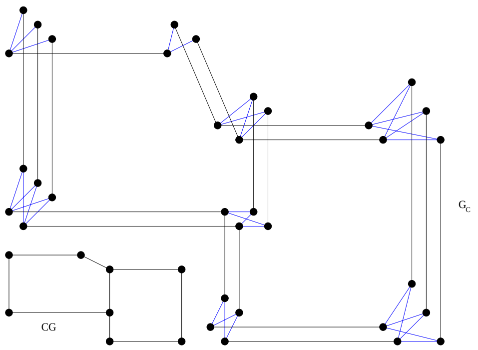
Sechstes Übungsblatt

Ausgabe: 5. Juli 2011
Abgabe: keine, Besprechung in einer der Übungen

1 Via-Minimierung

Betrachten Sie den Beispiel-Konfliktgraphen sowie den zugehörigen Clustergraphen aus Abb. 6.8 des Skriptes:



Aufgabe: Berechnen Sie an Hand einer Zweifärbung des Clustergraphen (z. B. der in Abb. 6.11 angegebenen) eine Zweifärbung des Konfliktgraphen (nach Festlegung der Repräsentanten). Bestimmen Sie die sich daraus ergebende Viareduktionsfunktion v_{red}, und lösen Sie die resultierende Instanz von Max-Via-Reduktion. Welche Realisierung des Layouts ergibt sich dadurch?

Hinweis: Das dabei auftretende Mixed-Max-Cut-Problem kann durch ‚scharfes Hinsehen' gelöst werden.

2 Entfernen von Rechtskreisen

In Schritt 2 des Algorithmus für das kantendisjunkte Menger-Problem werden in einem planaren Graphen G mit fester Einbettung einfache Kreise $C_1, \ldots, C_\ell$ wie folgt konstruiert:

Sei F die Menge der Facetten und f_0 die Äußere Facette von G. Bezeichne weiter $\text{dist}(f)$ die Länge eines kürzesten Weges vom der Facette f entsprechenden Dualknoten zum f_0 entsprechenden Dualknoten, und $\ell := \max_{f \in F} \text{dist}(f)$. Für $1 \leq i \leq \ell$ sei C_i die Vereinigung der einfachen Kreise in G so, dass $\text{dist}(f) \geq i$ für alle Facetten f im Inneren und $\text{dist}(f) < i$ für alle Facetten f im Äußeren eines Kreises aus C_i gilt.

Aufgabe: Geben Sie einen Algorithmus mit linearer Laufzeit an, der zu einem gegebenen Graphen G mit fester Einbettung die Kantenmengen $C_1, \ldots, C_\ell$ bestimmt.

3 Kantendisjunkte Wege, ein Spezialfall

Geben Sie einen einfachen Algorithmus an, der folgendes Problem in Linearzeit löst:

Gegeben ein planarer Graph G mit fester Einbettung und ausgezeichneten Knoten s und t, die an der Äußeren Facette liegen, bestimme eine maximale Anzahl von paarweise kantendisjunkten s-t-Wegen in G.

Hinweis: Es genügt eine Vorgehensweise ähnlich wie bei einer Graphsuche, die die gegebene Einbettung des planaren Graphen ausnutzt.

4 Dualität von Suchen

Sei G ein ungerichteter, zusammenhängender, planar eingebetteter Graph, G^* der zugehörige Dualgraph und $e = (u, v)$ eine orientierte Kante.

Algorithm 1: Right-First-Kanten-DFS

Lege e auf einen Stapel;
while *Stapel nicht leer* **do**
    Betrachte oberste Kante (x, y);
    if y *inzident zu nichtorientierter Kante* **then**
        Orientiere im Gegenuhrzeigersinn bzgl. y nächste nichtorientierte Kante $y \to w$ und lege diese auf den Stapel
    else
        Entferne (x, y) vom Stapel
    end
end

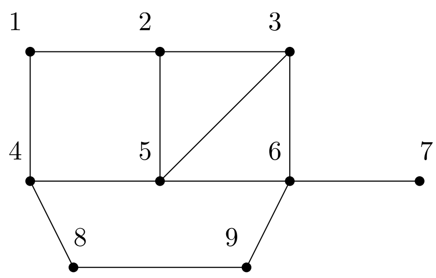
Algorithm 2: Left-First-Kanten-BFS

Orientiere alle zu u inzidenten Kanten $u \to w$ und hänge diese, beginnend bei e, im Uhrzeigersinn bzgl. u an eine Warteschlange;
while *Warteschlange nicht leer* **do**
    Betrachte erste Kante (x, y);
    if y *inzident zu nichtorientierter Kante* **then**
        Orientiere alle solche Kanten $y \to w$ und hänge diese im Uhrzeigersinn bzgl. y an die Warteschlange
    else
        Entferne (x, y) aus der Warteschlange
    end
end

In beiden Kantensuchen wird je eine Reihenfolge $R = (e_1, \ldots, e_m)$ der Kanten festgelegt. Die zu e „rechte" Facette sei f_1, und die Dualkante $e^* = (f_1, f_2)$ zu e sei orientiert ausgehend von f_1.

Wir betrachten eine Right-First-Tiefensuche in G, beginnend bei Kante e, mit Kantenfolge $R = (e = e_1, \ldots, e_m)$ und eine Left-First-Breitensuche in G^*, beginnend bei Kante e^*, mit Kantenfolge $R^* = (e^* = e_1^*, \ldots, e_m^*)$. Die Reihenfolgen R und R^* heißen dual, falls e_i Dualkante zu e_i^* für alle $1 \leq i \leq m$.

(a) Bestimmen Sie für den zweidimensionalen Würfel Q_2 sowie untenstehenden Graphen die Reihenfolgen R und R^* bei beliebiger Startkante e. (**Hinweis:** Die Reihenfolgen sind dual.)



(b) Geben Sie einen Graphen an, für den R und R^* nicht dual sind. (**Hinweis:** Betrachten Sie Graphen mit Brücken und Kreisen.)