

Lösungsskizze für das 11. Übungsblatt

Marco Gaertler

31. Januar 2005

7. Aufgabe

Wir definieren ein allgemeines Greedy-Verfahren \mathcal{A} , um unabhängige Mengen zu finden.

Algorithm 1: Greedy Verfahren \mathcal{A}

Input: ein Graph $G = (V, E)$ und
eine Knotenreihenfolge $\text{ord}: \{1, \dots, n := |V|\} \rightarrow V$

Output: eine unabhängige Menge IS von G

Data: eine Knotenmarkierung $N : V \rightarrow \{0, 1\}$

initialisiere IS mit \emptyset

initialisiere N mit $N(v) \leftarrow 0$ für $v \in V$

for $i \leftarrow 1, \dots, n$ **do**

```
1   |    $v \leftarrow \text{ord}(i)$ 
    |   if  $N(v) = 0$  then
2   |       |    $\text{IS} \leftarrow \text{IS} \cup \{v\}$ 
3   |       |    $N(v) \leftarrow 1$ 
    |       |   foreach  $w \in \{u \in V \mid \{u, v\} \in E\}$  do
4   |       |       |    $N(w) \leftarrow 1$ 
```

Bedeutung der Knotenmarkierung N : Ein Knoten v ist ein Kandidat für IS genau dann, wenn $N(v) = 0$ ist.

Lemma 1 *Es gilt folgende Invariante: zu jedem Zeitpunkt enthält die Menge IS eine unabhängige Menge.*

Beweis: Wir nehmen an, dass die Invariante zu einem Zeitpunkt verletzt ist. O.B.d.A. sei der Knoten v der erste Knoten, der zu IS hinzugefügt wird, der die Eigenschaft verletzt. Dann war IS eine unabhängige Menge, und $IS \cup \{v\}$ ist es nicht mehr (Schritt 2). Das bedeutet, dass es einen Knoten $u \in IS$ gibt, so dass $\{u, v\} \in E$. Da aber $u \in IS$ ist, müssen u und seine Nachbarschaft markiert worden sein (Schritt 3 bzw. 4). Da u vor v betrachtet wurde, gilt nach der Abarbeitung von u , dass $N(v) = 1$. Das ist ein Widerspruch zu der Tatsache, dass v zu IS hinzugefügt werden kann, da im Schritt 1 überprüft wird, ob $N(v) = 0$, und nur dann ein Vereinigungsschritt ausgeführt wird. \square

Lemma 2 *Die Schleife im Schritt 1 benötigt $\mathcal{O}(\deg(v))$ Zeit, also hat der Algorithmus \mathcal{A} eine Laufzeit von $\mathcal{O}(|E|)$.*

Lemma 3 *Es gilt folgende Invariante: zu jedem Zeitpunkt i haben alle Knoten v mit $\text{ord}^{-1}(v) < i$ eine Eins in der Knotenmarkierung N . Insbesondere ist am Ende des Algorithmus $N \equiv 1$.*

Beweis: Wir nehmen an, dass die Invariante zum Zeitpunkt i verletzt ist. D.h. es gibt einen Knoten v mit $\text{ord}^{-1}(v) < i$, so dass $N(v) = 0$. Da v vor $\text{ord}(i)$ bzgl. ord betrachtet wird, müsste für den Knoten v schon die `if`-Abfrage (Schritt 1) stattgefunden haben. Da während des ganzen Algorithmus nie die Markierung eines Knotens wieder auf 0 gesetzt wird, muß auch während der `if`-Abfrage $N(v) = 0$ gewesen sein. Das impliziert aber, dass der zugehörige `then`-Teil ausgeführt wurde. Dann wird in Schritt 3 die Markierung $N(v)$ auf 1 gesetzt. Das ergibt den Widerspruch zu $N(v) = 0$ zum Zeitpunkt i .

Beim letzten Knoten w gilt, dass entweder $N(w) = 0$ oder $N(w) = 1$. Im Fall von $N(w) = 0$, wird $N(w)$ auf 1 gesetzt während Schritt 3. D.h. nach der Abarbeitung von w ist $N \equiv 1$. \square

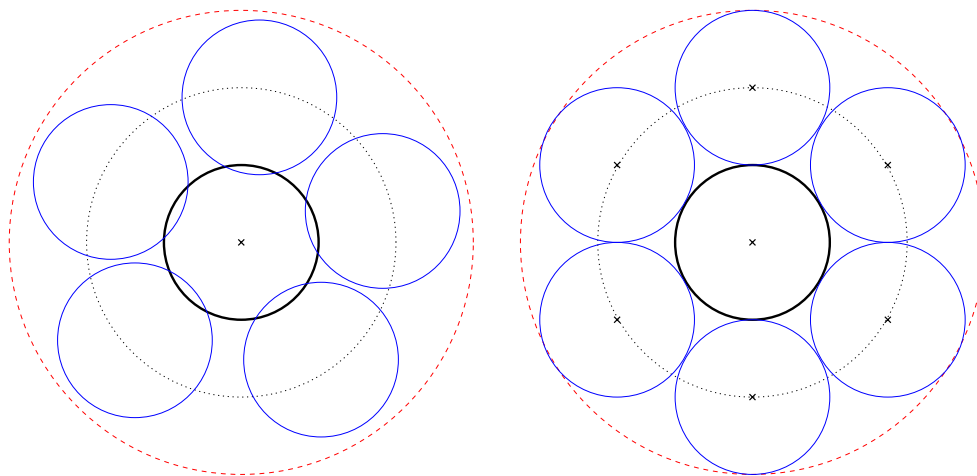
Satz 4 *Der Algorithmus \mathcal{A} berechnet in $\mathcal{O}(|E|)$ eine maximale unabhängige Menge von G .*

(a) Der Algorithmus \mathcal{A} mit einer beliebigen Knotenreihenfolge genügt.

Der Algorithmus \mathcal{A} berechnet nach Satz 4 eine maximale unabhängige Menge in polynomialer Zeit. Es genügt also zu zeigen, dass er als Approximationsalgorithmus eine (relative) Gütegarantie von 5 hat.

Lemma 5 *Die Nachbarschaft eines Knotens enthält eine unabhängige Menge der Größe höchstens 5.*

Beweis: Betrachte dazu Bild 1.



(a) Nachbarschaft mit einer unabhängigen Menge der Größe 5

(b)

Abbildung 1: Unabhängige Menge in der Nachbarschaft eines Knotens

Bild 1(a) zeigt eine Instanz eines Einheitskreisscheibengraphen, der einen Knoten hat (fetter, schwarzer Kreis in der Mitte), dessen Nachbarschaft eine unabhängige Menge der Größe 5 (blaue Kreise) hat. Bild 1(b) zeigt den Versuch, sechs schnittfreie Kreise zu platzieren. Leider schneiden die blauen Kreise den fetten, schwarzen Kreis nicht mehr, sondern berühren ihn nur. D.h. die blauen Kreise sind nicht in der Nachbarschaft des fetten, schwarzen Kreises enthalten. \square

Satz 6 *Es gilt $\mathcal{R}_{\mathcal{A}} = 5$.*

Beweis: Wir betrachten den Schnitt von $IS_{\mathcal{A}}$, welches von \mathcal{A} berechnet wurde, und IS_{opt} , welches eine kardinalitätsmaximale unabhängige Men-

ge ist. Wir müssen nun die Größe von $IS_{\text{opt}} \setminus (IS_{\text{opt}} \cap IS_{\mathcal{A}})$ abschätzen. Auf Grund der Maximalität von $IS_{\mathcal{A}}$ gilt:

$$\forall v \in IS_{\text{opt}} \setminus (IS_{\text{opt}} \cap IS_{\mathcal{A}}) : \exists w \in IS_{\mathcal{A}} \setminus (IS_{\text{opt}} \cap IS_{\mathcal{A}}) : \{v, w\} \in E .$$

Aus Lemma 5 folgt die Ungleichung

$$|IS_{\text{opt}} \setminus (IS_{\text{opt}} \cap IS_{\mathcal{A}})| \leq 5 \cdot |IS_{\mathcal{A}} \setminus (IS_{\text{opt}} \cap IS_{\mathcal{A}})| ,$$

welche $|IS_{\text{opt}}| \leq 5 \cdot |IS_{\mathcal{A}}|$ impliziert. □

- (b) Sortiere die Knoten so, dass die x- und y-Koordinaten nicht absteigend sind, wobei die x-Koordinate die höhere Priorität habe.

Der Algorithmus \mathcal{A} mit dieser Knotenreihenfolge genügt.

Es genügt zu zeigen, dass \mathcal{A} eine bessere relative Gütegarantie als 5 hat.

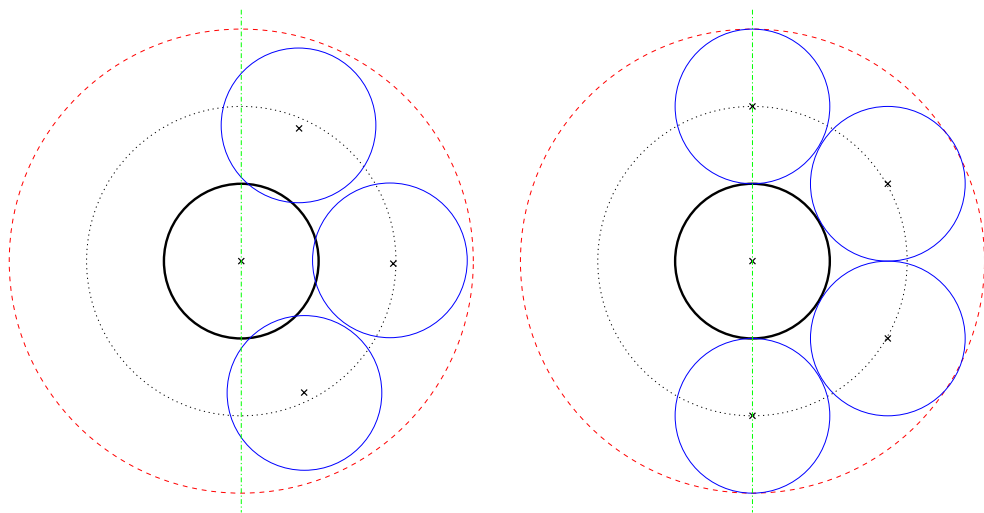
Satz 7 *Es gilt, dass \mathcal{A} mit obiger Knotenreihenfolge eine Gütegarantie von $\mathcal{R}_{\mathcal{A}} = 3$ hat. Insgesamt wird $\mathcal{O}(\max\{|E|, |V| \log |V|\})$ Zeit benötigt.*

Beweis: Man argumentiert ähnlich wie im Beweis zu Satz 6, allerdings ist es recht einfach zu sehen, dass Lemma 5 verbessert werden kann: Jeder Knoten hat in seiner noch nicht abgearbeiteten Nachbarschaft eine unabhängige Menge der Größe höchstens 3. Auch der Beweis läuft analog zu dem von Lemma 5, man betrachte dazu die Abbildung 2. □

- (c) Ja, die Idee von Teilaufgabe (b) kann auch ohne Koordinaten durchgeführt werden. Die Kernaussage ist dabei, dass es stets einen Knoten gibt, dessen noch abzuarbeitende Nachbarschaft eine unabhängige Menge der Größe höchstens drei enthält. Das Verfahren kann wie folgt umgewandelt werden:

Lemma 8 *Das Verfahren \mathcal{A}' erfüllt folgende Invarianten:*

- *es vergehen höchstens $(n - 1)$ Durchlauf der **while**-Schleife, bei der nicht der **else**-Teil der **if**-Abfrage ausgeführt wird.*
- *die **if**-Bedingung kann in $\mathcal{O}(\Delta^4)$, wobei $\Delta = \max_{v \in V} \deg(v)$ ist, getestet werden*
- *der **else**-Teil kann in $\mathcal{O}(|V|)$ abgearbeitet werden*



(a) nicht betrachtete Nachbarschaft
mit einer unabhängigen Menge der
Größe 3

(b)

Abbildung 2: Unabhängige Menge in der noch nicht bearbeiteten Nachbar-
schaft eines Knotens

Algorithm 2: Erweitertes Greedy Verfahren \mathcal{A}'

Input: ein Graph $G = (V, E)$

Output: eine unabhängige Menge IS von G

Data: eine Knotenmarkierung $N : V \rightarrow \{0, 1\}$, eine Knotenmarkierung $C : V \rightarrow \{0, 1\}$

initialisiere IS mit \emptyset

initialisiere N mit $N(v) \leftarrow 0$ für $v \in V$

initialisiere C mit $C(v) \leftarrow 0$ für $v \in V$

while $\exists v \in V : N(v) = C(v) = 0$ **do**

if $N' := \{w \in V \mid \{w, v\} \in E \wedge N(w) = 0\}$ eine unabhängige Menge der Größe 4 enthält **then**

$C(v) \leftarrow 1$

else

 IS \leftarrow IS $\cup \{v\}$

$N(v) \leftarrow 1$

for $w \in \{u \in V \mid \{u, v\} \in E\}$ **do**

$N(w) \leftarrow 1$

for $w \in V$ **do**

$C(w) \leftarrow 0$

- nach $\mathcal{O}(n \cdot \Delta^4)$ verringert sich die Anzahl der Knoten v mit $N(v) = 0$ um eins.
- die Bedingung in der **while**-Schleife kann in $\mathcal{O}(n)$ getestet werden.
- zu jedem Zeitpunkt gilt, dass alle komplett abgearbeiteten Knoten und deren Nachbarschaft eine Eins in N haben
- zu jedem Zeitpunkt ist IS eine unabhängige Menge, die maximal ist bzgl. aller Knoten w mit $N(w) = 1$
- am Ende des Algorithmus gilt $N \equiv 1$

Satz 9 Es gilt $\mathcal{R}_{\mathcal{A}'} = 3$ und \mathcal{A}' hat eine Laufzeit von $\mathcal{O}(n^6)$.