tilde{\varphi}_2)$ ein nicht ganzzahliger zulässiger Fluss auf \mathcal{N}_G , dann existiert auch ein ganzzahliger zulässiger Fluss φ und wir können in polynomieller Zeit ein Gutes-FAA aus $\tilde{\varphi}_2$ konstruieren.

Bemerkung. Wenn wir nicht darauf bestehen, dass ein berechneter zulässiger Fluss auf \mathcal{N}_G ganzzahlig ist, dann lässt sich ein zulässiger Fluss wie in Abschnitt 2.4 erwähnt durch lineare Programmierung in polynomineller Zeit finden und das Entscheidungsproblem, ob ein Graph eine SLTR hat, läge so in P.

Es ist uns noch nicht gelungen, einen Beweis von Vermutung 1 zu finden. Wir werden zuerst einen Weg besprechen, um ein FAA aus einer nicht-ganzzahligen Lösung zu konstruieren. Wir Vermuten, dass es sich bei diesem FAA um ein Gutes-FAA handelt. Danach werden wir zwei mögliche Beweisansätze betrachten. Um die Argumentation einfacher zu gestalten, werden wir das Zwei-Fluss-Problem manchmal als Drei-Fluss Problem, mit einer Lösung $\varphi = (\varphi_s, \varphi_e, \varphi_z)$ betrachten, indem wir Schnyder-, Ecken-, und Zuweisungs-Fluss eigene Typen und somit Quellen und Senken zuweisen.

Proposition 4.4. Sei \mathcal{N}_G^* ein Netzwerk bei dem wir Schnyder-, Ecken-, und Zuweisungs-Fluss jeweils eigene Quellen und Senken zuweisen und das sonst analog zu \mathcal{N}_G konstruiert wird. Dann existiert ein zulässiger (ganzzahliger) Fluss $\varphi^* = (\varphi_s^*, \varphi_e^*, \varphi_z^*)$ auf \mathcal{N}_G^* genau dann, wenn ein zulässiger (ganzzahliger) Fluss $\varphi = (\varphi_1, \varphi_2)$ auf \mathcal{N}_G existiert.

Beweis: Wir müssen den FAA-Fluss aus \mathcal{N}_G in zwei Typen aufteilen. In der Bemerkung nach Proposition 4.2 haben wir schon einen Weg beschrieben, dies zu tun. In \mathcal{N}_G^* existieren jeweils von der Ecken- und Zuweisungs-Quelle Kanten zum ersten Knoten der Winkeldreiecke. Die Winkel-Senke ist Senke 2 und wir trennen die Dummy-Senke von Senke 2 und machen sie zur Zuweisungs-Senke. Sonst sind \mathcal{N}_G und \mathcal{N}_G^* gleich. Die Bedarfe werden entsprechend aufgeteilt. Somit sind die einzigen von Ecken- und Zuweisungs-Fluss gemeinsam nutzbaren Kanten die ersten Kanten in den Winkeldreiecken. Für den Schnyder-Fluss verändert sich nichts.

Angenommen wir haben einen zulässigen (ganzzahligen) Fluss $\varphi^* = (\varphi_s^*, \varphi_e^*, \varphi_z^*)$ auf \mathcal{N}_G^* , dann induziert $(\varphi_s, \varphi_e + \varphi_z)$ auch einen zulässigen (ganzzahligen) Fluss auf \mathcal{N}_G . Sei nun $\varphi = (\varphi_1, \varphi_2)$ ein zulässiger (ganzzahliger) Fluss auf \mathcal{N}_G . Dann gilt $\varphi_1 = \varphi_s^*$. Wir müssen also φ_2 aufteilen. Ein (ganzzahliger) FAA-Pfad, der ein Winkeldreieck betritt, verlässt es entweder durch einen Dummy-Knoten oder führt weiter in das Gebiet. Wir können φ_2 somit in φ_e^* und φ_z^* aufteilen. Es bleibt zu zeigen, dass die Bedarfe erfüllt sind. In \mathcal{N}_G führt eine Kante mit Kapazität drei von jedem inneren Gebiet zu Senke 2, die von φ_2 gesättigt ist. Dieser Anteil des FAA-Flusses wird nach Konstruktion zu φ_e^* . Somit bleibt in jedem inneren Gebiet $|f| - 3$ Zuweisungs-Fluss. Die Bedarfe sind somit erfüllt. Die Ecken-Kompatibilität ergibt sich wie im Beweis zu Theorem 4.3. \square

Insbesondere gilt dieser Zusammenhang für eine beliebige Kombination von Ecken-, Schnyder- und Zuweisungs-Fluss zu zwei Flüssen, und wir müssen die Netzwerke nur leicht abwandeln. Die Flüsse lassen sich aber auch nach dieser Veränderung, von einem auf ein anderes Netzwerk übertragen.

Proposition 4.5. *Für jede beliebige Kombination von Schnyder-, Ecken- und Zuweisungs-Fluss zu zwei Flüssen auf einem zu \mathcal{N}_G^* analog konstruierten Netzwerk existiert ein zulässiger (ganzzahliger) Fluss φ genau dann, wenn ein zulässiger (ganzzahliger) Fluss $\varphi^* = (\varphi_s^*, \varphi_e^*, \varphi_z^*)$ auf \mathcal{N}_G^* existiert. Ein zulässiger (ganzzahliger) Fluss φ existiert also insbesondere genau dann, wenn eine zulässige (ganzzahlige) Lösung für eine beliebige andere Kombination der Flüsse existiert.*

Beweis: Wie wir einen zulässigen (ganzzahligen) Fluss φ aus φ^* konstruieren ist klar. Betrachten wir denn Fall, dass wir Schnyder- und Zuweisungs-Fluss zu φ_1 kombinieren. Wir fügen zwischen der Zuweisungs-Quelle und den Winkeldreiecken für jedes innere Gebiet einen zusätzlichen Knoten ein. Zu diesen Knoten führt eine Kante mit Kapazität $|f| - 3$ von der Zuweisungs-Quelle und Kanten zu den Winkeln der entsprechenden inneren Gebiete mit Kapazität 1 (vergleiche Abbildung 4.8). Sei nun $\varphi = (\varphi_1, \varphi_2)$ ein zulässiger (ganzzahliger) Fluss. Für den Zuweisungs-Fluss $\varphi_2 = \varphi_z$ verändert sich nichts. φ_2 lässt in jedem inneren Gebiet genau Platz für Ecken-Pfade mit Kapazität 3 frei. Wir können einen Pfad in φ_1 nun Zuweisungs-Fluss oder Schnyder-Fluss zuweisen, indem wir überprüfen, ob er das kleine Quadrat von einem Kanten-Knoten oder einem Winkeldreieck aus betritt. Durch jedes innere Gebiet führen Ecken-Pfade mit Auslastung 3 und somit Schnyder-Pfade mit Auslastung $|f| - 3$. Die Ecken-Kompatibilität zeigt man wieder analog zum Beweis von Theorem

4.3. Die Kombination aus Schnyder- und Ecken-Fluss folgt nach Aerts und Felsner [AF15]. Hier reicht wieder der Beutel im Zuweisungs-Fluss, um die Korrektheit des Netzwerkes zu erhalten. \square

Proposition 4.6. *Angenommen es existiert eine zulässige nicht-ganzzahlige Lösung $\tilde{\varphi}$ auf \mathcal{N}_G und \mathcal{N}_G lässt keine ganzzahlige Lösung zu, dann müssen die drei Flüsse $\tilde{\varphi}_e, \tilde{\varphi}_z$ und $\tilde{\varphi}_s$ alle nicht-ganzzahlig sein. Insbesondere muss mindestens ein Teilnetzwerk um ein inneres Gebiet existieren, auf dem alle drei Flüsse nicht-ganzzahlig sind.*

Beweis: Sei der Ecken-Fluss $\tilde{\varphi}_e$ ganzzahlig. Dann existieren auf \mathcal{N}_G keine Kanten die von zwei Flusstypen genutzt werden, da sich $\tilde{\varphi}_z$ und $\tilde{\varphi}_s$ nach Konstruktion keine Kanten teilen können. Somit erhalten wir nach Theorem 2.20 für alle drei Flüsse eine ganzzahlige Lösung. Wir können somit annehmen, dass $\tilde{\varphi}_e$ nicht-ganzzahlig ist. Wir definieren nun den Schnyder-Fluss als Fluss von Typ 1 und den Zuweisungs-Fluss als Typ 2 und werden den Ecken-Fluss lokal entweder dem einen oder dem anderen zuordnen.

Betrachten wir ein Teilnetzwerk um inneres Gebiet, auf dem $\tilde{\varphi}_s$ ganzzahlig ist, dann ordnen wir lokal den Ecken-Fluss Typ 2 zu. Wir behalten so trotzdem die Kontrolle über die Anzahl der zugewiesenen Winkel in diesem Gebiet und die Ecken-Kompatibilität, falls wir eine neue Lösung berechnen. Analog ordnen wir den Ecken-Fluss Typ 1 zu, falls der Zuweisungs-Fluss ganzzahlig ist. Somit existieren keine Kanten, die in unserer Lösung von beiden Flüssen gleichzeitig genutzt werden. Nach Theorem 2.20 existieren zulässige ganzzahlige Flüsse von Typ 1 und Typ 2. Dies liefert einen zulässigen ganzzahligen Fluss auf \mathcal{N}_G , was ein Widerspruch zur Annahme ist, dass dieser nicht existieren kann. \square

Betrachten wir zunächst den zweiten Teil von Vermutung 1. Die nächste Proposition liefert eine Methode, um aus einem nicht ganzzahligen zulässigen Fluss ein FAA zu konstruieren.

Proposition 4.7. *Sei $\tilde{\varphi}$ ein nicht ganzzahliger zulässiger Fluss auf \mathcal{N}_G und sei W die Menge der vom Zuweisungsfluss $\tilde{\varphi}_z$ genutzten inneren Winkel von G . Dann existiert eine Teilmenge $\phi \subseteq W$, sodass aus jedem Gebiet f genau $|f| - 3$ Winkel in ϕ enthalten sind und in der jeder Knoten v höchstens einmal vorkommt. ϕ ist also ein FAA auf G .*

Beweis: Sei $\tilde{\varphi}$ ein nicht ganzzahliger zulässiger Fluss auf \mathcal{N}_G . Wir definieren das gerichtete Netzwerk \mathcal{F}_z mit Quelle s und Senke t , einem Beutel B_f für jedes innere Gebiet f , einem Knoten für jeden inneren Winkel $(f, v) \in W$ und einem Knoten für jeden Dummy-Knoten. Zuerst fügen wir Kanten mit Kapazität $|f| - 3$ von der Quelle zu jedem Beutel ein. Dann folgen Kanten von den Beuteln B_f zu den Winkeln von f , von den Winkeln (f, v) zu den Dummy-Knoten v^* und zuletzt eine Kante von jedem Dummy-Knoten zu Senke, jeweils mit Kapazität 1. Die Kanten zu den Beuteln bilden einen Schnitt mit Kapazität $\sum_{f \in F_{in}} (|f| - 3)$ und $\tilde{\varphi}_z$ induziert einen zulässigen nicht-ganzzahligen Fluss φ^* auf \mathcal{F}_z . Die Stärke eines maximalen s - t -Fluss in \mathcal{F}_z ist somit $\sum_{f \in F_{in}} (|f| - 3)$. Nach Theorem 2.20 existiert also ein ganzzahliger zulässiger Fluss φ auf \mathcal{F}_z , mit $|\varphi| = |\varphi^*| = |\tilde{\varphi}_z| = \sum_{f \in F_{in}} (|f| - 3)$.

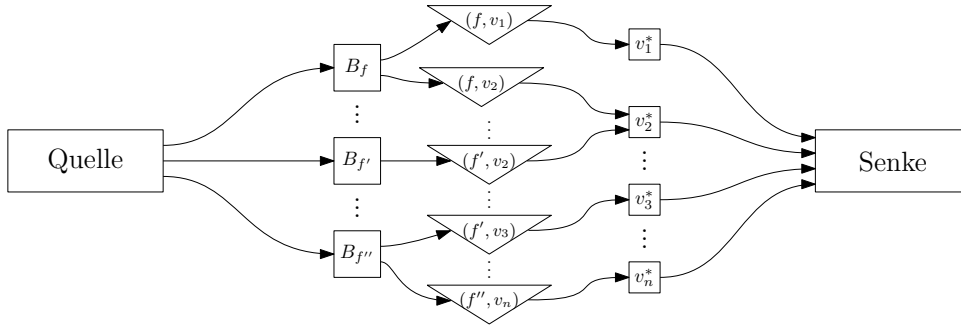


Abbildung 4.8 – Skizze des Netzwerkes \mathcal{F}_z . Die Kanten von der Quelle zu einem Beutel B_f hat Kapazität $|f| - 3$ und alle anderen haben Kapazität 1.

Der Fluss φ weist nun jedem inneren Gebiet f genau $|f| - 3$ Winkel zu und jeder Knoten v kann nur einmal zugewiesen werden. Wenn wir noch die Knoten hinzunehmen, die per Konstruktion von \mathcal{N}_G dem äußeren Gebiet zugewiesen sind, dann erhalten wir ein FAA auf G . \square

Wenn wir zeigen könnten, dass ein wie in Proposition 4.7 konstruiertes ϕ ein Gutes-FAA ist, folgt Vermutung 1, da die Existenz eines Guten-FAAs ϕ nach Theorem 4.3 auch die Existenz eines ganzzahligen zulässigen Flusses φ für \mathcal{N}_G impliziert. Die Ergebnisse aus Kapitel 5 legen nahe, dass es sich um ein Gutes-FAA handelt. Formulieren wir dies als eine zweite Vermutung.

Vermutung 2 *Ein wie in Proposition 4.7, aus einem zulässigen Fluss auf \mathcal{N}_G konstruiertes FAA ϕ ist ein Gutes-FAA von G und induziert somit eine SLTR.*

Beispiel 4.8. Es ist nicht möglich mit beliebigen Winkeln aus W zu beginnen und Schritt für Schritt für jedes Gebiet $|f| - 3$ Winkel wählen. Betrachte den planaren Graphen G aus Abbildung 4.9. Die beiden SLTRs auf der linken Seite haben dieselben Aufhängungen, implizieren jedoch andere FAAs und somit auch andere zulässige ganzzahlige Flüsse auf \mathcal{N}_G . Seien φ und φ' diese Flüsse und f_r, f_g und f_b die drei eingefärbten Gebiete. Betrachten wir die Zuweisungs-Flüsse φ_z und φ'_z . Dann gilt

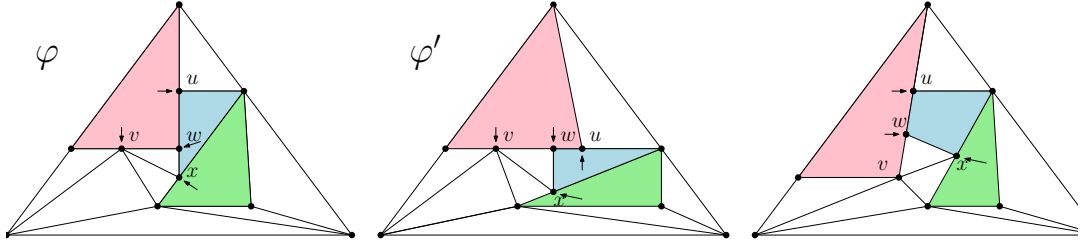
$$|\varphi_z(f_r, v)| = |\varphi_z(f_r, u)| = |\varphi_z(f_b, w)| = |\varphi_z(f_g, x)| = 1$$

$$|\varphi_z(f_r, v)| = |\varphi_z(f_r, w)| = |\varphi_z(f_b, u)| = |\varphi_z(f_g, x)| = 1.$$

Der Fluss $\tilde{\varphi} = \frac{\varphi + \varphi'}{2}$ ist ebenfalls zulässig und es folgt:

$$|\tilde{\varphi}_z(f_r, v)| = |\tilde{\varphi}_z(f_g, x)| = 1 \text{ und } |\tilde{\varphi}_z(f_r, w)| = |\tilde{\varphi}_z(f_r, u)| = |\tilde{\varphi}_z(f_b, w)| = |\tilde{\varphi}_z(f_b, u)| = \frac{1}{2}.$$

Somit liegen all diese Winkel in W . Wir können allerdings nicht einfach beginnen in einem Gebiet die benötigte Anzahl an Winkeln auszuwählen. In Abbildung 4.9 führt dies auf der rechten Seite zu keinem FAA und somit auch zu keiner SLTR. Die Konstruktion des Netzwerkes im Beweis von Proposition 4.7 ist somit sinnvoll.


 Abbildung 4.9 – Bei der Auswahl der Winkel aus W ist Vorsicht geboten.

4.2.1 Minimale Schnitte in \mathcal{N}_G

Wir wollen in diesem Abschnitt einen ersten möglichen Beweisansatz von Vermutung 1 besprechen. Die Idee ist, dass wir unter der Annahme, dass nur eine nicht-ganzzahlige Lösung existiert, einen minimalen Schnitt in einem Teilnetzwerk von \mathcal{N}_G erzeugen, und so zu einem Widerspruch gelangen.

Angenommen, es existiert ein Netzwerk \mathcal{N}_G , für das nur eine nicht ganzzahlige Lösung existiert. Sei $\tilde{\varphi}$ dieser nicht ganzzahlige zulässige Fluss und ϕ ein wie in Proposition 4.7 aus $\tilde{\varphi}$ konstruiertes FAA für G . Sei φ_z der eindeutige Zuweisungs-Fluss, der dieses FAA auf \mathcal{N}_G kodiert und $\overline{\mathcal{N}}_G$ das Teilnetzwerk von \mathcal{N}_G , aus welchem alle Kanten, die von φ_z ausgelastet sind, gelöscht wurden. Die Bedarfe sind weiterhin $|E_{in}|$ und $3|F_{in}|$ für den Schnyder- und Ecken-Fluss. Nach Proposition 4.5 können wir φ_s und φ_e zusammenfassen und mit φ_1 bezeichnen. Wir suchen also nach einem zulässigem ganzzahligem Fluss $\varphi_1 = \varphi_s + \varphi_e$ auf $\overline{\mathcal{N}}_G$ mit Bedarf $|E_{in}| + 3|F_{in}|$, da dann auch nach Bemerkung 4.6 eine ganzzahlige Lösung (φ_s, φ_e) folgen würde. Wir hätten somit einen ganzzahligen Fluss auf \mathcal{N}_G konstruiert, was zu einem Widerspruch führt.

Nach dem Max-Flow Min-Cut Theorem existiert ein zulässiger Fluss auf $\overline{\mathcal{N}}_G$ genau dann, wenn es keinen (Kanten-)Schnitt in $\overline{\mathcal{N}}_G$ mit Kapazität kleiner als $|E_{in}| + 3|F_{in}|$ gibt, der Quelle und Senke trennt. Bevor wir fortfahren, wollen wir einige Kantentypen aus \mathcal{N}_G benennen.

- E_{Δ} = Die äußeren Kanten in den Winkeldreiecken.
- E_{∇} = Die inneren Kanten in den Winkeldreiecken.
- S_* = Die Kanten von den Dummy-Knoten zur Dummy-Senke.
- V_* = Die Kanten von den Winkeldreiecken zu den Dummy-Knoten.
- E_{\rightarrow} = Die Kanten von Quelle 1 zu den Kanten-Knoten.
- F_{\square} = Die Kanten von den kleinen Quadraten zu inneren Gebieten f .
- V_{\rightarrow} = Die Kanten von den Knoten-Knoten zu Senke 1.
- e_d = Die Kante von der Dummy-Senke zu Senke 2

Sowohl $\mathcal{S}_1 = E_\Delta \cup E_{\rightarrow}$, als auch $\mathcal{S}_2 = F_\square \cup V_{\rightarrow} \cup \{e_d\}$ sind minimale Schnitte in \mathcal{N}_G . Für beide Menge gilt $|\mathcal{S}_1| = |\mathcal{S}_2| = |E_{in}| + \sum f \in F_{in}$ und sie trennen die Quellen (\mathcal{S}_1) bzw. die Senken (\mathcal{S}_2) vom Rest des Netzwerkes ab. Wenn wir nur von den Kanten aus E_Δ , die in $\overline{\mathcal{N}}_G$ übrig sind, sprechen, schreiben wir \overline{E}_Δ . Für die zu diesen korrespondierenden Kanten im Inneren ihrer Winkeldreiecke schreiben wir \overline{E}_∇ . Für die Teilmengen von V_* und S_* in $\overline{\mathcal{N}}_G$ schreiben wir \overline{S}_* und \overline{V}_* . Die restlichen Mengen sind vollständig in $\overline{\mathcal{N}}_G$ enthalten.

Seien E_z die von φ_z genutzten Kanten, die wir aus \mathcal{N}_G entfernen. Dann folgt $|\mathcal{S}_1 \cap E_z| = |E_\Delta \cap E_z| = |\varphi_z|$. Somit ist $\overline{\mathcal{S}}_1 = \mathcal{S}_1 \setminus E_z = \overline{E}_\Delta \cup E_{\rightarrow}$ ein Schnitt in $\overline{\mathcal{N}}_G$. Analog ist $\overline{\mathcal{S}}_2 = F_\square \cup V_{\rightarrow}$ ein Schnitt. Für die Kapazität von $\overline{\mathcal{S}}_1$ können wir folgern

$$c(\overline{\mathcal{S}}_1) = c(\overline{E}_\Delta) + c(E_{\rightarrow}) = c(E_\Delta) - |\varphi_z| + c(E_{\rightarrow}) = 3|F_{in}| + |E_{in}|,$$

und wieder folgt analog $c(\overline{\mathcal{S}}_2) = 3|F_{in}| + |E_{in}|$.

Falls es sich hierbei um minimale Schnitte handelt, dann würde dies bedeuten, dass eine ganzzahlige Lösung für $\overline{\mathcal{N}}_G$ existiert, mit deren Hilfe wir, zusammen mit φ_z , eine ganzzahlige zulässige Lösung für \mathcal{N}_G konstruieren könnten, was wiederum ein Widerspruch zu unserer Annahme wäre. Es muss also einen kleineren Schnitt \mathcal{S}_{min} , mit $|\mathcal{S}_{min}| \leq 3|F_{in}| + |E_{in}| - 1$, geben.

Bemerkung. Falls wir nach demselben Schema diejenigen Kanten aus \mathcal{N}_G entfernen, welche von einem Zuweisungs-Fluss gesättigt sind, der einem FAA entspricht, das keine SLTR induziert, dann muss so ein minimaler Schnitt \mathcal{S}_{min} existieren. Sonst würde ein Widerspruch zu Theorem 3.17 entstehen.

Falls Vermutung 2 korrekt ist, dann könnte so ein Schnitt nicht existieren. Nehmen wir jedoch für den Moment an, dass \mathcal{S}_{min} wie oben beschrieben existiert, dann können wir die folgenden Beobachtungen festhalten.

Behauptung 3 *Falls \mathcal{S}_{min} existiert, dann muss auch ein minimaler Schnitt \mathcal{S}_{min}^* in $\overline{\mathcal{N}}_G$ existieren, sodass er nur Kanten von einem der vier Typen $\overline{E}_\nabla, F_\square, V_{\rightarrow}$ und E_{\rightarrow} enthält.*

Betrachten wir die Kanten in \mathcal{S}_{min} . Die vier Kantentypen aus denen \mathcal{S}_{min}^* bestehen soll, sind in Abbildung 4.10 eingezeichnet. Wir werden argumentieren, dass wir Kanten, die nicht zu einer der vier Mengen gehören, eindeutig durch eine Kante aus diesen Mengen ersetzen können. Kanten auf einem Pfad von der Quelle, durch die korrespondierende Kante in \overline{E}_Δ zu einer Kante in \overline{E}_∇ , können in \mathcal{S}_{min}^* durch diese ersetzen. Ebenso können Kanten zwischen zwei Winkeldreiecken, oder von einem Winkeldreieck zu einem kleinen Quadrat³ in \mathcal{S}_{min}^* durch die entgegen dem Uhrzeigersinn nächste Kante in \overline{E}_∇ ersetzt werden. Kanten zwischen einem Kanten-Knoten und einem Knoten-Knoten oder einem kleinen Quadrat, können in \mathcal{S}_{min}^* durch die zuvor kommende Kante aus E_{\rightarrow} ersetzt werden. Abschließend können Kanten von

³Die Kante zwischen Winkeldreieck und kleinem Quadrat ließe sich ebenfalls durch die Kante aus F_\square ersetzen.

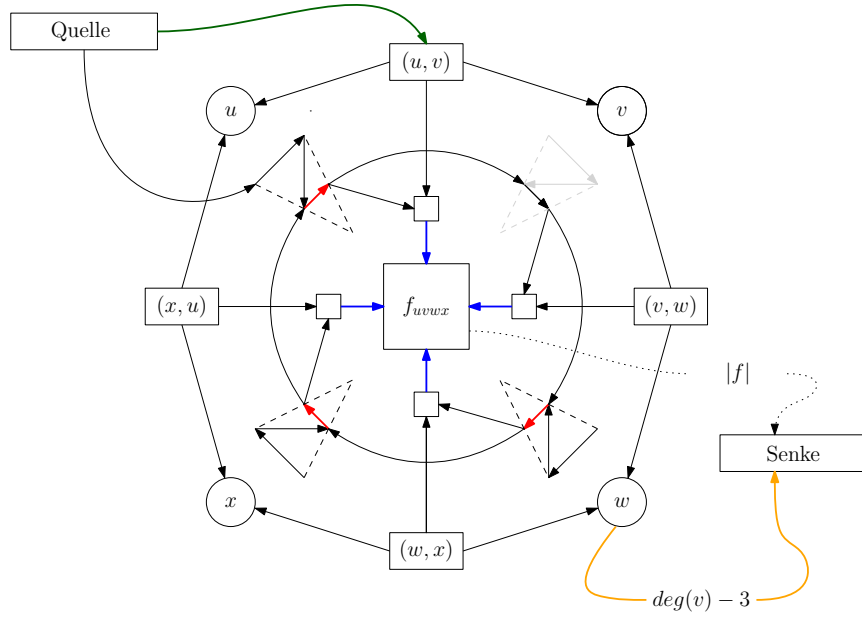


Abbildung 4.10 – Die vier Kantentypen \overline{E}_∇ (rot), F_\square (blau), V_\rightarrow (orange) und E_\rightarrow (grün), aus denen sich, nach Behauptungen 3 und 4, ein minimaler Schnitt in $\overline{\mathcal{N}}_G$ zusammensetzen müsste.

einem inneren Gebiet zu Senke in \mathcal{S}_{min}^* durch das Hinzufügen von allen Kanten aus F_\square an diesem Gebiet ersetzt werden.

Behauptung 4 Falls \mathcal{S}_{min} existiert, dann muss auch ein minimaler Schnitt \mathcal{S}_{min}^* in $\overline{\mathcal{N}}_G$ existieren, sodass er nur Kanten von einem der vier Typen \overline{E}_∇ , F_\square , V_\rightarrow und E_\rightarrow enthält. Er enthält aus jeder der Mengen \overline{E}_∇ , F_\square , V_\rightarrow und E_\rightarrow mindestens eine, aber aus keiner der Mengen alle Kanten.

Beweis: Falls ein solcher ein Schnitt \mathcal{S}_{min}^* existiert, dann kann \mathcal{S}_{min}^* nicht alle Kanten in \overline{E}_∇ enthalten. Sonst könnten wir aus $\mathcal{S}_{min}^* \cup (E_\Delta \cap E_z)$ einen Schnitt \mathcal{S} mit der gleichen Kapazität konstruieren, indem wir die Kanten $E_\nabla \cap \mathcal{S}_{min}^*$ durch die korrespondierenden Kanten in E_Δ ersetzen. Es folgt $\mathcal{S} \supseteq \mathcal{S}_1$, was ein Widerspruch ist. Falls \mathcal{S}_{min}^* jedoch keine Kante aus \overline{E}_∇ enthält, dann muss \mathcal{S}_{min}^* alle Kanten aus F_\square enthalten, weil \mathcal{S}_{min}^* ein Schnitt ist. Falls \mathcal{S}_{min}^* alle Kanten aus F_\square enthält, dann können wir annehmen, dass \mathcal{S}_{min}^* auch alle Kanten aus V_\rightarrow enthält. Es folgt $\mathcal{S}_{min}^* \cup \{e_d\} \supseteq \mathcal{S}_2$, was erneut ein Widerspruch ist. Angenommen \mathcal{S}_{min}^* enthält keine Kante aus F_\square , dann muss er alle Kanten aus \overline{E}_∇ und E_\rightarrow enthalten und $\mathcal{S}_{min}^* \cup (E_\Delta \cap E_z)$ wäre erneut ein Schnitt mit Kapazität $\geq |\mathcal{S}_1|$.

Es bleibt die Mengen E_\rightarrow und V_\rightarrow zu betrachten. Aus $\mathcal{S}_{min}^* \cap E_\rightarrow = \emptyset$ folgt $F_\square \subset \mathcal{S}_{min}^*$ und aus $\mathcal{S}_{min}^* \cap V_\rightarrow = \emptyset$ folgt $E_\rightarrow \subset \mathcal{S}_{min}^*$. Im Fall $E_\rightarrow \subset \mathcal{S}_{min}^*$ müssen um jedes innere Gebiet mindestens drei Kanten in \mathcal{S}_{min}^* liegen, womit wir wieder mindestens Kardinalität $|\mathcal{S}_1|$ erreichen. Wir betrachten als letztes V_\rightarrow . Hierbei ist zu beachten, dass für Knoten v mit $\deg(v) \leq 3$ keine Kante in \mathcal{N}_G existiert. Es gelte $V_\rightarrow \subset \mathcal{S}_{min}^*$,

dann muss \mathcal{N}_G noch mindestens $|E_{\rightarrow}| - |V_{\rightarrow}|$ Kanten enthalten, um den Schnyder-Fluss zu unterbrechen, was ein Widerspruch ist. Es gelte nun $\mathcal{S}_{min}^* \cap V_{\rightarrow} = \emptyset$. Wir benötigen erneut mindestens $|E_{in}|$ Kanten in \mathcal{S}_{min}^* , um den Schnyder-Fluss zu unterbrechen und wir erhalten somit einen letzten Widerspruch. Behauptung 4 ist somit richtig. \square

Schnyder- und Ecken-Fluss in $\overline{\mathcal{N}}_G$ können nur über die Kanten von Typ E_{\rightarrow} , V_{\rightarrow} und F_{\square} , beziehungsweise \overline{E}_{∇} und F_{\square} , fließen. Der einzige Kantentyp, der in beiden vorkommt, ist F_{\square} . Diese Kanten wurden in Netzwerk 3 eingefügt, um die Ecken-Kompatibilität zu erzwingen. Wir finden genau dann keinen ganzzahligen Fluss auf $\overline{\mathcal{N}}_G$, wenn es keinen Ecken-Kompatiblen Schnyder-Wood σ zu dem nach Proposition 4.7 erzeugten FAA ϕ gibt. Somit muss jeder zulässige Schnyder-Fluss (der einem beliebigen Schnyder Wood auf G entspricht) in mindestens einem Gebiet die kleinen Quadrate so auslasten⁴, dass aus mindestens einem freien Winkel keinen Ecken-Pfad führen kann. Alle kleinen Quadrate zwischen einem freien Winkel und dem im Uhrzeigersinn nächsten müssen in diesem Fall vom Schnyder-Fluss gesättigt sein. So ein kleines Quadrat ist in Abbildung 4.11) rot eingefärbt.

Wir nennen eine Kante aus F_{\square} an einem kleinen Quadrat ein *blockierendes* Quadrat, falls ein zulässiger ganzzahliger Schnyder-Fluss existiert, sodass dieses kleine Quadrat das erste nach einem freien Winkel ist, der durch den Schnyder-Fluss von der Senke abgeschnitten wird.

Behauptung 5 *Falls \mathcal{S}_{min} existiert, dann existiert ein Schnitt \mathcal{S}_{min}^* wie in Behauptung 4, welcher alle blockierenden Quadrate enthält.*

Falls ein blockierendes Quadrat nicht in \mathcal{S}_{min}^* enthalten ist, enthält \mathcal{S}_{min}^* die korrespondierenden Kanten aus E_{\rightarrow} und die gegen den Uhrzeigersinn nächste Kante aus E_{∇} . Da es sich um ein blockierendes Quadrat handelt, kann der Ecken-Fluss aus diesem Winkeldreieck nicht vorher das Gebiet verlassen. Wir können somit die Kante am Winkeldreieck in \mathcal{S}_{min}^* durch das blockierende Quadrat ersetzen.

Uns ist es jedoch bis jetzt nicht gelungen, mit diesem Ansatz zu einem Widerspruch zu gelangen und wir beenden die Überlegungen zu minimalen Schnitten an diesem Punkt.

⁴Genauer gesagt, sprechen wir hier von den Kanten aus F_{\square} .

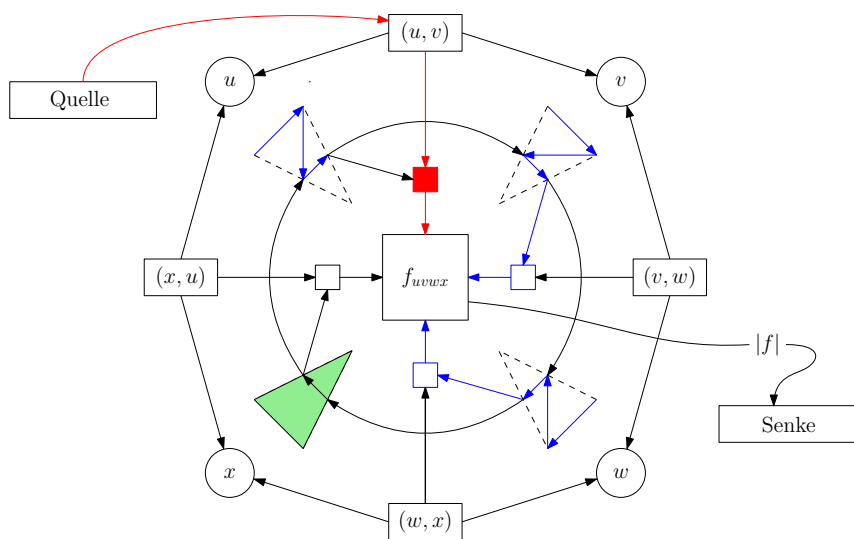


Abbildung 4.11 – Der Schnyder-Fluss (rot) ist nicht Ecken kompatibel zum FAA ϕ (grün). Somit muss ein blockiertes Quadrat (hier in rot) in \mathcal{N}_G existieren und durch mindestens einen freien Winkel kann kein Ecken-Pfad führen.

4.2.2 Alternierende Zyklen auf \mathcal{N}_G

Wir werden nun einen zweiten Beweisansatz erläutern. Sei $\tilde{\varphi} = (\tilde{\varphi}_s, \tilde{\varphi}_F)$ wieder ein (nicht-ganzzahliger) zulässiger Fluss auf \mathcal{N}_G . Wir können diesen Fluss in Pfade aufteilen. Diese Pfade werden bei einer nicht-ganzzahligen Lösung zumindest zum Teil fraktional sein⁵. Die Idee, die wir verfolgen wollen, ist, dass wir auf Teilmengen von Kanten in \mathcal{N}_G den Fluss vertauschen können, und wieder einen zulässigen Fluss erhalten. Das Ziel ist es, einen zulässigen Fluss zu konstruieren, der weniger oder gar keinen fraktionalen Anteil mehr hat.

Die formelle Definition dieses Ansatzes erfolgt nach einer Methode für α_s -Orientierungen. Ein ganzzahliger Schnyder-Fluss auf \mathcal{N}_G entspricht genau einer α_s -Orientierung auf dem Abschluss von $G + G^*$. Wir definieren nach Felsner die *essentiellen* Zykel auf dem Abschluss von $G + G^*$ bezüglich der α_s -Orientierungen [Fel04].

Definition 4.9 (essenzieller Zykel). Einen Zykel C in $G + G^*$ nennen wir *essentiell*, falls gilt:

- C1 C ist einfach und es existieren keine Kanten, die zwei Knoten von C verbinden, die aber nicht zu C gehören.
- C2 Alle Kanten, die C mit einem Knoten in seinem Inneren verbinden, sind in jeder α_s -Orientierung gleich orientiert.
- C3 Es existiert eine α_s -Orientierung mit C als gerichtetem Zykel.

⁵Der Fluss der auf so einem Pfad von der Quelle zur Senke gelangt ist nicht-ganzzahlig.

Nach Felsner können wir die Richtung der Kanten eines gerichteten essentiellen Zyklus einer α_s -Orientierung umkehren und erhalten erneut eine α_s -Orientierung. So kann man aus einem Schnyder Wood auf G einen anderen erzeugen. Mit Hilfe dieser Methode lässt sich das folgende Theorem beweisen [Fel04].

Theorem 4.10. *Die Menge der Schnyder Woods eines intern-3-zusammenhängenden ebenen Graphen G mit Aufhängungen a_1, a_2, a_3 bildet einen distributiven Verband.*

Das nächste Lemma schränkt die Länge essentieller Zykel ein.

Lemma 4.11 (Lemma 17 [Fel04]). *Sei G ein intern-3-zusammenhängender ebener Graph G mit Aufhängungen a_1, a_2 und a_3 . Die möglichen Längen von essentiellen Zykeln C von α_s -Orientierungen sind 4, 6, 8, 10 und 12.*

Beschränken wir unsere Suche für den Moment auf den Schnyder-Fluss $\tilde{\varphi}_s$. Mithilfe der folgenden Definitionen können wir einen Zusammenhang zwischen dem Schnyder-Fluss auf \mathcal{N}_G und der Umkehrung von essentiellen Zykeln einer α_s -Orientierung herstellen.

Definition 4.12 (V- und F-Schritte). Wir definieren zwei Mengen von Kanten in \mathcal{N}_G . Wir nennen $\{(e_1, v_1), (e_2, v_1)\}$ einen *V-Schritt* mit Kapazität a , falls gilt:

$$\tilde{\varphi}_s(e_1, v_1) = a > 0 \text{ und } \tilde{\varphi}_s(e_2, v_1) \leq 1 - a.$$

Analog nennen wir $\{(e_1, q_1), (q_1, f), (q_2, f), (e_2, q_2)\}$ einen *F-Schritt* mit Kapazität a , falls gilt:

$$\min(\tilde{\varphi}_s(e_1, q_1), \tilde{\varphi}_s(q_1, f)) = a > 0 \text{ und } \max(\tilde{\varphi}_s(q_2, f), \tilde{\varphi}_s(e_2, q_2)) \leq 1 - a.$$

Zwei Schritte folgen aufeinander, wenn der letzte Knoten des ersten dem ersten Knoten des zweiten entspricht (vergleiche Abbildung 4.12).

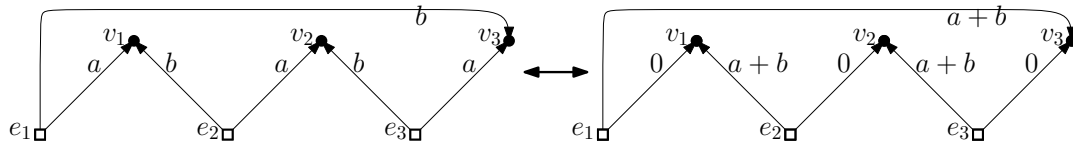


Abbildung 4.12 – Wir können einen Schnyder-Zykel umkehren, wenn er nur aus V-Schritten besteht. Die Kapazitäten werden nicht überschritten und durch die Knoten v und e fließt gleich viel Schnyder-Fluss. Merke, dass hier zwei Schnyder-Zykeln mit Kapazitäten a und b abgebildet sind.

Definition 4.13 (Schnyder-Zykel und alternierende Zykel). Sei $E_C = \{S_1 \cup \dots \cup S_n\}$, sodass alle S_i V- oder F-Schritte sind und S_i und S_{i+1} zyklisch aufeinander folgen. Wir nennen E_C einen *Schnyder-Zykel* und seine Kapazität $|E_C|$ entspricht der minimalen Kapazität aller enthaltenen V- und F-Schritte. Wir nennen Schnyder-Zykel und die weiter unten definierten Zykel in Ecken- und Zuweisungs-Fluss *alternierende Zykel* auf $\tilde{\varphi}$.

Ein Schnyder-Zykel mit Kapazität 1 in \mathcal{N}_G entspricht einem gerichteten Zykel einer α_s -Orientierung. Insbesondere finden wir also zu essentiellen Zykeln bezüglich α_s -Orientierungen, auch Schnyder-Zykel mit Kapazität 1 in korrespondierenden Schnyder-Flüssen.

Bemerkung. Wir können den fraktionalen Anteil des Schnyder-Fluss vollständig mit Schnyder-Zyklen der passenden Kapazitäten überdecken, ohne dabei neue fraktionale Teile zu erzeugen. Dies ist möglich, da wir den Fluss in Pfade aufteilen können. Da die Kanten von der Quelle und zu Senke gesättigt sind, reicht es die Pfadstücke zwischen den Kanten-Knoten und Gebiets-Knoten bzw. Knoten-Knoten zu betrachten. Aus diesen fraktionalen Pfadstücken können wir nun Schnyder-Zykel bilden. Dies wirft die folgenden Fragen auf, die wir allerdings hier nicht beantworten werden.

- Ist es möglich die Schnyder-Zykel so zu bilden, dass zu jedem ein korrespondierender essentieller Zykel auf $G + G^*$ existiert, wenn wir zulassen, dass auch ganzzahlige Pfade aufgeteilt werden?
- Angenommen dies ist möglich, muss dann $\tilde{\varphi}_s$ eine Linearkombination aus ganzzahligen Schnyder-Flüssen sein?

Die nächste Definition zeigt einen Weg auf, wie man mithilfe der alternierenden Zykel einen neuen zulässigen Fluss konstruieren kann.

Definition 4.14 (Umkehrung eines Schnyder-Zykels). Sei $E_C = \{S_1 \cup \dots \cup S_n\}$ Schnyder-Zykel mit Kapazität $a = |E_C|$. Wir können einen Schritt S_i umkehren, indem wir Fluss mit Kapazität a vom ersten Teil der Kanten auf den zweiten Teil verschieben. Die *Umkehrung* von E_C erfolgt nun, indem wir jeden Schritt $S_i \subset E_C$ umkehren (vergleiche Abbildung 4.12).

Die Schnyder-Zykel, die nur aus V-Schritten bestehen, können wir umkehren, indem wir den Fluss in $\tilde{\varphi}_s$ wie in Abbildung 4.12 verändern. Die Kapazitäten der Kanten werden nicht überschritten und durch die Knoten-Knoten und Kanten-Knoten fließt gleich viel Schnyder-Fluss wie zuvor. Der resultierende Fluss ist somit weiterhin zulässig.

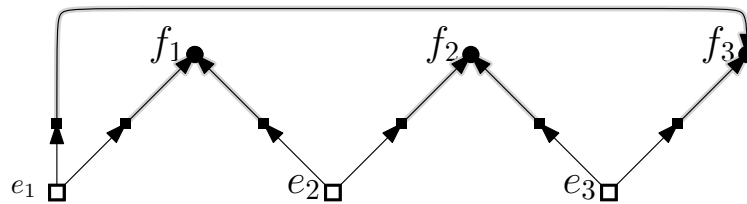


Abbildung 4.13 – Wir können einen Schnyder-Zykel der F-Schritte enthält nicht umkehren, ohne auch den FAA-Fluss anzupassen, da wir sonst die Kapazitäten der Kanten überschreiten. Die grau unterlegten Kanten sind von $\tilde{\varphi}$ gesättigt.

Betrachten wir einen Schnyder-Zykel, der nur aus F-Schritten besteht. So ein Fluss ist in Abbildung 4.13 dargestellt. Das Problem bei der Umkehrung ist, dass die Kanten

zwischen den kleinen Quadraten und inneren Gebieten von einem zulässigen Fluss $\tilde{\varphi}$ gesättigt werden. Somit müssen wir, wenn wir den Zykel umkehren wollen, auch den FAA-Fluss $\tilde{\varphi}_F$ anpassen. Wir definieren analog zu oben alternierende Zykeln auf dem Ecken- und Zuweisungsfluss. Beispiele sind in Abbildung 4.14 zu sehen.

Definition 4.15 (Z-Zykel und C-Zykel). Wir nennen $E_C = \{S_Z \cup S'_Z\}$, mit $S_Z = \{(s_2, w_1), (w_1, w_2), (w_2, v_1^*), (v_1^*, s_d)\}$ und $S'_Z = \{(s_2, w'_1), (w'_1, w'_2), (w'_2, v_1'^*), (v_1'^*, s_d)\}$ einen *Z-Zykel* mit Kapazität a , falls gilt:

$$\min(\tilde{\varphi}_s(\{S_Z\}) = a > 0 \text{ und } \max(\tilde{\varphi}_s(\{S'_Z\}) \leq 1 - a.$$

Wir nennen $E_C = \{C_Z \cup C'_Z\}$, mit $C_Z = \{(s_2, w_1^1), (w_1^1, w_2^1), (w_2^1, w_3^1), (w_3^1, w_4^1), (w_4^1, w_3^2), (w_3^2, w_4^2), \dots, (w_4^k, q), (q, f), (f, t_2)\}$ und analogem C'_Z einen *C-Zykel* mit Kapazität a , falls gilt:

$$\min(\tilde{\varphi}_s(\{C_Z\}) = a > 0 \text{ und } \max(\tilde{\varphi}_s(\{C'_Z\}) \leq 1 - a.$$

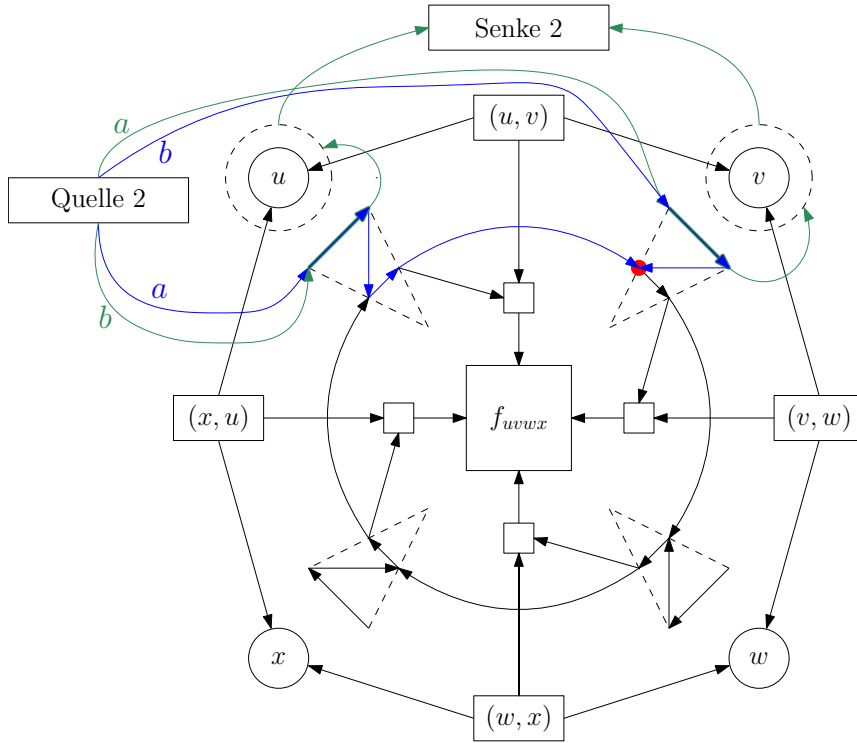


Abbildung 4.14 – Ein Beispiel von zwei C-Zykeln (blau) und zwei Z-Zykeln (grün). Merke, dass es sich hier jeweils einmal um Zykel mit Kapazität a und jeweils einmal um Zykel mit Kapazität b handelt. Die C-Zykeln führen in diesem Beispiel nur bis zu einem Winkeldreieck (roter Punkt). Falls $a + b = 1$ gilt und durch einen der beiden, von den Z-Zykel genutzten Dummy-Knoten kein weiterer Zuweisungs-Pfad führt, können wir jeweils einen der beiden Zykeln umkehren und erhalten einen ganzzahligen Fluss.

Bemerkung. Die hier eingeführten Definitionen führen zu deutlich längeren gleichgerichteten Kantenfolgen als zuvor. Es können allerdings auch kürzere Formen von Z- und C-Zykel auftreten. Ein Z-Zykel kann schon in einem Dummy-Knoten enden. Ein C-Zykel kann entweder in einem inneren Gebiet, oder am Knoten w_3 eines Winkel-dreiecks enden. In keinem der beiden Fälle müssen die Längen von erstem und zweitem Teil übereinstimmen.

Analog zu oben ist es möglich, den fraktionalen Anteil von Ecken- und Zuweisungsfluss jeweils mit alternierenden Zykeln zu überdecken. In den Lösungen für \mathcal{N}_G , die mit dem in Kapitel 5 vorgestellten Programm errechnet wurden, finden sich in allen drei Fluss-Typen solche Zykeln. Durch schrittweises Umkehren war es uns auf kleineren Beispielen möglich so ganzzahlige Lösungen zu konstruieren. Ein Ausschnitt aus so einem Beispiel ist in Abbildung 4.14 zu sehen.

Beispiel 4.16. Kommen wir zurück zu einen Schnyder-Zykel den wir umkehren wollen. Betrachten wir den Fall, dass der Zykel nur durch einen Gebiets-Knoten führt. Wir müssen also einen oder mehrere Ecken-Zykel finden, die wir parallel zum Schnyder-Zykel umkehren können, sodass die Kapazitäten an den kleinen Quadraten

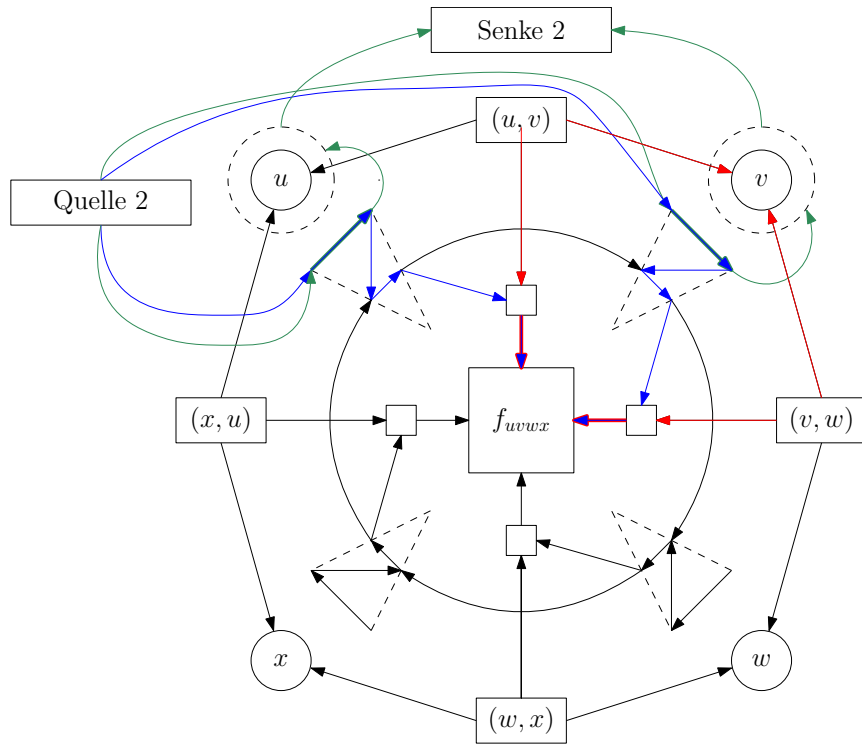


Abbildung 4.15 – Falls durch einen der Dummy-Knoten kein weiterer Zuweisungs-Pfad führt können wir die Zykeln so umkehren, dass wir einen ganzzahligen Fluss erhalten. Alle eingezeichneten Flüsse haben Stärke $1/2$. Die dicken Kanten sind somit ausgelastet.

nicht überschritten werden. In Abbildung 4.15 ist so eine Situation dargestellt. Nehmen wir an, dass auf allen eingefärbten Kanten Fluss mit Stärke $1/2$ fließt. Wie man hier sieht, müssen wir nun aber zusätzlich noch mindestens auf die äußeren Winkelkanten achten, an denen sich Zuweisungs- und Ecken-Zykel überschneiden. Wie in Proposition 4.6 festgehalten, sind diese Fälle die relevanten – wenn nur zwei Flusstypen nicht-ganzzahlig sind können wir eine ganzzahlige Lösung konstruieren. Für die alternierenden Zykel in der Abbildung ist die gleichzeitige Umkehrung möglich, falls durch einen der beiden Dummy-Knoten kein weiterer Pfad läuft.

In einer nicht-ganzzahligen Lösung auf einem Netzwerk, dass eine ganzzahlige Lösung zulässt müsste es einen Weg geben über das Umkehren von alternierenden Zykeln zu einer ganzzahligen Lösung zu kommen. Nehmen wir jedoch an, dass es keine ganzzahlige Lösung gibt, dann könnten wir auf diesem Weg zu einem Widerspruch zu gelangen und somit Vermutung 1 zu beweisen. Wir haben dies für einige Sonderfälle gezeigt, doch im Allgemeinen noch keinen Weg gefunden. Wir formulieren darum eine weitere Frage:

- Falls wir einen (essentiellen) Schnyder-Zykel E_C gegeben haben, existiert dann eine Menge aus C- und Z-Zykeln (und potentiell weiteren Schnyder-Zykeln), sodass die gemeinsame Umkehrung dieser Zykeln dazu führt, dass der fraktionale Flussanteil kleiner wird?

Weil wir mit unseren Mitteln diese Frage nicht beantworten konnten, beenden wir an diesem Punkt die Erläuterung der Beweisansätze und müssen Vermutung 1 und Vermutung 2 unbewiesen lassen.

5 Das Programm

In diesem Kapitel werden wir auf die Implementierung von Netzwerk 3 aus dem vorherigen Abschnitt eingehen. Das im Folgenden beschriebene Programm baut auf Vermutung 1 – dass sich aus jedem nicht ganzzahligen zulässigen Fluss auf Netzwerk 3 ein Gutes-FAA extrahieren lässt – auf. Der Code wurde in SageMath geschrieben und ist auf Anfrage erhältlich [S⁺18]. Algorithmus 1 gibt einen Überblick der durchgeführten Schritte bei der Suche nach einem Guten-FAA für einen gegebenen ebenen intern-3-zusammenhängenden Graphen mit Aufhängungen $\{a_1, a_2, a_3\}$.

Algorithmus 1 Berechnung eines Guten-FAA

```

1: procedure GFAA( $G, f_{aus}, \{a_1, a_2, a_3\}$ )
2:   if FAA( $G, \{a_1, a_2, a_3\}$ )  $\neq$  None then
3:      $\{d_1, d_2\} \leftarrow$  demands for  $\mathcal{N}_G$ 
4:     initialize  $\mathcal{N}_G$  ▷ nach Netzwerk 3
5:      $\varphi = (\varphi_1, \varphi_2) \leftarrow$  two-flow( $\mathcal{N}_G$ )
6:     if  $\varphi \neq$  None then
7:       if  $\varphi$  is integer then
8:          $\phi \leftarrow$  FAA( $\varphi_2$ )
9:         return  $\phi$ 
10:      else
11:         $\varphi_z \leftarrow$  FAA-flow( $\varphi_2$ ) ▷ Fluss nach Proposition 4.7
12:         $\mathcal{N} \leftarrow \mathcal{N}_G \setminus \{\text{edges used by } \varphi_z\}$ 
13:         $\tilde{\varphi} \leftarrow$  int-one-flow( $\mathcal{N}$ ) ▷ Ecken- und Schnyder-Fluss zu  $\varphi_z$ 
14:        if  $|\tilde{\varphi}| = d_1 + d_2 - |\varphi_z|$  then
15:           $\phi \leftarrow$  FAA( $\varphi_z$ )
16:          return  $\phi$ 
17:        else ▷ Nur erreichbar, falls Vermutung 2 falsch
18:           $\varphi = (\varphi_1, \varphi_2) \leftarrow$  int-two-flow( $\mathcal{N}_G$ )
19:          if  $\varphi \neq$  None then ▷ Sonst Gegenbeispiel zu Vermutung 1
20:             $\phi \leftarrow$  FAA( $\varphi_2$ )
21:            return  $\phi$ 
22:          end if
23:        end if
24:      end if
25:    end if
26:  end if
27: end procedure

```

Die Kontrolle, ob für G ein FAA existiert, ist optional, lässt sich jedoch, zum Beispiel wie zuvor über ein 1-Fluss-Problem, in polynomineller Zeit auf einem deutlich kleineren Netzwerk bestimmen. Dies spart Zeit, falls zu Aufhängungen keine FAAs auf G existieren. Das Multi-Fluss-Problem auf \mathcal{N}_G zu gegebenen Bedarfen (d_1, d_2) wird mithilfe des in SageMath enthaltenen Solvers *Glps* für Lineare Programmierung gelöst, welcher ein Paar von Flussgraphen (φ_1, φ_2) ausgibt, falls eine zulässige Lösung existiert und sonst nichts [M⁺12, S⁺18]. In den Zeilen 13 und 19 wird, im Gegensatz zu Zeile 5, nur nach ganzzahligen Lösungen gesucht. Aus einer ganzzahligen Lösung kann man ein FAA ϕ aus φ_2 extrahieren, indem man die Zuweisungs-Pfade durch die Dummy-Senke zurück verfolgt. Wir betreten jeden passierten Dummy-Knoten v^* aus einem Winkel (f, v) . Diese Winkel ergeben die Zuweisungen ϕ .

Die Überprüfung ab Zeile 12 ist unter der Annahme, dass Vermutung 1 stimmt, nicht notwendig. Es könnte hier ein FAA aus φ_2 extrahiert und ausgegeben werden. Wir gewährleisten jedoch so die Korrektheit des Algorithmus, weil wir den Beweis von Vermutung 1 noch nicht gefunden haben. Bei Tests ergaben sich mit diesem Ansatz immer noch deutlich kürzere Berechnungszeiten als bei der Suche nach ausschließlich ganzzahligen Lösungen. Zeile 17 sollte nach Vermutung 2 nicht erreicht werden. Sie stellt jedoch sicher, dass Algorithmus 1 immer dann ein Gutes-FAA ausgibt, wenn ein ganzzahliger Fluss auf Netzwerk 3 existiert.

Bemerkung. In der Implementierung wurden die Kanten innerhalb von Gebieten f mit $|f| = 3$ weggelassen, da die einzig mögliche Lösung hier ist, dass nur drei Ecken-Pfade durch das Gebiet laufen. Die Bedarfe werden dementsprechend angepasst.

5.1 Visualisierung

Nehmen wir an, wir haben für einen Graphen G ein Gutes-FAA ϕ gefunden. Für eine SLTR müssen wir eine zu ϕ passende Einbettung von G finden. Wir werden den in Abschnitt 3.1 erörterten Ansatz über harmonische Funktionen nutzen, um eine SLTR von G zu erhalten.

Wir wollen nun eine Einbettung $f : V \rightarrow \mathbb{R}^2$ von G ähnlich der Gummiband-Repräsentation berechnen, die ϕ respektiert. Sei $S \subseteq V$ die Menge der Knoten von f_{aus} . Nach Abschnitt 3.1 gelten die folgenden harmonischen Gleichungen für zugewiesene (oben) und nicht zugewiesene Knoten (unten).

$$\begin{aligned} f(v) &= \lambda_v f(u) + (1 - \lambda_v) f(w), \text{ mit } \lambda_v \in (0, 1) \\ f(v) &= \sum_{u \in N(v)} \lambda_{uv} f(u), \text{ mit } \sum_{u \in N(v)} \lambda_{uv} = 1 \text{ und } \lambda_{uv} > 0 \end{aligned}$$

Um zu einer gegebenen Gewichtsfunktion λ eine Lösung zu finden, können wir diese Gleichungen um die Bilder der Aufhängungen $f(A) = f(\{a_1, a_2, a_3\})$ erweitern und als Matrix schreiben.

$$M_\lambda(\vec{v}_x, \vec{v}_y) = \begin{pmatrix} f^{(A)_x} & f^{(A)_y} \\ 0 & 0 \end{pmatrix}, \text{ mit } (M_\lambda)_{vw} = \begin{cases} -\lambda_{vw} & \text{falls } (v, w) \in E, \\ \sum_{u \in N(v)} \lambda_{uw} & \text{falls } v = w, \\ 0 & \text{sonst.} \end{cases}$$

Wenn wir nun die Pseudo-Inverse berechnen, erhalten wir eine Einbettung.

$$f(V) = M_\lambda^{-1} \begin{pmatrix} f^{(A)_x} & f^{(A)_y} \\ 0 & 0 \end{pmatrix}.$$

Wir wollen nun, inspiriert von den *iterativen Tutte Einbettungen* nach Felsner und Scheucher, diese Rechnung mehrmals durchführen und Schritt für Schritt die Gewichtung λ anpassen [FS17]. Wünschenswert wäre es, wenn sich die Zeichnung nach einer gewissen Anzahl an Schritten nur noch so wenig verändert, dass wir den Algorithmus abbrechen können und die letzte Zeichnung ausgeben.

5.1.1 Möglichkeiten bei der Wahl von λ

Setzen wir im ersten Durchlauf $\lambda = 1$, erhalten wir eine klassische Gummiband-Repräsentation, die ϕ respektiert. Wir wollen nun anhand dieser Einbettung λ verändern, um Iteration für Iteration, eine „schönere“ Einbettung zu erhalten. Halten wir zwei Punkte fest, die wir als Bewertungsmaßstab für eine schöne Einbettung berücksichtigen können.

- Es gibt keine zu großen oder zu kleinen Gebiete.
- Es existieren keine zu kurzen Kanten.

Sehr lange Kanten lassen sich, wie in Beispiel 5.2, nicht immer vermeiden. Es gibt SLTRs, wie in Beispiel 5.1, bei denen alle inneren Knoten zugewiesen sind. Dies macht eine gute Wahl der λ kompliziert. Der Ansatz nach Felsner und Scheucher, bei dem λ als monoton steigende Funktion proportional zu Größe der an eine Kante angrenzenden Gebiete und ihrer Länge gewählt wird, konvergiert selten nicht und liefert so auch keine schönen Zeichnungen [FS17]. Allgemein wurden besonders SLTRs mit wenigen Kanten betrachtet, da für diese die oben erwähnten Einschränkungen stärker auftreten. Wir wollen das implementierte Schema anhand von zwei Beispielen erklären.

Beispiel 5.1. Bei der in Abbildung 5.1 links zu sehenden SLTR sind alle Knoten, bis auf die Aufhängungen, einem Gebiet zugeordnet. Somit liegt jeder Knoten auf einer Gerade und es existieren nur Gleichungen vom Typ

$$f(v) = \lambda_v f(u) + (1 - \lambda_v) f(w), \text{ mit } \lambda_v \in (0, 1).$$

Um von der linken zur rechten Zeichnung zu gelangen, wollen wir das Gebiet unten in der Mitte verkleinern, doch die drei angrenzenden Kanten kommen in keiner der

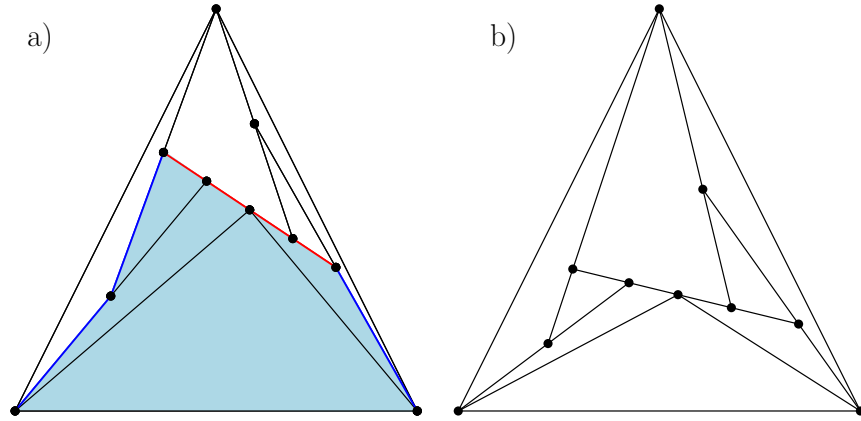


Abbildung 5.1 – a) Eine Einbettung mit $\lambda=1$. b) Eine Einbettung nach bei der wir nur zu e adjazente Gebiete für $\lambda(e)$ berücksichtigen. c) Eine Einbettung nach dem Schema aus Beispiel 5.2.

Gleichungen zur Bestimmung unserer Einbettung $f(V)$ vor. Die Kanten, die uns helfen können, das Segment in rot nach unten zu bewegen und somit das untere Dreieck zu verkleinern, sind in blau eingefärbt. Die Wahl der λ erfolgt in Schritt j , nach folgendem Schema für jedes Segment S des Guten-FAA. Seien G_1, G_2 die Mengen der Gebiete, die vollständig auf einer der beiden Seiten des Segments liegen (vergleiche die hellblaue Fläche). Nun berechnen wir deren Gesamtfläche A_1, A_2 . Seien nun P_1, P_2 Mengen von (kurzen) Pfaden, die von den Endpunkten des Segments zu den Aufhängungen, die in G_i liegen, führen. Wir fordern, dass die Pfade selbst ebenfalls nur auf den jeweiligen Seiten liegen. Auf den Pfaden $p \in P_i$ passen wir nun λ an. Die Pfade p, p' , die in unserem Beispiel nach unten führen, sind genau die blau eingefärbten Kanten in Abbildung 5.1. So gelingt es uns, Segmente wie in der Abbildung zu verschieben. Wir erhöhen für jede Kante $e \in p$ nun $\lambda(e)$ um a_i , mit:

$$a_i = \frac{1}{j} \left(\frac{A_i}{\text{Länge}(p) * |P_i|} \right)^{1.2}.$$

Die Formel ist heuristisch gewählt. Wir brechen entweder ab, wenn die Einbettung konvergiert oder wir eine feste Anzahl an Schritten durchgeführt haben. Aufgrund der Einbindung von kurzen Pfaden auf G , ist der Rechenaufwand sehr hoch, selbst wenn wir diese bei der Initialisierung abspeichern.

Beispiel 5.2. Betrachten wir einen effizienteren Ansatz. Wir setzen bei der Initialisierung $\lambda_0(e) = 1$ für jede Kante von G . Nun multiplizieren wir die Kanten an den Gebieten $f \in F_{in}$ mit $A(f) > A(f_{max}) * (1 + \epsilon)$ mit einer Konstante $c \in \mathbb{N}$. Für diese Kanten gilt somit $\lambda_{i+1}(e) = \lambda_i(e) * c$. Wir wählen $\epsilon = 0,1$ und $c = 2$. In Abbildung 5.2 sind die Einbettung von zwei Graphen nach 100 Schritten zu sehen.

Wir haben bei der Implementierung eine Kombination aus beiden Ansätzen gewählt. Für kleine Graphen wird zuerst das erste Schema durchgeführt. Falls dieses nicht

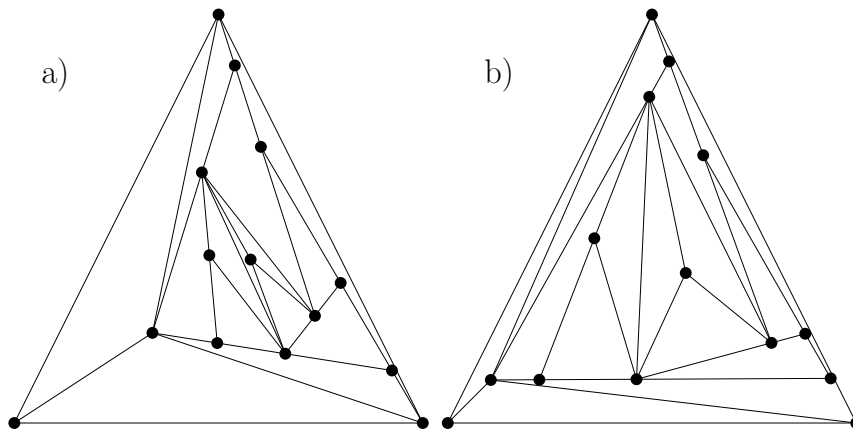


Abbildung 5.2 – Zwei Zeichnungen der gleichen SLTR, nach dem Schema aus Beispiel 5.1 (a) und nach Beispiel 5.1 (b).

konvergiert fahren wir mit dem zweiten fort. Für Graphen mit mehr als 15 Knoten dauert die Berechnung allerdings so lange, dass wir standardmäßig das zweite Schema durchführen. Es sei angemerkt, dass es auch hier Graphen gibt, deren Einbettungen nach diesen Berechnungen weniger schön werden, als wenn wir einfach $\lambda = 1$ setzen und nur einen Schritt durchführen. Im Durchschnitt kommen wir aber zu deutlich ansprechenderen Ergebnissen. Am Ende des Kapitels in Abbildung ?? sind einige so berechnete Einbettungen festgehalten.

Bemerkung. Wir haben viele an den Parametern der Berechnungen verändert, um so zu heuristisch schönen Ergebnissen zu gelangen. Die Einschränkungen, die wir durch die Zuordnung der Knoten vornehmen, scheint es allerdings deutlich schwerer zu machen, einen Algorithmus zu finden der für sehr viele unterschiedliche Graphen immer hinreichend schöne Zeichnungen liefert.

5.2 Experimentelle Rechnungen

Es folgt eine kurze statistische Betrachtung der Verteilung von Graphen mit SLTRs. Hier würde eine gleichmäßige Wahl von (intern-)3-zusammenhängenden Graphen die aufschlussreichsten Resultate liefern. Ein Algorithmus zur zufälligen Erstellung 3-zusammenhängender planarer Graphen lässt sich zum Beispiel nach einem Ansatz von Fusy implementieren [Fus09]. Als Teilschritt der Erstellung eines uniformen Samplers für planare Graphen werden hier 3-zusammenhängende planare Graphen mit gleichverteilter Wahrscheinlichkeit erzeugt. Die Implementierung ist jedoch aufgrund der notwendigen Auswertung von Erzeugendenfunktionen kompliziert. Diese Analyse beschränkt sich daher auf pseudo-zufällig erzeugte Graphen.

Es folgt eine kurze Beschreibung des von uns benutzten Samplers. Wir beginnen mit einem K_4 . Nun wird in jedem Schritt mit noch zu wählenden Wahrscheinlichkeiten eine der folgenden vier Operationen durchgeführt.

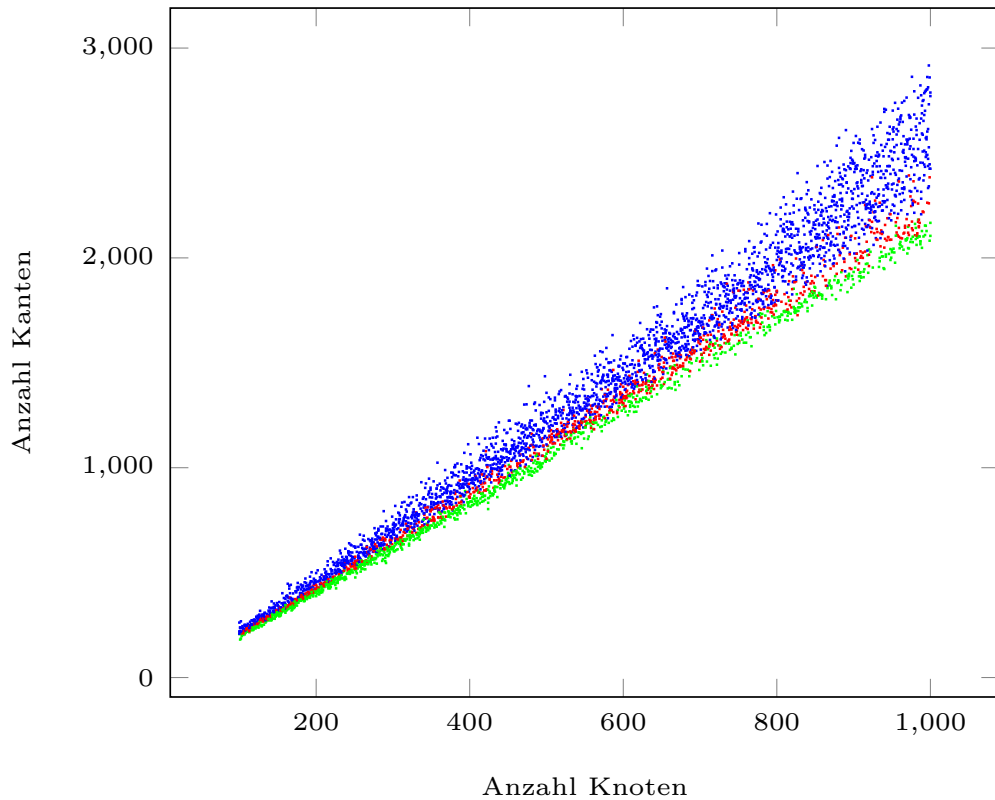


Abbildung 5.3 – Ergebnisse von experimentellen Rechnungen des Programms auf pseudo-zufälligen planaren 3-zusammenhängenden Graphen mit gegebenen Aufhängungen. Jeder Punkt entspricht einem Graphen mit SLTR (blau), nur FAA (rot) und ohne FAA (grün).

- PG1 Ein Knoten v mit $\deg(v) \geq 4$ wird in v_1, v_2 geteilt und eine Kante (v_1, v_2) eingefügt. Nun werden die zyklisch sortierten Nachbarn in zwei Teile N_1, N_2 getrennt und mit v_1 beziehungsweise v_2 verbunden.
- PG2 Eine Knoten wird auf einer Kante eingefügt und mit einem in einem angrenzenden Gebiet liegenden Knoten verbunden.
- PG3 Ein Knoten wird in ein Gebiet eingefügt und mit mindestens drei der am Gebiet liegenden Knoten verbunden.
- PG4 Es wird eine zufällige Kante in ein Gebiet mit mehr als drei Knoten eingefügt.

Nach jeder dieser Operationen ist G_i weiterhin planar und die Erzeugung kann bei der gewünschten Knotenzahl angehalten werden. Abschließend wird zufällig ein äußeres Gebiet und aus diesem die Aufhängungen gewählt. Die Parameter der pseudo-zufälligen Erzeugung sind durch drei aufsteigende natürliche Zahlen (a, b, c) , mit $a \leq b \leq c \leq 1000$, gegeben und lassen sich anpassen. In jedem Schritt wird eine der

vier Möglichkeiten PG1, PG2, PG3 und PG4 mit Verteilung $a/1000$, $(a+b)/1000$, $(a+b+c)/1000$ und $(1000-a-b-c)/1000$ ausgewählt. Wir benutzen zur Gewinnung der statistischen Daten Werte, die sich um $(a, b, c) \approx (380, 650, 990)$ bewegen.

In Abbildung 5.3 sind die Ergebnisse für so erzeugte Graphen zwischen 100 und 1000 Knoten dargestellt, mit jeweils fünf Graphen pro Knotenanzahl. Ein Punkt in der Abbildung entspricht einem Graphen. Die Farben stehen für eine SLTR (blau), nur ein FAA (rot) oder einen Graphen mit keinem von beiden (grün). Wie nicht anders zu erwarten, bilden die Farben drei Strahlen, die sich mit wachsender Knotenzahl zunehmend vermischen. Die Parameter des Samplers wurden so gewählt, um gerade an diesem Übergang viele Graphen zu erzeugen. Für Graphen mit im Verhältnis großen Kantenzahlen finden sich fast immer SLTRs und für Graphen mit wenigen Kanten existieren nur selten oder nie FAAs.

Da wir uns am Ende von Kapitel 4 ausführlich mit nicht-ganzzahligen Lösungen beschäftigt haben, wollen wir auch zu diesen eine statistische Einordnung vornehmen. In Abbildung 5.4 ist eine solche zu sehen. Hier entspricht jeder Punkt einem zulässigen Fluss auf einem Netzwerk \mathcal{N}_G zu pseudo-zufälligen Graphen G . Die Koordinaten entsprechen der der Kantenzahl¹ von \mathcal{N}_G und der Kantenzahl des fraktionalen Flusses. Insgesamt wurden jeweils die Lösungen für 1000 Graphen auf 500 Knoten berechnet die SLTRs besitzen. Es wurde erneut der oben beschriebene Sampler genutzt. Da wir bei der Implementierung die Kanten im Inneren von Dreiecken in \mathcal{N}_G weggelassen haben, müssen hier die Punkte auf der rechten Seite Graphen mit tendenziell mehr Gebieten entsprechen, die keine Dreiecke sind. In den berechneten Resultaten korreliert ein höherer Anteil von fraktionalem Fluss, mit einer größeren Anzahl von unterschiedlichen ganzzahligen zulässigen Flüssen und somit auch SLTRs auf G .

Wie wir sehen, werden oft ganzzahlige Lösungen zurück gegeben, selbst wenn wir nicht auf Ganzzahligkeit bestehen. Hinzu kommt, dass der Anteil des fraktionalen Flusses oft relativ klein ist.

¹ Hierbei werden Kanten mitgezählt, die in einer zulässigen Lösung immer gesättigt oder in diesem Fall leer sind.

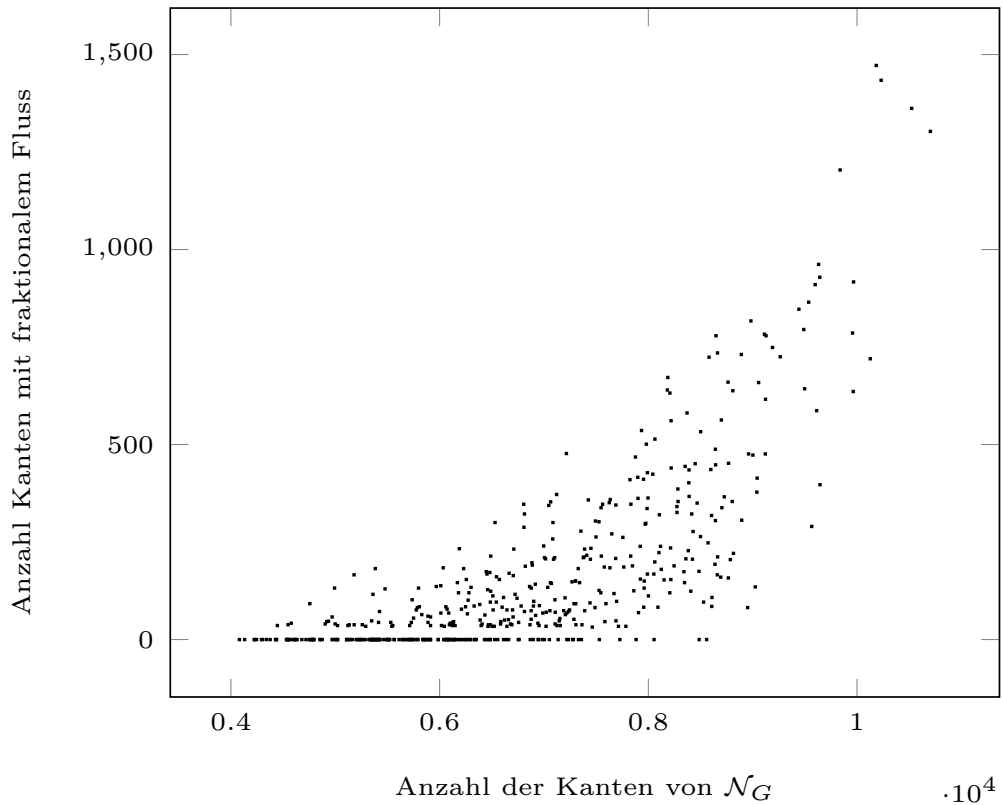


Abbildung 5.4 – Ergebnisse von experimentellen Rechnungen des Programms auf pseudo-zufälligen planaren 3-zusammenhängenden Graphen mit SLTRs zur Veranschaulichung des fraktionalen Anteils der errechneten Lösungen.

5.3 Dokumentation

Zum Abschluss des Kapitels folgt eine Dokumentation des Programms. Die Datei `program.sage` muss in SageMath geladen werden [S⁺18]. Die beiden in `program.sage` implementierten Funktionen sind:

```
sltr(graph, face, suspensions, non_int, check_int, plotting, ipe)
```

Gibt, falls möglich, ein Gutes-FAA zurück und erstellt optional eine Zeichnung.

return = GFAA : Eine Gutes-FAA als Liste von Gebieten mit Zuweisungen, `[[[1,4,6,8],[4]], ...]`, oder `None`.

graph : Ein planarer intern-3-zusammenhängender Graph

face : Das äußere Gebiet als Kantenfolge `[[v1, v2], ... , [vk, v1]]`, mit `None` werden alle möglichen äußeren Gebiete überprüft.

suspensions : Die Aufhängungen als Liste `[a1, a2, a3]`, mit `None` werden alle möglichen Aufhängungen überprüft. Falls ein äußeres Gebiet übergeben wurde müssen auch die Aufhängungen übergeben werden.

non_int : default `True`, sonst wird nur nach ganzzahligen Lösungen gesucht.
check_int : default `True`, sonst wird die Überprüfung ab Zeile 12 ausgelassen.
plotting : default `False`, sonst wird eine Zeichnung ausgegeben.
ipe : default `None`, bei Übergabe eines Strings **string** wird eine ipe-Datei **string.ipe** erstellt.

random_sltr(vertices, non_int, check_int, plotting, ipe, cut)

Es wird ein pseudo-zufälliger Graph generiert und, falls möglich, ein Gutes-FAA zurückgegeben. Optional kann eine Zeichnung ausgegeben werden.

return = **[graph, GFAA]** Ein Paar aus Graph und GFAA oder **[graph, None]**.

vertices : Die Anzahl der Knoten des zu erzeugenden Graphen

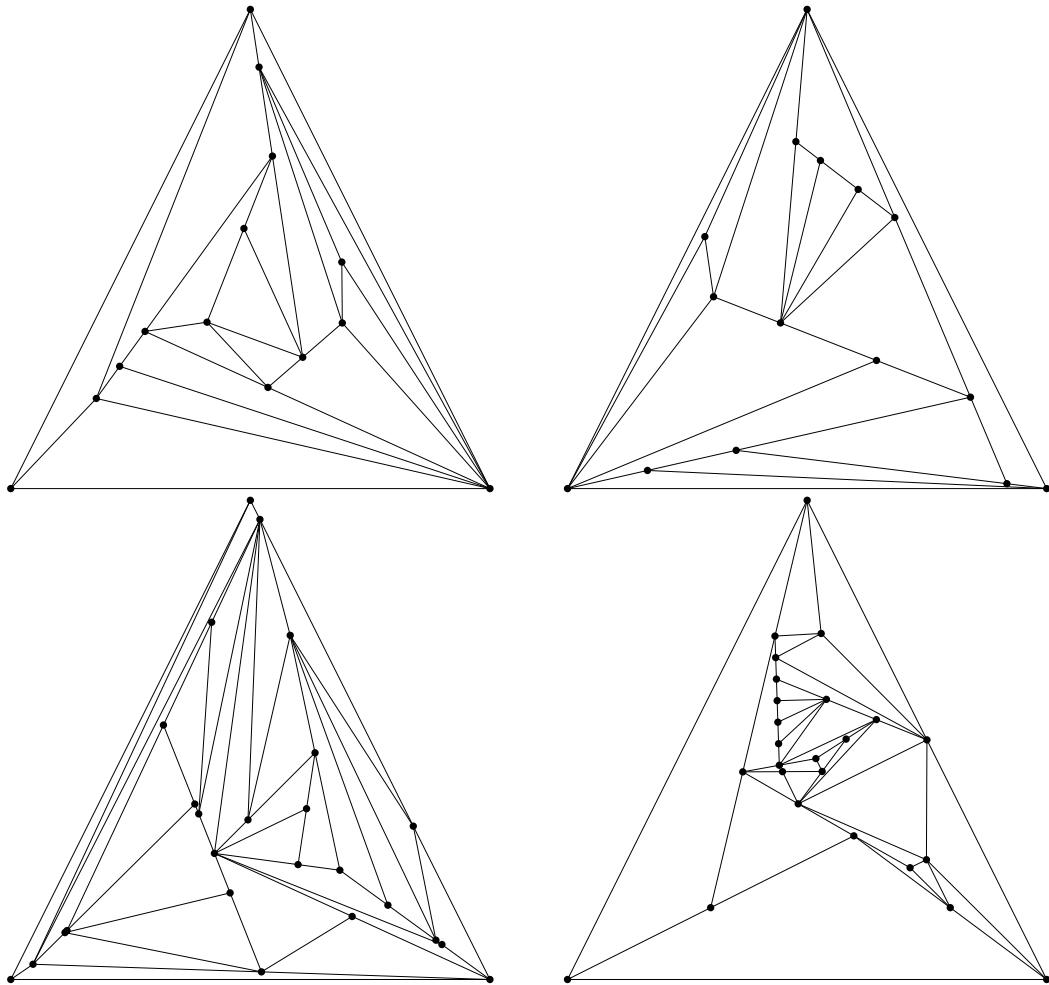
non_int : default `True`, sonst wird nur nach ganzzahligen Lösungen gesucht.

check_int : default `True`, sonst wird die Überprüfung ab Zeile 12 ausgelassen.

plotting : default `True`, sonst wird keine Zeichnung ausgegeben.

ipe : default `None`, bei Übergabe eines Strings **string** wird eine ipe-Datei **string.ipe** erstellt.

cut : default `None`, optional lassen sich die Parameter der pseudo-zufälligen Erzeugung durch drei aufsteigende natürliche Zahlen **[a, b, c]** anpassen, mit $a \leq b \leq c \leq 1000$. Es wird der in Abschnitt 5.2 beschriebene Sampler genutzt. Die normale Einstellung ist **[300, 600, 990]**. In jedem Schritt wird eine der vier Möglichkeiten PG1, PG2, PG3 und PG4 mit Verteilung $a/1000$, $(a+b)/1000$, $(a+b+c)/1000$ und $(1000-a-b-c)/1000$ ausgewählt.



6 Fazit

Wir haben uns mit der Charakterisierung von geradlinigen Dreiecks-Darstellungen (SLTRs) beschäftigt und in Kapitel 3 sowohl notwendige als auch hinreichende Bedingungen für ihre Existenz erarbeitet und uns dabei an Aerts und Felsner orientiert [AF15].

Es wird ein Zusammenhang zwischen SLTRs und Ecken kompatiblen Paaren aus einem Schnyder Wood und einem FAA auf ebenen Graphen bewiesen. Auf diesem Ergebnis aufbauend, haben wir in Kapitel 4 ein Netzwerk zur Lösung eines Gerichteten-Multi-Fluss-Problems konstruiert, auf dem ein zulässiger ganzzahliger Fluss einer SLTR entspricht. In Kapitel 5 haben wir mithilfe dieses Netzwerks einen Algorithmus entwickelt, der die maschinelle Berechnung von SLTRs ermöglicht.

Aerts und Felsner zeigten, dass ein FAA ein System aus harmonischen Gleichungen induziert, dessen Lösung in Abhängigkeit einer Gewichtsfunktion $\{\lambda_e\}_{e \in E}$ die Zeichnung einer SLTR ergibt [AF13b]. Wir haben diesen Ansatz in Kapitel 5 genutzt, um nach der Berechnung eines Guten-FAA eine Zeichnung zu erstellen und haben dabei unterschiedliche Ansätze bei der Wahl der Gewichtsfunktion $\{\lambda_e\}$ verfolgt.

Experimentellen Berechnungen, mit dem in Kapitel 5 beschriebenen Programm, deuten darauf hin, dass eine nicht-ganzzahlige Lösungen des Gerichteten-Multi-Fluss-Problems ausreicht, um eine SLTR zu bestimmen. In Abschnitt 4.2 haben wir uns deswegen mit nicht-ganzzahligen Lösungen des Fluss-Problems beschäftigt und einige dahingehende Beobachtungen gesammelt. Jedoch ist es uns nicht gelungen, die folgende Frage positiv zu beantworten: Liegt die Erkennung von SLTRs in P?

Literaturverzeichnis

- [AF13a] AERTS, Nieke ; FELSNER, Stefan: Henneberg steps for triangle representations. In: *The Seventh European Conference on Combinatorics, Graph Theory and Applications* Springer, 2013, S. 503–509
- [AF13b] AERTS, Nieke ; FELSNER, Stefan: Straight line triangle representations. In: *International Symposium on Graph Drawing* Springer, 2013, S. 119–130
- [AF15] AERTS, Nieke ; FELSNER, Stefan: Straight Line Triangle Representations via Schnyder Labelings. In: *J. Graph Algorithms Appl.* 19 (2015), Nr. 1, S. 467–505
- [EIS75] EVEN, Shimon ; ITAI, Alon ; SHAMIR, Adi: On the complexity of time table and multi-commodity flow problems. In: *16th Annual Symposium on Foundations of Computer Science (sfcs 1975)* IEEE, 1975, S. 184–193
- [Fár48] FÁRY, István: On straight-line representation of planar graphs. In: *Acta Sci. Math.* 11 (1948), S. 229–233
- [Fel01] FELSNER, Stefan: Convex drawings of planar graphs and the order dimension of 3-polytopes. In: *Order* 18 (2001), Nr. 1, S. pp. 19–37
- [Fel04] FELSNER, Stefan: Lattice structures from planar graphs. In: *The electronic journal of combinatorics* 11 (2004), Nr. 1, S. 15
- [Fel12] FELSNER, Stefan: *Geometric graphs and arrangements: some chapters from combinatorial geometry*. Springer Science & Business Media, 2012
- [FF56] FORD, Lester R. ; FULKERSON, Delbert R.: Maximal flow through a network. In: *Canadian Journal of Mathematics* 8 (1956), S. 399–404
- [FS17] FELSNER, Stefan ; SCHEUCHER, Manfred: Arrangements of pseudocircles: triangles and drawings. In: *International Symposium on Graph Drawing and Network Visualization* Springer, 2017, S. 127–139
- [Fus07] FUSY, Éric: *Combinatoire des cartes planaires et applications algorithmiques*, LIX Ecole Polytechnique, Diss., 2007
- [Fus09] FUSY, Éric: Uniform random sampling of planar graphs in linear time. In: *Random Structures & Algorithms* 35 (2009), Nr. 4, S. 464–522

- [FZ08] FELSNER, Stefan ; ZICKFELD, Florian: On the number of planar orientations with prescribed degrees. In: *the electronic journal of combinatorics* 15 (2008), Nr. 1, S. 77
- [Hu63] HU, T C.: Multi-commodity network flows. In: *Operations research* 11 (1963), Nr. 3, S. 344–360
- [KV18] KORTE, Bernhard ; VYGEN, Jens: *Combinatorial optimization*. 6. Springer, 2018
- [LR99] LEIGHTON, Tom ; RAO, Satish: Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms. In: *Journal of the ACM (JACM)* 46 (1999), Nr. 6, S. 787–832
- [LV99] LOVÁSZ, László ; VESZTERGOMBI, Katalin: Geometric representations of graphs. In: *Paul Erdos and his Mathematics* (1999)
- [M⁺12] MAKHORIN, Andrew u. a.: *GNU Linear Programming Kit (Version 8.3)*, 2012. – <http://www.gnu.org/software/glpk/>
- [Mil02] MILLER, Ezra: Planar graphs as minimal resolutions of trivariate monomial ideals. In: *Documenta Mathematica* 7 (2002), S. 43–90
- [S⁺18] STEIN, W. A. u. a. ; THE SAGE DEVELOPMENT TEAM (Hrsg.): *Sage Mathematics Software (Version 8.3)*. The Sage Development Team, 2018. – <http://www.sagemath.org>
- [Sch89] SCHNYDER, Walter: Planar graphs and poset dimension. In: *Order* 5 (1989), Nr. 4, S. pp. 323–343
- [Sch90] SCHNYDER, Walter: Embedding planar graphs on the grid. In: *Proceedings of the first annual ACM-SIAM symposium on Discrete algorithms* (1990), S. 138–148
- [Tut63] TUTTE, William T.: How to draw a graph. In: *Proceedings of the London Mathematical Society* 3 (1963), Nr. 1, S. 743–767
- [Whi33] WHITNEY, Hassler: 2-isomorphic graphs. In: *American Journal of Mathematics* 55 (1933), Nr. 1, S. 245–254