Nov 8

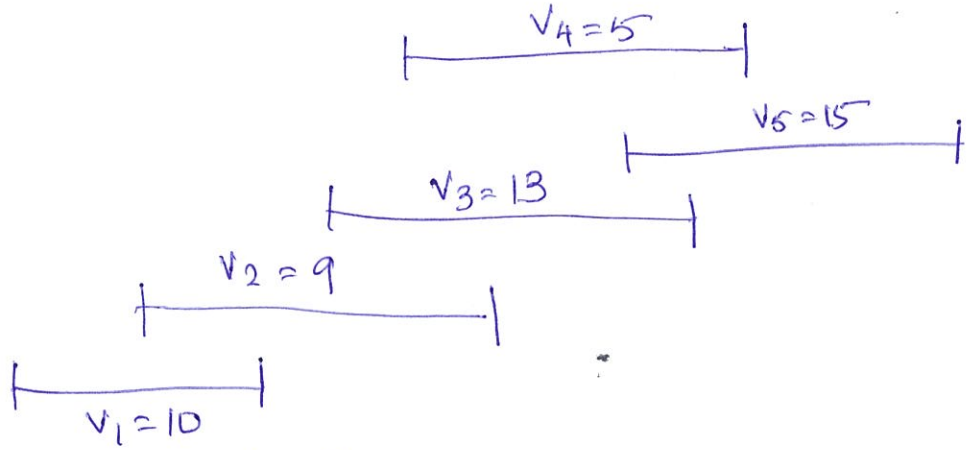
$M[0] = 0$

for $j = 1 \dots n$

$M[j] = \max\{v_j + M[p(j)],\ M[j-1]\}$

$M[0 \dots n]$ have access to $p(1), \dots, p(n)$

$n = 5$



$p(4) = 1$

$p(5) = 2$

$p(3) = 1$

$p(2) = 0$

$p(1) = 0$

	0	1	2	3	4	5
$j=0$	0					
$j=1$	0	10				
$j=2$	0	10	10			
$j=3$	0	10	10	23		
$j=4$	0	10	10	23	23	
$j=5$	0	10	10	23	23	25

p(j)

j-1

$M[0] = 0$

$M[1] = \max\{v_1 + M[0], M[0]\}$
$= \max\{10 + 0, 0\} = 10$

$M[2] = \max\{v_2 + M[0], M[1]\}$
$= \max\{9 + 0, 10\} = 10$

$M[3] = \max\{v_3 + M[1], M[2]\}$
$= \max\{13 + 10, 10\} = 23$

$M[4] = \max\{v_4 + M[1], M[3]\}$
$= \max\{5 + 10, 23\} = 23$

$M[5] = \max\{v_5 + M[2], M[4]\}$
$= \max\{15 + 10, 23\} = 25$

Compute an optimal solution

O_j : an optimal solution for $[j]$

$n = 5$ $\quad 5 \overset{?}{\in} O_5$: $25 > 23 \Rightarrow 5 \in O_5$

$p(5) = 2$: Consider $O_5 \setminus \{5\} = O_2 \subseteq [2]$

$2 \overset{?}{\in} O_2$: $9 < 10 \Rightarrow 2 \notin O_2$

Consider $O_2 = O_1 = [1] \Rightarrow 1 \in O_1$

$\Rightarrow \{1, 5\}$ is an optimal solution

MSchedule $(n; M, p)$

If $n = 0$ return ϕ

If $v_n + M[p(n)] > M[n-1]$

return $\{n\} \cup$ M Schedule $(p(n); M, p)$

else return MSchedule $(n-1; M, p)$.

SUBSET SUM Problem

Input: n integers $w_1, w_2, \dots, w_n$; $w_i > 0$

Budget: $W \geq 0$

Output: A subset $S \subseteq [n]$ s.t

(i) $\sum_{i \in S} w_i \leq W$

(ii) max $w(S) = \sum_{i \in S} w_i$

Ex: $n = 3$ $w_1 = 1$; $w_2 = 3$; $w_3 = 3$

(i) $W = 7 \Rightarrow$ opt soln $\{1, 2, 3\}$

(ii) $W = 6 \Rightarrow$ opt soln $\{2, 3\}$

(iii) $W = 5 \Rightarrow$ opt soln $\{1, 2\}$ or $\{1, 3\}$

Simpler Q: max $|S|$ (instead of $w(S)$)

Greedy algo: Sort in increasing order of w_i & pick as many as you can without exceeding the budget.

Ex: Prove this is optimal (Greedy stays ahead)

Original problem: $w(S)$

Try 1: Greedy algo : counter example $W = 6$
Greedy picks $\{1, 2\}$ but optimal is $\{2, 3\}$

NOTE: no known greedy ~~ds~~ algos