FPTAS für das Restricted Shortest Path-Problem

Rasmus Diederichsen   Sebastian Höffner

Universität Osnabrück

4. Dezember 2015

Inhalt

Problemstellung

Gegeben

- azyklischer Graph $G = (V, E)$
- $(u, v) \in E$ hat gewicht c und Verzögerung t

Single Source Shortest Path

Berechne vom Startknoten aus alle nach Kosten kürzesten Wege zu allen anderen ▶ Dijkstra

All Pairs Shortest Path

Kürzeste Wege zwischen allen Knotenpaaren ▶ Floyd

Das Problem

Gegeben

- azyklischer Graph $G = (V, E)$
- $(u, v) \in E$ hat gewicht c und Verzögerung t

Restricted Shortest Path

Finde nach Kosten kürzesten Weg von a nach b mit Verzögerung $\leq T$. **NP**-schwer.

Exakte Lösung

Algorithmus

Dynamische Programmierung (ähnlich wie Knapsack). Kanten (i,j) mit $i < j$, da azyklisch.

Algorithmus

$$g_1(c) = 0,\ \text{Für } c = 0, \ldots, OPT,$$
$$g_j(0) = \infty,\ \text{Für } j = 2, \ldots, n,$$
$$g_j(c) = \min \left\{ g_j(c-1), \min_{k \mid c_{kj} \le c} \{ g_k(c - c_{kj}) + t_{kj} \} \right\}$$
$$\text{Für } j = 2, \ldots, n;\ c = 1, \ldots, OPT$$

Exakte Lösung

Laufzeit

$$g_1(c) = 0, \text{ Für } c = 0, \dots, OPT,$$
$$g_j(0) = \infty, \text{ Für } j = 2, \dots, n,$$
$$g_j(c) = \min \left\{ g_j(c-1), \min_{k | c_{kj} \leq c} \{ g_k(c - c_{kj}) + t_{kj} \} \right\}$$
$$\text{Für } j = 2, \dots, n; \; c = 1, \dots, OPT$$

- $\mathcal{O}(OPT \cdot n \cdot \text{Aufwand pro } (c, j))$
  - Pro (c, j) evtl. alle Vorgänger betrachten
  - $\mathcal{O}\left(n^2 OPT\right) = \mathcal{O}\left(|E| OPT\right)$
- Pseudopolynomiell

Exakte Lösung

Terminierung

Man weiß $OPT = \min\{c \mid g_n(c) \leq T\}$

- Setze OPT, sobald erstes c mit $g_n(c) \leq T$ gefunden.

Das FPTAS

Test für Grenzen von *OPT*

Wir suchen zunächst Verfahren, dass untere und obere Schranken für *OPT* findet.

- Wünsch-dir-was: Polynomieller Algorithmus $TEST(k)$, sodass

$$TEST_{magic}(k) = \begin{cases} 1 & \text{falls } OPT \geq k \\ 0 & \text{falls } OPT < k \end{cases}$$

  - Binäre Suche auf $0, \ldots, UB$
  - Leider **NP**-schwer

Das FPTAS

Test für Grenzen von *OPT*

$TEST_{magic}(k)$ kann nicht existieren, also schwächer:

Eigenschaften von $TEST(k)$

$$TEST(k) = \begin{cases} 1 & \text{falls } OPT \geq k \\ 0 & \text{falls } OPT < k(1+\epsilon) \end{cases}$$

$TEST(K)$

- Skaliere und runde Kantengewichte als $\hat{c}_{ij} = \lfloor \frac{c_{ij}(n-1)}{k\epsilon} \rfloor$
- Wende exakten Algorithmus an, bis $g_n(c) \leq T$ gefunden ist für $c < \frac{n-1}{\epsilon}$ oder $c \geq \frac{n-1}{\epsilon}$.