

Approximative Algorithmen

Aufgabenblatt 1

Die Aufgaben werden besprochen in der zweiten Semesterwoche.

Aufgabe 1

DOMINATING SET bezeichnet (meistens) das Problem, in einem gegebenen ungerichteten Graphen eine kleinstmögliche Menge von Knoten zu finden, sodass deren abgeschlossene Nachbarschaft alle Knoten enthält. Geben Sie eine saubere Formulierung von DOMINATING SET als Optimierungsproblem an.

Aufgabe 2

Eine r -Färbung für einen Graphen $G = (V, E)$ ist eine Abbildung $f: V \rightarrow \{1, \dots, r\}$, sodass $f(u) \neq f(v)$ für alle $(u, v) \in E$.

Betrachten Sie folgenden Approximationsalgorithmus für INDEPENDENT-SET, angewendet auf einen Graphen mit vorgegebener r -Färbung f :

```
A(V,E,f){
  max := 1
  max_c := 0
  forall(i=1:r){
    V_i := { x : f(x)=i }
    k_i := |V_i|
    if(k_i > max_c){
      max := i
      max_c := k_i
    }
  }
  return V_max
}
```

1. Bestimmen Sie die Leistungsgüte von Algorithmus A durch eine geeignete (einfache) Abschätzung.
2. Geben Sie ein nicht-triviales worst-case Beispiel für $r = 3$.
3. Wie könnte man den Algorithmus (jedenfalls in der Praxis) verbessern, wenn man wüsste, dass der vorgelegte Graph in mehrere Zusammenhangskomponenten zerfällt?
4. Was bedeuteten Ihre Überlegungen aus der vorigen Teilaufgabe für den Sonderfall $r = 2$? Was wissen Sie darüber hinaus über 2-färbbare Graphen?

Aufgabe 3

EDGE COVER, das Kantenüberdeckungsproblem, erhält als Eingabe einen Graphen und gefragt ist eine kleinstmögliche Menge von Kanten, sodass jeder Knoten des Graphen mit mindestens einer der gewählten Kanten inzidiert.

Geben Sie eine saubere Formulierung von EDGE COVER als Optimierungsproblem an.

Zeigen Sie, dass EDGE COVER in Polynomzeit gelöst werden kann.

Sie dürfen dabei voraussetzen, dass größtmögliche Matchings in Polynomzeit bestimmt werden können.

Aufgabe 4

Der in der Vorlesung vorgestellte Approximationsalgorithmus für MAXSAT ist geradezu trivial. Überlegen Sie:

1. Inwieweit hängt das Ergebnis (2-Approximation) ab von der Wahl der Belegung (hier: alle Variablen auf Null)?
2. Inwiefern könnte man das ausnutzen, um einen randomisierten Algorithmus zu entwickeln, der zumindest in der Praxis besseres Verhalten zeigen dürfte als der in der Vorlesung vorgestellte?