RWTH Aachen **WS 2008/09**
Lehrgebiet Theoretische Informatik Übungsblatt 7
Kneis—Langer—Richter—Rossmanith 4.12.2008

♡ Übung zur Vorlesung Parametrisierte Algorithmen ♡

Tutoraufgabe T20

Zeigen Sie, dass das Problem MaxCut mit dem Parameter Baumweite in FPT ist.

Lösungsvorschlag

Wir lösen MaxCut mittels dynamischer Programmierung auf einer schönen Baumzerlegung. Für einen Bag B mit den Knoten $\{v_1, \dots, v_l\}$ generieren wir dabei Tabelleneinträge für alle $S \subseteq \{v_1, \dots, v_l\}$, wobei S als die Menge der Knoten interpretiert wird, die in die erste Seite des Schnittes aufgenommen werden. Für jedes S speichern wir jeweils die Anzahl der Kanten, die im Teilgraphen unter B geschnitten werden.

Sei B immer der aktuelle Bag. Für ein Blatt der Baumzerlegung ist der Wert von S einfach die Anzahl der Kanten zwischen Knoten aus B, die zwischen S und $B \setminus S$ verlaufen.

Für ein Introduce des Knotens v muss für den Tabelleneintrag $S = S' \cup \{v\}$, die Kanten zwischen v und $B \setminus S$ zu dem Wert von S' im vorherigem Bag B' addiert werden. Für alle Einträge S, die v nicht enthalten, müssen alle Kanten zwischen v und S zu dem Eintrag S in B' addiert werden.

Für ein Forget des Knotens v ist der Eintrag S einfach das Maximum der Einträge S und $S \cup \{v\}$ im vorherigen Bag B'.

Für einen Joinbag werden für den Eintrag S die Einträge an den Stellen S in den beiden Kinderbags addiert, die Kanten zwischen S und $B \setminus S$ werden dabei aber doppelt gezählt und müssen wieder subtrahiert werden.

Offensichtlich findet die dynamische Programmierung eine optimale Lösung, ein formaler Beweis kann z.B. leicht per Induktion über die Anzahl der Bags erfolgen. Die Laufzeit ist durch $O(2^w \cdot p(|V|))$ beschränkt, da die Tabellen höchstens 2^w Einträge besitzen und jeder Eintrag in polynomieller Zeit aus den vorherigen Einträgen berechnet werden kann.

Tutoraufgabe T21

Finden Sie MSO-Formeln für:

- die Existenz eines Pfades zwischen s und t
- die Existenz eines Kreises, auf dem s und t liegen
- die Existenz eines Hamiltonkreises.

Lösungsvorschlag

Ein Hinweis vorweg: Man kann beweisen, daß man für das Hamiltonkreisproblem einen Quantor über *Kanten* benötigt, oder in anderen Worten: Die Existenz eines Hamiltonkreises ist nicht in MS_1 ausdrückbar. Wenn Sie eine Lösung ohne Kantenquantor gefunden haben, ist diese also höchstwahrscheinlich falsch. Doch nun zur eigentlichen Lösung.

Die folgende Teilformel ist genau dann erfüllt, wenn es in der Kantenmenge E einen Pfad von s nach t gibt. Die erste Zeile stellt sicher, daß genau eine Kante inzident zu s ist, die zweite Zeile daß genau eine Kante zu t inzident ist. Die dritte und vierte Zeile schließlich sind dann wahr, wenn jeder Knoten ungleich s und t, in genau zwei Kanten oder in gar keiner Kante vorkommt. E enthält also einen Pfad von s nach t und möglicherweise noch weitere disjunkte Kreise.

$$\begin{aligned}
CON(E,s,t) \quad := \quad & \exists e \in E\,(s \in e \wedge \nexists e' \in E\,(e \neq e' \wedge s \in e')) \\
\wedge \quad & \exists e \in E(t \in e \wedge \nexists e' \in E(e \neq e' \wedge t \in e')) \\
\wedge \quad & \forall u(s \neq u \wedge t \neq u \wedge \exists e \in E.u \in E) \rightarrow \\
& \exists e' \in E(e \neq e' \wedge u \in e' \wedge \nexists e'' in E(e'' \neq e \wedge e'' \neq e' \wedge u \in e''))
\end{aligned}$$

Die folgende Formel ist wahr, wenn u in höchstens einer der beiden Kantenmengen auftritt.

$$\begin{aligned}
Dis(E,E',u) \quad := \quad & \exists e(u \in e \wedge e \in E) \rightarrow \nexists e'(u \in e' \wedge e' \in E') \\
& \wedge \exists e'(u \in e' \wedge e' \in E') \rightarrow \nexists e(u \in e \wedge e \in E)
\end{aligned}$$

Mit diesen Hilfsformeln können wir leicht die verlangten Formeln konstruieren.

- $$\begin{aligned}
Path(E,s,t) \quad := \quad & CON(E,s,t) \wedge (\forall u \forall v(\exists e \in E.u \in e \wedge \exists e \in E.v \in e) \\
& \rightarrow \exists E'(CON(E',u,v) \wedge e \in E' \rightarrow e \in E))
\end{aligned}$$

  Diese Formel erzwingt, daß E ein Pfad von s nach t ist, da einerseits s und t in E verbunden sind, E nur einen Pfad von s nach t und weitere knotendisjunkte Kreise enthält und zudem jedes Knotenpaar in E verbunden ist. Die Lösung für diese Teilaufgabe ist also $\exists E Path(E,s,t)$.

- Wir konstruieren eine Formel für einen Kreis mit Hilfe von $Path(E,s,t)$. Dazu reicht es nun aus, zwei disjunkte Pfade zu konstruieren.

  $$\begin{aligned}
Cycle(C,s,t) \quad := \quad & \exists E \exists E' \forall e(e \in E \rightarrow e \notin E') \wedge (e \in E' \rightarrow e \notin E) \\
& \wedge Path(E,s,t) \wedge Path(E',s,t) \\
& \wedge (e \in C \rightarrow e \in E \vee e \in E') \\
& \wedge \forall u(u \neq s \wedge u \neq t) \rightarrow Dis(E,E',u)
\end{aligned}$$

- Die Formel für einen Hamiltonkreis ist somit einfach:

  $$\exists E \forall u \forall v.Cycle(E,s,t)$$

Tutoraufgabe T22
Wie ist die Baumweite aller Graphen beschränkt, die

- kein Dreieck
- kein Viereck
- keine 3-Klaue

als Minor enthalten?

Lösungsvorschlag

- kein Dreieck: Diese Graphen sind Wälder und haben somit Baumweite eins.
- kein Viereck: Diese Graphen enthalten keine Kreise (mit möglichen Querkanten)der Länge vier oder größer. Solche Graphen erhält man, wenn in einem Graphen einzelne Knoten oder Kanten durch Dreiecke ersetzt werden. Eine Polizistenstrategie kommt daher mit 3 Polizisten aus.
- Der Graph enthält nur Pfade und Kreise, die Baumweite ist also höchstens zwei.

Hausaufgabe H13
Finden Sie einen parametrisierten Algorithmus, der für einen Graphen mit Baumweite k die größte Clique findet!

Lösungsvorschlag
Wir wissen, daß jede Clique komplett in einem Bag enthalten sein muss. Es reicht also aus, für jeden Bag für jede Teilmenge seiner Knoten zu testen, ob sie eine Clique ist. Dies ist in $O(2^w \cdot poly(|V|)$ Schritten möglich.

Hausaufgabe H14
Gegeben ist ein Graph $G = (V, E)$. Es gibt einen (Knoten-)Separator $S \subseteq V$ mit der Eigenschaft, daß $V = S \cup A \cup B$ ist, wobei A, B und S disjunkt sind und jeder Pfad von einem Knoten aus A zu einem Knoten in B einen Knoten aus S enthält (S separiert also A und B).
Beweisen Sie: $tw(G) \leq |S| + \max\{tw(G[A]), tw(G[B])\}$
Hierbei ist $tw(G)$ die Baumweite des Graphen G.

Lösungsvorschlag
Sei T_A (T_B) eine Baumzerlegung minimaler Weite für $G[A]$ ($G[B]$). Wir erhalten eine Baumzerlegung T für G folgendermaßen: Wir übernehmen die Bags und Kanten aus T_A und T_B. Anschließend fügen wir jedem Bag von T_A und T_B alle Knoten aus S hinzu. Abschließend wird eine Kante zwischen einem beliebigen Bag von T_A zu einem beliebigen Bag von T_B hinzugefügt. Natürlich enthält jeder Bag von T höchstens $|S| + \max\{tw(G[A]), tw(G[B])\} + 1$ Knoten. Es bleibt zu zeigen, daß T eine gültige Baumzerlegung ist.
Offensichtlich kommt jeder Knoten aus G in einem Bag von T vor.

Da S ein Separator ist, gibt es keine Kante zwischen A und B in G. Jede Kante von S nach A (nach B) kommt in einem Bag vor, da es einen Bag in T_A (T_B) gibt, der den Endpunkt der Kante aus A (B) enthält, und da diesem Bag alle Knoten aus S hinzugefügt wurden. Kanten innerhalb von A, innerhalb von B und innerhalb von S sind offensichtlich in einem der Bags enthalten.

Wenn zwei Bags einen Knoten u enthalten, so auch alle Bags dazwischen, da für Knoten aus A und B keine Änderungen gemacht wurden und S in jedem Bag enthalten ist.