

Approximationsalgorithmen

Jan Johannsen

Vorlesung im Sommersemester 2007

Einordnung

Algorithmik

- ▶ Entwurf und
- ▶ Analyse von **Algorithmen**

Komplexitätstheorie

- ▶ Analyse der Komplexität von **Problemen**
- ▶ Einteilung in **Klassen** ähnlicher Komplexität
- ▶ Untersuchung der Struktur der Klassen

Analyse des Verbrauchs an Ressourcen

- ▶ Zeit
- ▶ Speicher
- ▶ ...

Abhängig von der Größe des Input

- ▶ Komplexität im *worst case*

Asymptotische Analyse – vernachlässigt

- ▶ konstante Faktoren
- ▶ endliche Anfangsstücke

Komplexität von Problemen

Obere Schranken

- ▶ durch Angabe eines Algorithmus
- ▶ und dessen Analyse

Untere Schranken

- ▶ durch Analyse des Problems
- ▶ Beweis, dass jeder effizientere Algorithmus versagt

Beispiel: **Sortieren**

- ▶ HeapSort, MergeSort brauchen $O(n \log n)$ Vergleiche
- ▶ untere Schranke: $\Omega(n \log n)$ Vergleiche
- ▶ \rightsquigarrow Komplexität: $\Theta(n \log n)$

Problem: Erreichbarkeit

Instanz: Graph G , Ecken s, t .

Lösung: Weg von s nach t .

Entscheidung: Gibt es einen Weg von s nach t ?

Suche: Finde einen Weg von s nach t .

Optimierung: Finde einen *kürzesten* Weg von s nach t .

Zählen: Bestimme die *Anzahl* der Wege von s nach t .

Optimierungsprobleme

Eigenschaften:

- ▶ mehrere potentielle Lösungen zu jedem Input
- ▶ mit Kosten- / Nutzenmaß versehen
- ▶ gesucht: Lösung mit minimalen Kosten / maximalem Nutzen

Beispiele:

- ▶ kürzeste Wege
- ▶ Travelling Salesperson
- ▶ Scheduling
- ▶ ...

Approximation

Viele Optimierungsprobleme sind **NP**-schwer

- ▶ effizienter Algorithmus nur, wenn $P = NP$
- ▶ nur exponentielle Algorithmen zur **optimalen** Lösung

Deshalb: **Approximationsalgorithmen**

- ▶ effizient
- ▶ liefern eine nicht notwendig optimale Lösung
- ▶ Abweichung vom Optimum a priori abzuschätzen

Approximierbarkeit

NP-schwere Optimierungsprobleme unterscheiden sich bzgl. ihrer **Approximierbarkeit**:

- ▶ es gibt beliebig gute Approximationsalgorithmen
- ▶ es gibt Approximationsalgorithmen einer gewissen Güte, aber keine besseren.
- ▶ es gibt überhaupt keine guten Approximationsalgorithmen

Ziele:

- ▶ Methoden zum Entwurf
- ▶ Grundlagen zur Untersuchung dieser Unterschiede