1. Home Exercises - Efficient Algorithms

Samstag, 19. Oktober 2024 16:00

Ex. 1

a) $m = 2 \quad t = [6, 5, 4, 3, 3]$

$f_A = [1, 2, 2, 1, 1]$

$\Rightarrow L_{f_A}(1) = 12$

$L_{f_A}(2) = 9$

$f_{OPT} = [1, 2, 1, 2, 2]$

$\Rightarrow L_{f_{OPT}}(1) = 10$

$L_{f_{OPT}}(2) = 11$

b) given: $n \geq m+1$

to show: $\text{makespan}(f) \geq 2 \cdot t(m+1)$

solution:

Let $L_m(i)$ denote the Load of machine i after job m gets assigned. Then holds

$$\text{makespan}(f) \geq \min_{i \in \{1,\dots,m\}} L_m(i) + t(m+1).$$

If $n = m+1$, then there would be equality. After assigning m jobs to m machines, every machine got exactly one job assigned. Because of the descending order of the jobs hold

$$\min_{i \in \{1,\dots,m\}} L_m(i) = t(m)$$

and

$$t(m) \geq t(m+1).$$

As a result we get

$$\text{makespan}(f) \geq \min_{i \in \{1,\dots,m\}} L_m(i) + t(m+1)$$

$$= t(m) + t(m+1) \geq t(m+1) + t(m+1)$$

$$= 2 \cdot t(m+1) \quad \square$$

c)

Case 1 ($n \leq m$): One job per machine

$\Rightarrow$ $\text{makespan}(f_A) = \text{makespan}(f_{OPT}) = t^*$

Case 2 ($n > m$):
From the lecture, we know that

$\text{makespan}(f_A) \leq \frac{I}{m} + t(j_{max})$ and

$\text{makespan}(f) \geq \frac{I}{m}$ hold.

From b), we know that

$\text{makespan}(f) \geq 2 \cdot t(m+1)$

$\Leftrightarrow t(m+1) \leq \frac{\text{makespan}(f)}{2}$ holds.

Because of the descending order
$t(j_{max}) \leq t(m+1)$ holds for
$n \geq m+1$.

Therfore

$$\text{makespan}(f_A) \leq \frac{I}{m} + t(j_{max})$$

$$\leq \frac{I}{m} + t(m+1)$$

$$\leq \text{makespan}(f_{OPT}) + \frac{\text{makespan}(f_{OPT})}{2}$$

$$= \frac{3}{2}\,\text{makespan}(f_{OPT})$$

$$\Leftrightarrow \frac{\text{makespan}(f_A)}{\text{makespan}(f_{OPT})} \leq \frac{3}{2}$$

Ex 2.

$S = \{1,2,3,4,5,6,7,8,9,10\}$

$T = \{ T_1 = \{1,2,3\},$
$T_2 = \{1,3,4\},$
$T_3 = \{2,3,6,9\},$
$T_4 = \{6,7,8\},$
$T_5 = \{9\},$
$T_6 = \{9,10\},$
$T_7 = \{5,9,10\},$
$T_8 = \{1,5,7\} \}$

$n = 8$

$m = 10$

Init

$C = \{\}$

1. Iteration

Pick T_3 (only set with $|T_i| > 3$)

$C = \{3\}$

$S = \{1, 4, 5, 7, 8, 10\}$

$T = \{ T_1 = \{1\}, T_2 = \{1, 4\}, T_3 = \{\}, T_4 = \{7, 8\}, T_5 = \{\}, T_6 = \{10\}, T_7 = \{5, 10\}, T_8 = \{1, 5, 7\} \}$

2. Iteration

Pick T_8 (only set with $|T_i| > 2$)

$C = \{3, 8\}$

$S = \{4, 8, 10\}$

$T = \{ T_1 = \{\},$

$T_2 = \{ \quad 4\}$,
$T_3 = \{ \quad \}$,
$T_4 = \{ \quad 8\}$,
$T_5 = \{ \quad \}$,
$T_6 = \{ \quad 10\}$,
$T_7 = \{ \quad 10\}$,
$T_8 = \{ \quad \} \}$

3. Iteration

Pick T_2 (tied $|T_i| = 1$ and smallest index)

$C = \{3, 8, 2\}$
$S = \{8, 10\}$
$T = \{ T_1 = \{ \quad \}$,
$T_2 = \{ \quad \}$,
$T_3 = \{ \quad \}$,
$T_4 = \{ \quad 8\}$,
$T_5 = \{ \quad \}$,
$T_6 = \{ \quad 10\}$,
$T_7 = \{ \quad 10\}$,

$T_8 = \{\ \ \ \ \ \ \}\}$

4. Iteration

Pick T_4 (tied $|T_i| = 1$ and smallest index)

$C = \{3, 8, 2, 4\}$

$S = \{10\}$

$T = \{ T_1 = \{\ \ \ \ \ \},$
$T_2 = \{\ \ \ \ \ \},$
$T_3 = \{\ \ \ \ \ \ \ \},$
$T_4 = \{\ \ \ \ \ \ \ \},$
$T_5 = \{\ \ \},$
$T_6 = \{\ \ 10\},$
$T_7 = \{\ \ \ \ \ 10\},$
$T_8 = \{\ \ \ \ \ \ \ \}\}$

5. Iteration

Pick T_6 (tied $|T_i| = 1$ and smallest index)

$C = \{3, 8, 2, 4, 6\}$

$S = \{\}$

$T = \{ T_1 = \{ \}, T_2 = \{ \}, T_3 = \{ \}, T_4 = \{ \}, T_5 = \{ \}, T_6 = \{ \}, T_7 = \{ \}, T_8 = \{ \} \}$

End

Ex. 3

a)

```
GreedyMatching (E)
    C = ∅
    M = ∅
    for every {u, v} ∈ E do
        if u ∈ C or v ∈ C
            continue
        else
```

$C = C \cup \{u, v\}$

$M = M \cup \{\{u, v\}\}$

return M

b)

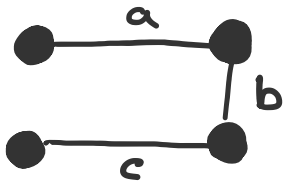
- Initialization of sets: $O(1)$
- Loop iterates over all edges: $O(|E|)$
- Check for intersection by iterating over elements of C: $O(|V|)$

$\Rightarrow O(1) + O(|E|) + O(|V|) = O(|E| + |V|)$

c)

Let M^* denote the best Matching in G and let M_A denote a matching computed by our algorithm.

The worst thing that can happen while performing our algorithm is, that in case there are 4 Vertices in M^* like illustrated here:



In case our algorithm picks edge b, all 4 vertices get "consumed" such that they won't be part of any future edges. M^* would contain a and c. From that observation we can derive that

$$\frac{1}{2}|M^*| \leq |M_A|$$

holds.

That means

$$\frac{|M_A|}{|M^*|} = \frac{APX}{OPT} \geq \frac{1}{2}.$$