Load Balancing

Bianca-Mihaela Stan, grupa 232

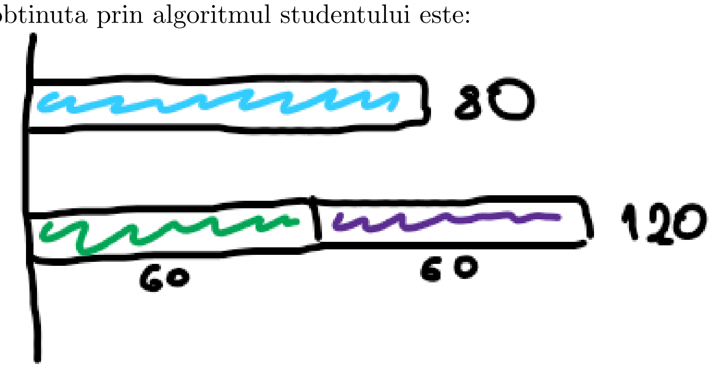
March 2021

1 Load Balance - 3p

1) Fie o iterație a problemei Load Balancing (Cursul 2, slide-ul 16) pentru 2 mașini. La seminarul de algoritmi aproximativi unul dintre studenți propune un algoritm de rezolvare si sustine ca acesta este 1.1 aproximativ. El ruleaza algoritmul pe un set de n activitati si obtine o incarcatura de 80 pe una dintre masini, respectiv 120 pe cealalta. Este posibil ca factorul lui de aproximare sa fie corect:

a) tinand cont ca rezultatul obtinut anterior a fost făcut pe un set de activități, fiecare cu timpul de lucru cel mult 100 (0.5p)

Da, este posibil. Exemplu: avem 3 activitati, A_1, A_2 si A_3 cu $t_1 = 80$, $t_2 = 60$ si t_3=60. Organizarea obtinuta prin algoritmul studentului este:



In acest caz, organizare este chiar optima, deci se verifica

$$OPT(I) \leq ALG(I) \leq 1,1 * OPT(I)$$

pentru acest I. Deci exista cazuri in care algoritmul studentului poate fi 1,1-aproximativ. Acesta nu este o demonstratie ca algoritmul este cu singuranta 1,1-aproximativ, insa exista o posibilitate sa fie.

b) tinand cont ca rezultatul obtinut anterior a fost făcut pe un set de activități, fiecare cu timpul de lucru cel mult 10 (0,5p)

Vom demonstra ca in cazul optim diferenta dintre incarcaturile celor 2 masini este maxim 10.
Presupunem prin absurd ca in cazul optim diferenta dintre masini este mai mare decat 10. Fie:

- L_1 : incarcatura uneia dintre masini in acest caz
- L_2 : incarcatura celeilalte masini in acest caz
- $OPT_1 = L_1$: optimul in acest caz

Cu proprietatea ca $L_1 > L_2 + 10$. Luam ultima activitate de pe masina cu incarcatura L_1 si o punem pe cealalta masina. Notam aceasta activitate cu A, iar timpul asociat ei cu t_A.

$$t_A \leq 10$$

In final, incarcaturile vor deveni:

$$L_1' = L_1 - t_A$$

si

$$L_2' = L_2 + t_A$$

$$L_1' = L_1 - t_A < I_1$$

$$L_2' = L_2 + t_A \leq L_2 + 10 < L_1$$

Deci, am gasit o solutie mai buna. => contradictie => in cazul optim diferenta maxima dintre incarcaturile masinilor este 10.

Fie acum cazul studentului:

- L_1 : incarcatura uneia dintre masini in acest caz
- L_2 : incarcatura celeilalte masini in acest caz

In cazul nostru, daca suma incarcaturilor celor 2 masini este 200 iar diferenta maxima dintre ele este 10, atunci:

$$L_1 + L_2 = 200$$

$$L_1 - L_2 \leq 10$$

$$L_1 - (200 - L_1) \leq 10$$

$$2 * L_1 \leq 210$$

$$L_1 \leq 105$$

Deci, pentru ca algoritmul nostru sa fie 1,1-aproximativ, rezultatul trebuie sa fie intre 105 si 105*1,1 = 115. Algoritmul nostru da un rezultat de 120. Ceea ce inseamna ca nu poate fi 1,1 aproximativ.

3) Fie algoritmul Ordered-Scheduling Algorithm (Cursul 2, slide-ul 42) care implica algoritmul descris anterior (slide 19) in care adaugam o preprocesare in care sortăm descrescător activitățile după timpul de desfășurare. Th. 2 afirmă că acest algoritm este 3/2 aproximativ. Arătați ca acest factor de aproximare poate fi îmbunătățit la 3/2-1/(2m).

Reamintim algoritmul:

```
notatii:
- J(i) = submultimea tuturor activitatilor care au fost programate sa se desfasoare pe masina i
- L_i = timpul de lucru al masinii i
- t_i = timpul necesar pentru activitatea i

sortez activitatile in ordinea descrescatoare a timpilor
load_balance(m, t_1, t_2, ..., t_n)
    for i de la 1 la m
        L_i = 0
        J(i)=multimea vida
    for j de la 1 la n
        i = arg (min {L_k| k in {1,..m}}) // i este masina cu incarcatura cea mai mica la momentul j
        J(i)=J(i)+{j}
        L_i+=t_j
```

Fie:

- k = indicele masinii care da incarcatura maxima

- j = indicele ultimei activitati adaugate la masina k
- load' (M_i) = load-ul masinii i dupa ce s-au asignat primele j-1 activitati
- t_{max} = timpul de lucru al activitatii cu timpul de lucru maxim

Daca j $\leq$ m inseamna ca numarul de activitati $\leq$ numarul de masini => fiecare masina va avea maxim o activitate. In particular, masina k va avea fix o activitate.
=> ALG = $L_k = t_j \leq t_{max} \leq OPT$

Daca j > m inseamna ca numarul de activitati > numarul de masini (n > m).
Rezultatul obtinut de algoritmul nostru este incarcatura maxima de pe o masina, adica incarcatura de pe masina k:

$$ALG = L_k \qquad (0)$$

Stim ca pe masina k, ultima activitate este activitatea j. De asemenea, stim ca inainte de a pune activitatea j pe masina k, aceasta era cea mai putin incarcata masina. Adica, daca era cea mai mica incarcatura, era mai mica sau egala cu media incarcaturilor:

$$load'(M_k) \leq \frac{1}{m} \sum_{1 \leq p \leq m} load'(M_p) \qquad (2)$$

Dar noi stim ca suma incarcaturilor de pe masini inainte de a pune activitatea j este chiar suma activitatilor de la 1 la j-1:

$$\sum_{1 \leq p \leq n} load'(M_p) = \sum_{1 \leq q \leq j-1} t_q \qquad (2)$$

Evident:

$$t_1 + t_2 + ... + t_{j-1} \leq t_1 + t_2 + ... + t_{j-1} + t_{j+1} + ... + t_n$$

Adica:

$$\sum_{1 \leq q \leq j-1} t_q \leq \sum_{1 \leq q \leq n} t_q - t_j$$

Deci putem inlocui in (2) astfel:

$$load'(M_k) \leq \frac{1}{m}(\sum_{1 \leq q \leq n} t_q - t_j)$$

Stim ca:

$$load'(M_k) = L_k - t_j$$

Deci, inlocuind in (1):

$$L_k \leq t_j + \frac{1}{m}(\sum_{1 \leq q \leq n} t_q - t_j)$$

$$L_k \leq t_j(1 - \frac{1}{m}) + \frac{1}{m} \sum_{1 \leq q \leq n} t_q \qquad (3)$$

Evident:

$$\frac{1}{m} \sum_{1 \leq q \leq n} t_q \leq max(\frac{1}{m} \sum_{1 \leq q \leq n} t_q, t_{max})$$

Iar din lema 1 stim ca:

$$max(\frac{1}{m} \sum_{1 \leq q \leq n} t_q, t_{max}) \leq OPT$$

Deci:

$$\frac{1}{m} \sum_{1 \leq q \leq n} t_q \leq OPT$$

Asa ca putem inlocui in (3):

$$L_k \le t_j(1 - \frac{1}{m}) + OPT \qquad (4)$$

Cum j>m rezulta si activitatile sunt sortate descrescator rezulta:

$$t_j < t_m$$

si

$$t_j \le t_{m+1}$$

De unde rezulta ca:

$$t_j \le \frac{t_m + t_{m+1}}{2} \qquad (5)$$

Din lema 3 stim ca daca n>m (adevarat in cazul nostru, am arata mai sus) si activitatile sunt sortate descrescator in funcie de timp (am facut asta in preprocesare), atunci:

$$t_m + t_{m+1} \le OPT$$

Inlocuind in (5), avem:

$$t_j \le \frac{OPT}{2}$$

Inlocuim acum in (4):

$$L_k \le \frac{OPT}{2}(1 - \frac{1}{m}) + OPT$$

Adica:

$$L_k \le OPT(\frac{3}{2} - \frac{1}{2m})$$

Din (0) rezulta:

$$ALG \le OPT(\frac{3}{2} - \frac{1}{2m})$$

Evident:

$$ALG \ge OPT$$

pentru ca altfel optimul nu ar fi intr-adevar optim.
Deci:

$$OPT \le ALG \le OPT(\frac{3}{2} - \frac{1}{2m})$$