

Übungszettel 1

Aufgabe 1.1

4+2 Punkte

Wir erinnern uns, dass ein Matching $M \subseteq E$ eines Graphen $G = (V, E)$ eine Teilmenge der Kanten ist, sodass es keinen Knoten gibt, der zu mehr als einer Kante aus M inzident ist. Im Problem MAXIMUM MATCHING ist das Ziel, gegeben einen Graphen $G = (V, E)$ ein Matching M zu bestimmen, sodass $|M|$ maximal ist.

- (a) Zeigen Sie, dass die Ausgabe eines beliebigen inklusionsmaximalen Matchings eine 2-Approximation für MAXIMUM MATCHING liefert.
- (b) Zeigen Sie, dass der Algorithmus im Allgemeinen keine bessere als eine 2-Approximation liefert.

Aufgabe 1.2

2+3+5 Punkte

Sei $G = (V, E)$ ein ungerichteter Graph. Eine Menge $I \subseteq V$ ist ein Independent Set, falls für alle $v, w \in I$ gilt, dass $\{v, w\} \notin E$. Wir betrachten den folgenden Greedy-Algorithmus.

Algorithm 1 MINDEG-GREEDY INDEPENDENT SET

```
 $I := \emptyset; G' := G;$   
while  $V(G') \neq \emptyset$  do  
  Wähle den Knoten  $v \in V(G')$  mit minimalem Grad in  $G'$ .  
   $I = I \cup \{v\};$   
  Lösche  $v$  und alle benachbarten Knoten aus  $G'$ .  
end while
```

Sei Δ der maximale Grad eines Knotens von G . Wir dürfen annehmen, dass $\Delta > 0$, denn andernfalls ist $I = V$ trivialerweise.

- (a) Zeigen Sie, dass I am Ende ein Independent Set ist.
- (b) Sei I^* ein optimales Independent Set, und sei k_i die Anzahl der Knoten aus I^* , die in Iteration i gelöscht werden. Zeigen Sie, dass $k_i \leq \Delta$ gilt, und folgern Sie, dass der Algorithmus Δ -approximativ ist.
- (c) Wir nehmen an, dass $\Delta > 2$. Zeigen Sie, dass der Algorithmus sogar $\left(\frac{\Delta}{2} + 1\right)$ -approximativ ist.

Hinweis: Sei d_i der Grad des in Iteration i gewählten Knotens. Zählen Sie die Kanten zwischen I^ und $V \setminus I^*$ auf zwei Arten und nutzen Sie die Abschätzung aus (b).*

Aufgabe 1.3

5 Punkte

Wir betrachten erneut das in der Vorlesung behandelte SET COVER Problem, nehmen aber nun zusätzlich an, dass jedes Element der Grundmenge $x \in S$ in höchstens $f \geq 1$ vielen Teilmengen $S_1, \dots, S_m \subseteq S$ vorkommt. Entwerfen Sie einen f -Approximationsalgorithmus für diesen Fall. Beweisen Sie die Korrektheit und den Approximationsfaktor des Algorithmus.

Hinweis: Ziehen Sie wiederholt für ein noch nicht abgedecktes Element e von den Kosten aller Mengen $S_i \ni e$ denselben Betrag ab (welchen?) und nehmen Sie jede Menge ins Cover auf, sobald ihre Restkosten Null sind. Die Summe der Abzüge ist eine untere Schranke für OPT; nutzen Sie dann die beschränkte Häufigkeit der Elemente.