Testen eines Graphen auf Planarität

Proseminar “Graph Drawing”

Markus Kaiser

Technische Universität München

2012-05-22

Grundlegendes

- Graph G, Knotenmenge V, Kantenmenge E
- $G = (V, E)$
- $E \subseteq V^2$, $|V| = n$, $|E| = m$

- einfach: Keine Schleifen oder Mehrfachkanten
- zusammenhängend: Es existiert ein Pfad von jedem Knoten zu jedem Knoten
- Baum: Zyklenfreier zusammenhängender Graph
- k-zusammenhängend: Zusammenhängend nach Entfernen von $k-1$ Knoten

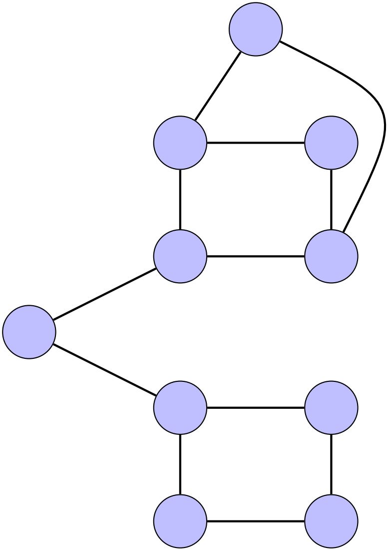
Grundlegendes

- Graph G, Knotenmenge V, Kantenmenge E
- $G = (V, E)$
- $E \subseteq V^2$, $|V| = n$, $|E| = m$

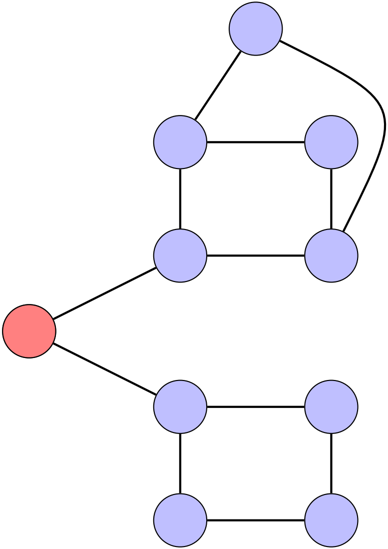
- einfach: Keine Schleifen oder Mehrfachkanten
- zusammenhängend: Es existiert ein Pfad von jedem Knoten zu jedem Knoten
- Baum: Zyklenfreier zusammenhängender Graph
- k-zusammenhängend: Zusammenhängend nach Entfernen von $k - 1$ Knoten

Grundlegendes

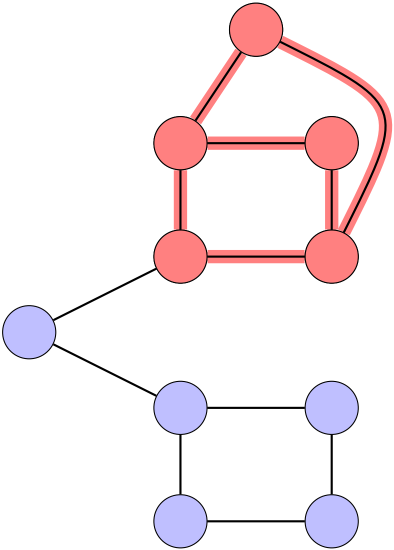
- Artikulationsknoten (*cut vertex*)
- Block
- Pfad
- Zyklus
- Facette (*face*)
- Bipartitheit



- Artikulationsknoten (*cut vertex*)
- Block
- Pfad
- Zyklus
- Facette (*face*)
- Bipartitheit

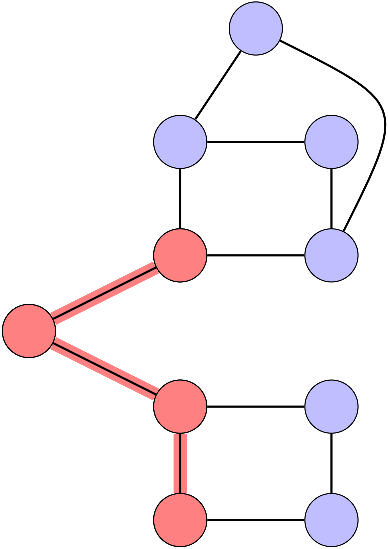


- Artikulationsknoten (*cut vertex*)
- Block
- Pfad
- Zyklus
- Facette (*face*)
- Bipartitheit



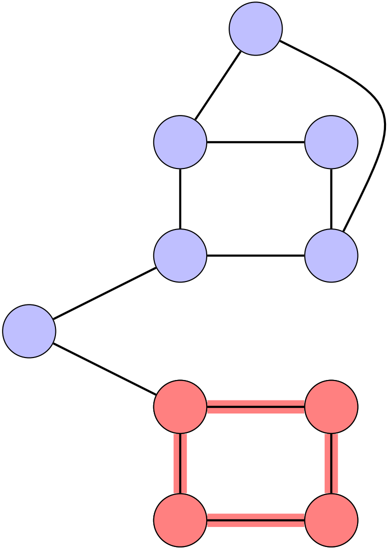
Grundlegendes

- Artikulationsknoten (*cut vertex*)
- Block
- Pfad
- Zyklus
- Facette (*face*)
- Bipartitheit



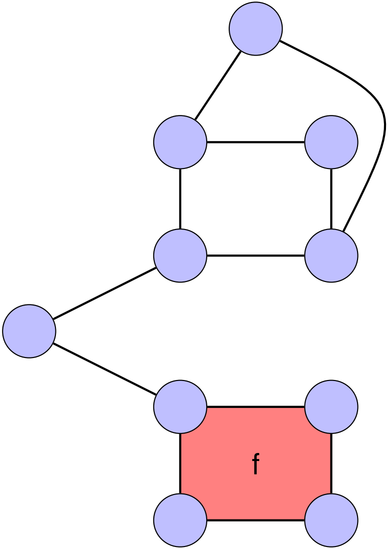
Grundlegendes

- Artikulationsknoten (*cut vertex*)
- Block
- Pfad
- Zyklus
- Facette (*face*)
- Bipartitheit



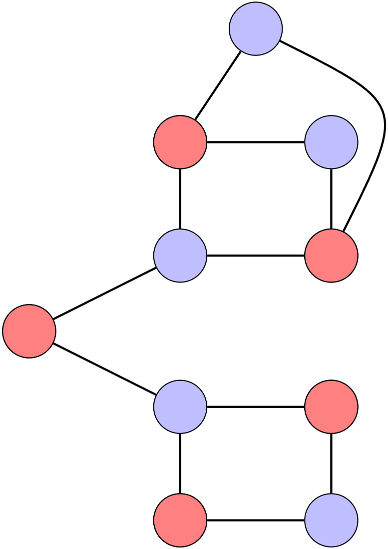
Grundlegendes

- Artikulationsknoten (*cut vertex*)
- Block
- Pfad
- Zyklus
- Facette (*face*)
- Bipartitheit



Grundlegendes

- Artikulationsknoten (*cut vertex*)
- Block
- Pfad
- Zyklus
- Facette (*face*)
- Bipartitheit



- offene Jordankurve

- Eine Zeichnung ordnet in $\mathbb{R}^2$ jedem Knoten Koordinaten und jeder Kante eine offene Jordankurve zu
- äquivalent: Gleiche Reihenfolge der Kanten an Knoten
- planar: Keine sich überschneidenden Kurven

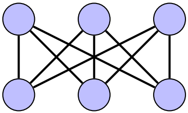
- offene Jordankurve

- Eine Zeichnung ordnet in $\mathbb{R}^2$ jedem Knoten Koordinaten und jeder Kante eine offene Jordankurve zu
- äquivalent: Gleiche Reihenfolge der Kanten an Knoten
- planar: Keine sich überschneidenden Kurven

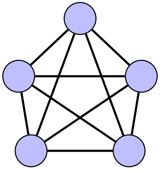
Theorem

Ein Graph ist genau dann planar wenn...

- *Kuratowski (1930): er keinen Teilgraphen enthält der eine Unterteilung von K_5 oder $K_{3,3}$ ist.*
- *Wagner (1937): er nicht K_5 oder $K_{3,3}$ als Minoren enthält.*



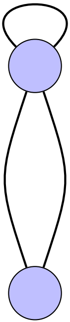
$K_{3,3}$



K_5

Vorüberlegungen

- Schleifen entfernen
- Parallele Kanten entfernen
- Zusammenhangskomponenten getrennt betrachten
- Blöcke getrennt betrachten
- Knoten von Grad 1 entfernen
- Knoten von Grad 2 durch Kanten ersetzen

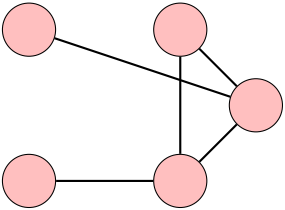


Teilgraphen sind…

- planar wenn $|E| < 9$ oder $|V| < 5$
- nicht planar wenn $|E| > 3|V| - 6$

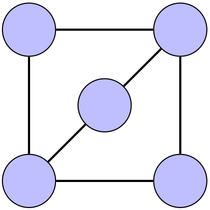
Vorüberlegungen

- Schleifen entfernen
- Parallele Kanten entfernen
- Zusammenhangskomponenten getrennt betrachten
- Blöcke getrennt betrachten
- Knoten von Grad 1 entfernen
- Knoten von Grad 2 durch Kanten ersetzen



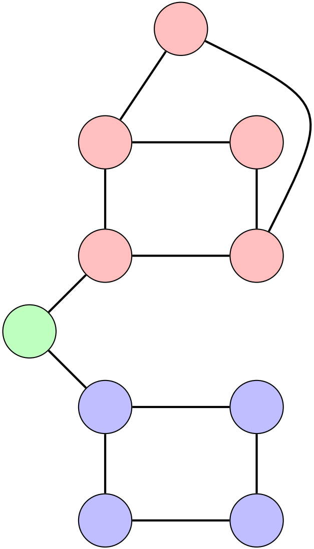
Teilgraphen sind...

- planar wenn $|E| < 9$ oder $|V| < 5$
- nicht planar wenn $|E| > 3|V| - 6$



Vorüberlegungen

- Schleifen entfernen
- Parallele Kanten entfernen
- Zusammenhangskomponenten getrennt betrachten
- Blöcke getrennt betrachten
- Knoten von Grad 1 entfernen
- Knoten von Grad 2 durch Kanten ersetzen

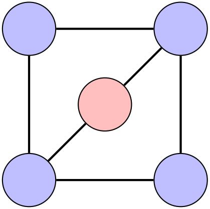


Teilgraphen sind...

- planar wenn $|E| < 9$ oder $|V| < 5$
- nicht planar wenn $|E| > 3|V| - 6$

Vorüberlegungen

- Schleifen entfernen
- Parallele Kanten entfernen
- Zusammenhangskomponenten getrennt betrachten
- Blöcke getrennt betrachten
- Knoten von Grad 1 entfernen
- Knoten von Grad 2 durch Kanten ersetzen

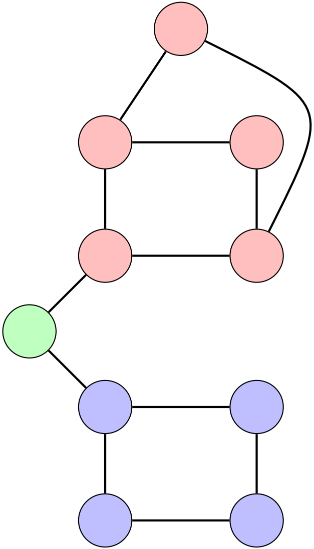


Teilgraphen sind…

- planar wenn $|E| < 9$ oder $|V| < 5$
- nicht planar wenn $|E| > 3|V| - 6$

Vorüberlegungen

- Schleifen entfernen
- Parallele Kanten entfernen
- Zusammenhangskomponenten getrennt betrachten
- Blöcke getrennt betrachten
- Knoten von Grad 1 entfernen
- Knoten von Grad 2 durch Kanten ersetzen



Teilgraphen sind…

- planar wenn $|E| < 9$ oder $|V| < 5$
- nicht planar wenn $|E| > 3|V| - 6$

- Knotenbasiert
  - Ein Graph bleibt planar wenn man einzelne Knoten entfernt
  - Prozess kann umgekehrt werden
- Kreisbasiert
  - Nur Graphen mit Zyklen können nicht planar sein
  - Zyklen trennen Ebene in zwei Bereiche
  - Teilgraphen sind entweder innen oder außen

LR-Algorithmus

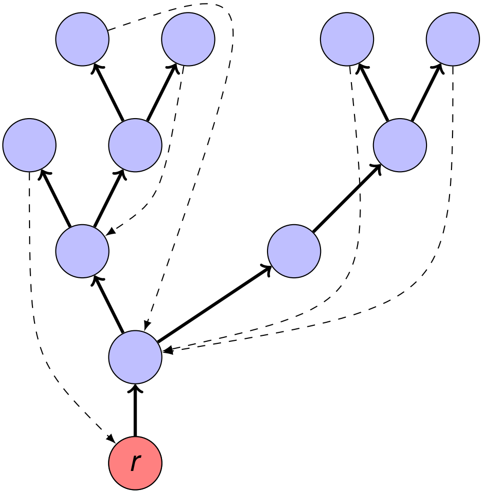
- Kreisbasiert
- Kantenweises betrachten des Graphen
- Weiterentwicklung von *Hopcroft und Tarjan* (1974)

- Original von *de Fraysseix, Ossona de Mendez und Rosenstiehl* (2006)
- Beschrieben von *Ulrik Brandes* (2009)

Tiefensuche

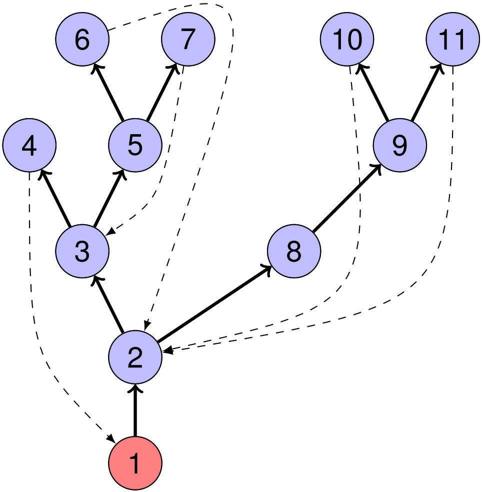
- Wurzel
- DFS-Baum
- Rückwärtskante
- Orientierung

- DFS-Index
- strikt höher/tiefer
- lowpoint
  $lowpt([5]) = 2$

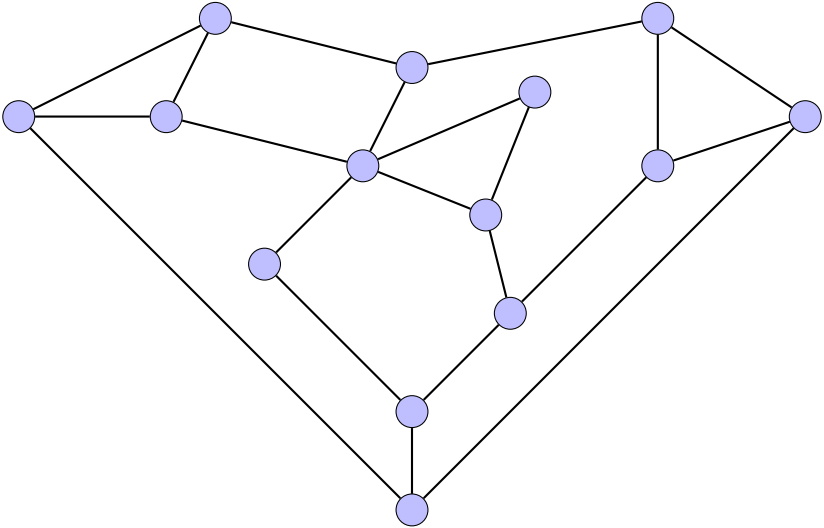


- Wurzel
- DFS-Baum
- Rückwärtskante
- Orientierung

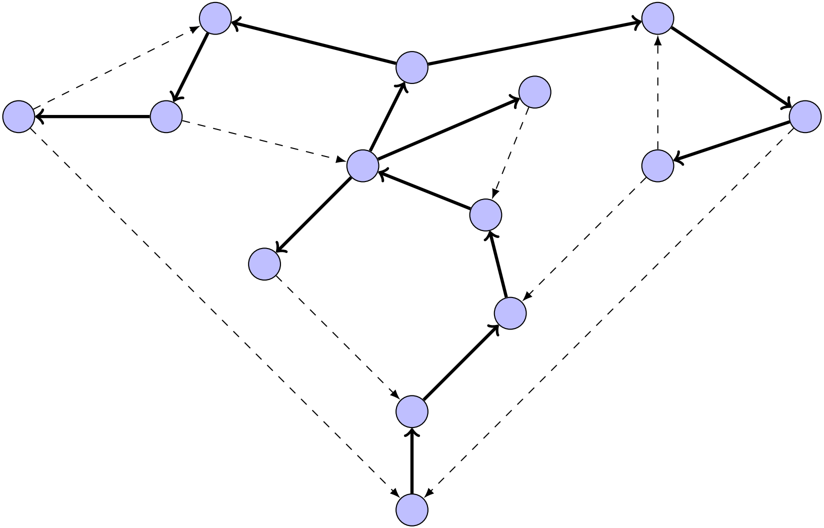
- DFS-Index
- strikt höher/tiefer
- lowpoint
  $lowpt([5]) = 2$



Beispiel

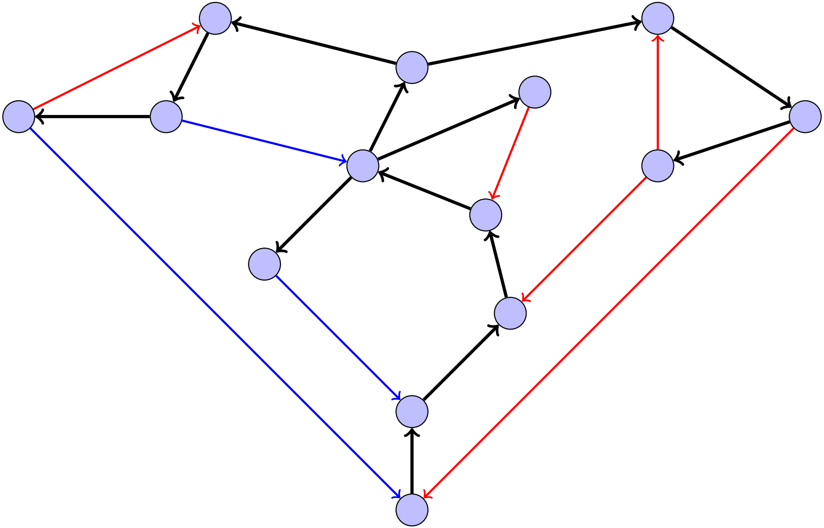


Beispiel



DFS

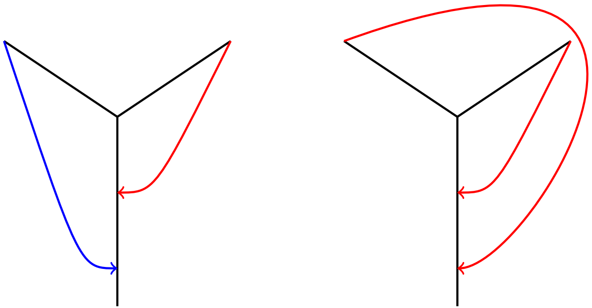
Beispiel



LR-Zerlegung

LR-Zerlegung

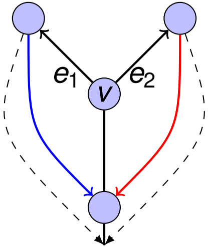
- Jede Rückwärtskante erzeugt einen Zyklus
- LR-Zerlegung teilt Zyklen in solche im und gegen den Uhrzeigersinn
- Quelle außen
- Verschachtelung nur bei gleicher Orientierung



Definition

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



Theorem

Ein Graph ist genau dann planar, wenn er eine LR-Zerlegung zulässt.

- Planare Zeichnung $\rightarrow$ LR-Zerlegung intuitiv

- Andere Richtung: Konstruktiv
- Probleme ergeben sich bei überlappenden Kreisen
- Welche Kreise liegen außen, welche innen?
- Bedingungen abhängig von Baumkanten

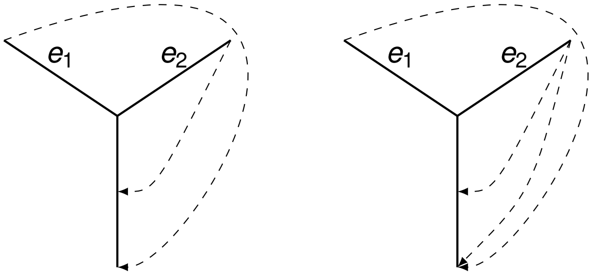
Beobachtung

Die Baumkante e_1 muss außerhalb von Baumkante sein e_2 wenn

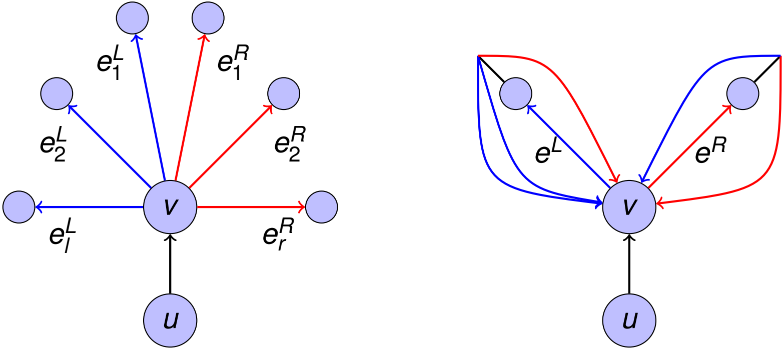
- $lowpt(e_1) < lowpt(e_2)$

oder

- $lowpt(e_1) = lowpt(e_2)$ und e_2 eine Sehne besitzt



Verschachtelung



Verschachtelung

Definition

Seien $e_1^L, \ldots, e_l^L$ die linken ausgehenden Baumkanten und $e_1^R, \ldots, e_r^R$ die rechten ausgehenden Baumkanten eines Knotens v und u sein Elternknoten. Dann ist die LR-Ordnung im Uhrzeigersinn

$$\begin{aligned}&(u, v),\\ &L(e_l^L), e_l^L, R(e_l^L), \ldots, R(e_1^L),\\ &L(e_1^R), \ldots, L(e_r^R), e_r^R, R(e_r^R)\end{aligned}$$

Dabei ist e_1 außerhalb von e_2 usw.

$L(e)$ bezeichnet die linken eingehenden Rückwärtskanten deren Kreise sich e teilen.

Lemma

Eine LR-Ordnung stellt eine Planarisierung einer LR-Zerlegung dar.

Beweisidee

Beweis durch Widerspruch

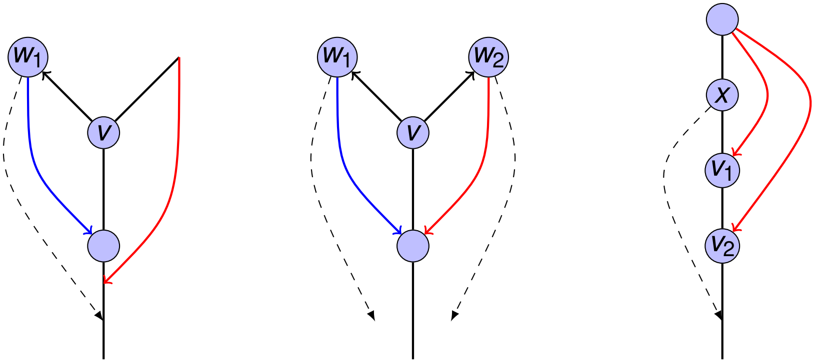
- Baumkanten sind immer planar
- Überschneidungen wenn dann nur in Rückwärtskanten
- Fallunterscheidung
  - selbe Klasse
  - verschiedene Klassen

Algorithmus

- Testen auf Planarität entspricht Testen auf LR-Zerlegung
- LR-Zerlegung stellt Randbedingungen

- Welche Bedingungen gibt es?
  - Überlappende Kreise
  - Gabelungen
- Können alle Bedingungen gleichzeitig erfüllt werden?

Algorithmus



Korollar

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. Dann gilt:

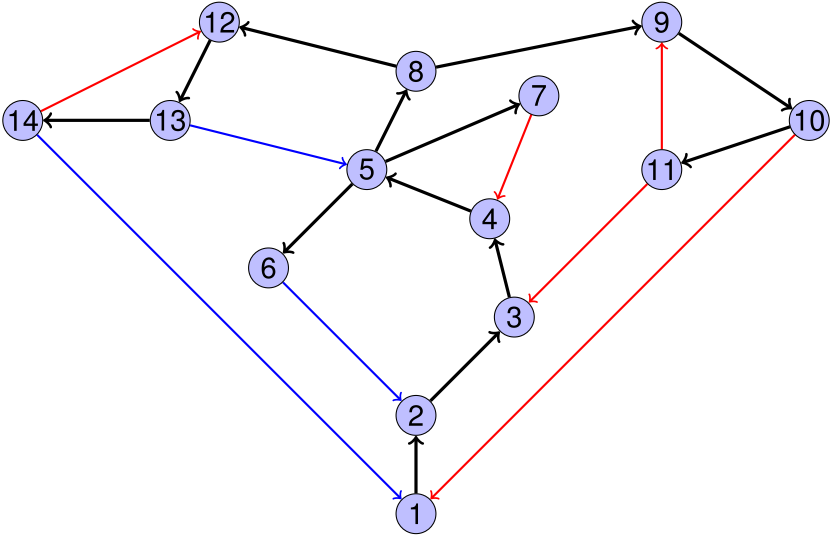
- b_1 und b_2 gehören zu verschiedenen Klassen wenn $lowpt(w_2) < v_1$ und $lowpt(w_1) < v_2$
- b_1 und b_2 gehören zur selben Klasse wenn es eine Kante $e' = (x, y)$ gibt, sodass $x \in C(b_1) \cap C(b_2)$ und $y \notin C(b_1) \cap C(b_2)$ und $lowpt(y) < min\{v_1, v_2\}$

$C(b)$ bezeichnet Knoten auf dem Zyklus, der durch b geschlossen wird.

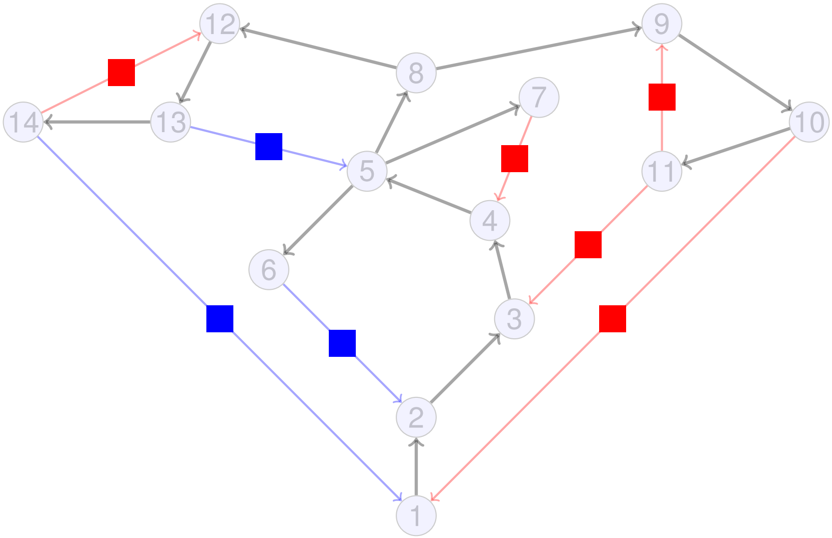
Algorithmus

- Die Bedingungen lassen sich in einem Constraint Graphen darstellen
- Rückwärtskanten werden Knoten
- Abhängigkeiten werden Kanten mit Beschriftung

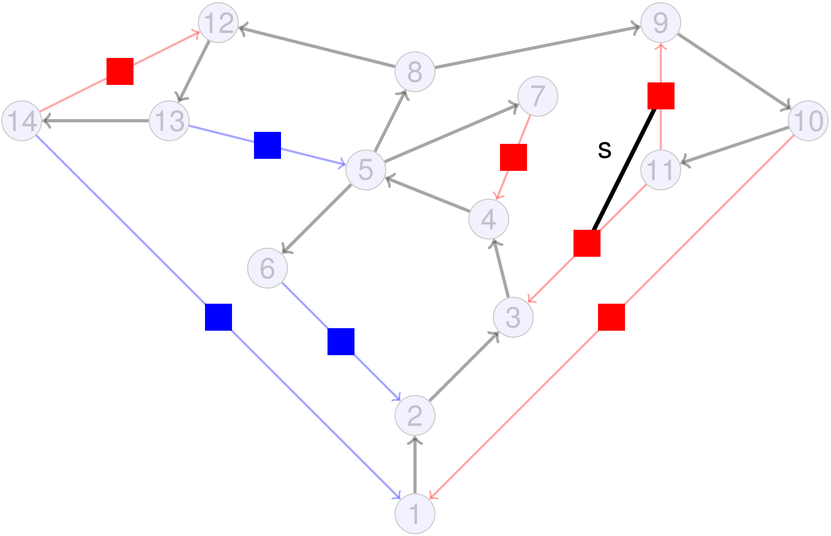
Constraintgraph



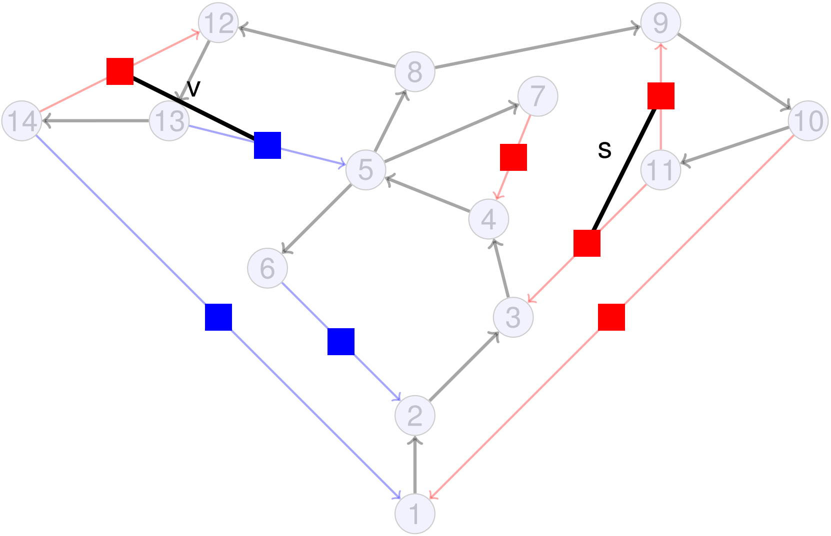
Constraintgraph



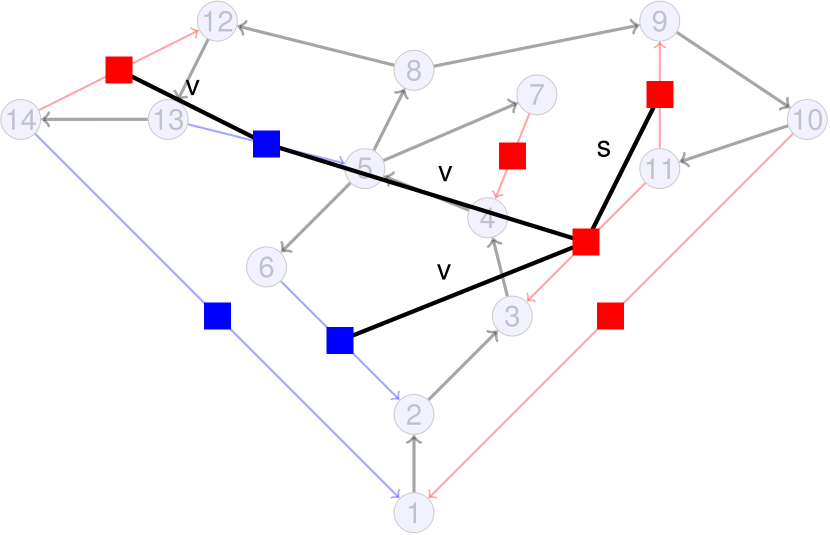
Constraintgraph



Constraintgraph



Constraintgraph



Algorithmus

- Erstellen einer DFS-Orientierung
- Berechnen von lowpoints
- Erzeugen des Constraint Graphs
- Ist dieser Widerspruchsfrei ist der Graph planar

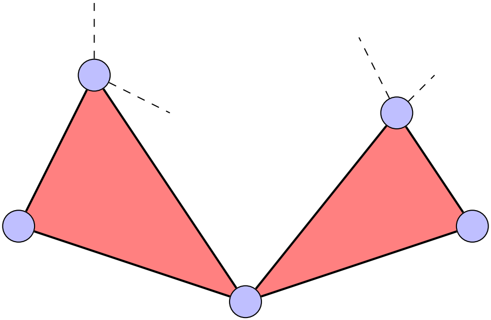
- Erzeugen des Constraint Graphen polynomieller Aufwand
- Kann in $\mathcal{O}(n)$ implementiert werden indem der Graph nicht explizit erzeugt wird

Danke für die Aufmerksamkeit!

Fragen?

Backup

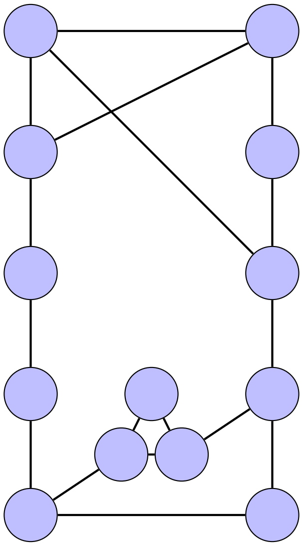
- Es existiert immer eine Kette von Planaren Graphen
- Einmal geschlossene Facetten bleiben geschlossen
- Die Reihenfolge von Knoten um eine Facette bleibt erhalten



Segmente

- Jede Kante eines Blocks ist Teil von mindestens einem Kreis
- Kreise enthalten Segmente
  - Kanteninduzierte Teilgraphen
  - Mindestens zwei Aufhängungnen (*attachments*)
- Kompatible Segmente können auf der selben Facette gezeichnet werden.

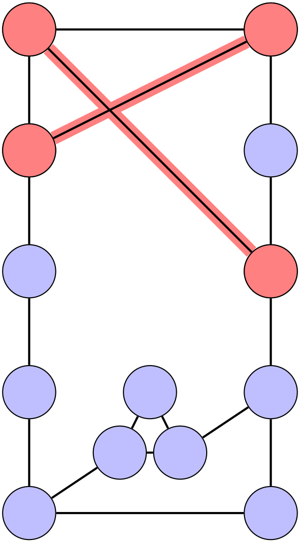
- Sehne (*chord*)
- Teilgraph als Segment



Segmente

- Jede Kante eines Blocks ist Teil von mindestens einem Kreis
- Kreise enthalten Segmente
  - Kanteninduzierte Teilgraphen
  - Mindestens zwei Aufhängungnen (*attachments*)
- Kompatible Segmente können auf der selben Facette gezeichnet werden.

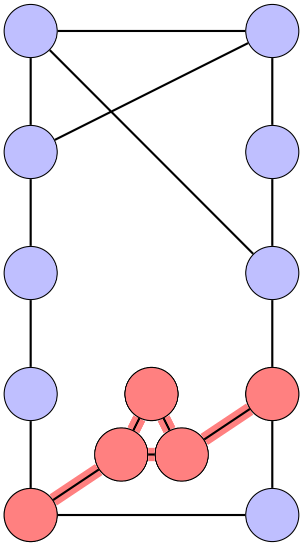
- Sehne (*chord*)
- Teilgraph als Segment



Segmente

- Jede Kante eines Blocks ist Teil von mindestens einem Kreis
- Kreise enthalten Segmente
  - Kanteninduzierte Teilgraphen
  - Mindestens zwei Aufhängungnen (*attachments*)
- Kompatible Segmente können auf der selben Facette gezeichnet werden.

- Sehne (*chord*)
- Teilgraph als Segment



Lemma

Zwei Segmente sind genau dann kompatibel, wenn ihre Aufhängungen nicht verflochten (interleaved) sind.

Theorem

Ein 2-zusammenhängender Graph mit Kreis C ist planar genau dann wenn

- *Der Verflechtungsgraph der Segmente von C bipartit ist*
- *Für alle Segmente P der Graph $P \cup C$ planar ist*

Kompatibilität

Lemma

Zwei Segmente sind genau dann kompatibel, wenn ihre Aufhängungen nicht verflochten (interleaved) sind.

Theorem

Ein 2-zusammenhängender Graph mit Kreis C ist planar genau dann wenn

- *Der Verflechtungsgraph der Segmente von C bipartit ist*
- *Für alle Segmente P der Graph $P \cup C$ planar ist*

Algorithmus

Rekursives Betrachten der Kreise und Teilgraphen mit drei Fällen:

- **Trivialer Fall** Der Graph G ist ein einfacher Kreis C. Tritt nur am Anfang auf.
- **Basisfall** Der Kreis C enthält ein einziges Segment, das ein Pfad ist.
- **Rekursionsfall** Ein Kreis mit mehreren Segmenten oder Subgraphen ist in G enthalten. Anwendung des Theorems und rekursive Abarbeitung der Subgraphen.

- $\mathcal{O}(n)$ Rekursionen mit Verflechtungsgraphen in $\mathcal{O}(n^2)$
- Gesamtkomplexität $\mathcal{O}(n^3)$