

Testen eines Graphen auf Planarität

Markus Kaiser

Technische Universität München
markus.kaiser@in.tum.de

Zusammenfassung. Der Test auf Planarität eines Graphen hat bedeutende Konsequenzen für Visualisierung und Graphalgorithmen. Ich definiere das Problem, stelle mit dem Satz von Kuratowski eine verbreitete Charakterisierung vor und erarbeite eine weniger bekannte Lösung des Problems anhand der Tiefensuche, das Links-Rechts-Planaritätskriterium.

1 Planarität

Ein Graph $G = (V, E)$ ist ein Tupel der endlichen *Knotenmenge* V und *Kantenmenge* E aus Paaren von Knoten (u, v) . Im Weiteren werden nur *einfache* Graphen ohne Mehrfachkanten oder Schleifen, also Kanten mit selben Anfangs- und Endknoten betrachtet, da diese keinen Einfluss auf die Planarität eines Graphen haben. Außerdem bezeichnet $n = |V|$ und $m = |E|$.

Eine *Zeichnung* eines Graphen ordnet jedem Knoten Koordinaten auf einer Ebene und jeder Kante eine Kurve zwischen ihren Endknoten zu. Eine Zeichnung heißt *planar* wenn sich keine Kanten schneiden. Zwei Zeichnungen heißen *äquivalent*, wenn ihre Reihenfolge der Kanten um jeden Knoten gleich ist. Diese Äquivalenzklasse von Zeichnungen wird *Einbettung* genannt und ist planar falls eine ihrer Zeichnungen planar ist.

Definition 1. *Ein Graph heißt planar, wenn für ihn eine planare Zeichnung existiert.*

Zeichnungen unterteilen die Ebene in *Flächen* mit f als der Anzahl der Flächen. Es existieren abgeschlossene *innere* Flächen und genau eine offene *äußere* Fläche. Der eulersche Polyedersatz stellt einen Zusammenhang her zwischen der Anzahl Knoten, Kanten und Flächen in einer planaren Zeichnung.

Theorem 1 (Eulerscher Polyedersatz). *Für zusammenhängende planare Graphen gilt in jeder Zeichnung $n - m + f = 2$*

Daraus ergibt sich direkt ein Kriterium zur Planaritätsprüfung anhand der Kantenzahl.

Lemma 1. *Für jeden planaren Graphen gilt $m \leq 3n - 6$*

Beweis. Jede Kante kann maximal zwei Flächen begrenzen. Außerdem wird eine Fläche von mindestens drei Kanten begrenzt. Daraus ergibt sich $f \leq \frac{2m}{3}$ und $m \leq 3n - 6$.

Dieses Lemma stellt außerdem für planare Graphen einen linearen Zusammenhang zwischen Kanten und Knoten her, es gilt $m \in \mathcal{O}(n)$, während für allgemeine Graphen nur $m \in \mathcal{O}(n^2)$ gilt.

1.1 Die Sätze von Kuratowski und Wagner

In den 1930er Jahren haben Kuratowski[Kur30] und Wagner[Wag37] erste Charakterisierungen für planare Graphen gefunden. Diese Charakterisierungen basieren auf *Unterteilungen* und *Minoren*. Eine Unterteilung eines Graphen entsteht durch beliebig häufiges Einfügen von neuen Knoten auf vorhandenen Kanten. Ein Graph ist Minor eines anderen Graphen, wenn er sich durch Löschen von Knoten oder Kanten oder Kantenkontraktion gewinnen lässt.

Theorem 2. Ein Graph ist genau dann planar wenn er keinen Teilgraphen (Kuratowski) enthält, der eine Unterteilung von K_5 oder $K_{3,3}$ ist und er nicht Minor (Wagner) eines dieser Graphen ist.

K_5 und $K_{3,3}$ sind also die kleinstmöglichen nicht planaren Graphen. Dieses Kriterium ist einfach zu verstehen, würde aber bei direkter Anwendung zu mindestens exponentiellen Algorithmen führen und ist deswegen zum Testen auf Planarität ungeeignet.

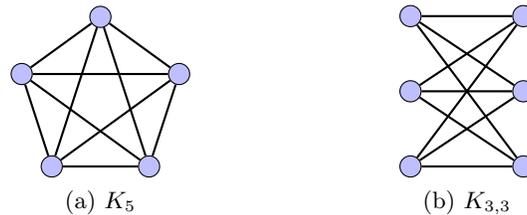


Abb. 1: Minimal nicht planare Graphen K_5 und $K_{3,3}$.

1.2 Geschichte der Planaritätsprüfung

Während die Formulierung des Problems einfach ist und Kuratowski und Wagner in den dreißiger Jahren eine Charakterisierung für planare Graphen finden konnten, war längere Zeit kein effizienter Algorithmus zum Testen auf Planarität bekannt. Erst 1961 fanden Auslander und Parter[AP61] einen Polynomialzeitalgorithmus basierend auf der Unterteilung von Kreisen. 1974 stellten Hopcroft und Tarjan[HT74] einen ersten Linearzeitalgorithmus vor, der auf dem Betrachten einzelner Pfade im Graphen aufbaut. Diese Algorithmen verwenden jedoch meist komplexe Datenstrukturen und sind relativ schwer zu implementieren.

Im Folgenden soll das *LR-Planaritätskriterium* vorgestellt werden, das auf der Tiefensuche basiert und ursprünglich von de Fraysseix, Osona de Mendez und Rosenstiehl 2006[dFdMR06] veröffentlicht und 2009 von Brandes[Bra09] näher beschrieben wurde.

2 Das LR-Planaritätskriterium

Die Planarität eines Graphen wird durch die Beschaffenheit seiner Kreise bestimmt. Während zyklensfreie Graphen immer planar sind, können sich Kreise schneiden. Zyklen schließen eine Fläche auf der Ebene. Um Überschneidungen zu vermeiden müssen Teilgraphen in Berührung zu diesem Kreis entweder komplett innerhalb oder komplett außerhalb gezeichnet werden.

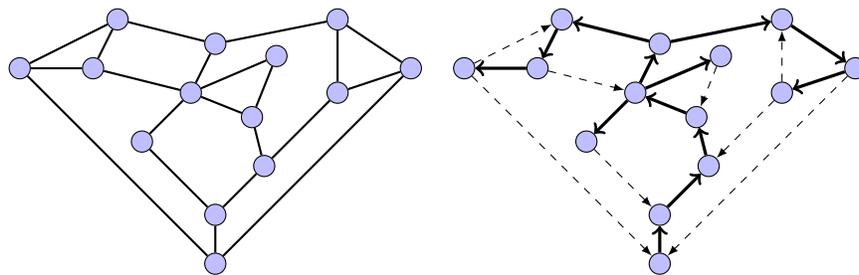
Weiterhin existieren genau zwei Möglichkeiten einen Kreis auf der Ebene zu zeichnen, im und gegen den Uhrzeigersinn. Entscheidet man sich bei einem Kreis

für eine Orientierung werden so möglicherweise Beschränkungen für alle anderen Kreise erzeugt, da sich Regeln für die planare Einbettung ergeben. Das Testen auf Planarität soll mit Hilfe der *LR-Zerlegung* zurückgeführt werden auf den Test, ob sich diese Regeln widersprechen.

In einem Graphen existieren jedoch potentiell exponentiell viele einfache Kreise. Um diese Zahl einzuschränken wird eine *DFS-Orientierung* des Graphen erzeugt und es wird gezeigt, dass allein eine Betrachtung der Rückwärtskanten eine Aussage über Planarität zulässt.

2.1 Tiefensuche

Eine Tiefensuche auf einem ungerichteten Graphen $G = (V, E)$ erzeugt eine *DFS-Orientierung* $G = (G, V)$ wobei jeder Kante die Richtung zugewiesen wird, in der sie durchlaufen wird. Die Tiefensuche ergibt außerdem eine Partition $E = T \uplus B$ der Kanten in Baumkanten T und Rückwärtskanten B . Der Graph (V, T) ist maximal zyklensfrei, das Hinzufügen einer beliebigen Kante aus B erzeugt einen Zyklus.



(a) Beispielgraph

(b) DFS-Orientierung

Abb. 2: Planarer Beispielgraph (aus [Bra09])

Lemma 2. Sei $G = (V, T \uplus B)$ ein DFS-orientierter Graph.

1. Alle Kreise in G werden durch genau eine Rückwärtskante induziert.
2. Zwei Kreise sind entweder unabhängig oder teilen sich einen Pfad in T .

Teilen sich zwei Kreise einen Pfad in T , so ist ihr *Gabelungspunkt* definiert als der letzte Knoten der Teil beider Kreise ist. Ihre *Gabelung* bezeichnet die letzte gemeinsame Baumkante und die jeweils folgenden Kanten. Da Kreise nur durch Rückwärtskanten erzeugt werden, muss für jede dieser Gabelungen eine Verschachtelungsreihenfolge um ihren Gabelungspunkt gefunden werden.

Hierfür wird der Begriff des *Lowpoint* benötigt. Der Lowpoint $lowpt(v)$ eines Knotens ist derjenige Knoten, der mittels einer Rückwärtskante von ihm selbst oder eines seiner Kinder erreicht werden kann und der Wurzel am nächsten ist (dessen DFS-Index also am kleinsten ist). Existiert kein solcher Knoten oder nur solche, die von der Wurzel weiter entfernt sind als v , so ist der Lowpoint eines Knotens der Knoten selbst. Der Lowpoint einer Kante $e = (u, v)$ ist definiert als $lowpt(e) = \min\{lowpt(v), u\}$.

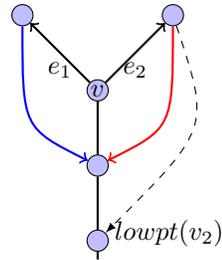


Abb. 3: Lowpoint

2.2 LR-Zerlegung

Es gibt zwei Klassen von Kreisen in einer Zeichnung eines planaren Graphen, diejenigen im und diejenigen gegen den Uhrzeigersinn. Teilen sich zwei Kreise einen Baumpfad so können sie sich potentiell überschneiden. Solche Überschneidungen können nur vermieden werden, indem ein Kreis komplett innerhalb des anderen gezeichnet wird. Abbildung 4 führt intuitiv zur Aussage wann Kreise ineinander *verschachtelt* sein können.

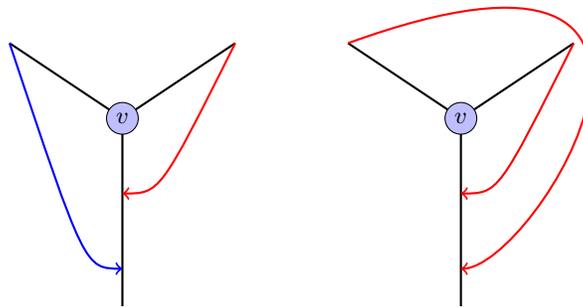


Abb. 4: Mögliche Konfigurationen für Kreise mit gemeinsamen Baumpfaden ohne symmetrische Fälle.

Beobachtung 1 *In einer planaren Zeichnung eines DFS-orientierten Graphen sind zwei Kreise mit gemeinsamem Baumpfad genau dann verschachtelt, wenn sie gleich orientiert sind.*

Diese Beobachtung macht jedoch keine Aussage darüber, welcher Kreis der Innere ist. Hierfür wird angenommen, dass die Wurzel der DFS-Orientierung immer an der äußeren Fläche der Zeichnung liegt. Da ein Kreis immer durch

genau eine Rückwärtskante induziert wird, können zwei solche Kreise c_1 und c_2 mit ihren jeweiligen Rückwärtskanten b_1 und b_2 identifiziert werden. Nimmt man außerdem an, dass $\text{lowpt}(b_1) < \text{lowpt}(b_2)$, dann muss c_2 innerhalb von c_1 liegen. Da die Lowpoints unterschiedlich sind, muss es einen Baumpfad geben, der sie verbindet. Dieser Pfad kann aber nicht Teil des inneren Kreises sein, da die Wurzel der DFS-Orientierung die äußere Fläche berühren soll.

Beobachtung 2 *In einer planaren Zeichnung eines DFS-orientierten Graphen bei dem die Wurzel die äußere Fläche berührt werden Kreise die sich Baumpfade teilen anhand ihrer Lowpoints verschachtelt.*

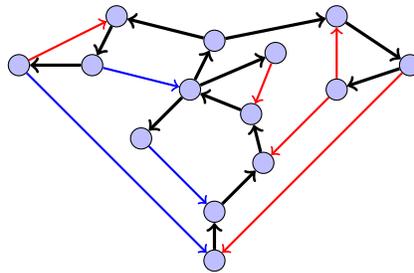


Abb. 5: LR-Zerlegung des Beispielgraphen aus Abb. 2

Definition 2 (LR-Zerlegung). *Sei $G = (V, T \uplus B)$ ein DFS-orientierter Graph. Eine Zerlegung $B = L \uplus R$ seiner Rückwärtskanten heißt LR-Zerlegung wenn für jeden Knoten mit ausgehenden Kanten e_1, e_2 gilt:*

- *Alle Rückwärtskanten von e_1 mit ihrem Ende über $\text{lowpt}(e_2)$ gehören zur einen und*
- *Alle Rückwärtskanten von e_2 mit ihrem Ende über $\text{lowpt}(e_1)$ gehören zur anderen Klasse.*

2.3 LR-Ordnung

Es soll nun gezeigt werden, dass die Existenz einer LR-Zerlegung eine Charakterisierung der Planarität darstellt.

Theorem 3. *Ein Graph ist genau dann planar wenn er eine LR-Zerlegung zulässt.*

Die Notwendigkeit der Bedingung ist klar, da aus einer gegebenen planaren Zeichnung und einer DFS-Orientierung sofort eine LR-Zerlegung erzeugt werden kann, indem die Kanten in ihre jeweiligen Gruppen eingeteilt werden. Um zu

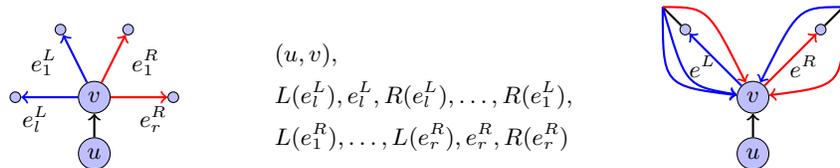
zeigen, dass die Bedingung hinreichend ist, wird im Folgenden aus einer LR-Zerlegung eine planare Einbettung erzeugt.

Da planare Einbettungen durch ihre Kantenreihenfolge um Knoten definiert sind, muss aus der LR-Zerlegung eine solche Reihenfolge konstruiert werden. Hierbei gilt, dass bei einer Gabelung linksorientierte Kanten im Uhrzeigersinn vor rechtsorientierten Kanten angeordnet werden müssen. Linke (rechte) Kanten müssen außerdem von innen nach außen (von außen nach innen) um ihren Knoten angeordnet werden.

Diese Orientierung ist bisher jedoch nur für Rückwärtskanten bekannt und muss auf Baumkanten erweitert werden. Dafür ist der Begriff der ausgerichteten LR-Zerlegung nötig. Eine LR-Zerlegung heißt *ausgerichtet*, wenn alle Rückwärtskanten einer Kante e , die in $lowpt(e)$ enden, zur selben Seite gehören. Jede LR-Zerlegung kann ausgerichtet werden, da für alle Kanten die in $lowpt(e)$ enden die selben Bedingungen gelten. Hat eine Baumkante Rückwärtskanten, dann wird sie der Seite derjenigen Kante zugewiesen, die den höchsten Rückkehrpunkt (analog zum Lowpoint) besitzt. Andererseits kann die Seite beliebig gewählt werden.

Für Baumkanten und Rückwärtskanten muss nun eine Verschachtelungsreihenfolge \prec festgelegt werden. Hierfür wird angenommen, dass alle Kanten rechte Kanten sind und wir betrachten eine Gabelung aus Knoten v und Kanten e_1 und e_2 , die v verlassen. Wenn beide Rückwärtskanten haben deren Lowpoint tiefer als v liegen, dann ist v Gabelungspunkt für zwei Kreise, die sich die Baumkante (u, v) teilen. Da beide Kreise rechtsorientiert sind müssen sie richtig verschachtelt werden. Da die Wurzel der Tiefensuche an der äußeren Fläche liegt, müssen wir $e_1 \prec e_2$ genau dann annehmen, wenn $lowpt(e_1) < lowpt(e_2)$. Haben beide den selben Lowpoint, jedoch e_2 noch eine höhere Rückwärtskante und e_1 nicht, so besitzt e_2 eine Sehne und wir nehmen $e_1 \prec e_2$ an, da zusätzliche Kreise in e_2 immer innerhalb des durch e_1 definierten Kreises liegen müssen. Wenn sowohl e_1 als auch e_2 eine Sehne haben wird die Reihenfolge beliebig gewählt, da beide in verschiedenen Klassen gezeichnet werden müssen um überschneidungsfrei sein zu können.

Definition 3 (LR-Ordnung). Zu einer gegebenen LR-Partition seien $e_1^L \prec \dots \prec e_l^L$ die linksorientierten und $e_1^R \prec \dots \prec e_r^R$ die rechtsorientierten ausgehenden Kanten von v . Wenn v nicht die Wurzel ist sei u seine Quelle. Die LR-Ordnung im Uhrzeigersinn um v ist dann gegeben durch:



$L(e)$ und $R(e)$ bezeichnen die linken und rechten eingehenden Kanten der Kreise die sich e teilen. Die Verschachtelung dieser Kanten verhält sich wie die Verschachtelung der Kanten an der entsprechenden Gabelung.

Lemma 3. Eine LR-Ordnung stellt eine planare Einbettung einer LR-Zerlegung dar.

Ein Beweis durch Widerspruch findet sich in [Bra09]. Die Existenz einer LR-Zerlegung ist also ein Planaritätskriterium.

3 Anwendung des Kriteriums

Das LR-Planaritätskriterium zeigt, dass das Testen auf Planarität einem Testen auf die Existenz einer LR-Zerlegung entspricht. Hier soll nun mit Hilfe des Kriteriums ein polynomieller Algorithmus konstruiert werden, [Bra09] beschreibt, wie ein Linearzeitalgorithmus auf Basis des Kriteriums implementiert werden kann.

Sei $G = (V, T \uplus B)$ eine DFS-Orientierung von G . Es muss also überprüft werden, ob B eine LR-Zerlegung zulässt. Dafür müssen für alle Gabelungen alle Bedingungen gleichzeitig erfüllt werden können. Aus Definition 2 ergeben sich direkt die folgenden Bedingungen für Kanten.

Lemma 4. Seien $b_1 = (u_1, v_1)$ und $b_2 = (u_2, v_2)$ zwei Rückwärtskanten mit überlappenden Zyklen und sei (u, v) , (v, w_1) , (v, w_2) ihre Gabelung. Außerdem sei $C(b)$ die Menge der Knoten, die sich auf dem durch b definierten Zyklus befinden. Dann gilt:

1. b_1 und b_2 gehören zu verschiedenen Klassen wenn $\text{lowpt}(w_2) < v_1$ und $\text{lowpt}(w_1) < v_2$
2. b_1 und b_2 gehören zur selben Klasse wenn es eine Kante $e' = (x, y)$ gibt, sodass $x \in C(b_1) \cap C(b_2)$ und $y \notin C(b_1) \cap C(b_2)$ und $\text{lowpt}(y) < \min\{v_1, v_2\}$

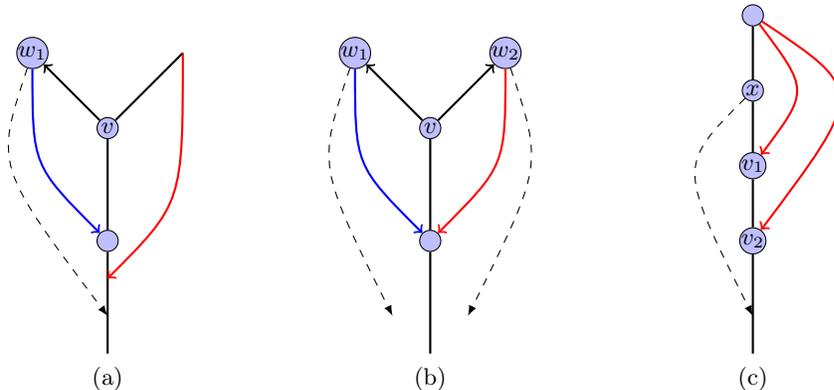


Abb. 6: Die Bedingungen für Rückwärtskanten ergeben sich durch drei Arten von Minimalkonfigurationen ([Bra09]). (a) und (b) zeigen Verschieden-Bedingungen, (c) zeigt eine Gleich-Bedingung

Unterliegt ein Paar von Rückwärtskanten sowohl einer Verschieden- als auch einer Gleich-Bedingung, kann keine LR-Partition existieren und der Graph ist nicht planar. Weitergehend können die Bedingungen in einem *Constraint-Graph* dargestellt werden, wobei Kanten mit Verschieden-Bedingungen eine andere Markierung bekommen als solche mit einer Gleich-Bedingung. Werden nun alle Kanten mit Gleich-Bedingungen kontrahiert muss der Graph auf Bipartitheit überprüft werden, um zu entscheiden, ob eine LR-Zerlegung existiert und damit, ob der Graph planar ist. Dieser Algorithmus hat quadratische Laufzeit ([Bra09]).

Literatur

- AP61. L. Auslander and S. V. Parter. On imbedding graphs in the sphere. *J. Math. and Mech.*, 114:517–523, 1961.
- Bra09. Ulrik Brandes. The left-right planarity test. Manuscript submitted for publication, 2009.
- dFdMR06. Hubert de Fraysseix, Patrice Ossona de Mendez, and Pierre Rosenstiehl. Trémaux trees and planarity. *Int. J. Found. Comput. Sci.*, 17(5):1017–1030, 2006.
- HT74. John Hopcroft and Robert Tarjan. Efficient planarity testing. *J. ACM*, 21(4):549–568, October 1974.
- Kur30. K. Kuratowski. Sur le Probleme des Courbes Gauches en Topologie. *Fundamenta Mathematicae*, 15:271–283, 1930.
- Wag37. K. Wagner. Über eine Eigenschaft der ebenen Komplexe. *Math. Ann.*, 114, 1937.