Seminar 5

Ştefan Popescu

April 2023

Vouchere de Vacanță

Cerință

(10p) George a intrat în concediu, şi ca atare doreşte să consume cât mai mult din pachetul său de vouchere de vacanţă. El şi-a făcut o listă cu n locaţii unde poate cheltui din suma alocată. Ştiind că George nu are de gând să cheltuiască din banii proprii decât pe transportul între locaţii şi că el se poate deplasa între oricare două locaţii, ajutaţi-l pe George să găsească o listă de locaţii pe care să le viziteze astfel încât să îşi folosească voucherele cât mai eficient.

Cerinţe:

Se cere un algoritm 1/2-aproximativ care rezolvă problema lui George în timp eficient din punct de vedere al complexităţii timp. (pseudocod + justificare)

Algoritmul va avea ca output lista de locaţii şi suma totală ce va fi cheltuită.

Notaţii şi indicaţii:

- Notăm cu S suma de bani pe care o are George sub formă de vouchere.
- Locaţiile sunt etichetate cu numerele $1, 2, ..., n$. Costurile celor n locaţii sunt notate cu $C_1, C_2, ..., C_n$
- Suma de bani S este alocată pe un card, nu sub formă de tichete. Din cont se poate extrage orice sumă disponibilă.
- Se lucrează cu numere întregi.

Soluție

Problema este în esență o variantă a *Problemei Rucsacului* în care valoarea și greutatea pentru fiecare obiect sunt egale. Se poate folosi algoritmul din curs, care este 1/2 aproximativ.

Camioane și Transporturi

Cerință

(15p) Avem n colete de transportat, fiecare având greutatea de $w_1, w_2, ..., w_n$. Pentru a le transporta, putem folosi un număr de camioane, fiecare având capacitatea de transport G. presupunem că $w \leq G$, pentru orice i. Ne dorim sa minimizăm numărul de camioane folosite. Considerăm următorul plan de încărcare a camioanelor:

Odată ce avem la dispoziție un camion pentru a fi încărcat, iterăm prin mulțimea coletelor, încărcându-le în camion, până când dăm peste primul colet ce nu mai încape. În acel moment considerat că am terminat de încărcat camionul curent și trecem la următorul camion, prima dată încărcând coletul care nu a mai încăput în cel precedent.

Cerințe:

a) Arătați, printr-un exemplu simplu, că metoda de mai sus nu furnizează soluția optimă. **(5p)**

b) Arătați totuși că soluția de mai sus este un algoritm 2-aproximativ pentru problema noastră. **(10p)**

Notații:

n - numărul de colete

G - capacitatea unui camion

W - Lista cu greutățile coletelor în ordinea în care sosesc spre a fi procesate.

OPT – soluția optimă care există: Numărul minim de camioane ce poate fi folosit pentru a transporta cele n colete de greutate totală G

ALG – soluția oferită de algoritmul nostru: numărul de camioane folosite conform algoritmului

$G[i]$ - greutatea cu care este încărcat camionul i în configurația dată de algoritmul nostru. Evident $G[i] \leq G$.

Soluție

a) Un exemplu ar putea fi $n = 3, G = 100, W = \{30, 80, 60\}$; soluțiile obținute sunt: $ALG = 3, OPT = 2$.

b) Presupunem că soluția dată de algoritm folosește $2p$ camioane. Vom arăta că soluția optimă folosește cel puțin p camioane:

Din algoritm se poate observa că $G[2i-1] + G[2i] \geq G, \forall i \in \{1, 2, ..., p\}$. Deci pentru fiecare pereche de camioane consecutive folosite de algoritmul nostru, o abordare optima ar folosi cel puțin 1 camion. Cum algoritmul nostru folosește exact p perechi de camioane, o abordare optimă ar folosi cel puțin p camioane.

Dacă soluția noastră folosește $2p + 1$ camioane:

Din analiza precedentă am putut observa că primele p perechi de camioane transportă o încărcătură totală de minim $p * G$. Pentru această încărcătură sunt necesare cel puțin p camioane. Însă mai este nevoie de un camion în plus pentru a transporta și $G[2p+1]$, pentru un total de minim $p+1$ camioane într-o configurație optimă, atunci când algoritmul nostru folosește $2p + 1$ camioane.

Cadourile lui Moș Crăciun

Cerință

(30p) O clasă de elevi este formată din m copii. La această clasă Moș Crăciun aduce n cadouri, fiecare dintre ele având o valoare notată $val_1, val_2, ..., val_n$. Moșul, grăbit fiind, lasă task-ul împărțirii cadourilor pe umerii profesorului de informatică. Acesta își alege următorul obiectiv pentru împărțirea cadourilor: sa maximizeze valoarea cadourilor primite de către copilul care primește cel mai puțin. (Altfel spus, copilul cel mai "vitregit" în urma împărțirii să primească totuși cadouri de o valoare cât mai bună / Să existe un "spread" cât mai bun al cadourilor). În timp ce se gândește la o soluție de implementare, profesorul își amintește de la cursurile de algoritmică din facultate că astfel de probleme de optim sunt NP-hard! Fiind cuprins de groază (dar și de o nostalgie pentru cursurile de algoritmică din facultate) el vă cere vouă ajutorul.

Notații:

OPT – soluția optimă care există: Valoarea totală a cadourilor primite de copilul cel mai "vitregit" în o configurație optima

ALG – soluția oferită de algoritmul vostru: Valoarea totală a cadourilor primite de copilul cel mai "vitregit" în urma algoritmului vostru

$C[k]$ - lista cadourilor primite de către copilul k la finalul algoritmului vostru

$W(K)$ – valoarea totala a cadourilor primite de către copilul k la finalul algoritmului vostru

$val(i)$ – valoarea cadoului i

$Val = \sum_{1 \leq i \leq n} val(i)$ – valoarea totală a cadourilor

Copiii vor fi indexați folosind variabile de forma k, q, k', q', etc... Cadourile vor fi indexate folosind variabile de forma i, j, i', j', etc....

Restricții: Considerăm că $n \geq m$ și că $val(i) \leq \frac{Val}{2m}$ pentru oricare cadou i.

Cerințe:

a) Descrieți un algoritm $\frac{1}{2}$ aproximativ pentru problema cadourilor in complexitate $\mathcal{O}(n \log m)$ **(10p)**

b) Fie k acel copil pentru care $ALG = W(K)$. Fie i ultimul cadou primit de un copil oarecare q ($q \neq k$). Care este relația între $W(K)$ și $W(Q) - val(i)$? Justificaţi. **(5p)**

c) Pe baza punctului b) arătați că $ALG \geq \frac{Val}{2m}$ **(5p)**

d) Demonstrați că algoritmul descris la punctul a) este $\frac{1}{2}$ aproximativ **(5p)**

e) Dați un exemplu format din minimum 2 copii și 4 cadouri pentru care algoritmul vostru nu găsește soluția optima. Spuneți care este soluția optima. Spuneți care este soluția dată de algoritmul vostru. **(5p)**

Soluție

a) **for** $k \in [1:m]$ **do**: *// initializare*

$C[k] \leftarrow \emptyset$; $W(K) \leftarrow 0$;

for $i \in [1:n]$ **do**: *// pentru fiecare cadou 'i'; $\mathcal{O}(n)$*

$q \leftarrow k$ cu proprietatea ca W(K) - minima; *//alegem pe cel mai "vitregit" copil; $\mathcal{O}(\log m)$ cu minheap sau priority queue*

$C[q] \leftarrow C[q] \cup \{i\}$ *//adaugam cadoul 'i' in lista de cadouri ale lui 'q'*

$W(Q) \leftarrow W(Q) + val(i)$ *//se actualizează valoarea totala a cadourilor lui 'q'.*

b) Deoarece cadoul i este atribuit copilului q, înseamnă că, la acel moment, q avea cadourile de valoare totală cea mai mică, deci în mod evident mai mică decât cea a lui k. Deci avem relația $W(K) \geq W(Q) - val(i)$. Aceasta poate fi rescrisă ca $W(K) + val(i) \geq W(Q)$

c) Fie k acel copil pentru care $ALG = W(K)$. Vom presupune prin absurd că $W(K) < \frac{Val}{2m}$. Deoarece $W(K) + val(i) \geq W(Q)$ și $val(i) \leq \frac{Val}{2m}$ avem:

$$
\begin{aligned}
Val = \sum_{1 \leq q \leq m} W(Q) &= W(K) + \sum_{1 \leq q \leq m}^{q \neq k} W(Q) \\
&\leq W(K) + (m-1)(W(K) + \frac{Val}{2m}) \\
&\leq m * W(K) + (m-1) * \frac{Val}{2m} \\
&< m * W(K) + \frac{Val}{2} \\
&< m * \frac{Val}{2m} + \frac{Val}{2} \\
&< \frac{Val}{2} + \frac{Val}{2} \\
&< Val(!)
\end{aligned}
$$

d) Știm că $OPT \leq \frac{Val}{m}$ (cazul unei egalități ar implica un "spread" perfect). Am demonstrat la punctul c) că $ALG \geq \frac{Val}{2m}$. Rezultă deci că $2 * ALG \geq \frac{Val}{m} \geq OPT$; Iar în final avem relația dorită: $\mathbf{ALG \geq \frac{1}{2} * OPT}$

e) * $m = 2$, $n = 4$; Cadourile au valorile: $2, 3, 1, 6$. [1] Soluția optimă implică faptul că primul copil primește cadoul 4 iar cel de-al doilea cadourile $1, 2, 3$. În acest caz ambii copii primesc cadouri în valoare de 6 unități. Algoritmul nostru va rezulta în alocarea primului copil cadourile $1, 3$, celui de-al doilea copil cadoul 2 iar cadoul 4 oricăruia dintre cei doi. În orice caz, copilul cel mai vitregit primește cadouri în valoare de doar 3 unități.

[1] Acest exemplu are o scăpare: nu respectă restricția $val(i) \leq \frac{Val}{2m}$