Assignment Problems

Guoming Tang

A Practical Problem

- **Workers**: A, B, C, D **Jobs**: P, Q, R, S.
- **Cost matrix:**

	Job P	Job Q	Job R	Job S
Worker A	1	2	3	4
Worker B	2	4	6	8
Worker C	3	6	9	12
Worker D	4	8	12	16

- **Given**: each worker need perform only one job and each job need be assigned to only one worker.
- **Question:** how to assign the jobs to the workers to minimize the cost?

Assignment Problem

- Teachers vs. Courses
  - Performance
  - Maximize the total performance
- Jobs vs. Machines
  - Time
  - Minimize the total time
- ……

Formulation of Assignment Problem

- **Consider *m workers to whom n jobs are assigned.***
- **The cost of assigning worker *i to job j is* c_{ij}*.***

- **Let**

$$x_{ij} = \begin{cases} 1, & \text{if job } j \text{ is assigned to worker } i \\ 0, & \text{if job } j \text{ is not assigned to worker } i \end{cases}$$

Formulation of Assignment Problem

- **Objective function:**

$$Minimize \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}$$

- **Constraints:**

$$\sum_{i=1}^{n} x_{ij} = 1 \qquad for\ j = 1,2,\dots n$$

$$\sum_{j=1}^{m} x_{ij} = 1 \qquad for\ i = 1,2,\dots m$$

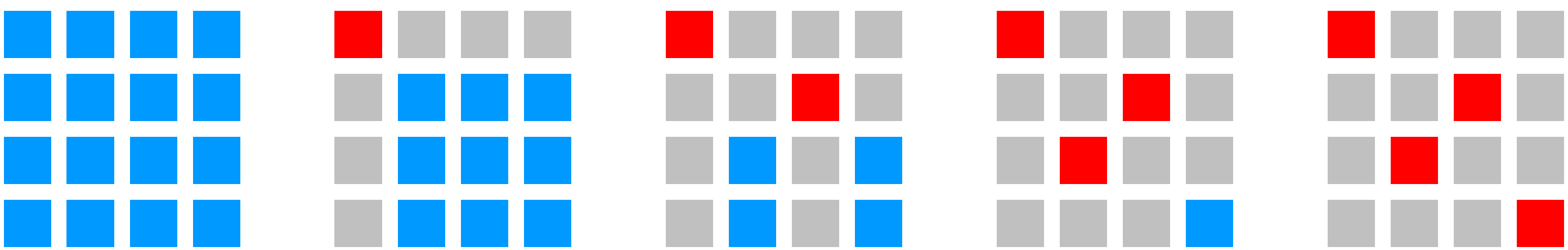
$$x_{ij} = 0\ or\ 1$$

Each worker is assigned one job

Each job is assigned one worker

Solution of Assignment Problem

- Brute-force method
  - Enumerate all candidates sets
  - *n*! possible assignment sets
  - Exponential runtime complexity



$4! = 24$ $10! = 3{,}628{,}800$ $100! \approx 9.3 \times 10^{157}$

Solution of Assignment Problem

- **Simplex method**
  - **Is it feasible to solve AP? Yes.**
  - **[The Integrality Theorem] If a transshipment problem: minimize *cx* subject to *Ax=b*, *x*≥0, such that all the components of b are integers, has at least one feasible solution, then it has an <u>integer-valued feasible solution</u>; if it has an optimal solution, then it has an <u>integer-valued optimal solution</u>.**

Solution of Assignment Problem

- **Simplex method**
  - More variables (an *n* assignments needs n^2 variables.)
  - More slack variables result in a sparse dictionary matrix, which may lead to more iterations.
  - Inefficient.

Solution of Assignment Problem

- **Network simplex method**
  - **Tree based network optimization method**
  - **Can apply to transshipment problem, maximum flows through networks**
  - **Works well in practice for assignment problems.**
  - **Is there any easier way to solve the assignment problem?**

Hungarian Method

Introduction

- A combinatorial optimization algorithm
  - Based on two Hungarian mathematicians' works
  - Developed and published by Harold Kuhn, 1955
- Polynomial runtime complexity
  - [Wikipedia] $O(n^4)$, can be modified to $O(n^3)$
- Much easier to implement

Hungarian Method

Process (1/5)

- **Assume the cost matrix**

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

- **For each row, subtract the minimum number in that row from all numbers in that row; do the same for each column.**
- **[*Network optimization*: Theorem 2.9] There exists an optimal solution, when # of assignments = minimum # of lines required to cover all 0s.**

Hungarian Method

Process(2/5)

- **How can we find the optimal solution?**
- **One efficient way in [*Network optimization*]**
  - **(a) Locate a row/column in c’ with exactly one 0, circle it and draw a vertical/horizontal line through it. If no such row/column exists, locate a row/column with the smallest number of 0s.**
  - **(b) Repeat (a) till every 0 in the matrix has at least one line through it.**
  - **(c) If # of lines equal to n, the circled 0s show the optimal solution.**
- **Apply it to c’.**

Hungarian Method

Process(3/5)

- **Why is it optimal?**

$$c' = \begin{pmatrix} \textcircled{0} & 0 & 2 & 1 \\ 2 & \textcircled{0} & 0 & 1 \\ 0 & 0 & 3 & \textcircled{0} \\ 3 & 2 & \textcircled{0} & 6 \end{pmatrix}$$

- **[*Network optimization*: Theorem 2.17]**
  Suppose *c* is the cost matrix and *c'* is the matrix obtained by adding a number *t* to each element in the *i*th row or to each element in the *i*th column. Then a solution is optimal with respect to *c'* if and only if it is optimal with respect to *c*.

Hungarian Method

Process(4/5)

- **Is that all? No.**

$$c' = \begin{pmatrix} 3 & 0 & 2 & 1 \\ 2 & 4 & 0 & 1 \\ 0 & 0 & 3 & 0 \\ 3 & 2 & 0 & 6 \end{pmatrix}$$

- **No matching can result in 0 total cost.**
- **Remember: (c) If # of lines equals to *n*, we find the optimal solution.**

Then what if # of lines is not equal to *n*?

Hungarian Method

Process(5/5)

- **Modify c' further:**

$$c' = \begin{pmatrix} 3 & 0 & 2 & 1 \\ 2 & 4 & 0 & 1 \\ 0 & 0 & 3 & 0 \\ 3 & 2 & 0 & 6 \end{pmatrix}$$

- **Subtract the smallest uncovered element from each uncovered element and add it to each doubly covered element.**

- **Check the solution.**

$$c'' = \begin{pmatrix} 2 & 0 & 2 & 0 \\ 1 & 4 & 0 & 0 \\ 0 & 1 & 4 & 0 \\ 2 & 2 & 0 & 5 \end{pmatrix}$$

Hungarian Method
the Whole Course

- 1. Given the cost matrix c (n×n), get modified c':
  - (a) For each row, subtract the minimum number in that row from all numbers in that row
  - (b) Do the same for each column.
- 2. Check if there exists an optimal solution:
  - (a) Locate a row/column in modified matrix with exactly one 0, circle it and draw a vertical/horizontal line through it. If no such row/column exists, locate a row/column with the smallest number of 0s.
  - (b) Repeat (a) until every 0 in the matrix has at least one line through it.
  - (c) If # of lines = n, the circled 0s show the optimal solution, end; if not, go to step 3.
- 3. Further modify c':
  - (a) Subtract the smallest uncovered element from each uncovered element and add it to each doubly covered element.
  - (b) Repeat step 2.

Hungarian Method

Special Considerations

- **Special considerations:**
  - **Hungarian method requires # of rows = # of columns. What if not?**
  - **Considering a case like assigning workers to jobs, what if worker *i* cannot do job *j* ?**
  - **Maximization objective, such as maximize the profits.**

- **Solutions:**
  - **Add dummy rows/columns with 0 assignment costs**
  - **Assign $c_{ij} = +M$**
  - **Create a loss matrix**

Hungarian Method
an Example (1/7)

- Teachers: A, B, C, D; Courses: P, Q, R
- Teachers' performance matrix

	P	Q	R
A	90	76	67
B	68	70	69
C	75	72	71
D	69	65	65

- Assign courses to teachers to **maximize** the sums of their performance.

Hungarian Method
an Example (2/7)

- **Transfer Maximum to Minimum:**
  - Create the loss matrix: subtract each score in each column from the highest score in that column.

	P	Q	R
A	0	0	4
B	22	6	2
C	15	4	0
D	21	11	6

→

	P	Q	R	S
A	0	0	4	0
B	22	6	2	0
C	15	4	0	0
D	21	11	6	0

Cost matrix c

- **Add a dummy column with 0 score loss.**

Hungarian Method

an Example (3/7)

- **Step 1. Given the cost matrix c, get modified c'**

$$c = \begin{pmatrix} 0 & 0 & 4 & 0 \\ 22 & 6 & 2 & 0 \\ 15 & 4 & 0 & 0 \\ 21 & 11 & 6 & 0 \end{pmatrix} \Rightarrow c' = \begin{pmatrix} 0 & 0 & 4 & 0 \\ 22 & 6 & 2 & 0 \\ 15 & 4 & 0 & 0 \\ 21 & 11 & 6 & 0 \end{pmatrix}$$

Hungarian Method
an Example (4/7)

- **Step 2. check if there exists an optimal solution**

$$c' = \begin{pmatrix} 0 & 0 & 4 & 0 \\ 22 & 6 & 2 & 0 \\ 15 & 4 & 0 & 0 \\ 21 & 11 & 6 & 0 \end{pmatrix}$$

- **# of lines ≠ 4, go to step 3.**

Hungarian Method

an Example (5/7)

- **Step 3. Further modify c'**

$$c' = \begin{pmatrix} 0 & 0 & 4 & 0 \\ 22 & 6 & 2 & 0 \\ 15 & 4 & 0 & 0 \\ 21 & 11 & 6 & 0 \end{pmatrix} \Rightarrow c'' = \begin{pmatrix} 0 & 0 & 8 & 4 \\ 18 & 2 & 2 & 0 \\ 11 & 0 & 0 & 0 \\ 17 & 7 & 6 & 0 \end{pmatrix}$$

- **# of lines ≠ 4, continue step 3.**

Hungarian Method

an Example (6/7)

- **Step 3. Further modify c''**

$$c'' = \begin{pmatrix} 0 & 0 & 8 & 4 \\ 18 & 2 & 2 & 0 \\ 11 & 0 & 0 & 0 \\ 17 & 7 & 6 & 0 \end{pmatrix} \Rightarrow c''' = \begin{pmatrix} 0 & 0 & 8 & 6 \\ 16 & 0 & 0 & 0 \\ 11 & 0 & 0 & 2 \\ 15 & 5 & 4 & 0 \end{pmatrix}$$

- **# of lines = 4, get the optimal solution.**

Hungarian Method

an Example (7/7)

- The optimal solution

$$c''' = \begin{pmatrix} 0 & 0 & 8 & 6 \\ 16 & 0 & 0 & 0 \\ 11 & 0 & 0 & 2 \\ 15 & 5 & 4 & 0 \end{pmatrix}$$

- Assignment:
  - A—P, B—R, C—Q or A—P, B—Q, C—R
- Maximum score:
  - Z = 90 +69 +72 = 90 +70 +71 = 231

Tools for AP & LP

- **A Parametric Visualization Software for the Assignment problem. University of Macedonia, Greece. (*http://users.uom.gr/~samaras/gr/yujor/yujor.htm*)**
- **LINGO: a comprehensive tool designed to help you build and solve linear, optimization models quickly, easily, and efficiently. (*http://www.lindo.com/*)**
- **MATLAB: *linprog* function for general LP; Toolbox *YALMIP* for (mixed) integer LP.**
- **……**

Thanks for Your Attention.

Assignment Problems

Guoming Tang