Approximationsalgorithmen

Marco Ammon

14. Oktober 2020

Inhaltsverzeichnis

1 Kombinatorisches Optimierungsproblem Π

1.1 Definition

$$\begin{aligned}
\mathcal{D} &= \text{Menge der Eingaben } I \\
\mathcal{S}(I \in \mathcal{D}) &= \text{Menge der zur Eingabe } I \text{ zulässigen Lösungen} \\
f : \mathcal{S}(I) \mapsto \mathbb{N}^{\neq 0} &= \text{Bewertungs-/Kosten-/Maßfunction} \\
\text{ziel} \in \{\min, \max\}
\end{aligned}$$

1. Beschränkung auf natürliche Zahlen, weil Vergleich reeller Zahlen bislang nicht beweisbar schnell funktioniert.
2. Ausschluss der 0 für spätere Definitionen sinnvoll (lässt sich durch Modifikation von f in der Regel trivial erreichen)

Gesucht ist zu $I \in \mathcal{D}$ eine zulässige Lösung $\sigma_{\text{opt}} \in \mathcal{S}(I)$, sodass

$$\text{OPT}(I) = f(\sigma_{\text{opt}}) = \text{ziel}\{f(\sigma) \mid \sigma \in \mathcal{S}(I)\}$$

1.2 Beispiele

TODO: TSP, Rucksackproblem, etc.

2 $t(n)$-Zeit-Approximationsalgorithmus A

Für Eingabe $I \in \mathcal{D}$ berechnet A in Zeit $t(|I|)$ eine Ausgabe $\sigma_I^A \in \mathcal{S}(I)$. Es gilt die Schreibweise $A(I) = f(\sigma_I^A)$.

3 Konstante Gütegarantie

3.1 Definition

1. A hat bei Eingabe I absolute Güte von
$$\kappa_A(I) = |A(I) - \text{OPT}(I)|$$
2. Die absolute Worst-Case-Güte von A abhängig von der Eingabelänge $n = |I|$ ist die Funktion
$$\kappa_A^{\text{wc}}(n) = \max\{\kappa_A(I) \mid I \in \mathcal{D}, |I| <= n\}$$
3. A garantiert eine absolute Güte von $\kappa_A : \mathbb{N} \mapsto \mathbb{N}$, falls für alle $n \in \mathbb{N}$ gilt:
$$\kappa_A^{\text{wc}}(n) \leq \kappa_A(n)$$
4. A hat eine absolute Abweichung von $\kappa'_A : \mathbb{N} \mapsto \mathbb{N}$, falls für unendlich viele n gilt
$$\kappa'_A(n) \leq \kappa_A^{wc}(n)$$
Eine unendlich große Menge $\mathcal{D}' \subseteq \mathcal{D}$ heißt $\kappa'_A(n)$-Zeugenmenge gegen A, wenn für alle $I \in \mathcal{D}'$ gilt:
$$\kappa_A(I) \geq \kappa'_A(|I|)$$

3.2 Unmöglichkeitsergebnis für das Rucksackproblem

Satz. *Falls $P \neq NP$, dann gibt es keine Konstante $n \in \mathbb{N}$, sodass es einen polynomiellen Approximationsalgorithmus A für das Rucksackproblem gibt mit*

$$|A(I) - \mathrm{OPT}(I)| \leq k$$

Widerspruchsbeweis. Unter der Annahme, dass A und k existieren, kann RUCKSACK in Polynomzeit exakt gelöst werden, was $P = NP$ zur Folge hat:

Konstruiere aus einer Instanz $I = \langle W, \mathrm{vol}, p, B\rangle$ eine neue Probleminstanz $I' = \langle W, \mathrm{vol}, p', B\rangle$ mit $p'(w) = (k+1) \cdot p(w)$. Eine zulässige Lösung σ für I ist auch eine zulässige Lösung für I'. Gleiches gilt aufgrund der Monotonie der Multiplikation auch für optimale Lösungen. Durch die Multiplikation aller Preise mit $k+1$ beträgt die „Lücke" zwischen den optimalen und der ersten nicht-optimalen Lösung für I' mindestens $k+1$.

Da A eine absolute Güte von k garantiert und in Polynomzeit terminiert, kann es nur eine optimale Lösung für I', welche auf optimal für I ist, zurückgeben. Damit ist das NP-vollständige RUCKSACK in Polynomzeit exakt lösbar. □

Die hierbei verwendete Vorgehensweise einer Selbstreduktion sowie das „Aufblasen" des Problems („Scaling", „Gap Amplification") lässt sich auch auf viele andere Probleme wie etwa SETCOVER anwenden. Folglich kann eine konstante Gütegarantie nur für vergleichsweise wenig Probleme erreicht werden.

4 Graphfärbbarkeit

4.1 Knotenfärbungsproblem

4.1.1 Definition

$$\begin{aligned}
\mathcal{D} &= \{\langle G\rangle \mid G = (V,E)\,\text{ein ungerichteter Graph mit mindestens einer Kante}\}\\
\mathcal{S}(\langle G\rangle) &= \{c_{\mathrm{V}} \mid c_{\mathrm{V}}\,\text{ist eine Knotenfärbung von}\,G\}\\
f(c_{\mathrm{V}}) &= |c_{\mathrm{V}}(V)|\\
\mathrm{ziel} &= \min
\end{aligned}$$

Die Größe der kleinsten möglichen Knotenfärbung ist die chromatische Zahl $\chi(G)$.

4.1.2 Algorithmen

GreedyCol

for all $u_i \in V$ **do**
  $c_{\mathrm{V}}(u_i) = \infty$
end for
for all $i \in [1, \ldots, |V|]$ **do**
  $c_{\mathrm{V}}(u_i) = \min\{\mathbb{N} \setminus \{c_{\mathrm{V}}(\Gamma(u_i))\}\}$
end for
return c_{V}

Satz. GREEDYCOL *berechnet in Zeit $\mathcal{O}(|V| + |E|)$ eine Knotenfärbung aus höchstens $\Delta(G) + 1$ Farben.*

Beweis. Da ein Knoten u maximal $\Delta(G)$ viele Nachbarn haben kann, muss in $[1, \ldots, \Delta(G) + 1]$ noch mindestens eine Farbe frei sein. □

Satz. GREEDYCOL *garantiert eine absolute Güte von*

$$\kappa_{\text{GREEDYCOL}}(G) = \text{GREEDYCOL}(G) - \text{OPT}(G) \leq \Delta(G) + 1 - 2 = \Delta(G) - 1$$

, weil die untere Schranke $\text{OPT}(G) \geq 2$ *für Graphen mit* $|V| \geq 2$ *gilt.*

Zeuge. $\Delta(G) - 1$*-Zeuge gegen* GREEDYCOL*: TODO (Abbildung 2.1)*

4.2 Kantenfärbungsproblem

4.2.1 Definition

$$\begin{aligned}
\mathcal{D} &= \{\langle G \rangle \mid G = (V, E)\, \text{ein ungerichteter Graph mit mindestens einer Kante}\} \\
\mathcal{S}(\langle G \rangle) &= \{c_{\text{E}} \mid c_{\text{E}}\, \text{ist eine Kantenfärbung von}\, G\} \\
f(c_{\text{E}}) &= |c_{\text{E}}(E)| \\
\text{ziel} &= \min
\end{aligned}$$

Die Größe der kleinsten möglichen Kantenfärbung ist der chromatische Index $\chi'(G)$.

4.2.2 Algorithmen

TODO: Übung

5 Relative Gütegarantie

5.1 Definition

1. A hat bei Eingabe I eine relative Güte von

$$\rho_A(I) = \max\left\{\frac{A(I)}{\text{OPT}(I)}, \frac{\text{OPT}(I)}{A(i)}\right\} \geq 1$$

2. Die relative worst-case-Güte von A ist die Funktion

$$\rho_A^{\text{wc}}(n) = \max\{\rho_A(I) \mid I \in \mathcal{D}, |i| \leq n\}$$

3. A garantiert eine relative Güte von $\rho_A : \mathbb{N} \mapsto \mathbb{N}$, falls für alle $n \in \mathbb{N}$ gilt

$$\rho_A^{\text{wc}}(n) \leq \rho_A(n)$$

4. A macht für die Eingabe $I \in \mathcal{D}$ einen relativen Fehler von

$$\varepsilon_A(I) = \frac{|A(I) - \text{OPT}(I)|}{\text{OPT}(I)} = |\frac{A(I)}{\text{OPT}(I)} - 1|$$

5. A garantiert einen relativen Fehler von $\varepsilon_A(n)$, falls für alle $\{I \mid I \in \mathcal{D}, |I| \leq n\}$ gilt

$$\varepsilon_A(I) \leq \varepsilon_A(n)$$

6. A hat eine relative Abweichung von $\rho'_A : \mathbb{N} \mapsto \mathbb{N}$, falls für unendlich viele n gilt

$$\rho_A^{\text{wc}}(n) \geq \rho'_A(n)$$

   Eine unendlich große Menge $\mathcal{D}' \subseteq \mathcal{D}$ heißt $\rho'_A(n)$-Zeugenmenge gegen A, wenn für alle $I \in \mathcal{D}'$ gilt

$$\rho_A(I) \geq \rho'_A(|I|)$$

Es folgen daraus direkt, dass

1. bei einem Minimierungsproblem $1 + \varepsilon_A(n) = \rho_A(n)$ ist.
2. bei einem Maximierungsproblem $1 - \varepsilon_A(n) = \frac{1}{\rho_A(n)}$ ist.
3. für alle Probleme $\varepsilon_A(n) \leq \rho_A(n) - 1$ ist.

Weiter lassen sich damit obere bzw. untere Schranken der Optimallösung aus einer approximierten Lösung angeben. Es folgt, dass

1. bei einem Minimierungsproblem gilt
$$\frac{1}{\rho_A(|I|)} \cdot A(I) \leq \mathrm{OPT}(I) \leq A(I) \leq \rho_A(|I|) \cdot \mathrm{OPT}(I)$$
2. bei einem Maximierungsproblem gilt
$$\frac{1}{\rho_A(|I|)} \cdot \mathrm{OPT}(I) \leq A(I) \leq \mathrm{OPT}(I) \leq \rho_A(|I|) \cdot A(I)$$
3. bei beiden Problemtypen mit der Beziehung
$$|A(I) - \mathrm{OPT}(I)| \leq \varepsilon_A(|I|) \cdot \mathrm{OPT}(I)$$
gilt
$$(1 - \varepsilon_A(|I|)) \cdot \mathrm{OPT}(I) \leq A(I) \leq (1 + \varepsilon_A(|I|)) \cdot \mathrm{OPT}(I)$$

6 Das metrische Traveling Salesperson Problem ΔTSP

6.1 Definition

$$\begin{aligned}
\mathcal{D} = \{\langle K_n, c\rangle \mid\ & K_n \text{ vollständiger Graph auf } n \text{ Knoten}, c : E \mapsto \mathbb{N}, \\
& \underbrace{\forall u, v, w \in V : c(u, v) \leq c(u, w) + c(w, v)}_{\text{Dreiecksungleichung}}\} \\
\mathcal{S}(\langle K_n, c\rangle) &= \{C \mid C = (v_{i_1}, v_{i_2}, \ldots, v_{i_n}, v_{i_i}) \text{ ist ein Hamiltonkreis}\} \\
f(c_{\mathrm{E}}) &= c(v_{i_n}, v_{i_1}) + \sum_{j=1}^{n-1} c(v_{i_j}, v_{i_{j+1}}) \\
\text{ziel} &= \min
\end{aligned}$$

6.2 Christofides' Algorithmus CH

Ein Matching M eines kantengewichteten Graphen G ist ein Teilgraph von G mit $\Delta(G) \leq 1$. Ist G ein vollständiger Graph mit $|V|$ gerade, dann gibt es perfekte Matchings. In einem perfekten Matching haben alle Knoten genau den Grad 1. Ein perfektes Matching mit kleinstmöglichem Gewicht wird als leichtestes Matching bezeichnet. Ein solches leichtestes Matching kann in $\mathcal{O}(n^{2.5} \cdot (\log n)^4)$ berechnet werden.

Als Multi-Graph wird ein Graph bezeichnet, der um mehrere Kanten zwischen den gleichen Knoten erweitert wurde.

Wird in einem Pfad jede Kante des (Multi-)Graph genau einmal besucht, so spricht man von einem Euler-Pfad. Bildet der Pfad einen Kreis, so nennt man ihn Euler-Kreis oder Euler-Tour. Haben alle Knoten von G geraden Grad, so existiert eine Euler-Tour in G. Diese lässt sich in $\mathcal{O}(|V| + |E|)$ berechnen.

Der Algorithmus von Christofides (CH) geht wie folgt vor:

berechne einen minimalen Spannbaum T_{CH} von $I = \langle K_n, c \rangle$
$S = \{v \in T_{\text{CH}} \mid \deg_{T_{\text{CH}}}(v) \text{ ungerade}\}$ $\triangleright$ $|S|$ ist gerade
berechne auf dem durch S induzierten Teilgraphen des K_n ein leichtestes Matching M_{CH}
berechne eine Euler-Tour $E = (u_1, u_2, \dots)$ auf $T_{\text{CH}} \uplus M_{\text{CH}}$ $\triangleright$ $T_{\text{CH}} \uplus M_{\text{CH}}$ kann Multi-Graph sein, alle Knoten haben geraden Grad
entferne Wiederholungen von Knoten in E, sodass man E' erhält
return E'

Satz. CH, *gestartet mit einer Eingabe auf* n *Knoten, garantiert eine relative Güte von* $\rho_{\text{CH}} \leq \frac{3}{2} - \frac{1}{n}$ *in einer Laufzeit von* $\mathcal{O}(n^{2.5} \cdot (\log n)^4)$.

Beweis. Sei R^* eine optimale Rundreise für I, d.h. $c(R^*) = \text{OPT}(I)$. Es gilt $\text{CH}(I) = c(E') \leq (\frac{3}{2} - \frac{1}{n}) \cdot c(R^*)$ zu zeigen.

1. Da R^* aus n Kanten besteht, muss durch ein Durchschnittsargument mindestens eine Kante e mit $c(e) \geq \frac{c(R^*)}{n}$ existieren. Wird diese aus R^* entfernt, so enthält man einen Spannbaum des K_n. Da T_{CH} minimal ist, gilt

$$c(T_{\text{CH}}) \leq c(R^*) - \frac{c(R^*)}{n} = \left(1 - \frac{1}{n}\right) \cdot c(R^*)$$

2. In beliebigen Bäumen ist die Anzahl der Knoten mit ungeradem Grad gerade.

3. Zur Vereinfachung werden die Knoten so umbenannt, dass $R^* = (u_1, u_2, \dots, u_n, u_1)$ ist. S kann dann als $S = \{u_{i_1}, \dots, u_{i_{|S|}}\}$ mit $i_1 < \dots < i_{|S|}$ geschrieben werden.

   Aus S kann ein Kreis $H = (u_{i_1}, \dots, u_{i_{|S|}}, u_{i_1})$ gebildet werden. Durch die Dreiecksungleichung ($|H| \leq n$ und jede „Abkürzung“ ist maximal gleich lang wie der Weg in R^*) gilt $c(H) \leq c(R^*)$.

   Es können zwei perfekte Matching M_1 und M_2 auf H berechnet werden, denn $|S|$ ist gerade. Weil M_{CH} minimal ist, folgt o.B.d.A. mit $c(M_1) \leq c(M_2)$ die Aussage

$$c(M_{\text{CH}}) \leq c(M_1) \leq \frac{1}{2} \cdot (c(M_1) + c(M_2)) = \frac{1}{2} \cdot c(H) \leq \frac{1}{2} \cdot c(R^*)$$

4. Da jeder Knoten in $T_{\text{CH}} \uplus M_{\text{CH}}$ geraden Grad hat, kann eine Euler-Tour E berechnet werden. Weil diese nur Kanten aus $T_{\text{CH}} \uplus M_{\text{CH}}$ benutzt, kann ihre Länge mit den vorherigen Ergebnissen wie folgt beschränkt werden:

$$c(E) = c(T_{\text{CH}} \uplus M_{\text{CH}}) \leq \left(1 - \frac{1}{n}\right) \cdot c(R^*) + \frac{1}{n} \cdot c(R^*) = \left(\frac{3}{2} - \frac{1}{n}\right) \cdot \text{OPT}(I)$$

5. Durch die Dreiecksungleichung kann E' nicht länger als E werden.

$\square$

Zeuge. *TODO*