Transportation Problem

Given m "supply" centers with supply quantities a_i; $i=1 \dots m$; & "demand" centers with demand quantities b_j; $j=1 \dots n$; Cost/unit of transporting from supply center i to demand center j: c_{ij} $\begin{matrix} i=1 \dots m \\ j=1 \dots n \end{matrix}$

We want the transportation plan that meets all demands w/o exceeding supply quantities And minimizes total Cost.

Formulation:

Let x_{ij} = quantity shipped from supply center i to demand center j $\begin{matrix} i=1 \dots m \\ j=1 \dots n \end{matrix}$

Constraints: 1) $x_{ij} \geq 0$

Supply side (2) $\sum_{j=1}^{n} x_{ij} \leq a_i \quad i=1 \dots m$

demand side (3) $\sum_{i=1}^{m} x_{ij} \geq b_j \quad j=1 \dots n$

$$\text{Min} \sum_i \sum_j c_{ij} x_{ij}$$

The problem is variously attributed to F.L. Hitchcock, T.C. Koopmans, and L. Kantarovitch.

Lemma 1: A necessary (and sufficient) condition for feasibility is:

$$\sum_{i=1}^{m} a_i \geq \sum_{j=1}^{n} b_j \quad \text{---- (I)}$$

total supply ≥ total demand.

Pf. Left as exercise

Lemma 2. If (I) holds, we can assume without loss, $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$

Pf If $\sum_{i=1}^{m} a_i > \sum_{j=1}^{n} b_j$, introduce a new "fictitious" demand center (n+1) with

$$\begin{cases} b_{n+1} = \sum_{i=1}^{m} a_i - \sum_{j=1}^{n} b_j > 0, \text{ and} \\ c_{i,n+1} = 0 \quad \forall i \end{cases}$$

We make this assumption from now on.

If $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$, then the problem is equivalent to:

$$x_{ij} \geq 0 \qquad \begin{matrix} i = 1 \cdots m \\ j = 1 \cdots n \end{matrix}$$

u_i : $\sum_{j=1}^{n} x_{ij} = a_i \qquad i = 1 \cdots m$ (P)

Unrestd

v_j : $\sum_{i=1}^{m} x_{ij} = b_j \qquad j = 1 \cdots n$

Unrestd

$$\text{Min} \sum_i \sum_j c_{ij} x_{ij}$$

Which is the "Standard" form of the Transportation Problem.

The problem data is represented as: $m = 3,\ n = 4$

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

c_{ij} — a_i — b_j

u_i, v_j unrestd

$$u_i + v_j \leq c_{i,j} \qquad \begin{matrix} i = 1 \cdots m \\ j = 1 \cdots n \end{matrix} \qquad (D)$$

$$\text{Max} \sum_{i=1}^{m} a_i u_i + \sum_{j=1}^{n} b_j v_j$$

Primal-Dual Algorithm:

This applies to problems where it is "easy" to find a feasible solution (D). Using this, we "enforce" Complementary slackness conditions (these hold at all times during the algorithm) and attempt to find a feas. solution to (P).

So. Maintain Dual feas. & Comp. Slackness at all times & work towards Primal feasibility

---

We describe this first with a numerical example that was used before.

One feas solution to (D) is given by:

$$u_i^0 = \min_{j=1}^{n} c_{i,j} \qquad i = 1 \cdots m$$

$$v_j^0 = \min_{i=1}^{m} \left[c_{i,j} - u_i^0 \right] \Rightarrow v_j^0 + u_i^0 \leq c_{ij} \quad \forall i,j.$$

This is our starting Dual feas. solution.

Now to our example

	0	0	3	0	V_j^0	
U_i^0 2	8	(2)	(5)	4	5	
1	(1)	3	7	2	5	a_i
1	2	(1)	10	(1)	5	
	4	3	4	4	15 → $\Sigma a_i = \Sigma b_j$	
	b_j					

Circled cells satisfy $U_i^0 + V_j^0 = C_{ij}$

"Enforcing" Complementary Slackness Condition

Means: For all non-circled entries, $X_{ij} = 0$ ← enforced

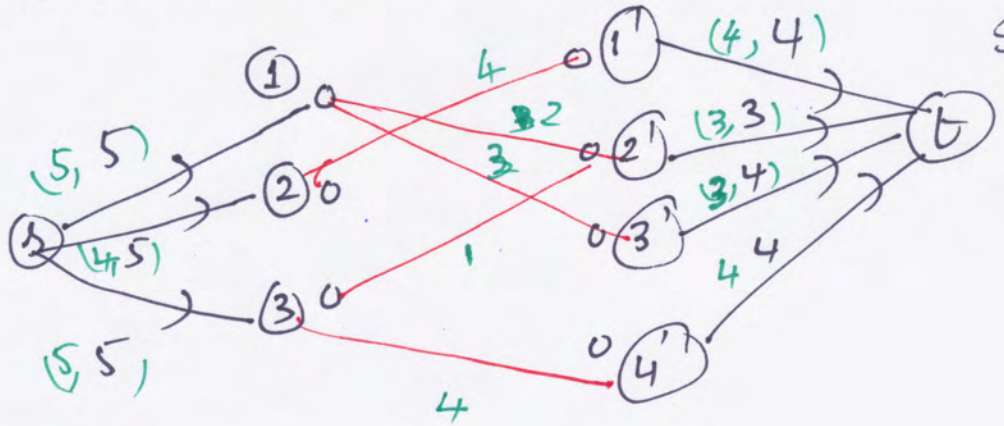
The resulting (P) is called Restricted Primal

We now try to find a feas. solution to (P) using only those X_{ij}'s that are permitted (not forced = 0)

If succes, we have opt. solution to (P) and (D) respectively. If not, we change solution to D (to an improved solution to D) and repeat.

This on the above example. Restricted graph

Solve Max Flow



Mincut: $\{s, ②, ①'\}$ $\{①, ③, ②', ③', ④', t\}$

$I = \{②\}$; $J = \{①'\}$

$\bar{I} = \{③, ①\}$ $\bar{J} = \{②', ③', ④'\}$

Let $\delta = \min_{\substack{i \in I \\ j \in \bar{J}}} [C_{ij} - u_i - v_j] > 0$

$= \min \{3-1-0, 7-1-3, 2-1-0\}$

$= 1$

Change dual solution as follows

$$u_i^{new} = \begin{cases} u_i^{old} & i \notin I \\ u_i^{old} + \delta & i \in I \end{cases}$$

$$v_j^{new} = \begin{cases} v_j^{old} & j \notin J \\ v_j^{old} - \delta & j \in J \end{cases}$$

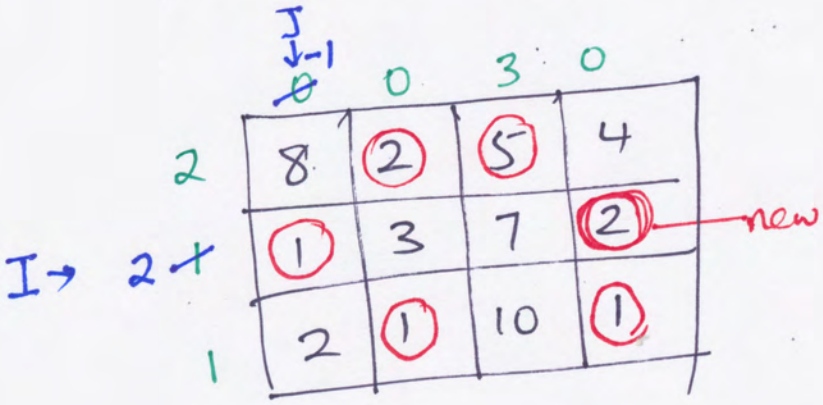
For $i \in I, j \in J$ / $i \in \bar{I}, j \in \bar{J}$: $u_i^{new} + v_j^{new} = u_i^{old} + v_j^{old}$

For $i \notin \bar{I}, j \in J$: $u_i^{new} + v_j^{new} = u_i^{old} + v_j^{old} - \delta \leq C_{ij}$ (since $\delta > 0$)

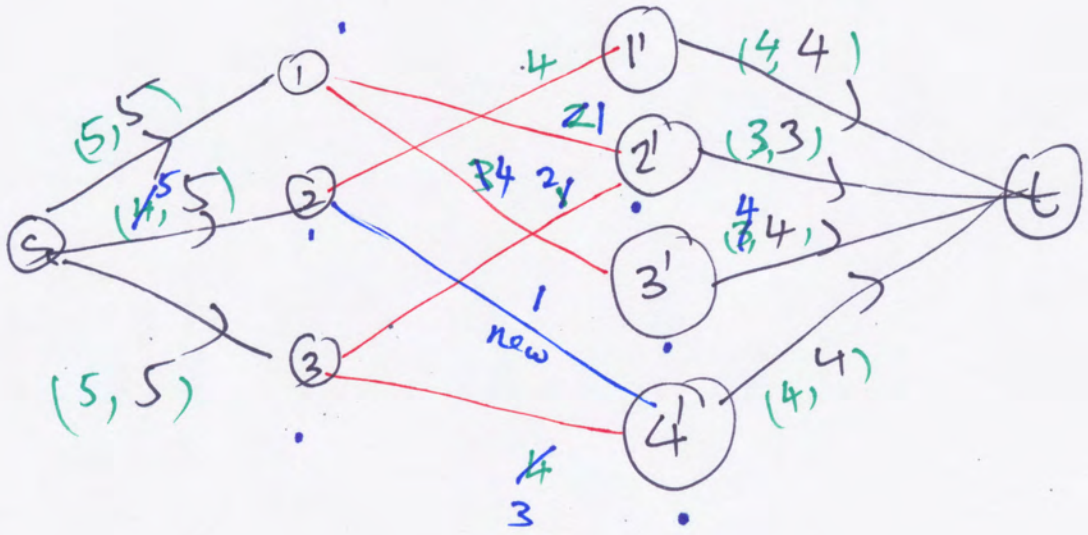
For $i \in I, j \in \bar{J}$: Choice of δ assures us that $u_i^{new} + v_j^{new} \leq C_{ij}$ also holds.

Hence $\{u_i^{new}, v_j^{new}\}$ are feasible to (D).

One new edge of the type $(I, \bar{J})$ enters the restricted primal. An edge of the type ~~$(\bar{I}, \bar{J})$~~ $(\bar{I}, J)$ may depart; but carries no flow (since reverse would be in residual graph if flow > 0). Now we repeat.



old u_i, v_j
new



Optimal X_{ij}

	1	4		5
4			1	5
	2		3	5
4	3	4	4	

If the total flow does not increase, set of reachable nodes increases in size. Hence, algorithm is finite

To make it polynomial we need additional tricks covered under "scaling". To make it strongly polynomial, we need further results. We will ~~come~~ back to these later.

If $a_i = 1$, $b_j = 1$, $\forall i, j$ and $m = n$, "clearly" this algorithm is Strongly polynomial. (Bipartite Matching algorithm). We will later generalize this to general graph matching problem.

Now to the use of duality theory to general Min cost flow problem as well as "primal" Algorithm, etc. in the next lecture.