DM204 – Spring 2011

Scheduling, Timetabling and Routing

Lecture 7

Timetabling: Reservations and Education

Marco Chiarandini

Department of Mathematics & Computer Science
University of Southern Denmark

Outline

1. Reservations without slack

2. Reservations with slack

Timetabling

- Transport Timetabling
- Personnel/Employee timetabling
  - Crew scheduling
  - Crew rostering
- Reservations
- Educational Timetabling
  - School/Class timetabling
  - University/Course timetabling
  - curriculum planning
  - project assignment
- Sports Timetabling
- Communication Timetabling

Outline

1. Reservations without slack

2. Reservations with slack

Reservations without slack
Interval Scheduling

Given:

- m parallel machines (resources)
- n activities
- r_j starting times (integers),
  d_j termination (integers),
  w_j or w_{ij} weight,
  M_j eligibility
- without slack $p_j = d_j - r_j$

Task: Maximize weight of assigned activities

Examples: Hotel room reservation, Car rental

Polynomially solvable cases

1. $p_j = 1$

Solve an assignment problem at each time slot

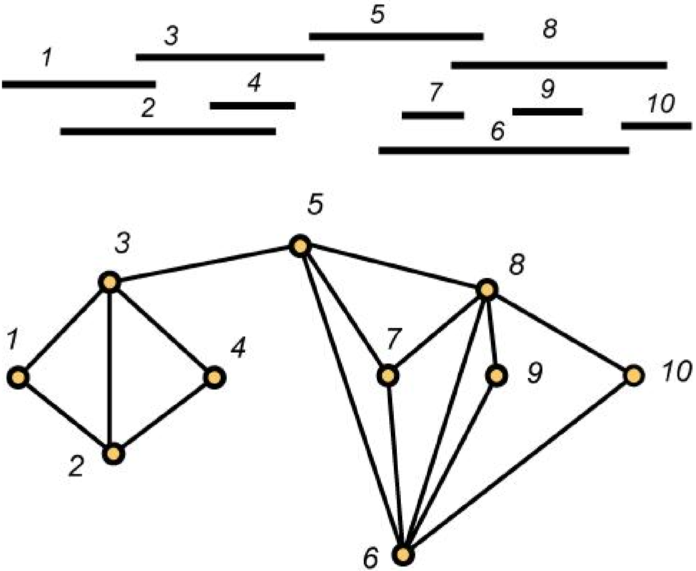
2. $w_j = 1$, $M_j = M$, Obj. minimize resources used

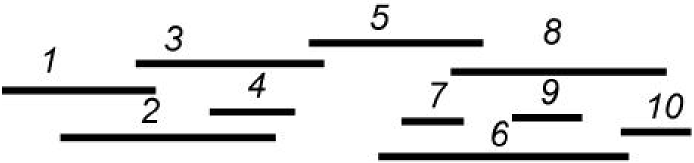
- Corresponds to coloring interval graphs with minimal number of colors
- Optimal greedy algorithm (First Fit):

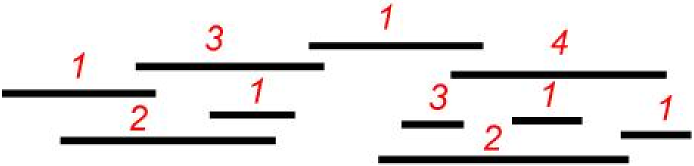
  order $r_1 \leq r_2 \leq \ldots \leq r_n$

Step 1 assign resource 1 to activity 1

Step 2 **for** j from 2 to n **do**
Assume k resources have been used.
Assign activity j to the resource with minimum feasible value from $\{1, \ldots, k+1\}$


5
3
8
1
4
7
9
2
10
6
5
3
8
1
7
4
9
10
2
6


5
3
8
1
4
7
9
2
10
6


1
3
4
1
1
3
1
2
1
2

3. $w_j = 1$, $M_j = M$, Obj. maximize activities assigned

- Corresponds to coloring max # of vertices in interval graphs with k colors
- Optimal k-coloring of interval graphs:

  order $r_1 \leq r_2 \leq \ldots \leq r_n$

  $J = \emptyset$, $j = 1$

Step 1 if a resource is available at time r_j then assign activity j to that resource;
include j in J; go to Step 3

Step 2 Else, select j^* such that $C_{j^*} = \max_{j \in J} C_j$
if $C_j = r_j + p_j > C_{j^*}$ go to Step 3
else remove j^* from J, assign j in J

Step 3 **if** $j = n$ STOP **else** $j = j + 1$ go to Step 1

Outline

Reservations with Slack

Given:

- m parallel machines (resources)
- n activities
- r_j starting times (integers),
  d_j termination (integers),
  w_j or w_{ij} weight,
  M_j eligibility
- with slack $p_j \leq d_j - r_j$

Task: Maximize weight of assigned activities

Heuristics

Most constrained variable, least constraining value heuristic

- $|M_j|$ indicates how much constrained an activity is
- ν_{it}: # activities that can be assigned to i in $[t-1, t]$

Select activity j with smallest $I_j = f\left(\frac{w_j}{p_j}, |M_j|\right)$

Select resource i with smallest $g(\nu_{i,t+1}, \ldots, \nu_{i,t+p_j})$ (or discard j if no place free for j)

Examples for f and g:

$$f\left(\frac{w_j}{p_j}, |M_j|\right) = \frac{|M_j|}{w_j/p_j}$$

$$g(\nu_{i,t+1}, \ldots, \nu_{i,t+p_j}) = \max(\nu_{i,t+1}, \ldots, \nu_{i,t+p_j})$$

$$g(\nu_{i,t+1}, \ldots, \nu_{i,t+p_j}) = \sum_{l=1}^{p_j} \frac{\nu_{i,t+l}}{p_j}$$