Indian Institute of Technology- Jodhpur

Graph Theory and Applications(GTA)
COURSE CODE: CSL7410

Lecture Scribing Assignment: Week 8

Done by:
Sachin Kumar
M20MA064

March 7, 2022

॥ त्वं ज्ञानमयो विज्ञानमयोऽसि ॥

Week 8

Chinese Postman Problem And Assignment problem*

Chinese Postman Problem:- the background of chinese postman problem is interesting. it is the problem that the chinese postman faces to travel along every road in a city in order to deliver letters with the least possible distance

8.1 Some Definition

8.1.1 Eulerian Circuit

Eulerian path is a path in graph that visits every edge excatly one time.Eulerian circuit is eulerian path which starts and ends on the same verter.

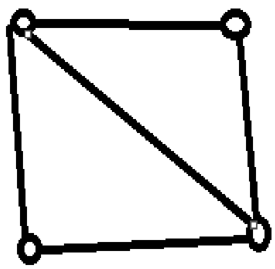
These are two case in chinese postman problem:-

(i) If all vertices has degree even then this graph also be eulerian circuit then nothing to solve.

(ii) if $\exists$ a vertex has odd degree then we make to eulerian circuit use arbitrary dummy edge

Example:-

*Lecturer: Anand Mishra. Scribe: Sachin Kumar (M20MA064).



this is not eulerian graph so we use dummy edge to make eulerian circuit

Weighted bipartite Matching Transversal:-

A transversal of an $N * N$ matrix consists of N positions one in each row and one in each column example:-

$$\begin{pmatrix} 11 & 12 & 13 \\ 14 & 15 & 16 \\ 17 & 18 & 19 \end{pmatrix}$$

Transversal will be $\{11, 15, 19\}$, $\{4, 12, 19\}$, $\{17, 15, 13\}$, $\{11, 1, 8, 16\}$, $\{14, 18, 13\}$, $\{17, 12, 16\}$

Solve assignment problem minimize

$\sum_i \sum_j c_{ij} \times x_{ij}$ such that

(i) $\sum_i x_{ij} = 1$

(ii) $\sum_j X_{ij} = 1$

(iii) $X_{ij} \in \{0, 1\}$

$$C = \begin{bmatrix} c_{00} & c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & \cdots & c_{2n} \\ \vdots & & & & \\ c_{n1} & c_{n2} & \cdots & & c_{nn} \end{bmatrix}$$

Hungarian algorithm:-

use Hungarian algorithm finds a maximum weight matching and a minimum cost cover.

step.1. find Row minimum

Step 2. Subtract row-minimum with every element

Step 3. find column-minimum

Step 4. Subtract column min with - every element.

Step 5. cover all zeroes with minimum number of horizontal and vertical lines and if number of required line equal to N then stop. if not. then next step

Step 6. find smallest uncovered entry

Step 7. subtract smallest uncovered with every element except zero and whose element where lines cut each other add smallest uncovered entry.

Example:- Step 1.

$$\begin{array}{cc} \begin{bmatrix} 15 & 40 & 45 \\ 20 & 60 & 35 \\ 20 & 40 & 25 \end{bmatrix} & \begin{array}{c} 15 \\ 20 \\ 20 \end{array} \end{array}$$

Step 2.

$$\begin{bmatrix} 0 & 25 & 30 \\ 0 & 40 & 15 \\ 0 & 20 & 5 \end{bmatrix}$$

Step 3.

$$\begin{bmatrix} 0 & 25 & 30 \\ 0 & 40 & 15 \\ 0 & 20 & 5 \end{bmatrix}$$

Step 4.

$$\begin{bmatrix} 0 & 5 & 25 \\ 0 & 20 & 10 \\ 0 & 0 & 0 \end{bmatrix}$$

number of required lines $= 2$ now next step

Step 5.

$$\begin{bmatrix} 0 & 5 & 25 \\ 0 & 20 & 10 \\ 0 & 0 & 0 \end{bmatrix}$$

uncovered smallest element $= 5$

Step 6.

$$\begin{bmatrix} 0 & 0 & 20 \\ 0 & 15 & 5 \\ 5 & 0 & 0 \end{bmatrix}$$

number of required line $= 3$

So optimal solution $= 20 + 40 + 25 = 85$.