Nov 15

# Weighted Interval Scheduling

**Input:** $n$ jobs/intervals : $i^{th}$ interval $(s_i, f_i, v_i)$

start time, finish time, val

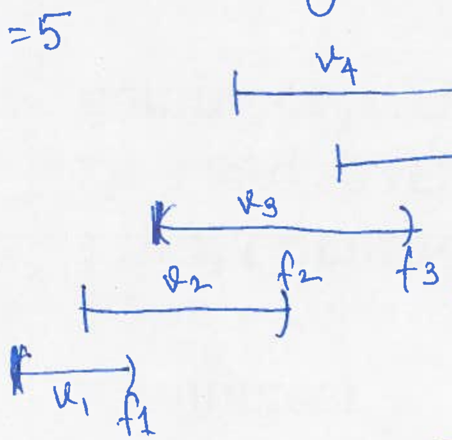
**Output:** A schedule $S \subseteq [n]$ (no conflict) that maximizes $v(S) = \sum_{i \in S} v_i$

Recall: $[s_i, f_i)$

---

**Interval scheduling:** $v_i = 1 \quad \forall i$

Ex: $n = 5$




$[m] \overset{def}{=} \{1, .., m\}$

Let $\Theta$ be an optimal schedule for $[5]$

Case 1: $5 \notin \Theta$

$\Rightarrow \Theta \subseteq [4]$

Claim 1: $\Theta$ is an optimal solution to $[4]$

Pf (idea):

(1) Since $\Theta$ does not have any conflicts, $\Theta$ is a schedule for $[4]$

(2) For contradiction assume $\exists$ a schedule $\Theta' \subseteq [4]$ s.t $v(\Theta') > v(\Theta)$

But $\Theta'$ is also a valid schedule for $[5]$ $\Rightarrow$ contradicts the assumption that $\Theta$ is optimal for $[5]$.

---

Case 2: $5 \in \Theta \Rightarrow 3, 4 \notin \Theta$ as 3 and 4 conflict with 5.

Claim: $\Theta \setminus \{5\}$ is an optimal schedule for $[2]$

Pf (idea): (1) By ↓, $\Theta \setminus \{5\} \subseteq [2]$
(2) $\Theta \setminus \{5\}$ has no conflict

$\Rightarrow \Theta \setminus \{5\}$ is a valid schedule for $[2]$

For contradiction assume $\exists$ schedule $\Theta' \subseteq [2]$ s.t $v(\Theta') > v(\Theta \setminus \{5\})$

Note: (i) $\Theta' \cup \{5\}$ is a valid schedule for $[5]$ {as 5 does not conflict with 1 or 2

(ii) $v(\Theta' \cup \{5\}) = v(\Theta') + v_5 > v(\Theta \setminus \{5\}) + v_5 = v(\Theta)$

$\Rightarrow$ contradicts optimality of $\Theta$ for $[5]$.

---

Assume: For now, only care about $v(\Theta)$ [will later see how to compute $\Theta$]

$OPT(i)$ = value of the optimal solution $[i]$

Goal: $OPT(n)$

---

$$OPT(5) = \max\{\underset{5 \notin \Theta}{OPT(4)},\ \underset{5 \in \Theta}{OPT(2) + v_5}\}$$

---

NEXT: Extend this logic to general $n$

Def: $\Theta_j$ be an optimal solution for $[j]$

$OPT(j) = v(\Theta_j)$

---

Case 1: $j \notin \Theta_j \Rightarrow OPT(j) = OPT(j-1)$

Case 2: $j \in \Theta_j$

Assume: $f_1 \le f_2 \le \cdots f_n$

{If not sort in $O(n \log n)$ time

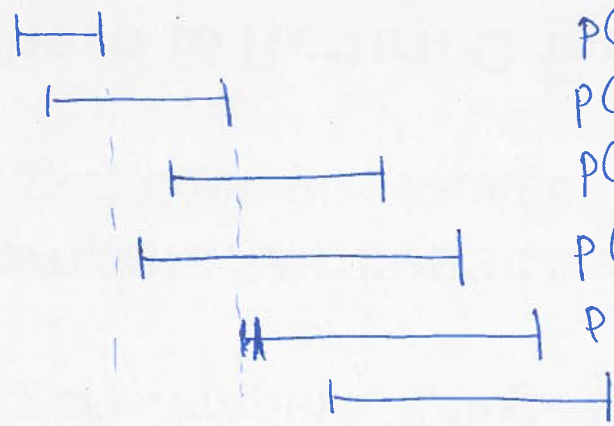
Def: $p(j)$ be the largest $i < j$

s.t. $i$ & $j$ do not conflict [If no such $i$ $\exists$, $p(j) = 0$]

Ex:

$n = 6$

1, 2, 3, 4, 5, 6



$p(1) = 0$

$p(2) = 0$

$p(3) = 1$

$p(4) = 1$

$p(5) = 2$

$p(6) = 2$

Note:

(i) $p(j)+1, \ldots, j-1$ conflict with $j$

(ii) NONE of $1, \ldots, p(j)$ conflict with $j$

$\Rightarrow OPT(j) = v_j + OPT(p(j))$

---

Overall: $OPT(j) = \max\{OPT(j-1),\ OPT(p(j)) + v_j\}$

Base case: ~~OPT~~ $OPT(0) = 0$.