Weighted Bipartite Matching

Dr. Rajiv Misra
Associate Professor
Dept. of Computer Science & Engg.
Indian Institute of Technology Patna
rajivm@iitp.ac.in

Preface

Recap of Previous Lecture:

- In the previous lecture, we have discussed König-Egerváry theorem, Independent sets, Covers i.e. edge cover, vertex cover, Maximum bipartite matching and Augmenting Path Algorithm.

Content of this Lecture:

- In this lecture, we will discuss Weighted Bipartite Matching, Transversal, Equality subgraph and Hungarian Algorithm.

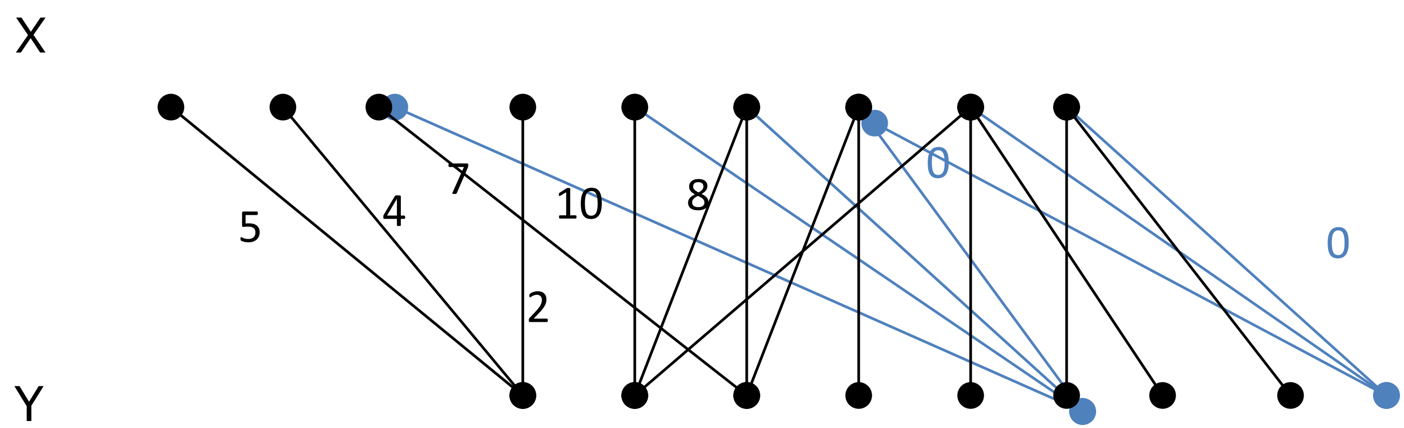
Weighted Bipartite Matching

- The results on maximum matching generalize to **weighted *X*,*Y*-bigraphs**, where we seek a matching of **maximum total weight**. If the graph is not all of $K_{n,n}$, then we can insert the missing edges and assign them weight 0. This does not affect the numbers we can obtain as the weight of a matching. Thus, it is assumed that the **given graph is $K_{n,n}$.**

- Since we consider only nonnegative edge weights, some maximum weighted matching is a perfect matching; thus we seek a perfect matching. We will solve both the maximum weighted matching problem and its dual.

Example

- Matching generalized by giving weights on edges; Seek for maximum weighted matching
- Fill up to $K_{n,n}$ with 0



Example: Weighted bipartite matching and its dual 3.2.5

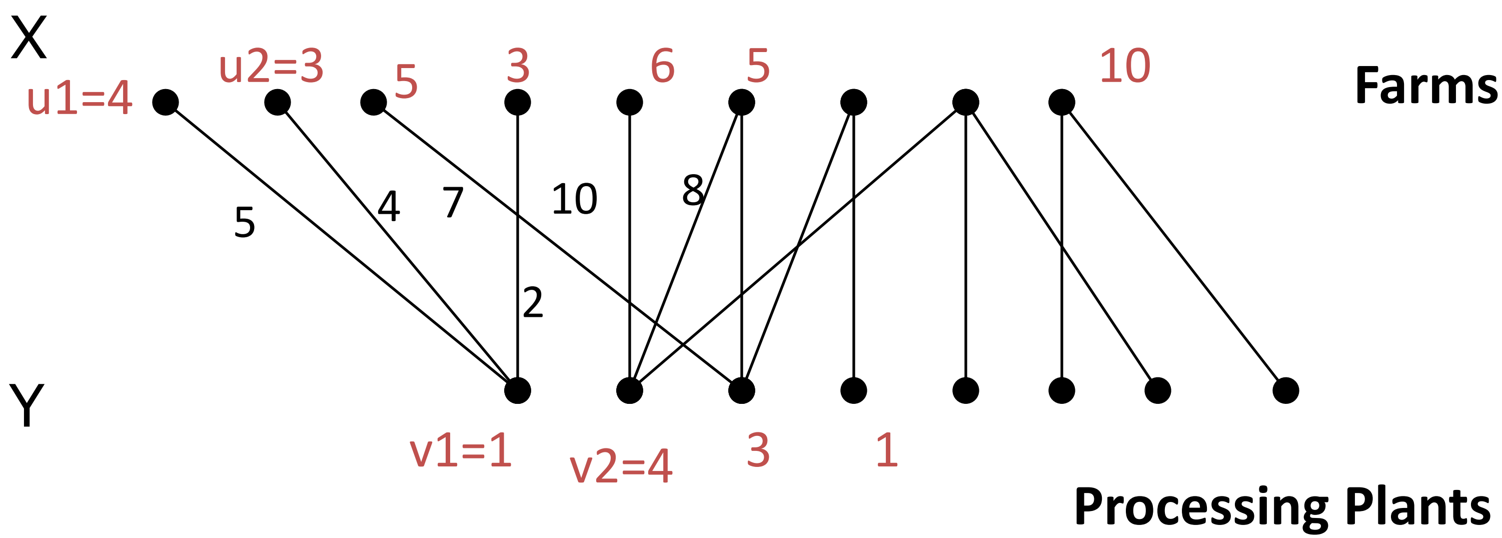
- A farming company owns ***n* farms** and ***n* processing plants**.
  - Each farm can produce corn to the capacity of one plant.
  - The profit that results from sending the output of farm i to plant j is $w_{i,j}$.
  - Placing weight $w_{i,j}$ on edge $x_i y_j$ gives us a weighted bipartite graph with partite sets $X=\{x_1,\ldots,x_n\}$ and $Y=\{y_1,\ldots,y_n\}$.
  - The company wants to select edges forming a matching to **maximize total profit**.

Example: Weighted bipartite matching and its dual continue

- The government claims that too much corn is being produced, so it will pay the company not to process corn.
  - The government will pay u_i if the company agrees not to use farm i and v_j if it agrees not to use plant j.
  - **If $u_i + v_j < w_{i,j}$**, then the company makes more by using the edge $x_i y_j$ than by taking the government payments for those vertices.
  - In order to stop all production, the government must offer amounts such that $u_i + v_j \geq w_{i,j}$ for all i, j. The governments wants to find such values to **minimize $\Sigma u_i + \Sigma v_j$.**

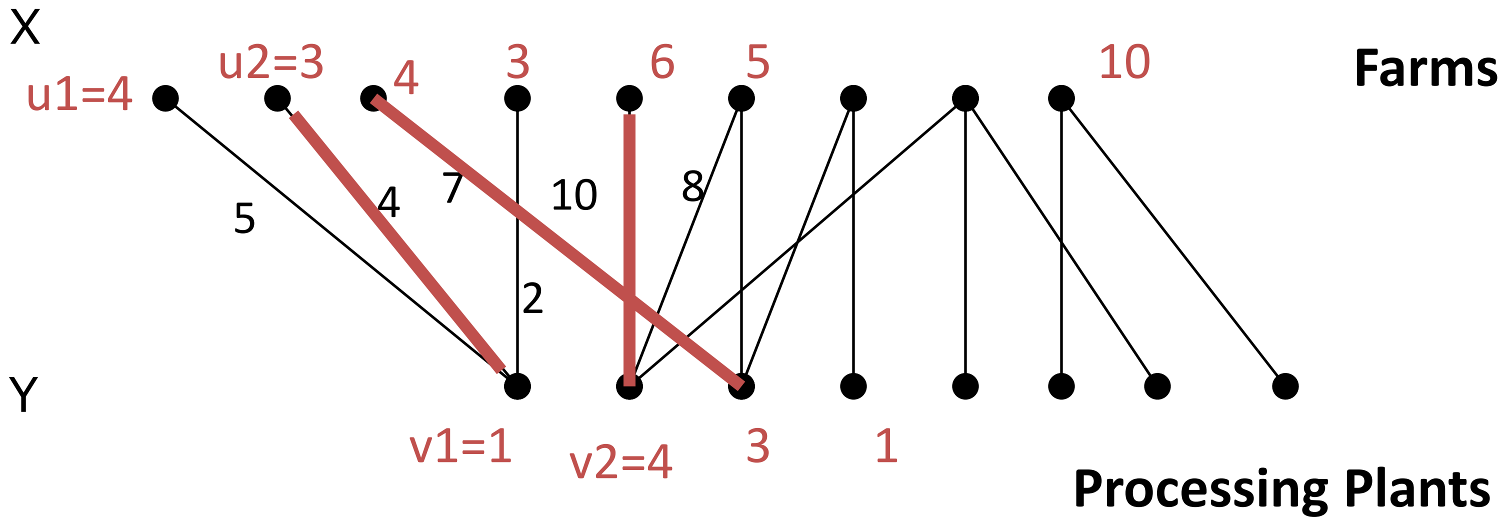
Example:

- **Dual:** farmer and government.
- Weights = profit; Cover = compensation
- $u_i + v_j \geq w_{i,j}$



Example:

- **Farmer:** maximize $\Sigma\ w_{i,j}$
- **Government:** minimize $\Sigma\ (u_i + v_j)$
- Cost(u, v) $v_j \geq w(M)$
- They are dual; Equality holds if M is consist of only $u_i + v_j = w_{ij}$



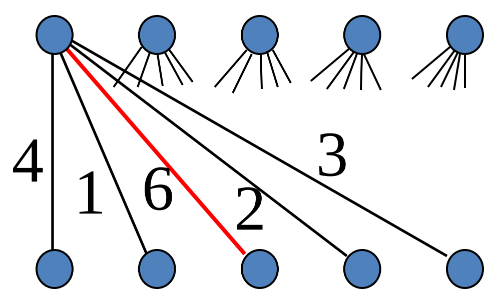
Transversal 3.2.6

- A ***transversal*** of an n-by-n matrix consists of n positions, one in each row and each column.
- Finding a transversal with maximum sum is the **Assignment Problem**.

$$\begin{pmatrix} 4 & 1 & 6 & 2 & 3 \\ 5 & 0 & 3 & 7 & 6 \\ 2 & 3 & 4 & 5 & 8 \\ 3 & 4 & 6 & 3 & 4 \\ 4 & 6 & 5 & 8 & 6 \end{pmatrix}$$

Assignment Problem

- This is the matrix formulation of the **maximum weighted matching** problem, where nonnegative weight $w_{i,j}$ is assigned to edge $x_i y_j$ of $K_{n,n}$ and
- We seek a perfect matching M to maximize the total weight $w(M)$.



$$\begin{pmatrix} 4 & 1 & 6 & 2 & 3 \\ 5 & 0 & 3 & 7 & 6 \\ 2 & 3 & 4 & 5 & 8 \\ 3 & 4 & 6 & 3 & 4 \\ 4 & 6 & 5 & 8 & 6 \end{pmatrix}$$

Minimum weighted Cover

- With these weights, a **(weighted) cover** is a choice of labels $u_i, \ldots, u_n$ and $v_j, \ldots, v_n$ such that $u_i + v_j \geq w_{i,j}$ for all i, j. The **cost $c(u, v)$** for a cover (u, v) is $\Sigma u_i + \Sigma v_j$.
- The **minimum weighted cover** problem is that of finding a cover of minimum cost.

$$\begin{array}{cc} & \begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \end{array} \\ \begin{array}{c} 6 \\ 7 \\ 8 \\ 6 \\ 8 \end{array} & \begin{pmatrix} 4 & 1 & \underline{6} & 2 & 3 \\ 5 & 0 & 3 & \underline{7} & 6 \\ 2 & 3 & 4 & 5 & \underline{8} \\ 3 & 4 & \underline{6} & 3 & 4 \\ 4 & 6 & 5 & \underline{8} & 6 \end{pmatrix} \end{array}$$

Duality of weighted matching and weighted Cover problems

Lemma: For a perfect matching M and *cover*(u, v) in a weighted bipartite graph G, also $c(u, v) = w(M)$ if and only if M consists of edges $x_i y_i$ such that $u_i + v_j = w_{i,j}$. In this case, M and (u,v) are optimal. 3.2.7

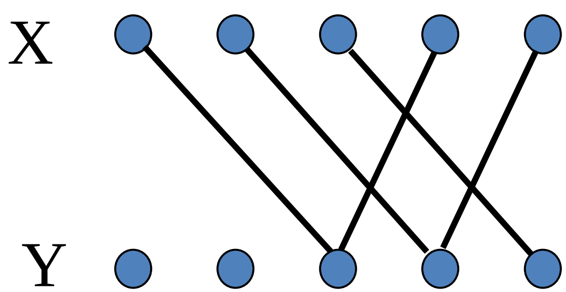
Proof:

- Since M saturates each vertex, summing the constraints $u_i + v_j \geq w_{i,j}$ that arise from its edges yields $c(u, v) = w(M)$, then equality must hold in each of the n inequalities summed.
- Finally, since $c(u, v) \geq w(M)$ for every matching and every cover, $c(u, v) = w(M)$ implies that there is no matching with weight greater than $c(u, v)$ and no cover with cost less than $w(M)$.

Definition: Equality subgraph 3.2.8

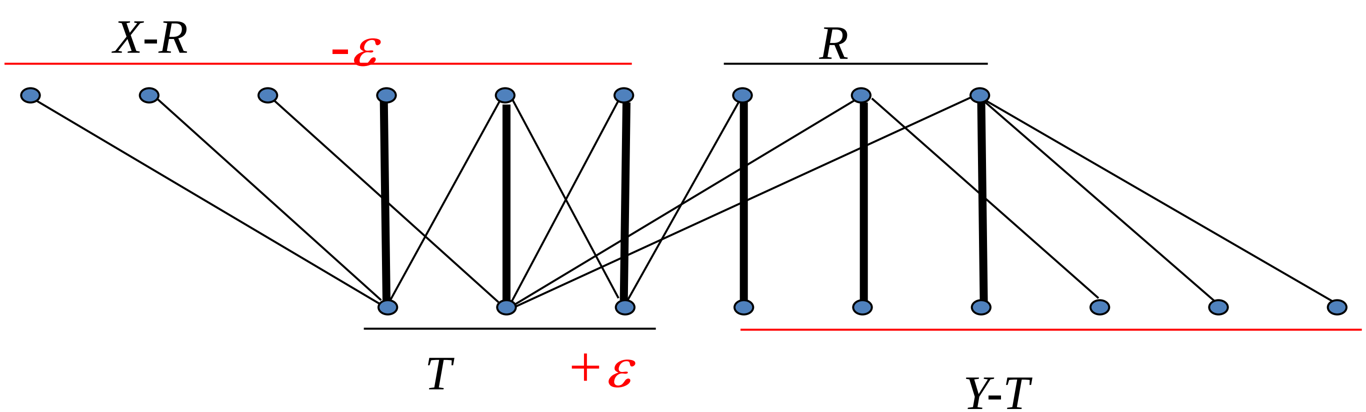
- The ***equality subgraph*** $G_{u,v}$ for a weighted cover (u, v) is the spanning subgraph of $K_{n,n}$ whose edges are the pairs x_iy_j such that $u_i+v_j=w_{i,j}$. In the cover **excess** for i, j is $u_i+v_j-w_{i,j}$

$$\begin{array}{cc} & \begin{array}{ccccc} & & Y & & \\ 0 & 0 & 0 & 0 & 0 \end{array} \\ \begin{array}{r} 6 \\ 7 \\ X\ \ 8 \\ 6 \\ 8 \end{array} & \begin{pmatrix} 4 & 1 & \underline{6} & 2 & 3 \\ 5 & 0 & 3 & \underline{7} & 6 \\ 2 & 3 & 4 & 5 & \underline{8} \\ 3 & 4 & \underline{6} & 3 & 4 \\ 4 & 6 & 5 & \underline{8} & 6 \end{pmatrix} \end{array}$$



Idea for Hungarian Algorithm Continue

- If $G_{u,v}$ has a perfect matching, then its weight is $\Sigma u_i + \Sigma v_j$, and by Lemma 3.2.7 we have the optimal solution.
- Otherwise, we find a matching M and a vertex cover Q of the same size in $G_{u,v}$ (by using the Augmenting Path Algorithm, for example). Let $R = Q \cap X$ and $T = Q \cap Y$. Our matching of size $|Q|$ consists of $|R|$ edges from R to Y-T and $|T|$ edges from T to X-R, as shown below.
  - To seek a larger matching in the equality subgraph, we change (u, v) to introduce an edge from X-R to Y-T while maintaining equality on all edges of M.



Input: A matrix of weights on the edges of $K_{n,n}$ with bipartition X,Y.

Idea: Iteratively adjusting the cover (u, v) until the equality subgraph $G_{u,v}$ has a perfect matching.

Initialization: Let (u, v) be a cover, such as $u_i = \max_j w_{i,j}$ and $v_j = 0$.

Hungarian Algorithm Continue

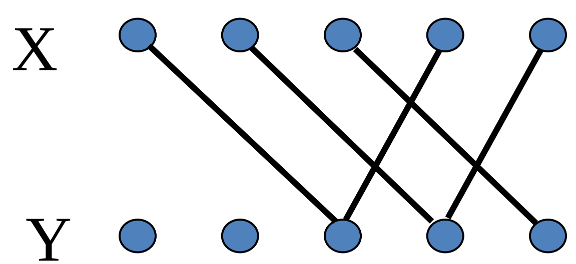
Iteration: Find a maximum matching M in $G_{u,v}$.

- If M is a perfect matching, stop and report M as a maximum weight matching.
- Otherwise,
  - Let Q be a vertex cover of size $|M|$ in $G_{u,v}$.
  - Let $R=X\cap Q$ and $T=Y\cap Q$.
  - Let $\varepsilon=\min\{u_i+v_j-w_{i,j}: x_i \in X-R, y_j \in Y-T\}$.
  - Decrease u_i by ε for $x_i \in X-R$, and
  - Increase v_j by ε for $y_j \in T$.
- Form the new equality subgraph and repeat.

Example

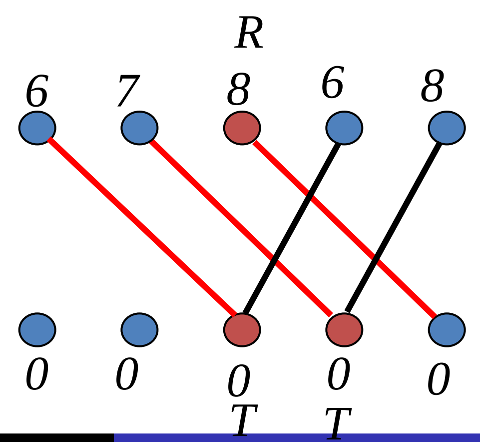
- Edges joining *X-R* and *Y-T* are not in $G_{u,v}$ and have positive excess.

$$
\begin{array}{ccc}
 & & \text{Y} \\
 & & \begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \end{array} \\
\text{X} & \begin{array}{c} 6 \\ 7 \\ 8 \\ 6 \\ 8 \end{array} & \begin{pmatrix} 4 & 1 & \underline{6} & 2 & 3 \\ 5 & 0 & 3 & \underline{7} & 6 \\ 2 & 3 & 4 & 5 & \underline{8} \\ 3 & 4 & \underline{6} & 3 & 4 \\ 4 & 6 & 5 & \underline{8} & 6 \end{pmatrix}
\end{array}
$$



$$
\begin{array}{ccc}
 & & \begin{array}{ccccc} & & T & T & \end{array} \\
 & & \begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \end{array} \\
R & \begin{array}{c} 6 \\ 7 \\ 8 \\ 6 \\ 8 \end{array} & \begin{pmatrix} 4 & 1 & \underline{6} & 2 & 3 \\ 5 & 0 & 3 & \underline{7} & 6 \\ 2 & 3 & 4 & 5 & \underline{8} \\ 3 & 4 & \underline{6} & 3 & 4 \\ 4 & 6 & 5 & \underline{8} & 6 \end{pmatrix}
\end{array}
$$

R∪T: Vertex Cover



Contd...

- Let ε be the minimum excess on the edges from $X\text{-}R$ to $Y\text{-}T$.
- Reducing u_i by ε for all $x_i \in X\text{-}R$ maintains the cover condition for these edges while bringing at least one into the equality subgraph.
- To maintain the cover condition for the edges from $X\text{-}R$ to T increase v_j by ε for $y_j \in T$

$$\begin{array}{ccc}
 & & \begin{matrix} & & T & T & \\ 0 & 0 & 0 & 0 & 0 \end{matrix} \\
\begin{matrix} \\ \\ R \\ \\ \\ \end{matrix} & \begin{matrix} 6 \\ 7 \\ 8 \\ 6 \\ 8 \end{matrix} & \begin{pmatrix} 4 & 1 & \underline{6} & 2 & 3 \\ 5 & 0 & 3 & \underline{7} & 6 \\ 2 & 3 & 4 & 5 & \underline{8} \\ 3 & 4 & \underline{6} & 3 & 4 \\ 4 & 6 & 5 & \underline{8} & 6 \end{pmatrix}
\end{array}$$

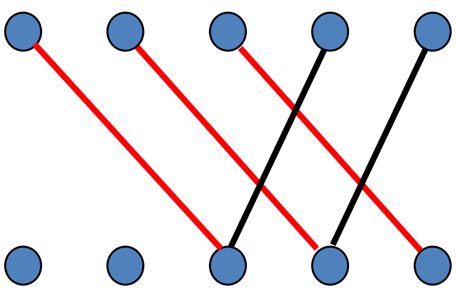
$$\begin{array}{ccc}
 & & \begin{matrix} & & T & T & \\ 0 & 0 & 0 & 0 & 0 \end{matrix} \\
\begin{matrix} \\ \\ R \\ \\ \\ \end{matrix} & \begin{matrix} 6 \\ 7 \\ 8 \\ 6 \\ 8 \end{matrix} & \begin{pmatrix} 2 & 5 & \underline{0} & 4 & 3 \\ 2 & 7 & 4 & \underline{0} & 1 \\ 6 & 5 & 4 & 3 & \underline{0} \\ 3 & 2 & \underline{0} & 3 & 2 \\ 4 & 2 & 3 & \underline{0} & 2 \end{pmatrix}
\end{array}$$

Matrix of excess

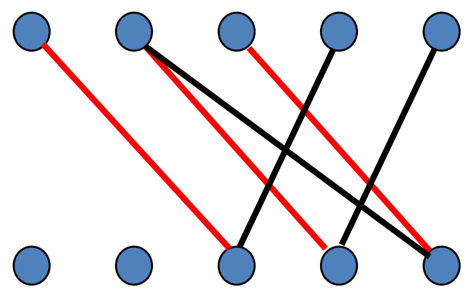
Min excess $\varepsilon = 1$

Contd...

$$
\begin{array}{cc|ccccc|}
 & & & & T & T & \\
 & & 0 & 0 & 0 & 0 & 0 \\
 & 6 & 4 & 1 & \underline{6} & 2 & 3 \\
 & 7 & 5 & 0 & 3 & \underline{7} & 6 \\
R & 8 & 2 & 3 & 4 & 5 & \underline{8} \\
 & 6 & 3 & 4 & \underline{6} & 3 & 4 \\
 & 8 & 4 & 6 & 5 & \underline{8} & 6
\end{array}
$$



$$
\begin{array}{cc|ccccc|}
 & & & & T & T & \\
 & & 0 & 0 & \boxed{1} & \boxed{1} & 0 \\
 & \boxed{5} & 4 & 1 & \underline{6} & 2 & 3 \\
 & \boxed{6} & 5 & 0 & 3 & \underline{7} & \underline{6} \\
R & 8 & 2 & 3 & 4 & 5 & \underline{8} \\
 & \boxed{5} & 3 & 4 & \underline{6} & 3 & 4 \\
 & \boxed{7} & 4 & 6 & 5 & \underline{8} & 6
\end{array}
$$



Equality subgraph is expanded

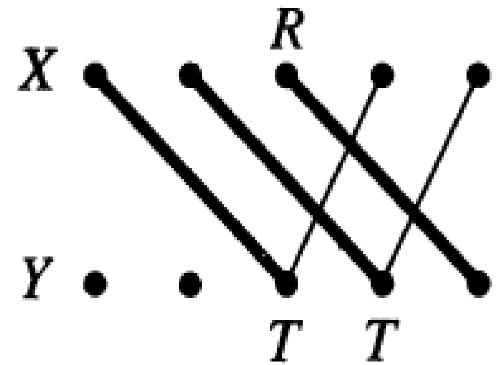
Contd...

- Repeat the procedure with the new equality subgraph; eventually we obtain a cover whose equality subgraph has a perfect matching.

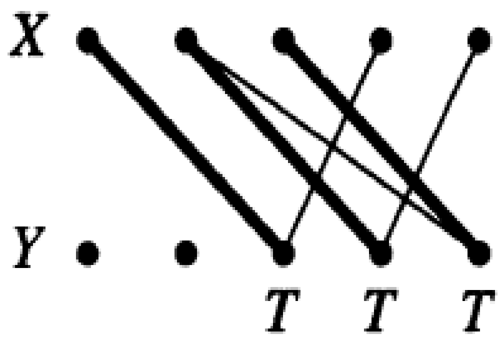
- The resulting algorithm was named the **Hungarian Algorithm** by Kuhn in honor of the work of König-Egerváry on which is based.

Contd...

$$
\begin{pmatrix} 4 & 1 & 6 & 2 & 3 \\ 5 & 0 & 3 & 7 & 6 \\ 2 & 3 & 4 & 5 & 8 \\ 3 & 4 & 6 & 3 & 4 \\ 4 & 6 & 5 & 8 & 6 \end{pmatrix}
\rightarrow
\begin{array}{c c c}
 & \begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \end{array} & \\
\begin{array}{c} 6 \\ 7 \\ 8 \\ 6 \\ 8 \end{array} & \begin{pmatrix} 2 & 5 & \underline{0} & 4 & 3 \\ 2 & 7 & 4 & \underline{0} & 1 \\ 6 & 5 & 4 & 3 & \underline{0} \\ 3 & 2 & 0 & 3 & 2 \\ 4 & 2 & 3 & 0 & 2 \end{pmatrix} & \begin{array}{c} \\ \\ R \\ \\ \\ \end{array} \\
 & \begin{array}{ccccc} & & T & T & \end{array} &
\end{array}
$$



$$
\begin{array}{c c}
 & \begin{array}{ccccc} 0 & 0 & 1 & 1 & 0 \end{array} \\
\begin{array}{c} 5 \\ 6 \\ 8 \\ 5 \\ 7 \end{array} & \begin{pmatrix} 1 & 4 & \underline{0} & 4 & 2 \\ 1 & 6 & 4 & \underline{0} & 0 \\ 6 & 5 & 5 & 4 & \underline{0} \\ 2 & 1 & 0 & 3 & 1 \\ 3 & 1 & 3 & 0 & 1 \end{pmatrix} \\
 & \begin{array}{ccccc} & & T & T & T \end{array}
\end{array}
$$



$$
\rightarrow
\begin{array}{c c}
 & \begin{array}{ccccc} 0 & 0 & 2 & 2 & 1 \end{array} \\
\begin{array}{c} 4 \\ 5 \\ 7 \\ 4 \\ 6 \end{array} & \begin{pmatrix} 0 & 3 & \underline{0} & 4 & 2 \\ \underline{0} & 5 & 4 & 0 & 0 \\ 5 & 4 & 5 & 4 & \underline{0} \\ 1 & \underline{0} & 0 & 3 & 1 \\ 2 & 0 & 3 & \underline{0} & 1 \end{pmatrix}
\end{array}
$$

Example: Weighted bipartite matching and its dual 3.2.5

- A farming company owns ***n* farms** and ***n* processing plants**.
  - Each farm can produce corn to the capacity of one plant.
  - The profit that results from sending the output of farm i to plant j is $w_{i,j}$.
  - Placing weight $w_{i,j}$ on edge $x_i y_j$ gives us a weighted bipartite graph with partite sets $X=\{x_1,\ldots,x_n\}$ and $Y=\{y_1,\ldots,y_n\}$.
  - The company wants to select edges forming a matching to **maximize total profit**.

Example: Weighted bipartite matching and its dual continue

- The government claims that too much corn is being produced, so it will pay the company not to process corn.
  - The government will pay u_i if the company agrees not to use farm i and v_j if it agrees not to use plant j.
  - **If $u_i + v_j < w_{i,j}$**, then the company makes more by using the edge x_iy_j than by taking the government payments for those vertices.
  - In order to stop all production, the government must offer amounts such that $u_i + v_j \geq w_{i,j}$ for all i, j. The governments wants to find such values to **minimize $\Sigma u_i + \Sigma v_j$.**

Hungarian Algorithm: Example

X: n farms

Y: n plants

$$\begin{array}{cc} & \begin{matrix}1 & 2 & 0 & 4 & 3\end{matrix} \\ \begin{matrix}6\\7\\8\\6\\8\end{matrix} & \begin{pmatrix}4 & 1 & 6 & 2 & 3\\5 & 0 & 3 & 7 & 6\\2 & 3 & 4 & 5 & 8\\3 & 4 & 6 & 3 & 4\\4 & 6 & 5 & 8 & 6\end{pmatrix}\end{array}$$

Contd...

- Adjust the cover;until we find a cover $C(u,v) = w(M)$
- Better viewed in Matrix

Contd...

Excess : government over paying.

$$
\begin{array}{c|c}
 & \begin{matrix} 0 & 0 & 0 & 0 & 0 \end{matrix} \\
\begin{matrix} 6 \\ 7 \\ 8 \\ 6 \\ 8 \end{matrix} &
\begin{pmatrix}
4 & 1 & 6 & 2 & 3 \\
5 & 0 & 3_{\mathbf{4}} & 7 & 6 \\
2 & 3 & 4 & 5 & 8 \\
3 & 4 & 6 & 3 & 4 \\
4 & 6 & 5 & 8_{\mathbf{0}} & 6
\end{pmatrix}
\end{array}
$$

Contd...

Excess matrix.

$$
\begin{array}{c c}
 & \begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \end{array} \\
\begin{array}{c} 6 \\ 7 \\ 8 \\ 6 \\ 8 \end{array} & \begin{pmatrix} 2 & 5 & 0 & 4 & 3 \\ 2 & 7 & 4 & 0 & 1 \\ 6 & 5 & 4 & 3 & 0 \\ 3 & 2 & 0 & 3 & 2 \\ 4 & 2 & 3 & 0 & 2 \end{pmatrix}
\end{array}
$$

Contd...

Target: all excess 0, both optimum.

$$\begin{array}{cc} & \begin{array}{ccccc} ? & ? & ? & ? & ? \end{array} \\ \begin{array}{c} ? \\ ? \\ ? \\ ? \\ ? \end{array} & \begin{pmatrix} 4 & 1 & 6 & 0 & 3 \\ 0 & 0 & 3 & 7 & 6 \\ 2 & 3 & 4 & 5 & 0 \\ 3 & 4 & 0 & 3 & 4 \\ 4 & 0 & 5 & 8 & 6 \end{pmatrix} \end{array}$$

Contd…

Equality subgraph S

Maximum matching of S

Vertex cover of S

$$\begin{array}{ccc} & \begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \end{array} & \\ \begin{array}{c} 6 \\ 7 \\ 8 \\ 6 \\ 8 \end{array} & \begin{pmatrix} 2 & 5 & 0 & 4 & 3 \\ 2 & 7 & 4 & 0 & 1 \\ 6 & 5 & 4 & 3 & 0 \\ 3 & 2 & 0 & 3 & 2 \\ 4 & 2 & 3 & 0 & 2 \end{pmatrix} & \begin{array}{c} \\ \\ R \\ \\ \\ \end{array} \\ & \begin{array}{ccccc} & & T & T & \end{array} & \end{array}$$

Contd...

ε : minimum excess

ε= 1

$$
\begin{array}{cc}
 & \begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \end{array} \\
\begin{array}{c} 6\ -1 \\ 7\ -1 \\ 8 \\ 6\ -1 \\ 8\ -1 \end{array} &
\left(\begin{array}{ccccc}
2\ -1 & 5\ -1 & 0\ -1 & 4\ -1 & 3\ -1 \\
2\ -1 & 7\ -1 & 4\ -1 & 0\ -1 & 1\ -1 \\
6 & 5 & 4 & 3 & 0 \\
3\ -1 & 2\ -1 & 0\ -1 & 3\ -1 & 2\ -1 \\
4\ -1 & 2\ -1 & 3\ -1 & 0\ -1 & 2\ -1
\end{array}\right)
\begin{array}{c} -\varepsilon \\ -\varepsilon \\ R \\ -\varepsilon \\ -\varepsilon \end{array} \\
 & \begin{array}{ccccc} & & T & T & \end{array}
\end{array}
$$

Equality subgraph S

Maximum matching of S

Vertex cover of S

Contd…

ε : minimum excess

ε= 1

$$
\begin{array}{ccc}
 & \begin{array}{ccccccc} 0 & 0 & 0 & & 0 & & 0 \end{array} & \\
\begin{array}{c} 5 \\ 6 \\ 8 \\ 5 \\ 7 \end{array} &
\left(\begin{array}{ccccccc}
1 & 4 & -1 & +1 & 3 & +1 & 2 \\
1 & 6 & 3 & +1 & -1 & +1 & 0 \\
6 & 5 & 4 & +1 & 3 & +1 & 0 \\
2 & 1 & -1 & +1 & 2 & +1 & 1 \\
3 & 1 & 2 & +1 & -1 & +1 & 1
\end{array}\right) &
\begin{array}{c} -\varepsilon \\ -\varepsilon \\ R \\ -\varepsilon \\ -\varepsilon \end{array} \\
 & \begin{array}{ccccccc} & & T & & T & & \\ & & +\varepsilon & & +\varepsilon & & \end{array} &
\end{array}
$$

Equality subgraph S

Maximum matching of S

Vertex cover of S

Contd...

ε : minimum excess
$\varepsilon = 1$

One more 0: Equality subgraph BIGGER

Only R and T get $+\varepsilon$, but R and T are saturated

More and More 0 for unsaturated vertex

$$\begin{array}{ccc}
 & \begin{array}{ccccc} 0 & 0 & 1 & 1 & 0 \end{array} & \\
\begin{array}{c} 5 \\ 6 \\ 8 \\ 5 \\ 7 \end{array} &
\begin{pmatrix} 1 & 4 & 0 & 4 & 2 \\ 1 & 6 & 4 & 0 & 0 \\ 6 & 5 & 5 & 4 & 0 \\ 2 & 1 & 0 & 3 & 1 \\ 3 & 1 & 3 & 0 & 1 \end{pmatrix} &
\begin{array}{c} -\varepsilon \\ -\varepsilon \\ R \\ -\varepsilon \\ -\varepsilon \end{array} \\
 & \begin{array}{ccccc} & & T & T & \end{array} & \\
 & \begin{array}{ccccc} & & +\varepsilon & +\varepsilon & \end{array} &
\end{array}$$

Equality subgraph S
Maximum matching of S
Vertex cover of S

Contd...

Next Iteration:

$\varepsilon = 1$

$$
\begin{array}{c c}
 & \begin{array}{ccccc} 0 & 0 & 1 & 1 & 0 \end{array} \\
\begin{array}{c} 5 \\ 6 \\ 8 \\ 5 \\ 7 \end{array} & \left(\begin{array}{ccccc} 1 & 4 & 0 & 4 & 2 \\ 1 & 6 & 4 & 0 & 0 \\ 6 & 5 & 5 & 4 & 0 \\ 2 & 1 & 0 & 3 & 1 \\ 3 & 1 & 3 & 0 & 1 \end{array}\right) \\
 & \begin{array}{ccccc} & & T & T & T \end{array}
\end{array}
$$

Equality subgraph S

Maximum matching of S

Vertex cover of S

Contd...

Result:
Matching and Cover

$$
\begin{array}{c c}
 & \begin{array}{ccccc} 0 & 0 & 2 & 2 & 1 \end{array} \\
\begin{array}{c} 4 \\ 5 \\ 7 \\ 4 \\ 6 \end{array} &
\left(\begin{array}{ccccc}
0 & 3 & 0 & 4 & 2 \\
0 & 5 & 4 & 0 & 0 \\
5 & 4 & 5 & 4 & 0 \\
1 & 0 & 0 & 3 & 1 \\
2 & 0 & 3 & 0 & 1
\end{array} \right)
\end{array}
$$

Equality subgraph S

Maximum matching of S

Vertex cover of S

Theorem: The Hungarian Algorithm finds a maximum weight matching and a minimum cost cover. 3.2.11

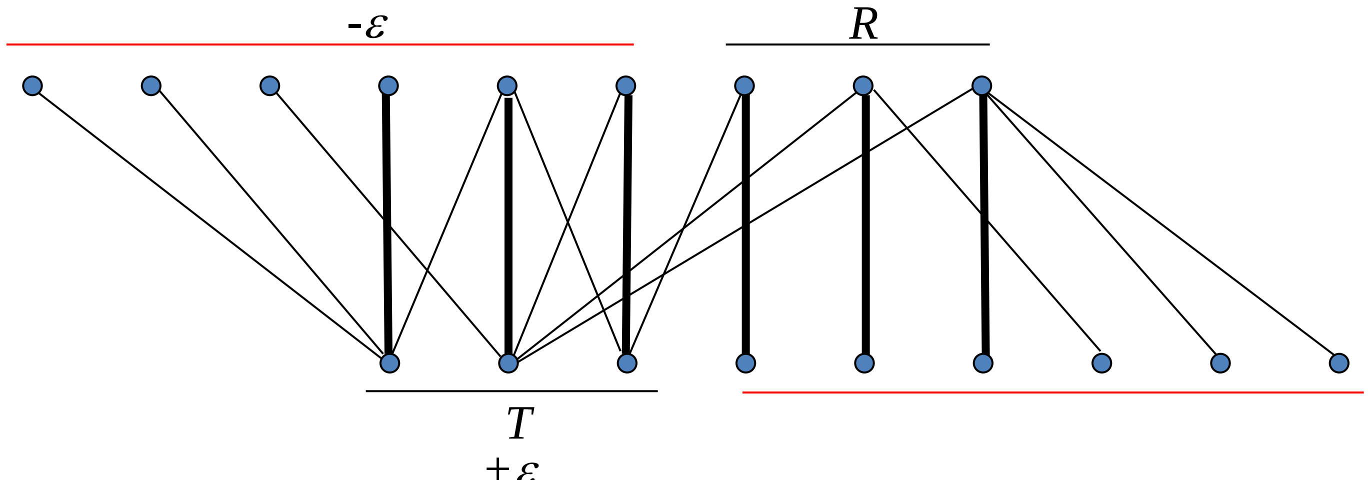
- The algorithm begins with a cover. It can terminate only when the equality subgraph has a perfect matching, which guarantees equal value for the current matching and cover.

- Suppose that (u, v) is the current cover and that the equality subgraph has no perfect matching.

- Let (u', v') denote the new lists of numbers assigned to the vertices. Because ε is the minimum of a nonempty finite set of positive numbers, $\varepsilon>0$.

Theorem 3.2.11 Continue

- We verify first that (u', v') is a cover.
  - The change of labels on vertices of X-R and T yields $u_i'+v_j'=u_i+v_j$ for edges x_iy_j from X-R to T or from R to Y-T.
  - If $x_i \in R$ and $y_j \in T$, then $u_i'+v_j'=u_i+v_j+\varepsilon$, and the weight remains covered.
  - If $x_i \in X$-R and $y_j \in Y$-T, then $u_i'+v_j'$ equals $u_i+v_j-\varepsilon$, which by the choice of ε is at least $w_{i,j}$.



Theorem 3.2.11 Continue

- The algorithm terminates only when the equality subgraph has a perfect matching, so it suffices to show that it does terminate.
- Suppose that the weights $w_{i,j}$ are rational. Multiplying the weights by their least common denominator yields an equivalent problem with integer weights.
- We can now assume that the labels in the current cover also are integers.
- Thus each excess is also an integer, and at each iteration we reduce the cost of the cover by an integer amount.
- Since the cost starts at some value and is bounded bellow by the weight of a perfect matching, after finitely many iterations we have equality.

Conclusion

- In this lecture, we have discussed Weighted Bipartite Matching, Transversal, Equality subgraph and Hungarian Algorithm.

- In upcoming lectures, we will discuss Stable Matchings, Gale-Shapley Algorithm and Faster Bipartite Matching.