Das Rundreiseproblem Approximationsalgorithmen für ΔTSP Stabilität von Approximationsalgorithmen Fazit

Das Rundreiseproblem und Stabilität von Approximationsalgorithmen

Florian Forster

Friedrich Alexander Universität Erlangen-Nürnberg

Seminar "Perlen der theoretischen Informatik", 2008-01-19 http://verplant.org/uni/perlen/

Gliederung

Das Rundreiseproblem
Das Rundreiseproblem
Das ΔTSP

Approximationsalgorithmen für ΔTSP Der 2APPR-Algorithmus Der Christofides-Algorithmus

Stabilität von Approximationsalgorithmen
Definition
Abstandsfunktionen für ΔTSP
Stabilität von 2APPR und Christofides

Motivation

- ► Für viele "spezielle" Probleme existieren gute Approximationsalgorithmen
- ► Ist es möglich diese Algorithmen auch für "normale" Instanzen zu nutzen?
- ► Wie gut sind diese Lösungen?

Definition

Das Rundreiseproblem wird auch "Traveling Sales**person** Problem" (**TSP**) genannt.

- ▶ Sei G = (V, E) ein vollständiger, ungerichteter Graph mit Kantengewichten
- Gesucht ist die kürzeste (billigste) Rundreise, so dass jeder Knoten genau einmal besucht wird (Hamiltonkreis)

NP-Vollständigkeit

- ► TSP ist **stark** NP-vollständig
- ▶ Approximationsalgorithmen mit relativer Güte kann es nur geben, wenn P = NP
- ▶ Es gibt exakte Algorithmen mit Laufzeit $O(n^2 \cdot 2^n) \ll O(n!)$

Varianten

- Asymmetischrs TSP: Hin- und Rückweg haben nicht notwendigerweise die gleichen Kosten
- ► TSP mit Dreiecksungleichung (Δ **TSP**): $|a \rightarrow c| \leq |a \rightarrow b \rightarrow c|$
- euklidisches TSP: Knoten haben euklidische Abstände (beinhaltet Dreiecksungleichung)
- ▶ ..

- ► Für alle Knoten gilt die Dreiecksungleichung ⇒ "Umwege sind teurer als direkt zu gehen"
- ► Problem immernoch NP-schwer
- Approximationsalgorithmen mit relativer Güte existieren:
 2APPR mit Güte 2,
 Christofides [2] mit Güte ³/₂
- ▶ Diese Algorithmen werden im Folgenden kurz vorgestellt

Approximationsalgorithmen

- ► Approximationsalgorithmen für Optimierungsprobleme berechnen "möglichst gute" Lösungen
- ► Die Abweichung vom Optimum ist nicht beliebig groß
- ► Laufzeit ist aber (verhältnismäßig) gering (also: polynomiell)

Approximationsgüte

- ► Das Verhältnis von schlechtester approximierter zur optimalen Lösung heißt **Approximationsgüte**
- Zum Beispiel:
 - ► Optimale Lösung/Rundreise: 100
 - ► Approximierte Lösung/Rundreise: 150
 - ightharpoonup \Rightarrow Güte mindestens 1,5

- ► Sei *H* ein minimaler Spannbaum von *G*
- ▶ Berechne eine Eulertour auf $H \uplus H$; überspringe doppelte Knoten
- ► Approximationsgüte 2 folgt aus der Minimalität des Spannbaums und der Dreiecksungleichung

Beweis für die Güte von 2APPR

2APPR besitzt eine relative Güte von 2:

► Die Kosten des minimalen Spannbaum sind kleiner als die kosten der optimalen Rundreise R_{OPT}:

$$cost(H) \le \left(1 - \frac{1}{|V|}\right) \cdot cost(R_{OPT})$$

Durch das überspringen von Kanten wird die Strecke nicht länger (Dreickesungleichung):

$$cost(A(x)) \le 2 \cdot cost(H) \le 2 \cdot cost(R_{OPT})$$

- ► Sei *H* ein minimaler Spannbaum von *G*
- ightharpoonup Sei V_u die Menge aller Knoten in H mit ungeradem Grad
- \blacktriangleright Sei M ein leichtestes Matching aller Knoten in V_u
- ▶ Berechne eine Eulertour auf $H \cup M$; überspringe doppelte Knoten
- ► Approximationsgüte $\frac{3}{2}$ folgt aus der Minimalität des Spannbaums und des leichtesten Matchings sowie der Dreiecksungleichung

Beweis für die Güte von Christofides

Christofides besitzt eine relative Güte von $\frac{3}{2}$:

- ▶ Die Kosten des minimalen Spannbaum sind kleiner als die kosten der optimalen Rundreise R_{OPT}
- ▶ Die Kosten des leichtesten Matchings sind kleiner als $\frac{1}{2} \cdot R_{\text{OPT}}$
- Durch das überspringen von Kanten wird die Strecke nicht länger (Dreickesungleichung):

$$cost(A(x)) \le cost(H) + cost(M) \le \frac{3}{2} \cdot cost(R_{OPT})$$

Approximationsalgorithmen für $\Delta\mathsf{TSP}$

Wichtige Bemerkung

Die Algorithmen selbst haben die Dreiecksungleichung nicht verwendet. Lediglich die Abschätzung der Approximationsgüte verwendet die Δ-Ungleichung!

Stabilität von Approximationsalgorithmen

Definition 1/2

- ▶ Definition nach [1]
- ▶ Sei L_{ϕ} eine Spezialisierung des (Optimierungs)Problems L
- ▶ Sei $A_{\phi}(x)$ ein Approximationsalgorithmus für L_{ϕ} mit relativer Güte δ_{ϕ}
- ▶ Sei $h_{\phi}(x)$ eine **Abstandsfunktion** für die gilt:
 - $h_{\phi}(x) = 0$ für $x \in L_{\phi}$
 - $h_{\phi}(x)$ effizient berechenbar
- ▶ Sei $L_{\phi,h,r}$ die Menge aller Probleminstanzen, deren Abstand (bezüglich h_{ϕ}) kleiner oder gleich r ist:

$$L_{\phi,h,r} = \{x \in L : h_{\phi}(x) \leq r\}$$

Stabilität von Approximationsalgorithmen Definition 2/2

- ▶ $A_{\phi}(x)$ heißt p-stabil bezüglich h_{ϕ} , wenn für jedes $r, 0 \not \leq r \leq p$ ein $\delta_{\phi,r} \in \mathbb{R}^{>1}$ existiert, so dass $A_{\phi}(x)$ ein Approximationsalgorithmus für $L_{\phi,h,r}$ mir relativer Güte $\delta_{\phi,r}$ ist.
- ▶ $A_{\phi}(x)$ heißt **stabil bezüglich** h_{ϕ} , wenn $A_{\phi}(x)$ für **alle** $p \in \mathbb{R}^+$ p-stabil bezüglich h_{ϕ} ist.
- ▶ $A_{\phi}(x)$ heißt instabil bezüglich h_{ϕ} , wenn $A_{\phi}(x)$ für kein $p \in \mathbb{R}^+$ p-stabil bezüglich h_{ϕ} ist.
- ▶ $A_{\phi}(x)$ heißt $(r, f_r(n))$ -quasistabil bezüglich h_{ϕ} , wenn $A_{\phi}(x)$ ein Approximationsalgorithmus für $L_{\phi,h,r}$ mir relativer Güte $f_r(n)$ ist.

Abstandsfunktionen für ΔTSP

Definition

► Funktion Disting =

$$\max \left\{0, \max \left\{\frac{\mathrm{cost}(u,v)}{\mathrm{cost}(u,w) + \mathrm{cost}(w,v)} - 1, u, v, w \in V\right\}\right\}$$

▶ Sei $W_{u,v} = \{w_1 = u, w_2, \dots, w_m = v\}$ ein einfacher Pfad von u nach v.

Funktion $Dist_{path} =$

$$\max\left\{0, \max\left\{\frac{\mathrm{cost}(u,v)}{\mathrm{cost}(W_{u,v})} - 1, u, v \in V\right\}\right\}$$

Stabilität von 2APPR und Christofides

Stabilität bezüglich $\mathrm{Dist}_{\mathrm{path}}$

- ► 2APPR und Christofides sind **stabil bezüglich** DIST_{path}
- ▶ Die Kosten des berechneten Hamiltonkreises $v_0 \rightarrow v_1 \rightarrow \ldots \rightarrow v_n \rightarrow v_0$ sind maximal (1+r) mal die Kosten von $v_0 \rightarrow_p v_1 \rightarrow_p \ldots \rightarrow_p v_n \rightarrow_p v_0$ wobei $u \rightarrow_p v$ der kurzeste **Pfad** von u nach v in G ist.
- ▶ 2APPR ist ein (2(1+r))-Approximationsalgorithmus für TSP
- ► Christofides ist ein $(\frac{3}{2}(1+r))$ -Approximationsalgorithmus für TSP

Stabilität von 2APPR und Christofides

Stabilität bezüglich $\mathrm{DiST}_{\mathrm{three}}$

► Jeder Pfad *p* mit *m* Kanten in *G* kann durch eine einzige Kante *e* mit

$$\cot(e) \leq (1+r)^{\lceil \log_2 m \rceil} \cdot \cot(p)$$

ersetzt werden.

- ▶ 2APPR ist $\left(r, 2\left(1+r\right)^{\lceil \log_2 n \rceil}\right)$ -quasistabil bezüglich DIST_{three}
- ► Christofides ist $\left(r, \frac{3}{2} \left(1+r\right)^{\lceil \log_2 n \rceil}\right)$ -quasistabil bezüglich $\text{DIST}_{\text{three}}$
- ▶ 2APPR und Christofides sind instabil bezüglich Dist_{three}

Der PMCA-Algorithmus

Ein stabiler Algorithmus bezüglich $\mathrm{DiST}_{\mathrm{three}}$

- ▶ In [1] wird der PMCA-Algorithmus vorgestellt
- ► Der Algorithmus ist stabil bezüglich Distitute
- ► Erreicht wird das durch ein "Pfad-Matching" und trickreiche "Abkürzungen"
- ▶ Die Approximationsgüte ist $\frac{3}{2}(1+r)^2$, die Laufzeit $O(n^3)$

Fazit 1/2

Das Rundreiseproblem

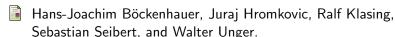
- ▶ Das allgemeine TSP kann man nicht gut approximieren
- ► Für das ∆TSP gibt es gute Approximationsalgorithmen
- ▶ Die Algorithmen verwenden die Dreiecksungleichung nicht, lediglich die Abschätzung braucht sie

Fazit 2/2

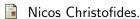
Stabilität von Approximationsalgorithmen

- \blacktriangleright 2APPR und Christofides sind stabil bezüglich Dist_{path} aber instabil bezüglich Dist_{three}
- ▶ 2APPR und Christofides sind quasistabil bezüglich DIST_{three}
- Mit PMCA existiert ein Approximationsalgorithmus für ΔTSP der stabil bezüglich Distitute ist

Literatur



Towards the notion of stability of approximation for hard optimization tasks and the traveling salesman problem. In CIAC '00: Proceedings of the 4th Italian Conference on Algorithms and Complexity, pages 72–86, London, UK, 2000. Springer-Verlag.



Worst-case analysis of a new heuristic for the travelling salesman problem.

In Algorithms and Complexity: New Directions and Recent Results, page 441. Academic Press, 1976.