Lecture 8

Approximation Algos.

Last Time

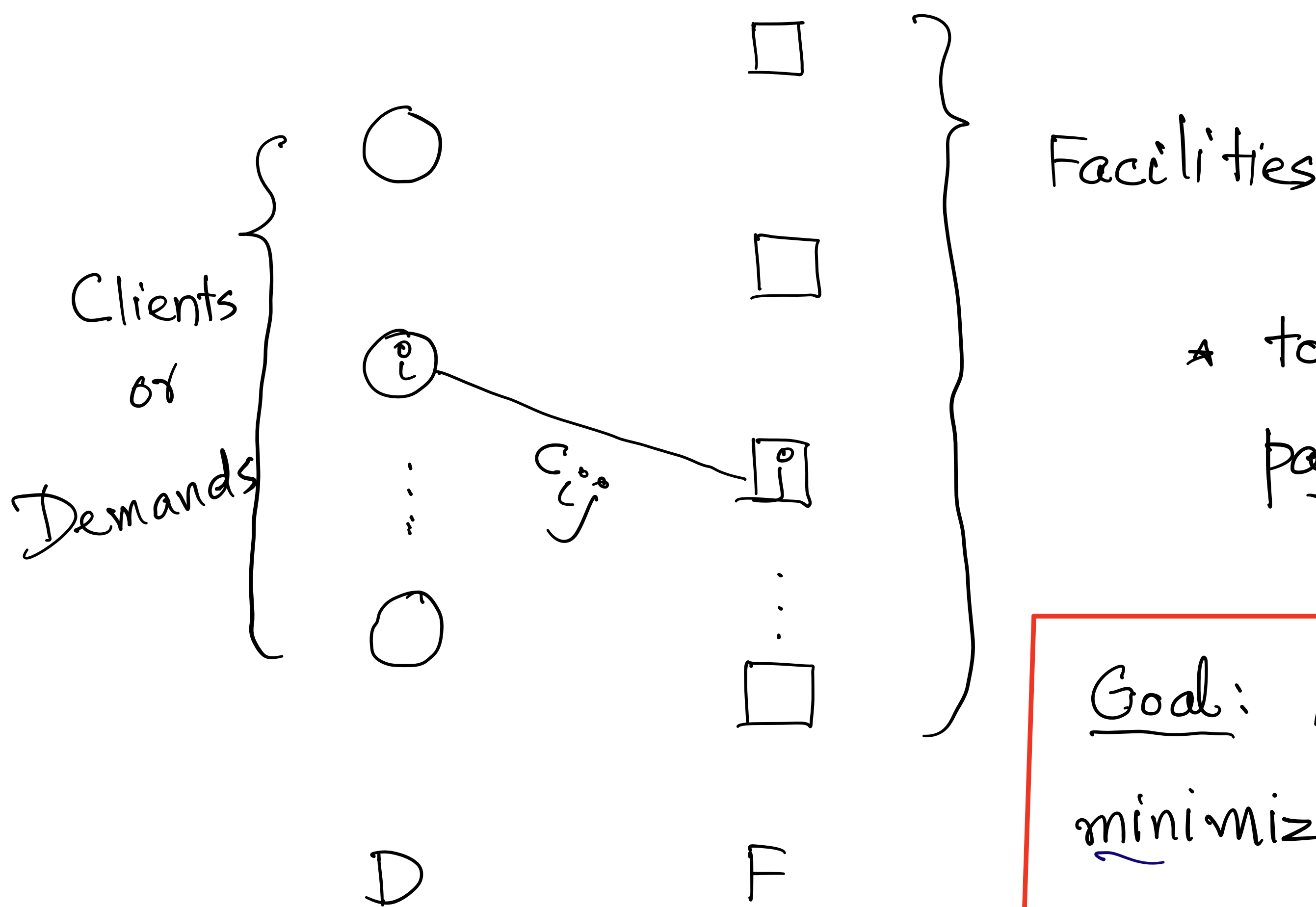
Dynamic Programming for Bin Packing

Today

Deterministic LP rounding

* Uncapacitated Facility Location
* Bin packing

Uncapacitated Facility Location (UFL)



* to assign client j to fac. i, we pay c_{ij}

* to open facility i, we pay a cost f_i

Each client must be assigned to some facility

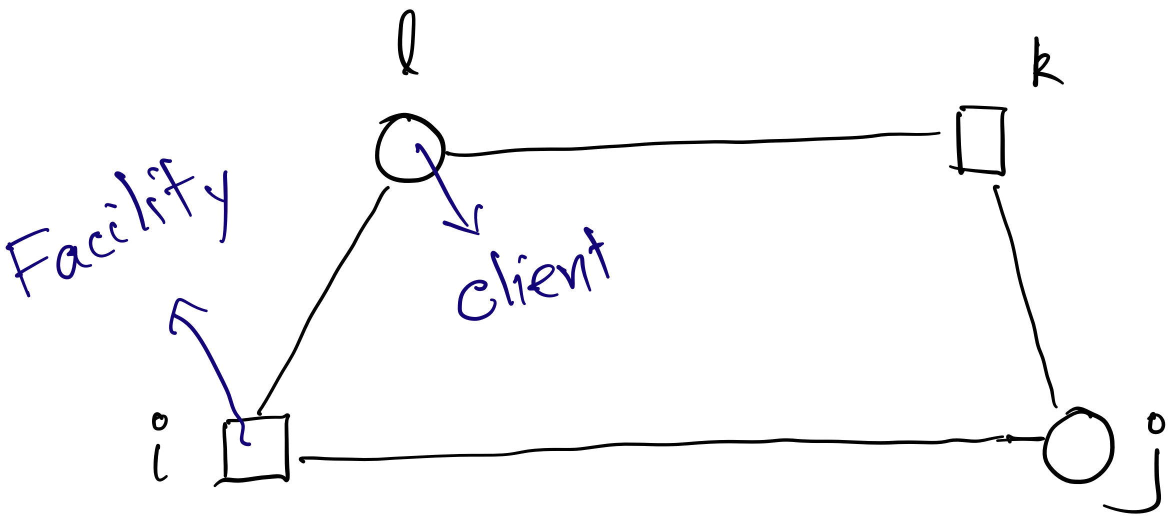
Goal: Find subset $F' \subseteq F$ minimizing

$$\sum_{i \in F'} f_i + \sum_{j \in D} \min_{i \in F'} c_{ij}$$

Today: Metric UFL

$$c_{ij} \leq c_{il} + c_{lk} + c_{kj}$$

* Facilities & clients are points in a metric space
* Distances obey Δ^{le} inequality



4-approx. algo. for metric UFL

Let's write an LP formulation.

$y_i \in \{0,1\}$ for $i \in F$

$x_{ij} \in \{0,1\}$ for $i \in F, j \in D \longrightarrow$ assignment of client j to fac. i

Objective: $\min \overbrace{\sum y_i f_i}^{\text{open. cost}} + \underbrace{\sum x_{ij} c_{ij}}_{\text{assgnmnt. cost}}$

Constraints:

Each client assigned to exactly 1 facility $\quad \sum_{i \in F} x_{ij} = 1 \;\; \forall j \in D$

A client is assigned only to an open facility $\quad \sum_{i \in F} x_{ij} y_i = 1$

$$x_{ij} \leq y_i \qquad \forall i \in F,\ j \in D$$

$$x_{ij} \geq 0 \qquad \forall i, j$$

$$y_i \geq 0 \qquad \forall i$$

Primal Program

$$\min \sum f_i y_i + \sum c_{ij} x_{ij}$$

s.t.

$$\sum_{i \in F} x_{ij} = 1 \quad \forall j \in D$$

$$x_{ij} \leq y_i \quad , \forall i \in F, j \in D$$

$$x_{ij} \geq 0$$

$$y_i \geq 0$$

Dual Program

- Introduce variable v_j $j \in D$
- variable w_{ij} , $i \in F, j \in D$

v_j - price paid by j

w_{ij} - price paid by j to open facility i

$$\max \sum_{j \in D} v_j$$

$$v_j - w_{ij} \leq c_{ij}$$

$$\sum_{j \in D} w_{ij} \leq f_i$$

Dual of the program

$$\max \sum_{j \in D} v_j$$

s.t.

$$\sum_{j \in D} w_{ij} \leq f_i \qquad \forall i \in F$$

$$v_j - w_{ij} \leq c_{ij} \qquad \forall i \in F, j \in D$$

$$v_j , w_{ij} \geq 0$$

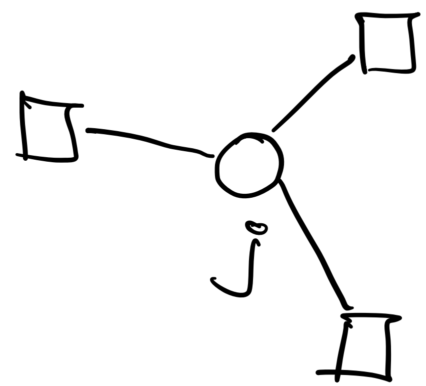
* By weak duality $\nearrow$ opt. value of LP soln.

$$\sum_{j \in D} v_j \leq Z^*_{LP} \leq OPT$$

* (x^*, y^*) - opt. soln. to primal
  (v^*, w^*) - " " " dual

* <u>Defn:</u> Facility i is nbr of client j if $x^*_{ij} > 0$.

  $$N(j) \triangleq \{ i \in F : x^*_{ij} > 0 \}$$



* $N^2(j) \triangleq \{ k \in D : \text{client } k \text{ neighbors some facility in } N(j) \}$ for $j \in D$.

Rounding Algo.

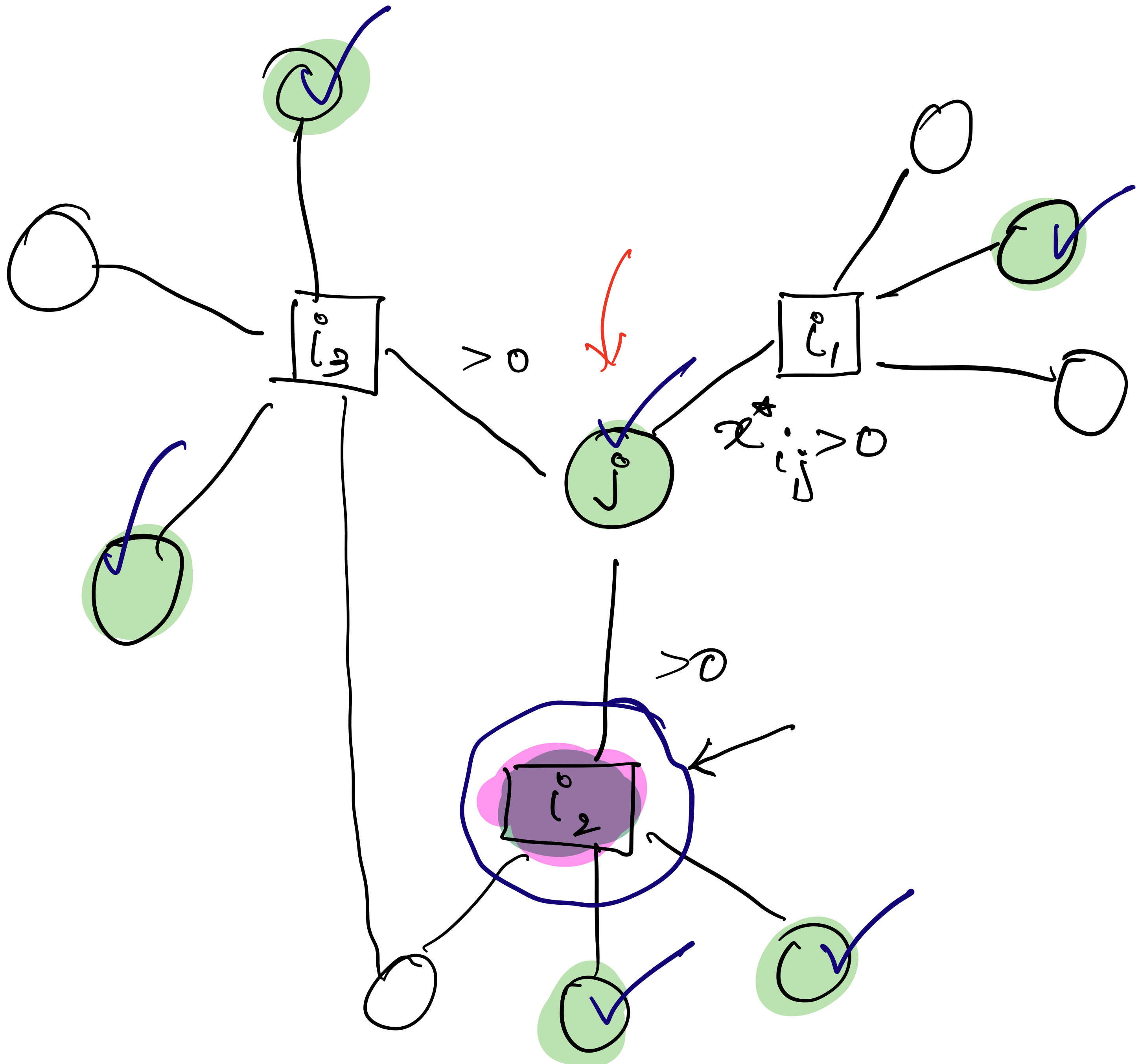
1. Solve LP and get optimal dual soln. (v^*, w^*).

2. $C \leftarrow D$; $k \leftarrow 0$

3. <u>while</u> $C \neq \phi$ <u>do</u>

   (a) $k \leftarrow k+1$ (b) $j_k \leftarrow \arg\min_{j \in C} v_j^*$

   (c) Choose cheapest facility $i_k \in N(j_k)$

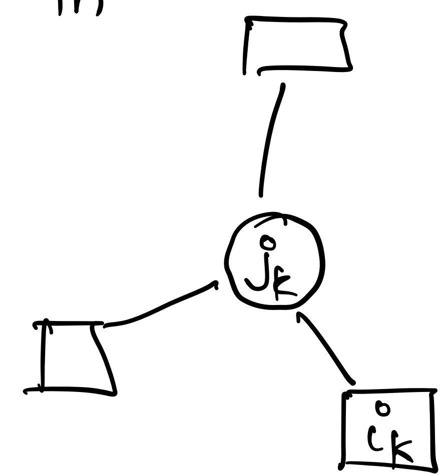


– unassigned clients

$$x^*_{i_1 j} + x^*_{i_2 j} + x^*_{i_3 j} = 1$$

(d) Assign $\mathring{j}_k$ & unassigned clients in $N^2(\mathring{j}_k)$ to $\mathring{i}_k$.

(e) $C \leftarrow C - \{\mathring{j}\} - N^2(\mathring{j}_k)$.



<u>Thm:</u> Algo. is a 4-approximation

| $\mathring{i}_k$ - fac. picked in k^{th}
| $\mathring{j}_k$ - client " " "

<u>Proof:</u>-

$$f_{\mathring{i}_k} = f_{\mathring{i}_k} \cdot \sum_{i \in N(\mathring{j}_k)} x^*_{i\mathring{j}_k} \leq \sum_{i \in N(\mathring{j}_k)} f_i \, x^*_{i\mathring{j}_k} \leq \sum_{i \in N(\mathring{j}_k)} f_i y_i^*$$

$$\downarrow$$

$$f_{\mathring{i}_k} \leq f_i \quad \forall\, i \in N(\mathring{j}_k)$$

$$\sum_{i \in F} x^*_{ij_k} = 1$$

$$\Rightarrow \sum_{i \in N(j_k)} x^*_{ij_k} = 1 \qquad \text{Why?}$$

$$x^*_{ij_k} = 0 \quad \text{if} \quad i \notin N(j_k)$$

$$\sum_{i \in F} f_i y^*_i$$ – facility opening cost of primal optimal soln.

$$\underbrace{\sum_k f_{i_k}}_{\text{Facility Cost}} \leq \sum_k \sum_{i \in N(j_k)} y_i^* f_i \leq \sum_{i \in F} y_i^* f_i \leq Z_{LP}^* \leq OPT$$

$$\boxed{\because \biguplus_k N(j_k) = F}$$

How about assignment cost ?

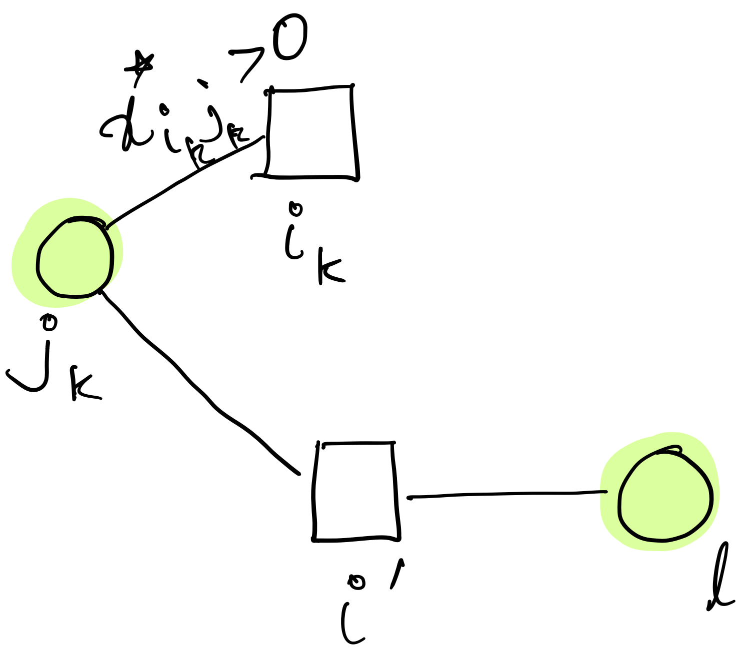
$\star$ If $x_{ij}^* > 0$, then $c_{ij} \leq v_j^*$

Complementary slackness : $x_{ij}^* > 0 \Rightarrow v_j^* - w_{ij}^* = c_{ij}$

$\Rightarrow v_j^* \geq c_{ij}$ since $w_{ij}^* \geq 0$

In iteration k,

consider unassigned client $\ell \in N^2(j_k)$



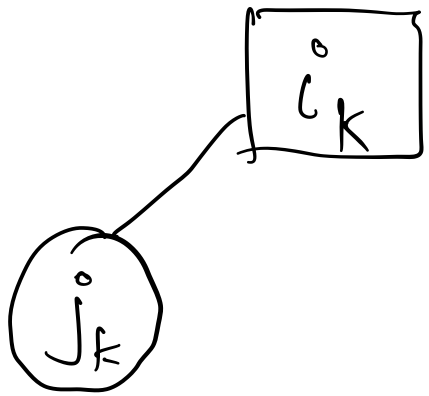
$$\left.\begin{array}{l}\text{Cost of assigning} \\ \ell \text{ to } i_k\end{array}\right\} = c_{\ell, i_k}$$

$$\leq c_{i_k, j_k} + c_{i', j_k} + c_{i', \ell}$$

$$\leq v^*_{j_k} + v^*_{j_k} + v^*_{\ell}$$

$$\leq 3 v^*_{\ell}$$

$$v^*_{\ell} \geq v^*_{j_k}$$



$$\left.\begin{array}{l}\text{Cost of}\\ \text{assigning } j_k\\ \text{to } i_k\end{array}\right\} = c_{i_k, j_k}$$

$$\leq v^*_{j_k}$$

$$\leq 3v^*_{j_k}$$

Total assignment cost $\leq 3 \cdot \sum_{j \in D} v_j^* \leq 3 \cdot Z_{LP}^*$

$\leq 3 OPT$

Facility opening cost $\leq OPT$ (done earlier)

$\therefore$ Total Cost $\leq 4 \cdot OPT$

$\blacksquare$

Bin Packing Problem

* n items, where $a_j \in (0, 1]$ is the size of j^{th} item

* Goal: Assign items to as few bins as possible; each bin has capacity ≤ 1.

$$\boxed{\text{Today: } OPT + O(\log^2 OPT) \text{ algo.}}$$

$(1+\varepsilon) OPT + 1$

Improvement from:

1. Integer programming replacing DP
2. Different grouping scheme — Today.

Firstly

Assume each item has

$$\text{size} \geq \frac{1}{\text{SIZE}(I)}$$

Harmonic Grouping

1. Sort items in descending order of sizes.

groups have total size ≥ 2 < 3

Let n_i be cardinality of G_i

- Drop G_1 & G_r
- From G_i, drop $n_i - n_{i-1}$ items $\quad n_i \geq n_{i-1}$
  & equalize lengths of rem. items

Claim: Let I' be bin-pack i/p obtained by applying harmonic grouping.

$$\#\text{distinct item sizes} \leq \frac{SIZE(I)}{2}$$

$$\left.\begin{array}{l}\text{total size of discarded} \\ \text{pieces}\end{array}\right\} = O(\log SIZE(I))$$

Total size in G_l & G_r $\leq b$

Reading Exercise: Section 4.3

(Ellipsoid Method)

Williamson - Shmoys.