

Approximations-algoritmer

Sidste gang

- Motivation
- Definitioner
- Approximations-algoritme for knudeoverdækning
- Approximations-algoritme for TSP med trekantsulighed
- Negativt resultat om generel TSP
- Approximations-algoritme for SET-COVERING

Denne gang

- Fuldt polynomiel-tids approximations skema (FPTAS) for SUBSET-SUM

Motivation

- Faglig indsigt: trods \mathcal{NP} -fuldstændighed kan vi finde vilkårligt god tilnærmet løsning i polynomiel tid.
- Eksamensrelevant!

1

Approximations-algoritmer

Ikke-eksakte løsningsmetoder, der giver garanti for hvor tæt på optimum man kommer.

- C — algoritmens løsningsværdi
- C^* — problemets optimale løsningsværdi

Minimeringsproblem ($C \geq C^*$)

Mål: gør $\frac{C}{C^*}$ så lille som muligt

Maximeringsproblem ($C \leq C^*$)

Mål: gør $\frac{C}{C^*}$ så lille som muligt

Generelt krav

Gør

$$\max\left(\frac{C}{C^*}, \frac{C^*}{C}\right)$$

så lille som mulig

$$\max\left(\frac{C}{C^*}, \frac{C^*}{C}\right) \leq \rho(n)$$

hvor $\rho(n)$ er "ratio bound". Bemærk $\rho(n) \geq 1$

2

Betegnelser

- **A** Approximations-algoritme
- **PTA** Polynomiel-Tids Approximations-algoritme
- **AS** Approximations Skema
- **PTAS** Polynomiel-Tids Approximations Skema
- **FPTAS** Fuldt Polynomiel-Tids Approximations Skema

- **Heuristik** Ingen garanti for løsningskvalitet

3

Subset-sum Problem

Subset-sum problem (delmængde sum):

- Givet mængde af heltal $S = \{x_1, x_2, \dots, x_n\}$ samt t
- Find en delmængde $S' \subseteq S$ så

$$\sum_{j \in S'} x_j \leq t$$

er størst mulig

For eksempel

$$S = \{1, 4, 16, 64, 256, 1040, 1093, 1284, 1344\}$$
$$t = 3754$$

har løsning $S' = \{1, 16, 64, 256, 1040, 1093, 1284\}$.

Fuldt polynomiel-tids approximations skema

4

Ekspontiel algoritme

Dynamisk programmering-lignende algoritme.

Lister over tal som kan opnås med en delmængde af S

- Liste L af positive heltal, f.eks. $L = \langle 1, 2, 3, 5, 9 \rangle$
- x positivt heltal, f.eks. $x = 2$
- $L + x = \langle 3, 4, 5, 7, 11 \rangle$

Lad

$$P_i = \{ \text{tal, der kan opnås som sum af } \{x_1, \dots, x_i\} \}.$$

så gælder recursionsligningen

$$P_i = P_{i-1} \cup (P_{i-1} + x_i)$$

Her kan foreningsmængden $L_1 \cup L_2$ udregnes i lineær tid da begge lister er sorteret i voksende orden (MERGE-LISTS).

Eksempel

Lad $S = \{4, 5, 7\}$, $t = 14$

$$P_1 = \{0, 4\}$$

$$P_2 = \{0, 4, 5, 9\}$$

$$P_3 = \{0, 4, 5, 7, 9, 11, 12, 19\}$$

5

Ekspontiel algoritme

Algoritme

```
EXACT-SUBSET-SUM( $S, t$ )
1  $n \leftarrow |S|$ 
2  $L_0 \leftarrow \langle 0 \rangle$ 
3 for  $i \leftarrow 1$  to  $n$  do
4    $L_i \leftarrow \text{MERGE-LISTS}(L_{i-1}, L_{i-1} + x_i)$ 
5   remove from  $L_i$  every element that is greater than  $t$ 
6 return the largest element in  $L_n$ 
```

Total køretid: nt .

Ekspontiel i input størrelsen $n \log t$.

F.eks: $t = 2^n$, køretid $n2^n = O(2^n)$, input n^2 .

Idé til forbedring

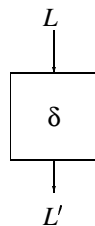
- fjern elementer i P_i som ligger "tæt" på hinanden
- behold det mindste af to tal tæt på hinanden

TRIM liste L

6

Trimining

trim parameter δ med $0 < \delta < 1$



- Fjern så mange elementer tæt på hinanden som muligt
- For ethvert fjernet element $y \in L$ eksisterer $z \in L'$ med $z \leq y$ så

$$\frac{y}{1 + \delta} \leq z$$
$$\left(y \leq (1 + \delta)z \quad \frac{y}{z} \leq (1 + \delta) \right)$$

Eksempel: $\delta = 0.1$

$$L = \langle 10, 11, 12, 15, 20, 21, 22, 23, 24, 29 \rangle$$

$$L' = \langle 10, 12, 15, 20, 23, 29 \rangle$$

7

Trimining

Algoritme:

```
TRIM( $L, \delta$ )
1  $m \leftarrow |L|$ 
2  $L' \leftarrow \langle y_1 \rangle$ 
3  $last \leftarrow y_1$ 
3 for  $i \leftarrow 2$  to  $m$  do
4   if  $y_i > last \cdot (1 + \delta)$  then
5     append  $y_i$  onto the end of  $L'$ 
5      $last \leftarrow y_i$ 
6 return  $L'$ 
```

Køretid: lineær.

8

Approximations algoritme

Givet instans (S, t) , tilladt fejl ϵ .

- vælg trimningsfaktor $\delta = \frac{\epsilon}{2n}$

APPROX-SUBSET-SUM(S, t, ϵ)

```

1  $n \leftarrow |S|$ 
2  $L_0 \leftarrow \langle 0 \rangle$ 
3 for  $i \leftarrow 1$  to  $n$  do
4    $L_i \leftarrow \text{MERGE-LISTS}(L_{i-1}, L_{i-1} + x_i)$ 
5    $L_i \leftarrow \text{TRIM}(L_i, \epsilon/2n)$ 
6   remove from  $L_i$  every element that is greater than  $t$ 
7 return  $z^A$  given by the largest element in  $L_n$ 

```

(Bogen kalder returnerede værdi z^* , hvilket ikke harmonerer med at z^A angiver optimal løsning)

Sætning

Fuldt polynomiel-tids approximations skema

- 1 Finder en lovlig løsning z^A
- 2 Relative fejl er mindre end ϵ

$$\frac{|z^* - z^A|}{z^*} \leq \epsilon$$

- 3 Algoritmen kører i polynomiel tid i n og $1/\epsilon$

Vi kan antage $\epsilon < 1$, da 2-approximation nem (overvej!)

9

Eksempel

Instans: $S = \{104, 102, 201, 101\}$, $t = 308$, $\epsilon = 0.40$
Vælger $\delta = \epsilon/2n = 0.40/8 = 0.05$.

Forløb:

line 2: $L_0 = \langle 0 \rangle$

line 4: $L_1 = \langle 0, 104 \rangle$

line 5: $L_1 = \langle 0, 104 \rangle$

line 6: $L_1 = \langle 0, 104 \rangle$

line 4: $L_2 = \langle 0, 102, 104, 206 \rangle$

line 5: $L_2 = \langle 0, 102, 206 \rangle$

line 6: $L_2 = \langle 0, 102, 206 \rangle$

line 4: $L_3 = \langle 0, 102, 201, 206, 303, 407 \rangle$

line 5: $L_3 = \langle 0, 102, 201, 303, 407 \rangle$

line 6: $L_3 = \langle 0, 102, 201, 303 \rangle$

line 4: $L_4 = \langle 0, 101, 102, 201, 203, 302, 303, 404 \rangle$

line 5: $L_4 = \langle 0, 101, 201, 302, 404 \rangle$

line 6: $L_4 = \langle 0, 101, 201, 302 \rangle$

Approximativ løsning: $z^A = 302$

Optimal løsning: $z^* = 307$, (faktisk afvigelse 2%)

10

Finder en lovlig løsning

- Kun lovlige summer i L , alle $\leq t$
- Naturligvis lovlig sum i sidste iteration

Relativ fejl mindre end ϵ

For ethvert fjernet element $y \in P$ eksisterer $z \in L$ så

$$\frac{y}{1 + \delta} \leq z \leq y$$

Ved induktion i i kan det vises at efter i iterationer gælder:

For ethvert fjernet element $y \in P_i$ eksisterer $z \in L_i$ så

$$\frac{y}{(1 + \delta)^i} \leq z \leq y$$

Gælder specielt for $z^* \in P_n$, dvs. der findes $z \in L_n$ så

$$\frac{z^*}{(1 + \delta)^n} \leq z \leq z^*$$

Må også gælde for z^A som er den største værdi i L_n så

$$\frac{z^*}{(1 + \delta)^n} \leq z^A$$

eller

$$\frac{z^*}{z^A} \leq (1 + \delta)^n$$

Vil vise at $(1 + \delta)^n \leq 1 + \epsilon$ når $\delta = \epsilon/2n$

11

Relativ fejl mindre end ϵ

Betragter $f(n, \epsilon) = (1 + \frac{\epsilon}{2n})^n$.

Formel (3.13) side 53 siger at

$$\lim_{n \rightarrow \infty} \left(1 + \frac{x}{n}\right)^n = e^x$$

og da $\frac{d}{dn} f(n, \epsilon) > 0$ er f voksende.

$$\left(1 + \frac{\epsilon}{2n}\right)^n \leq e^{\epsilon/2}$$

Formel (3.12) side 53 siger $e^x \leq 1 + x + x^2$ for $|x| < 1$ så

$$\left(1 + \frac{\epsilon}{2n}\right)^n \leq 1 + \epsilon/2 + (\epsilon/2)^2$$

og da $\epsilon < 1$ får vi:

$$\left(1 + \frac{\epsilon}{2n}\right)^n \leq 1 + \epsilon$$

Kombineret med formlen forrige side fås

$$\frac{z^*}{z^A} \leq 1 + \epsilon$$

og dermed

$$\frac{|z^* - z^A|}{z^*} = 1 - \frac{z^A}{z^*} \leq 1 - \frac{1}{1 + \epsilon} = \frac{1 + \epsilon - 1}{1 + \epsilon} \leq \epsilon$$

12

Kører i polynomiel tid i n og $1/\epsilon$

TRIM sletter et element $z \geq z'$ hvis

$$\frac{z}{z'} \leq (1 + \delta)$$

Så på ethvert tidspunkt vil elementerne i L_i overholde

$$z > (1 + \delta)z' = \alpha z' \quad \alpha = (1 + \delta) \text{ afstands-faktor}$$

Den tættest mulige liste ser ud som

$$0, 1, \alpha, \alpha^2, \alpha^3, \dots, \alpha^k \approx t$$

Antal elementer er $k + 2$, hvor k findes som:

$$\alpha^k = t \Leftrightarrow k \ln \alpha = \ln t \Leftrightarrow k = \frac{\ln t}{\ln \alpha} = \frac{\ln t}{\ln(1 + \delta)}$$

Formel (3.16) side 54 siger at for $\delta > -1$ gælder

$$\frac{\delta}{1 + \delta} \leq \ln(1 + \delta) \Leftrightarrow \frac{1}{\ln(1 + \delta)} \leq \frac{1 + \delta}{\delta}$$

så

$$k \leq \frac{(1 + \delta) \ln t}{\delta} = \frac{(1 + \epsilon/2n) \ln t}{\epsilon/2n}$$

Da $\epsilon < 1$ gælder at $(1 + \epsilon/2n) \leq 2$ så

$$k \leq \frac{4n \ln t}{\epsilon}$$

Køretid: $O(n(k + 2)) = O(\frac{1}{\epsilon} n^2 \ln t)$

13

Indsigt

Approximabilitet og reduktion af problemer

- Da SUBSET-SUM er \mathcal{NP} -fuldstændigt gælder f.eks.

$$\text{TSP} \leq_p \text{SUBSET-SUM}$$

- Vi kan approximere SUBSET-SUM vilkårligt godt i polynomiel tid
- Gælder tilsvarende resultat for TSP?

Reduktion gælder kun for afgørlighedsproblemer

Såfremt vi vil løse SUBSET-SUM-DECISION skal $\epsilon = 0$

14

Branch-and-bound som Approximations Skema

Minimeringsproblem: (givet $\epsilon > 0$)

- For ethvert underproblem (knode) lad ℓ være en nedre grænseværdi
- Lad z være hidtil bedste løsning
- Forkast underproblem hvis $\ell(1 + \epsilon) \geq z$
- Eksempel: $\epsilon = 0.1$, $\ell = 11$, $z = 12$. Forkast knode

Approximations skema. Relativ afvigelse $< \epsilon$.

Køretid: ???

Branch-and-bound i Polynomiel tid

Undersøg kun et polynomiel antal knuder $M = n^k$

- Best-first søgning
- Stop efter M knuder

Køretid polynomiel.

Løsningskvalitet: ???

15