CSL 356 Lecture 12, Aug 22

Optimization Problems

items = n

Each item, say x_i has an associated cost c_i and some profit p_i (some priority associated with x_i)

We are also constrained by some budget, say B.

We want to maximise the sum of p_i's s.t. the total cost $\leq B$.

In general there is an objective function.

Suppose y_i are $\{0, 1\}$ variables representing whether or not we buy the item i

Objective function

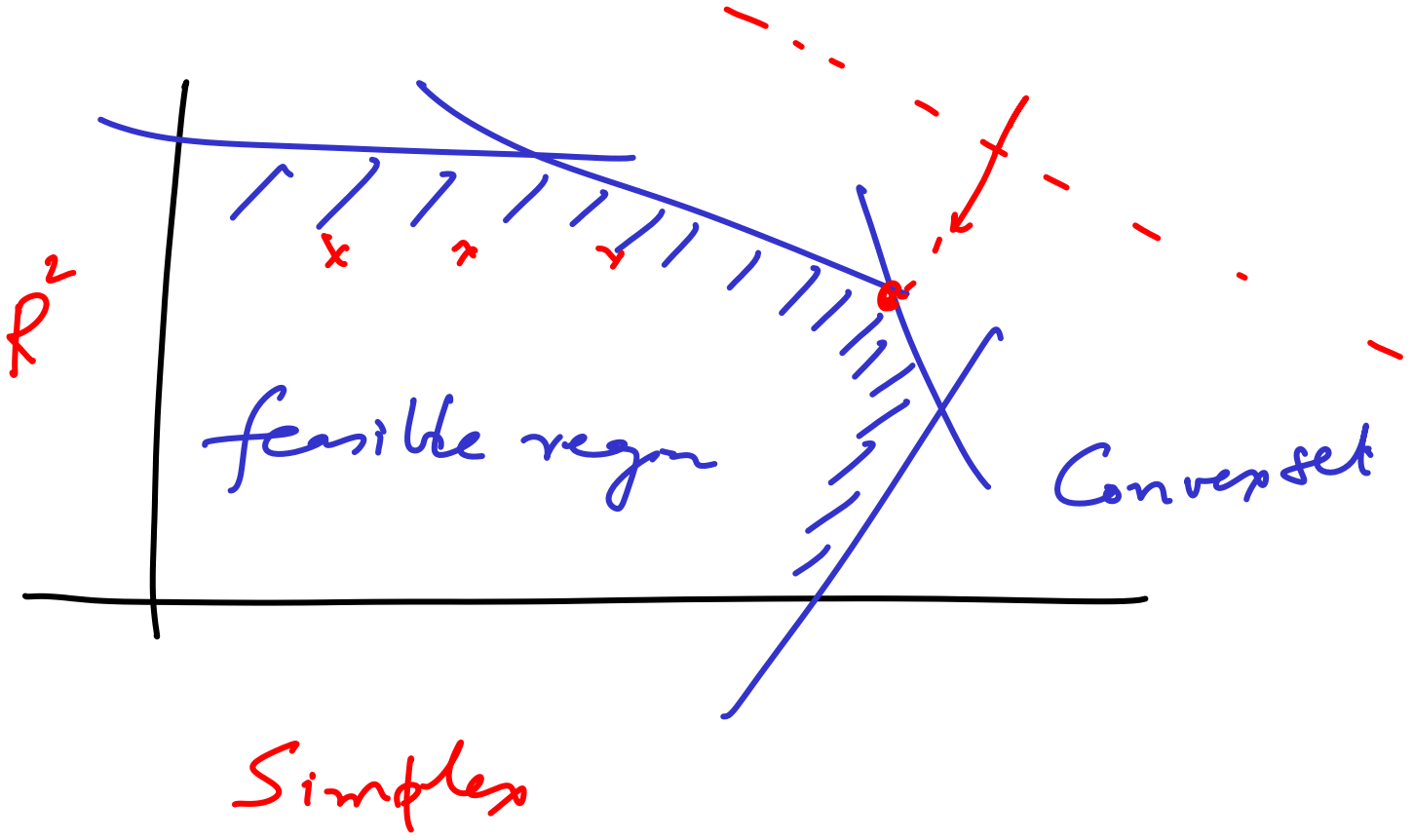
Objective fn

$$\max \sum_i y_i \cdot p_i \qquad y_i \in \{0,1\}$$

$$\text{s.t.} \sum y_i \cdot c_i \leq B \qquad 0 \leq y_i \leq 1$$

constraints

In Linear Programming we have a linear objective function and a set of linear constraints



Simplex

0-1 Knapsack problem is not known to have any polynomial time algorithms

---

Continued Lec 13 Aug 23

B = 15

	1	2	3	4
Profit	10	10	12	17
Cost	2	4	6	9
	5	2.5	2	< 2

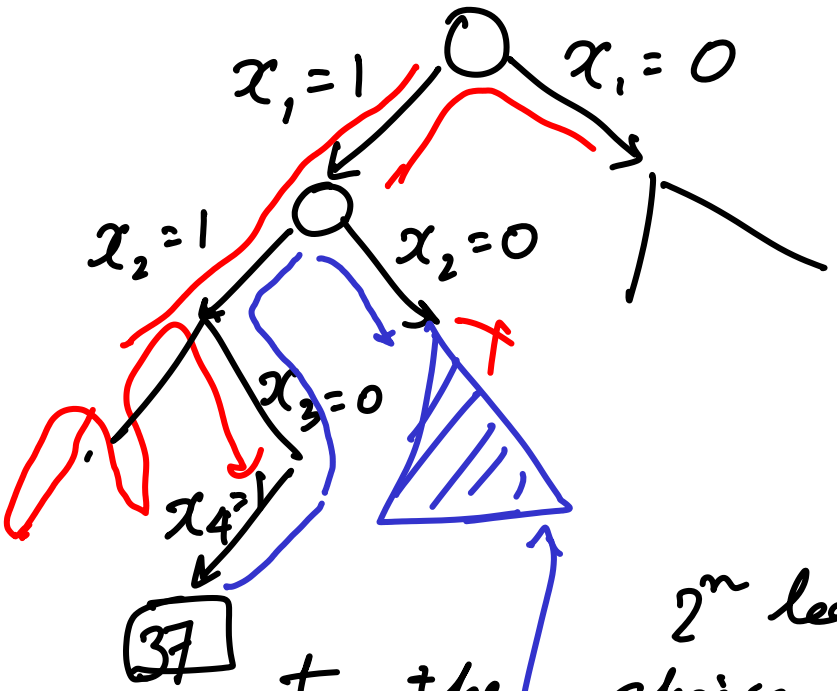
Exhaustive soln will yield

max profit : 10 + 10 + 17
(1, 2, 4)

For n objects, try all possible 2^n choices and for each choice check if feasible (total cost $\leq$ B)

→ Backtracking

→ Branch and Bound



Pruning some choices that won't yield anything better than the current best

2^n leaves corresponding to the choice vector

estimate of the best soln in this subtree

$10 \cdot + (x_3, x_4)$
(x_1)

most profitable : $\max_{x_3, x_4} \frac{\text{profit}}{\text{cost}}$

2 (15 : 13 x2 : 26

Maximum profit from the subtree

= 13 (remaining budget) × 2 (marginal profit)

+ 10 (from including a_1)

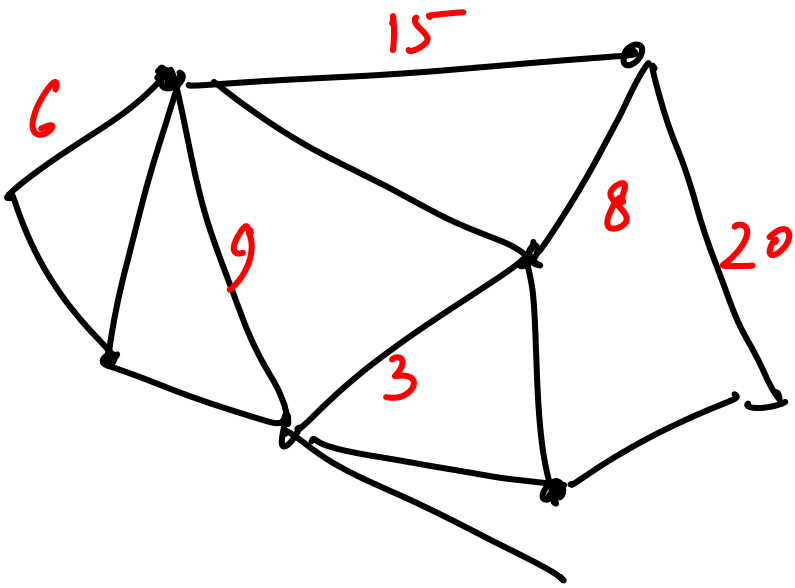
= 36

No provable gains in efficiency by using branch & bound

- still a good heuristic
- depend on the bounding fn.

Greedy : Doesn't work on the current example

(Both according to profits and marginal profit)



Find a subset of edges such that the sum of weights is maximised

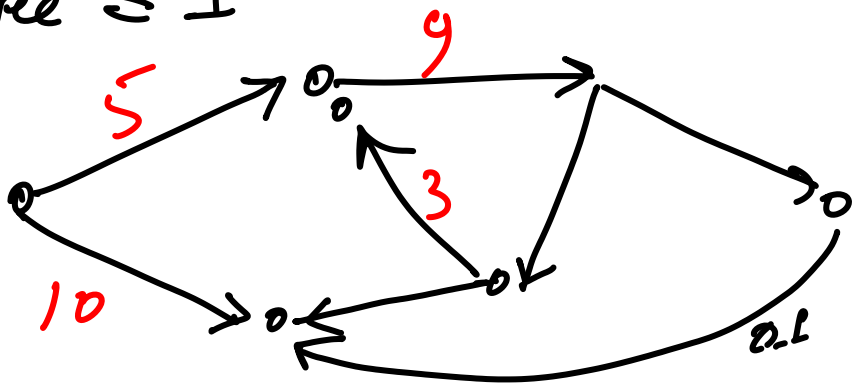
s.t. they don't induce a cycle

(Forest)

↓ it must be connected

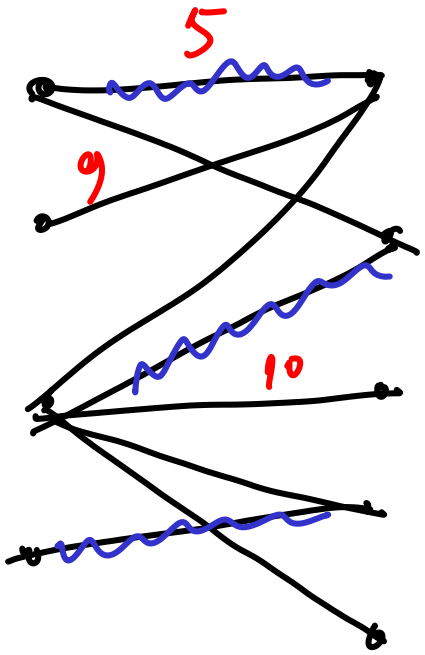
(tree)

Indegree ≤ 1



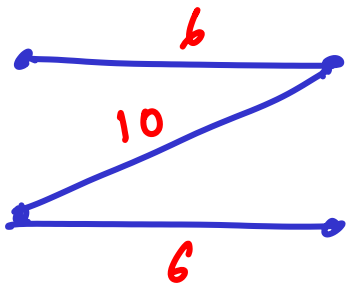
Directed graph

Pick a subset of edges, vertex has ..



Given a bipartite graph (weighted). Pick a subset of edges so that no vertex has degree ≥ 1

Maxm Matching problem



We have a ground set of n elements $e_1, e_2, \ldots e_m$ and let M be a family of subsets that are feasible solns. The elements have weights associated.

Find the maxm weighted subset in M. (M is not necessarily power set)