**Math 414  Lecture 22**
Not included in Exam 3.

**Assignment Algorithm**

**Input**: an $n \times n$ cost matrix $[c_{ij}]$.

**Output**: a minimum cost assignment $[x_{ij}]$.

**Row step**: From each row, subtract the row minimum.

**Column step**: From each column, subtract the column minimum. Each row and each column will now have a 0 entry.

**Cover steps**.

Loop:
- Find $n$ noncolinear 0’s or find < $n$ lines that cover the 0’s. Use the Hungarian Algorithm to do this.
- If there are $n$ noncolinear 0’s stop, the solution $[x_{ij}]$ with 1’s in these positions is optimal.
- If there are < $n$ lines which cover the 0’s, find the minimal uncovered entry $k$, add it to doubly covered entries, subtract it from uncovered entries. Repeat the loop.

```
\[ c_{ij} = \begin{bmatrix}
1 & 2 & 3 & 4 \\
4 & 3 & 2 & 1 \\
2 & 4 & 6 & 8 \\
5 & 5 & 6 & 6
\end{bmatrix} \rightarrow
\begin{bmatrix}
0 & 1 & 2 & 3 \\
3 & 2 & 1 & 0 \\
0 & 2 & 4 & 6 \\
0 & 0 & 1 & 1
\end{bmatrix} \rightarrow
\begin{bmatrix}
0 & 1 & 1 & 3 \\
0 & 2 & 0 & 0 \\
0 & 2 & 3 & 6 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]
```

Apply the Hungarian algorithm and continue.

```
Solution:
The person-job, $P_i-J_j$, pairings are:
$P_1-\_\_\_, P_2-\_\_\_, P_3-\_\_\_, P_4-\_\_\_.$
The minimum cost is $\Sigma c_{ij}x_{ij} = \$338.6$. Solve the assignment prob. for $[c_{ij}]$ the following matrix.
```

```
Row step: circle the minimum of each row (above), subtract this row min from each row (below).

\[
\begin{bmatrix}
6 & 4 & 1 & 4 & 0 \\
0 & 8 & 3 & 5 & 8 \\
4 & 1 & 0 & 6 & 7 \\
5 & 6 & 7 & 0 & 4 \\
0 & 7 & 3 & 5 & 1
\end{bmatrix}
\rightarrow
\begin{bmatrix}
6 & 3 & 1 & 4 & 0 \\
0 & 7 & 3 & 5 & 8 \\
4 & 0 & 0 & 6 & 7 \\
5 & 5 & 7 & 0 & 4 \\
0 & 6 & 3 & 5 & 1
\end{bmatrix}
\]
```

```
Column step: circle the minimum of each column (above), subtract this from each column (below).

\[
\begin{bmatrix}
6 & 3 & 1 & 4 & 0 \\
0 & 7 & 3 & 5 & 8 \\
4 & 0 & 0 & 6 & 7 \\
5 & 5 & 7 & 0 & 4 \\
0 & 6 & 3 & 5 & 1
\end{bmatrix}
\rightarrow
\begin{bmatrix}
6 & 3- & 1- & 4 & 0* \\
0* & 7- & 3- & 5 & 8 \\
4 & 0* & 0 & 6+ & 7+ \\
5 & 5- & 7- & 0* & 4 \\
0 & 6- & 3- & 5 & 1
\end{bmatrix}
\]
```

```
Cover step: cross out rows and columns and mark 0*’s via Hungarian algorithm. Circle the minimum uncrossed box. Decrement the uncrossed and increment the doubly crossed by this minimum.

\[
\begin{bmatrix}
6 & 3- & 1- & 4 & 0* \\
0* & 7- & 3- & 5 & 8 \\
4 & 0* & 0 & 6+ & 7+ \\
5 & 5- & 7- & 0* & 4 \\
0 & 6- & 3- & 5 & 1
\end{bmatrix}
\rightarrow
\begin{bmatrix}
6 & 3 & 1 & 4 & 0* \\
0* & 7 & 3 & 5 & 8 \\
4 & 0* & 0 & 6+ & 7+ \\
5 & 5 & 7 & 0* & 4 \\
0 & 6 & 3 & 5 & 1
\end{bmatrix}
\]
```

```
Solution: The person-job pairings are: $P_1-J_3$, $P_2-J_1$, $P_3-J_2$, $P_4-J_4$, $P_5-J_5$.
```
The minimum cost: Now transfer the stars of the previous solution to the original cost matrix \([c_{ij}]\). This gives an optimal assignment.

To get the minimum cost \(\Sigma c_{ij}x_{ij}\), total the starred costs.

\[
\begin{array}{cccc}
9 & 7 & 4^* & 7 & 3 \\
0^* & 8 & 3 & 5 & 8 \\
6 & 3^* & 2 & 8 & 9 \\
5 & 6 & 7 & 0^* & 4 \\
2 & 9 & 5 & 7 & 3^* \\
\end{array}
\]

\[4 + 0 + 3 + 0 + 3 = 10.\]

In the original matrix, we starred the positions of the 0*’s. Then we totaled the costs (with the original \(c_{ij}\)’s in each square).

**Networks and flows**

**Definitions.**

A **directed graph** is a set of nodes (vertices) connected by directed edges (arrows) which go from one node to another different node (no loops allowed, no multiple arrows between a given pair of nodes).

A **network** is a directed graph with a nonnegative number, the capacity, assigned to each edge. This is the maximal amount of material which can flow through the edge. There is exactly one source node which has no edge coming in to it; there is exactly one sink node which has no edge leaving it.

Think of a network as water pipes connected together at nodes. Water flows in the direction of the arrows from the source to the sink.

A **flow** is an assignment of a nonnegative number, the flow along the edge, to each edge of the graph. A flow must satisfy the following restrictions:

- The flow along an edge is \(\leq\) the capacity of the edge.
- For nodes other than the source and sink, the sum of the flows going in = the sum of the flows going out.

**Convention:** List capacities above edges and flows below.

**Lemma.** In any flow, the sum of the flows leaving the source = the sum of the flows entering the sink.

These sums (either one, they are the same) are the **volume** of the flow. A flow of largest possible volume is **maximal**.

- For each network, find a maximal flow and give its volume. For each edge, list the flow below the capacity.

![Network Diagram 1](image1)

**Definition.** A cut is a set of edges such that every path from the source to the sink must cross an edge of the cut. The capacity of a cut is the sum of the capacities of its edges. A cut of minimum capacity is a **minimum** cut.

- Give the capacity of each cut. Note, the second diagonal is down and backwards.

![Network Diagram 2](image2)

**Lemma.** In any network, the volume of any flow is \(\leq\) the capacity of any cut.

**Lemma.** If \(F\) is a flow and \(C\) is a cut and the flow along each edge of the cut = the edge’s capacity, then the flow is maximal.
**Theorem.** For any flow $F$ and cut $C$, if the volume of $F$ = the capacity of $C$ then $F$ is maximal and $C$ is minimal.

**Proof.** The capacity of the minimum cut

- ≤ the capacity of the cut $C$
- = the volume of the flow $F$ (by the hypothesis)
- ≤ the volume of the maximal flow
- ≤ (by the Lemma) the capacity of the minimum cut.

Hence all are equal.

**Corollary.** In any network,

- the volume of the maximal flow
- = the capacity of the minimum cut.

**Proof.** The maximum flow algorithm will show how to find a flow $F$ and a cut $C$ such that the volume of $F$ = the capacity of $C$. By the Theorem, $F$ is maximal and $C$ is minimal. Thus

- the volume of the maximum flow
- = the volume of $F$
- = the capacity of $C$
- = the capacity of the minimum cut.

Suppose the amount of water flowing into the node $x$ at the left (bottom network in the first column) increases by 2.

![Diagram](image1)

How much can the flow to adjacent nodes increase? How much can the flow from adjacent nodes decrease without changing the volume flowing out of $x$?

If the amount flowing into node $x$ increases by 6, how much can the flow from adjacent nodes decrease?

If the amount of water flowing into a node $x$ increases by 5. How much can the flow into adjacent nodes increase? The upper number = the capacity; the lower = the flow.

![Diagram](image2)

We are asking for potential individual increases; these increases are not simultaneously achievable.

*Remaining capacity = edge capacity − edge flow.*

The potential increase at an adjacent node $y$ is:

- $x \rightarrow y$ edges: $\min$(potential increase at $x$, remaining capacity).
- $y \rightarrow x$ edges: $\min$(potential increase at $x$, flow on the edge).