RWTH Aachen **WS 2008/09**
Lehrgebiet Theoretische Informatik Übungsklausur
Kneis—Langer—Richter—Rossmanith 03.02.2009

♡ Übung zur Vorlesung Parametrisierte Algorithmen ♡

Aufgabe 1 (10 Punkte)

Ein *perfect code* eines Graphens $G = (V, E)$ ist eine Menge von Knoten $V' \subseteq V$ mit der Eigenschaft, daß für jeden Knoten $v \in V$ genau ein Knoten in der Menge $N[v] \cap V'$ enthalten ist, wobei $N[v] = N(v) \cup \{v\}$ die geschlossene Nachbarschaft von v bezeichnet.
Das Problem Planar Perfect Code ist folgendermaßen definiert:

Input: Ein ungerichteter, planarer Graph $G = (V, E)$, eine Zahl k
Parameter: k
Question: Gibt es in G einen *perfect code* V' der Größe k?

Zeigen Sie, daß Planar Perfect Code in FPT ist.

Lösungsvorschlag

Wie schon aus der Tutoraufgabe T2 bekannt, gibt es in jedem planaren Graphen einen Knoten mit Grad höchstens fünf. Außerdem wissen wir, daß für jedes v entweder v oder genau einer seiner Nachbarn in V' sein muß. Weiterhin haben zwei Elemente in V' mindestens Abstand drei zueinander. Ein einfacher rekursiver Algorithmus geht auf der Eingabe (G, B, k) so vor (B ist hier eine Menge von verbotenen Knoten, die zu geringen Abstand haben):

1. Wenn $k = 0$, antworte „nein“, falls $V \neq \emptyset$, und „ja“, falls $V = \emptyset$.
2. Wenn $V = \emptyset$ ist, gebe „nein“ zurück.
3. Wähle einen Knoten $v \in V$, der in $G[V \setminus B]$ höchstens Grad fünf hat.
4. Für jedes $u \in N[v] \setminus B$ rufe sich selbst mit Eingabe $(G[V \setminus N[u]], B \cup N(N(u)), k-1)$ auf.
5. Wenn einer dieser Aufrufe „ja“ zurückgab, antworte „ja“, sonst antworte „nein“.

Da $N[v] \setminus B$ jeweils nur bis zu sechs Knoten enthält, sich der Parameter in jedem Schritt verringert, und jeder Schritt in polynomieller Zeit lösbar ist, erhalten wir einen parametrisierten Algorithmus mit Laufzeit 6^kpoly(n).
Wir müssen noch zeigen, daß der Algorithmus korrekt ist. Mit Induktion über k zeigen wir, daß der Algorithmus immer einen *perfect code* V' mit der Eigenschaft $B \cap V' = \emptyset$ findet, wenn ein solcher existiert. Aufgerufen mit $(G, \emptyset, k)$ löst der Algorithmus dann Planar Perfect Code auf der Eingabeinstanz.
Für $k = 0$ ergibt sich die Korrektheit direkt. Seien daher (G, B, k) mit $k > 0$ gegeben und existiere für G ein *perfect code* V' mit $V' \cap B = \emptyset$. Da G planar ist, ist auch jeder induzierte Untergraph von G planar, also insbesondere auch $G[V \setminus B]$, und dieser enthält einen

Knoten v mit höchstens fünf Nachbarn. Also hat v auch in G höchstens fünf Nachbarn, die nicht in B enthalten sind. Da V' ein *perfect code* mit der zusätzlichen Eigenschaft $V' \cap B = \emptyset$ ist, wissen wir, daß wegen $V' \setminus B = V'$ auch $|N[v'] \cap (V' \setminus B)| = 1$ ist. Sei u dieser Knoten. Es ist leicht einzusehen, daß $V' \setminus \{u\}$ ein *perfect code* für $G[V \setminus N[u]]$ der Größe $k-1$ ist, welcher weder die Knoten in B, noch – aufgrund des Mindestabstands – die Knoten in $N(N(u))$ enthält. Nach Induktionsvoraussetzung antwortet daher der entsprechende rekursive Aufruf in Schritt 4 mit „ja“, womit auch dieser Aufruf mit „ja“ antwortet.

Aufgabe 2 (10 Punkte)

Das Problem Buchdurchstechung ist folgendermaßen definiert:

Input:	Ein Alphabet Σ, ein Buch mit Zeichen in Σ, k Nadeln
Parameter:	k
Question:	Können Sie mit k Nadeln das Buch so durchstechen, daß jedes Zeichen aus Σ wenigstens einmal durchstochen wird?

Sie können annehmen, daß jede Buchseite durch eine $n \times m$-Matrix beschrieben werden kann, so daß jede Position (i, j) innerhalb einer Buchseite entweder leer oder mit höchstens einem Zeichen beschriftet ist. Eine Nadel durchsticht alle übereinanderliegenden Seiten immer genau auf derselben Position.

Zeigen Sie, daß Buchdurchstechung W[2]-schwer ist.

Lösungsvorschlag

Wir reduzieren Dominating Set, bekannterweise W[2]-schwer, parametrisiert auf Perfect Code:

Gegeben sei ein ungerichteter Graph $G = (V, E)$ und eine Zahl k, wobei oBdA $V = \{1, \dots, n\}$ sei. Wir konstruieren ein Buch über das Alphabet $\Sigma = V$ mit n Seiten der Größe $n \times 1$ wie folgt: Die Seite i $(1 \leq i \leq n)$ sei an Position $(v, 1)$ $(1 \leq v \leq n)$ genau dann beschriftet mit i, wenn $i \in N[v]$. Der Parameter k bleibt erhalten.

Die Reduktion ist offensichtlich in polynomieller Zeit durchführbar. Zu zeigen, daß G ein *dominating set* der Größe k genau dann hat, wenn man im konstruierten Buch mit k Nadeln jedes Zeichen wenigstens einmal durchstechen kann, ist nun einfach: Sei $D = \{u_1, \dots, u_k\}$ ein *dominating set* für G. Nadeln an den Positionen $(u_1, 1), \dots, (u_k, 1)$ durchstechen jedes Zeichen wenigstens einmal: Wenn $u \in V$ durch $u_i \in D$ dominiert wird, dann wird das entsprechende Zeichen auch durch die Nadel an Position $(u_i, 1)$ der Seite u durchstochen. Da D ein *dominating set* ist, sind alle Knoten v in der Nachbarschaft $N[u_i]$ mindestens eines $u_i \in D$, also wird auch jedes Zeichen wenigstens einmal durchstochen. Analog kann aus einer Lösung für Buchdurchstechung ein *dominating set* konstruiert werden: Seien $(u_1, 1), \dots, (u_k, 1)$ Nadelpositionen, so daß jedes Zeichen mindestens einmal durchstochen wird. Wähle $D := \{u_1, \dots, u_k\}$ und betrachte ein beliebiges $v \in V$. Dieses v wird von wenigstens einer Nadel durchstochen, nehmen wir mal an, die Nadel an Position $(u_i, 1)$ tut dies. Also ist nach Konstruktion des Buches $v \in N[u_i]$ und u wird von $u_i \in D$ dominiert. Es ist also D ein *dominating set* für G.

Aufgabe 3 (10 Punkte)

Zeigen Sie, daß Buchdurchstechung in FPT ist, wenn die Größe des Alphabets Σ ein zusätzlicher Parameter ist, indem Sie einen geeigneten Problemkern angeben.

Lösungsvorschlag

Sei $l := |\Sigma|$. Bekannterweise gibt es überhaupt nur 2^l verschiedene Teilmengen von Σ, und damit auch nur 2^l relevante Nadelpositionen im Buch, da es natürlich keinen Sinn macht, zwei Nadeln so zu platzieren, daß sie die gleiche Menge Zeichen durchstechen.

Diese höchstens 2^l verschiedenen Teilmengen kann man nun in einem Buch mit l Seiten mit höchstens 2^l Einträgen ausreichend abbilden, was bei unveränderten Parametern direkt einen Problemkern ergibt:

Sei die Instanz wie oben gegeben. Für eine Position (i, j) im Buch bezeichne $U_{i,j}$ die Menge aller Zeichen, die an einer Nadel an dieser Position durchstochen wird. Sei

$$\mathcal{U} := \{\, U_{i,j} \mid 1 \leq i \leq n, 1 \leq j \leq m \,\}$$

und dabei oBdA $\mathcal{U} = \{U_1, \ldots, U_s\}$. Die Problemkerninstanz besteht aus dem Alphabet Σ, den unveränderten Parametern k und l sowie aus einem Buch mit l Seiten der Größe $s \times 1$. Die Seite j $(1 \leq j \leq l)$ ist dabei an Position $(i, 1)$ genau dann mit j beschriftet, wenn $j \in U_i$ ist.

Das neue Buch hat damit nun die Größe $l \cdot s \leq l \cdot 2^l$, womit wir schon einmal die Größenanforderung der Problemkerninstanz erreicht haben. Da man außerdem für jede Nadelposition sehr leicht eine Entsprechung in der jeweils anderen Instanz finden kann (nämlich die, die jeweils der gleichen Menge Zeichen entspricht), ist auch die Korrektheit der Problemkernreduktion direkt ersichtlich.