NEWSPAPER DISTRIBUTION PROBLEM IN A CITY NETWORK.

A dessertation submitted in partial fulfilment of
of the requirements for the M.Tech(computer science) degree of
The Indian Statistical Institute.

bv

GAUTAM KUMAR DAS.

under the supervision of
DR. BIMAL KUMAR ROY.
computer science unit.
INDIAN STATISTICAL INSTITUTE.
203, Barrackpore Trunk Road.
Calcutta - 700035.

24 July 1997.

Indian Statistical Institute.
203, B. T. Road.
Calcutta - 700035.

Certificate of Approval.

This is to certify that the thesis titled, NEWSPAPER DISTRIBUTION PROBLEM IN A CITY NETWORK submitted by **Gautam Kumar** Das, towards partial fulfilments of the requirement for the degree of M.Tech in computer science at the Indian Statistical Institute, Calcutta, embodies the work done under my supervision.

Bimal Roy 31.7.97

Dr. Bimal Kumar Roy.
Computer Science Unit,
Indian Statistical Institute,
Calcutta - 700035.

Acknowledgements.

The very first person Whom I'd like to thank and whom I feel myself honoured to have worked with is my supervisor Dr. Bimal Kumar Roy, who is both an excellent teacher and a nice human being, and who has taught me a lot during my stay of two years.

I would like to thank all C.S.S.C staffs for helping me to avail of the computing facilities and offering their valuable advice, whenever I face any problem.

I am grateful to Padma Mahalingam, Rekha Menon and Vishal Hawa for helping me with LETEX, when I was writing this report. I am also grateful to Sounak Mishra for providing valuable suggestion related to my dissertation work.

Lastly, I would like to thank all my classmates, for making my two year's stint in ISI a memorable one.

Calcutta.
July 24, 1997. (Gautam Kumar Das)

Abstract

Suppose there are n no. of nodes(i.e junctions) in a city out of which there are m number of dumping nodes ($m \leq n$) where newspapers are to be supplied from the vehicle. The problem is to find the shortest route in terms of cost of edges where the starting node is the newspaper office itself & is required to traverse every dumping node at least once. (this total cost does not include the cost of the edges from the last dumping nodes to the newspaper office).

Contents

1 Introduction.

Heuristic algorithms with polynomial rates of growth in the number of variables can be used to provide approximate solutions to combinatorial problems. The question then arises as to what is the worst possible ratio of the value of the answer obtained by the heuristic to the value of the optimal solution. We will denote this worst-case ratio by R_w.

In this Dissertation paper we describe a heuristic algorithm for Newspaper Distribution Problem (NDP) with O(n^3) growth rate and for which $R_w < 2$ for all n. The algorithm involves as substeps the computation of a minimun spanning tree of the graph G formed by the shortest path algorithm among dumping nodes and the finding of a minimum cost perfect matching of a certain induced subgraph of G.

This problem was posed as an integer linear programming problem and was partially solved by Achutan et el[1].

2 Euler and Hamilton Graphs.

Let u,v $\in$ V be the vertices of the graph G = (V,E)

Definition -2.1 A u-v walk in G is $(u = u_0, e_1, u_1, e_2, u_2, ...e_n, u_n = v)$ where $u_i \in V$, $0 \leq i \leq n$ are vertices ; $e_i \in E$, $1 \leq i \leq n$ are edges and $e_i = \{u_{(i-1)}, u_i\}$.

u-v walk is closed if u=v; otherwise open.

definition -2.2 A u-v walk in which all the edges are distinct is called a

TRAIL. A trail that traverses every edges of G is called Euler trail of G. A Euler tour is a closed Euler trail. A graph is Eulerian if it contains an Euler tour.

Theorem -2.1 A nonempty connected graph is Eulerian iff it has no vertices of odd degree.

Corollary -2.1 A connected graph has an euler trail iff it has two vertices of odd degree.

Definition -2.3 A u-v walk in which all the vertices are distinct is called a PATH.

Definition -2.4 A cycle is a closed u-v walk in which all the vertices (except u=v) are distinct.

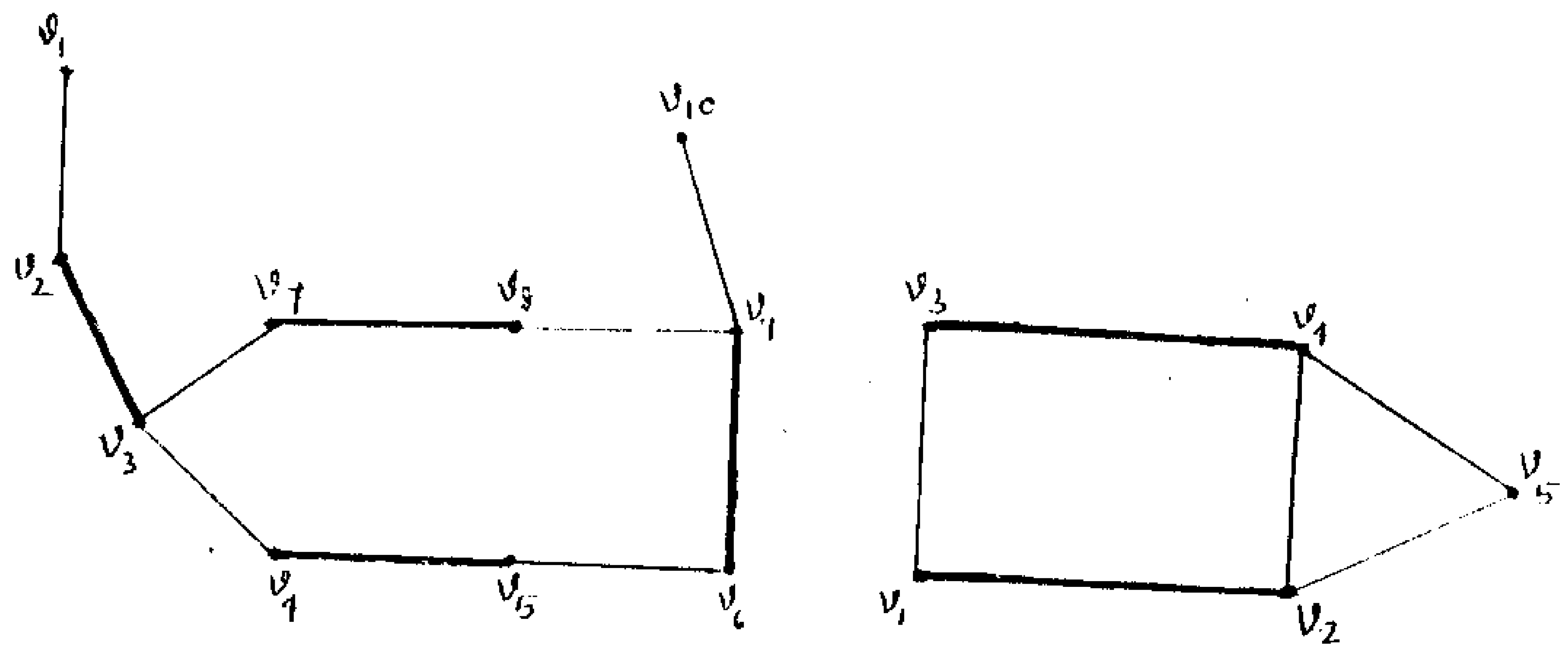
Definition -2.5 A cycle containing all the vertices of the graph is called a HAMILTONIAN CIRCUIT.

A graph is Hamiltonian if it contains a Hamilton circuit.

3 Matching.

A subset M of E is called a matching in G = (V,E) if its elements are links and no two are adjacent in G; the two ends of an edge in M are said to be matched under M. In the graph of figure - 3.1(a) , for example $M_1 = \{[v_2, v_3],[v_4, v_5],[v_7, v_8],[v_6, v_9]\}$ (shown as heavy lines in fig -3.1(a)) & $M_2 =$

$\{[v_1, v_2],[v_3, v_4],[v_5, v_6],[v_7, v_8],[v_9, v_{10}]\}$ are matching.



(a) (b)

figure - 3.1

A matching M saturates a vertex v and v is said to be M saturated, if some edge of M is incident with v. A node v is said to be exposed reletive to M if no edges of M meets v.

M is a maximum matching if G has no matching M' with |M'| > |M|. Clearly M_2 is maximum matching but M_1 is not. If every vertex of G is M saturated, then the matching M is perfect. So M_2 is also perfect matching but M_1 is not. Notice that matching in fig - 3.1(b) is maximum but not perfect. Clearly every perfect matching are maximum but not the converse.

Definition -3.1 Let G = (V,E) be a graph and M be a matching. An alternating path with respect to M is a path such that the first and last edges of the path are not matched and such that every second edge on the path is matched.

If the first and last vertices on the path are unmatched ,then the alternating path is called an augmenting path. So in Fig -3.1(a) $\{v_1, v_2, v_3, v_4, v_5, v_6, v_9, v_{10}\}$ is an augmenting path and $\{v_1, v_2, v_3, v_7, v_8, v_9, v_6, v_5\}$ is an alternating path but it is not augmenting path.

Lemma - 3.1 Let P be the set of edges on an augmenting path p = $[u_1, u_2, u_3, ..., u_{2n}]$ in a graph G with respect to the matching M. Then M' = M $\otimes$ P is a matching with cardinality |M| + 1.

Theorem - 3.1 A matching M in a graph G is maximum if and only if there is no augmenting path in G with respect to M.

4 Weighted Matching.

Given a graph G = (V,E) and a weight for each edge w : E ->R^+, In weighted matching we have to find a matching of minimum weight.

5 The classes NP_hard and NP_complete.

In measuring the complexity of an algorithm we shall use the input length as the parameter. An algorithm A is of polynomial complexity if there exits a polynomial P() such that the computing time of A is O(P(n)) for every input of size n.

Definition - 5.1 P is the set of all decision problem solvable by a deterministic algorithm in polynomial time. NP is the set of all decision problems solvable by a nondeterministic algorithm in polynomial time.

For a problem to be in NP, we simply require that if x is a yes instance of the problem, then there exists a concise (ie, of length bounded by polynomial in the size of x) certificate for x, which can be checked in polynomial time for validity.

We are now ready to define NP_hard and NP_complete classes of problems, first we define the notion of reducibility.

Definition - 5.2 Let A_1 and A_2 be recognition (ie, yes _ no) problems. We say that A_1 reduces in polynomial time to A_2 (denoted as $A_1 \propto^p A_2$) iff there exits a polynomial time algorithm a_1 for A_1 that uses several times as a subroutine at unit cost a (hypothetical) algorithm a_2 for A_2. We call a_1 a polynomial time reduction of A_1 to A_2.

proposition - 5.2 If A_1 polynomially reduces to A_2 and there is a polynomial time algorithm for A_2, then there is a polynomial time algorithm for A_1.

Definition -5.3 We say that a recognition problem A_1 polynomially transforms to another recognition problem A_2 if, given any string x, we can construct a string y within polynomial (in |x|) time such that x is a yes instance of A_1 iff y is a yes instance of A_2.

Definition - 5.4 A problem A is NP_hard iff for all $A' \in NP$, $A' \propto^p A$. A problem is NP_ complete iff A is NP_hard and A $\in$ NP.

By proposition - 5.1 if a problem A is NP_complete , then it has a formidable property: If there is an efficient algorithm for A, then there is an efficient algorithm for every problem in NP.

Definition - 5.5 Two problems A_1 and A_2 are said to be polynomially equivalent iff $A_1 \propto^p A_2$ and $A_2 \propto^p A_1$.

6 Triangle_Inequality.

Definition -6.1 Consider an nxn distance matrix $[d_{ij}]$ with positive real entries. As usual we assume that $[d_{ij}]$ is symmetric ie, $d_{ij} = d_{ji}$ for all i,j. and that $d_{jj} = 0$, for all j. We say that $[d_{ij}]$ satisfies triangle inequality iff $d_{ij} + d_{jk} \geq d_{ik}$

What the triangle inequality constraints essentially says is that going from city i to k through j can not be cheaper than going directly from i to k. This is very reasoable since the imposed visit to city j appears to be an additional constraints,which increases the cost.

0 2 3 5	0 2 3 3
2 0 7 1	2 0 4 1
3 7 0 3	3 4 0 3
5 1 3 0	3 1 3 0
(a)	(b)

The matrix in (a) does not satisfy the triangle ineuality because for example $d_{23} > d_{24} + d_{43}$. The matrix in (b) does. In fact this matrix is the closure of the previous one.

One inportatent class of distance matrices that automatically satishfy the triangle inequality are closure matrices. We say that the matrix $[d'_{ij}]$ is the closure of $[d_{ij}]$ if d'_{ij} is the length of the shortest path from i to j in the commmplete graph of n nodes {1,2,3,....,n} where the length of the edge [i,j] is d_{ij}.

Definition -6.2 The triangle inequality (or metric) TSP (abbreviated $\triangle$TSP) is the TSP restricted to matrices satisfying the triangle inequality.

Theorem - 6.1 The (recognition version of) $\triangle$TSP is NP_complete.

proof : - Given a graph (V,E) we construct an instance ($[d_{ij}]$,V) of |V|x|V| TSP with $d_{ij} = 1$ if $[v_i, v_j] \in$ E and 2 Otherwise. It is immidiate that this instance had a tour of cost |V| or less iff G is hamiltonian. Observe that any distance matrix with all entries either 1 or 2 such as $[d_{ij}]$, satisfies the triangle inequality. Thus HAMILTON CYCLE pollynomially transform to $\triangle$TSP. ♣

Definition - 6.3 Let $[d_{ij}]$ be an nxn distance matrix satisfying the triangle inequality. An Eulerian spanning graph is a Eulerian multigraph (graph with repetions of edges allowed) G = (V,E) with V ={ 1,2,3...,n}. The cost of G is $C(G) = \sum d_{ij}$ over [i,j] $\in$ E.

Theorem - 6.2 If G = (V,E) is an Eulerian spanning graph, then we can find a Hamiltonian path T of V with $C(T) \leq C(G)$ in O(|E|) time.

proof :- By hypothesis G has an Eulerian tour w, because w visits all nodes at least once , we can write $w = [\alpha_0, i_1, \alpha_1, i_2, ..., i_n, \alpha_n]$, where $T = (i_1, i_2, i_3, ..., i_n)$ is a tour and $\alpha_0, \alpha_1, ..., \alpha_n$ are sequences(possibly empty) of integers in { 1,2,3...,n} (We say that T is embeded in G). Now, the triangle inequality implies that

$d_{ik} \leq d_{ij_1} + d_{j_1 j_2} + ... + d_{j_{(m-1)} j_m} + d_{j_m k}$

for any $m \geq 1$. Consequently , the total length of w which is exactly C(G) can be no smaller than

$$d_{i_1 i_2} + d_{d i_2 i_3} + ... + d_{i_n i_1} = \text{C(T)}. \clubsuit$$

7 TSP for Complete graph satisfying triangle_inequality.

Consider the n_city TSP defined on the complete graph G=(V,E) where V is the set of vertices and E is the set of edges. Let the edge cost matrix be $[C_{ij}]$ which satisfies the triangle inequality. Let $T = (X, A_T)$ be the minimum spanning tree of the graph G & Let C(T) be the cost of T. Let $X^0(T) = \{ x_i | d_i(T) \text{ is odd } \}$ where $d_i(T)$ is the degree of vertex $x_i \in X$ w.r.to the tree T. The cardinality $|X^0(T)|$ of the set $X^0(T)$ is always even. Let $M_0 = (X^0(T), A_{M_0})$ be the minimun cost perfect matching of $<X^0(T)>$ & $C(M_0)$ be its cost.

Theorem - 7.1 A Hamiltonian circuit Φ_H of G can be found with cost $C(\Phi_H) \leq C(T) + C(M_0) < 3/2C(\Phi^*)$ where $C(\Phi^*)$ is the optimal value of the TSP tour Φ^*.

Lemma - 1 For an n_city TSP with n even, we have $C(M_0) \leq 1/2C(\Phi^*)$, where M_0 is the minimum cost perfect matching of the graph G defining the TSP and Φ^* is the optimal TSP tour.

8 Newspaper distribution problem (NDP) in a city network.

In order to tackle the problem we will do the following:-

8.1 > We will show that the given problem (NDP) is NP_complete.

8.2> We will give some heuristic algoritm for NDP.

8.3> Also we will establish the worst case upper bound of the heuristic algorithm.

Before proving 8.1 we will show that finding minimum cost hamiltonian path from a weighted complete graph whose weight matrix satishfies the triangle inequality (CHPT) is NP_complete . The proof that we will present here is simillar to the proof that TSP is NP_complete from Hamiltonian Circuit [4].

We know that Hamilton path problem (HP) for a general graph is NP_complete. In the proof we will reduce HP to CHPT.

8.1(a) : The (recognition version of) CHPT is NP_complete.

proof :- We will transform from HP to CHPT. Take any instance of HP i.e, Arbitrary graph G = (V,E). We will now construct an instance for CHPT from HP in two steps.

step - 1 Construct a weighted complete graph $G' = (V,[d_{ij}])$ where $d_{ij} = 1$ if $[v_i, v_j] \in$ E, $d_{ij} = 2$ if $[v_i, v_j] \notin$ E.

step - 2 Add a new node say v_{n+1} to G' and connect v_{n+1} to all other nodes with weight 2. i.e, $d_{i,n+1} = 2$ for $1 \leq i \leq n$.

Let this new graph be $G'' = (V', [d'_{ij}])$ where $V' = V \cup \{v_{n+1}\}$.

Clearly, G'' is a complete graph which satisfies the triangle inequality, because weights are either 1 or 2.

Suppose that there is a solution of CHPT in polynomial time. Then we find the solution of CHPT from each vertex $v \in V' - \{v_{n+1}\}$ and we take minimum weight hamiltonian path among them. Let it be C. This can be done in polynomial time. Notice that value of C can never be < (n+1) (clear from the construction of G''). So C may be either (n+1) or > (n+1).

If C = n+1, iff there is a hamiltonian path for the original graph G. So we can find a polynomial time algorithm for HP which is most unlikely.

if C > n+1, iff there is no hamiltonian path for the original graph G. So in this case also we find a polynomial time algorithm for HP.

So from above discussion we can conclude that CHPT has a hamiltonian path of cost n+1 iff original graph G has a hamiltonian path. So HP is polynomially transform to CHPT. ♣

8.1 : The (recognition version of) NDP is NP_complete.

proof :- Given a weighted complete graph G = (V,E) whose weight matrix satishfies triangle inequality where V = { 1, 2, 3,....,n}. Now construct an instance of NDP which contains |V| number of dumping nodes & shortest distance between any two dumping nodes is taken as the distance between the corresponding nodes in the complete graph. So it is immidiate that NDP has a tour of length ≤ L (some constant number) iff the complete graph has a hamiltonian path of length ≤ L. Thus CHPT polynomially transform to NDP. ♣

8.2 Proposed Heuristic Algorithm.

Input :- Adjacency matrix C[i,j] (edge cost) of a weighted graph having n number of nodes {1,2,3........n} of which m number of nodes are dumping.

Output :- Approximation to the minimum cost traversal where each dumping node has to be traversed at least once and not required to go back to the starting node.

step-1 > Find the all pairs of shortest path among dumping nodes. (so that we will get complete graph of size equals to number of dumping nodes with weight matrix D[i,j]).

step-2 > Find the minimum spanning tree T where input is the weight matrix D[i,j].

step-3 > Find the nodes of T having odd degree & find the complete matching M with minimum weight in the complete graph consisting of these nodes only. Let G be the multigraph(i.e, repetation of edges are allowed) with nodes {1,2,3.....,m} and edges those of T and those in M.

step-4 > Find an eulerian tour $\{1, 2', 3', ..., m', 1\}$ & then find the embeded tour $\{1, 2'', 3'', ..., m'', 1\}$ (ie, hamiltonian circuit).

step-5 > Take either $\{1, 2'', 3'', ..., m''\}$ or $\{1, m'', ..., 3'', 2''\}$ depending on whether edge $(1, 2'')$ or edge $(m'', 1)$ is minimum & calculate the total cost. ♣

8.3 WORST CASE UPPER BOUND.

A graph of n number of nodes {1,2,3....,n} are given out of which m number of nodes are dumping . Our problem is to find the order of traversal to each dumping node so that total cost is minimum.

Let $\{1', 2', 3', ..., m'\}$ be the order of traversal $(1 = 1')$ so that total cost $C = \sum_{i=1', j=i+1'}^{i<m'} C[i, j]$ (where C[i,j] is the minimum sum of cost of the edges from i to j and $\{2', 3', ..., m'\}$ is the some permutation of {2,3,....,m}) is minimum. In step -1 of our algorithm we find the all pairs of shortest path among m dumping nodes so that we get a weighted complete graph of size m whose weight matrix satisfies the triangle inequality. Now we claim that if we are going to find the minimum cost hamiltonian path to this weighted complete graph then we have to traverse this graph in the same order $\{1', 2', 3', ..., m'\}$ and same cost C will be obtained as in case of our problem in hand and vice versa.

Proof of the claim.

—> Suppose not

then let $\{1, 2'', 3'', ..., m''\}$ be the order of traversal of the weighted complete graph for finding the minimum cost hamiltonian path for which the total cost $C' < C$. Since i to j $(i = 1, 2'', 3'', ...m'' - 1$ and $j = 2'', 3'', ..., m'')$ is the shortest distance between dumping nodes i & j and also our target is to obtain minimum cost for our problem, so that $\{1, 2'', 3'', ...m''\}$ is also the order of traversal for our problem at hand. which is a contradiction to our initial assumtion (i,e $\{1, 2', 3', ..., m'\}$ is the order of traversal to our problem)

<— similar argument.

So from here we reach to the conclusion that the reletive error in finding minimum cost traversal in our problem is exactly same as the relative error of finding the hamiltonian path from a weighted complete graph whose weight matrix satisfies the triangle inequality.

Step-2,3,4,5 of the algorithm are for finding a minimum cost hamiltonian path from a weighted complete graph[6]. Let $C(\tilde{T})$ be the minimum cost hamiltonian path . In step-2 we find the minimum spanning tree T. Let C(T) is the total edge cost of the minimum spanning tree . Then it is easy to show that $C(T) \leq C(\tilde{T})$. Instead of step-3 if we replace it by creat a multigraph G by using two copies of T. Then we have $C(G) = 2C(T) \leq 2C(\tilde{T})$

In step-4 since after finding Eulerian tour , we are embedded it to get the hamiltonian circuit (we use triangle inequality here) , so that the output result at step-5 of our algorithm that we get $C(R) \leq 2C(T) \leq 2C(\tilde{T})$. Therefore relative error = $C(R)-C(\tilde{T})/C(\tilde{T}) \leq 1$.

i,e the result is guarenteed to be not farther than 100 percent from optimal.

Result can be improved if we take maximum matching with minimum weight among odd no. of nodes in the tree T as in the case of step-3 of our algorithm. ♣

9 Illustration of the proposed Heuristic Algorithm.

Let us apply the heuristic algorithm to the input graphs G = (V,E) shown below (Cost of edges $[v_i, v_j]$ is taken as infinity if there is no direct edge between v_i & v_j) .

example - 1

$$\begin{bmatrix} 0 & 3 & \infty & \infty & 1 \\ 3 & 0 & 5 & \infty & \infty \\ \infty & 5 & 0 & 4 & 1 \\ \infty & \infty & 4 & 0 & 2 \\ 1 & \infty & 1 & 2 & 0 \end{bmatrix}$$

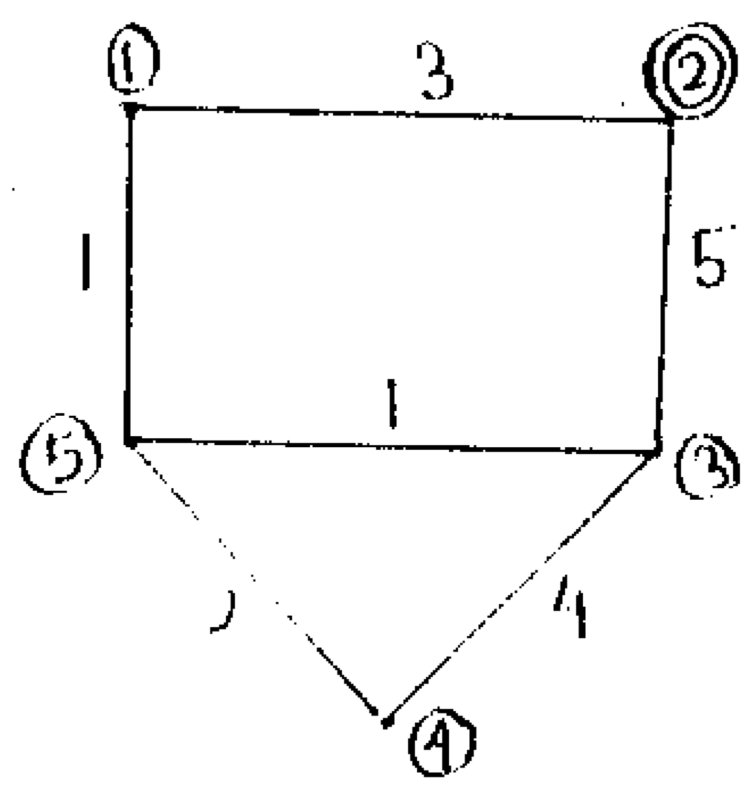


Fig - 9.1 Input Graph.

Suppose node 2 is not Dumping node. So all pair of shortest path among dumping nodes {1 3 4 5 } is given below :

$$\begin{bmatrix} 0 & 2 & 3 & 1 \\ 2 & 0 & 3 & 1 \\ 3 & 3 & 0 & 2 \\ 1 & 1 & 2 & 0 \end{bmatrix}$$

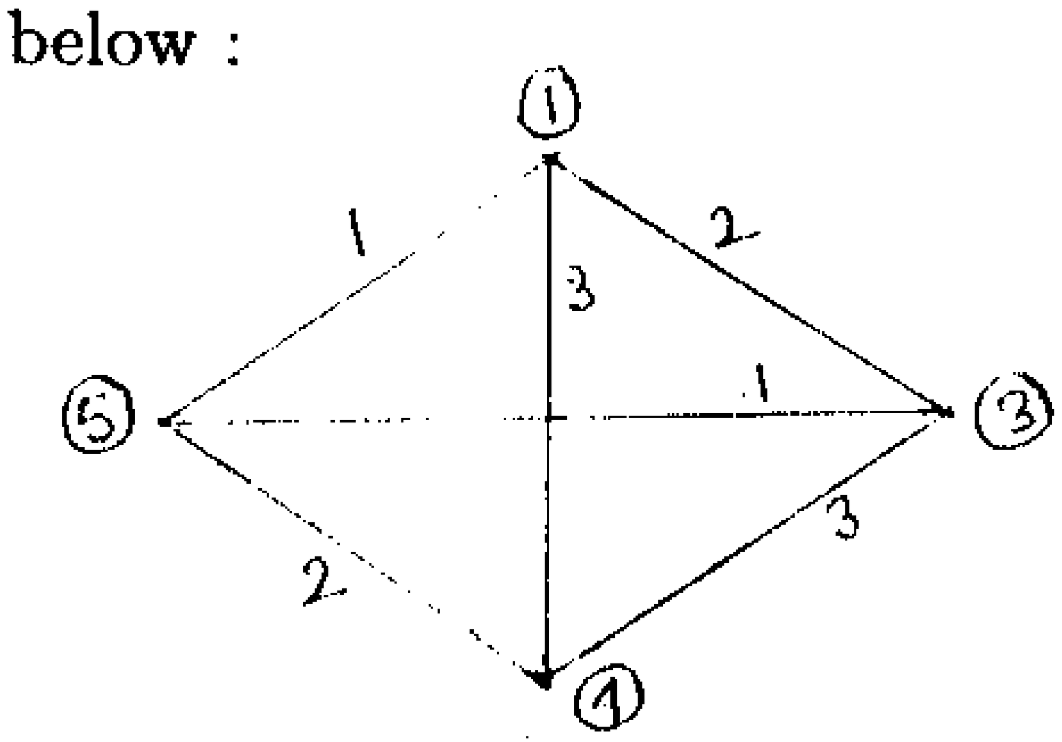
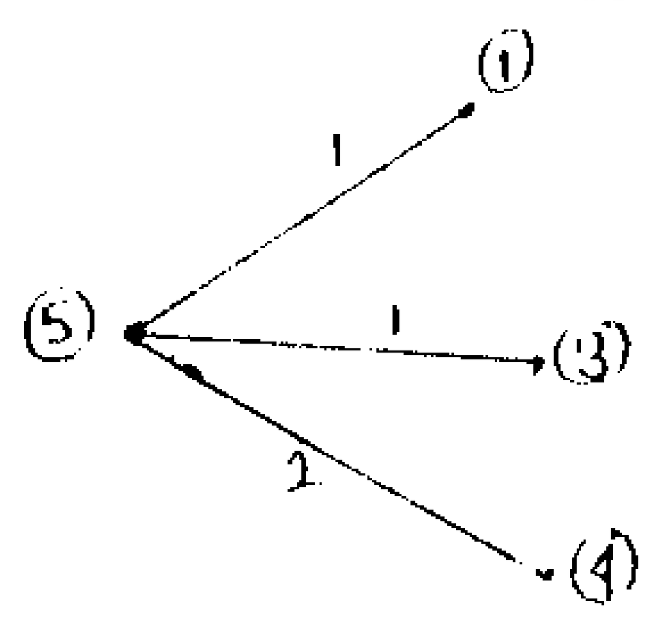


Fig- 9.2 shortest path among dumping node.

Minimum spanning tree of figure - 9.2 is shown below :

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 2 \\ 1 & 1 & 2 & 0 \end{bmatrix}$$



weight_matrix of MST   Fig - 9.3 MST.

Apply Step - 3 of our algorithm to the graph in Fig- 9.3 we get the diagram shown below :

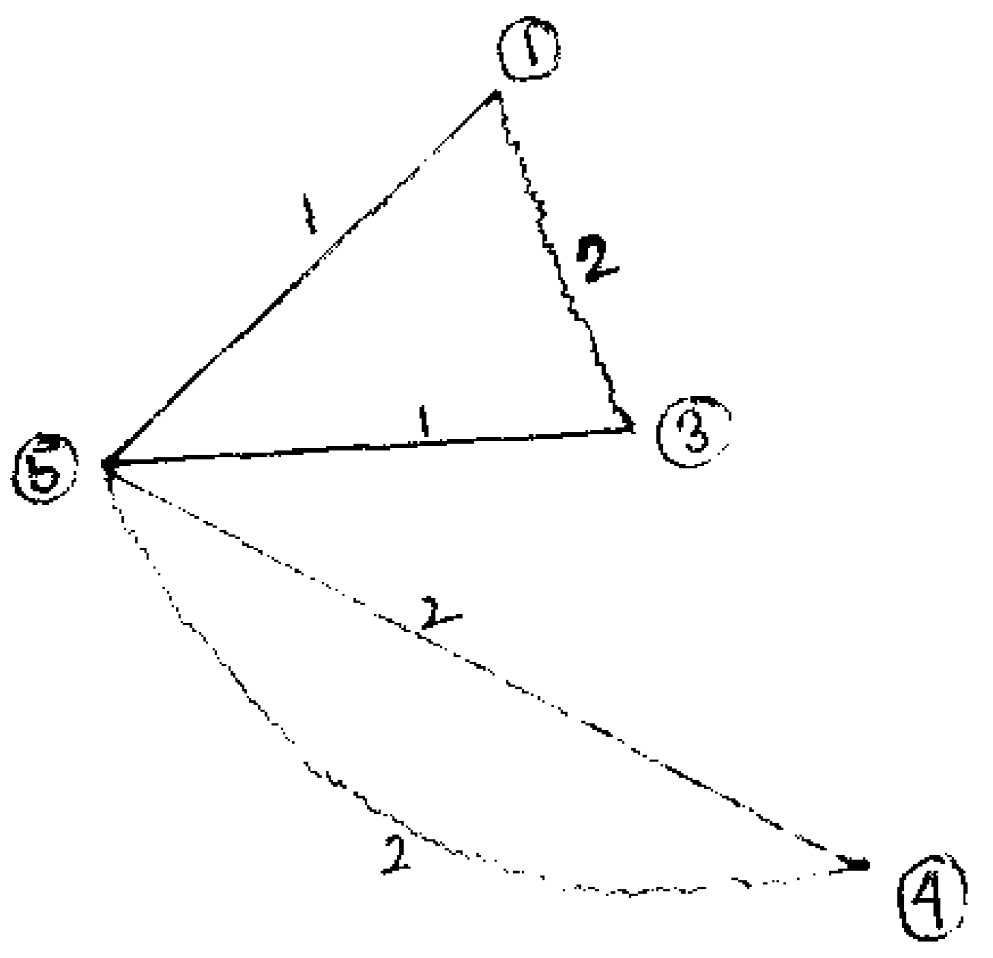


Fig - 9.4.

Eulerian tour of the Fig- 9.4 is shown below :

$$\{\ 1\ 3\ 5\ 4\ 5\ 1\ \}$$

Embeded hamiltonian circuit (by using triangle inequality) is given below :

$$\{\ 1\ 3\ 5\ 4\ 1\ \}$$

So total cost of the traversal { 1 3 5 4 }found by our algorithm = 5.
Note that optimal cost is also = 5.

example - 2

$$\begin{bmatrix} 0 & 5 & \infty & \infty & \infty & \infty & 1 \\ 5 & 0 & 2 & 1 & \infty & \infty & \infty \\ \infty & 2 & 0 & \infty & 3 & \infty & \infty \\ \infty & 1 & \infty & 0 & 6 & \infty & 4 \\ \infty & \infty & 3 & 6 & 0 & 4 & 3 \\ \infty & \infty & \infty & \infty & 4 & 0 & 2 \\ 1 & \infty & \infty & 4 & 3 & 2 & 0 \end{bmatrix}$$

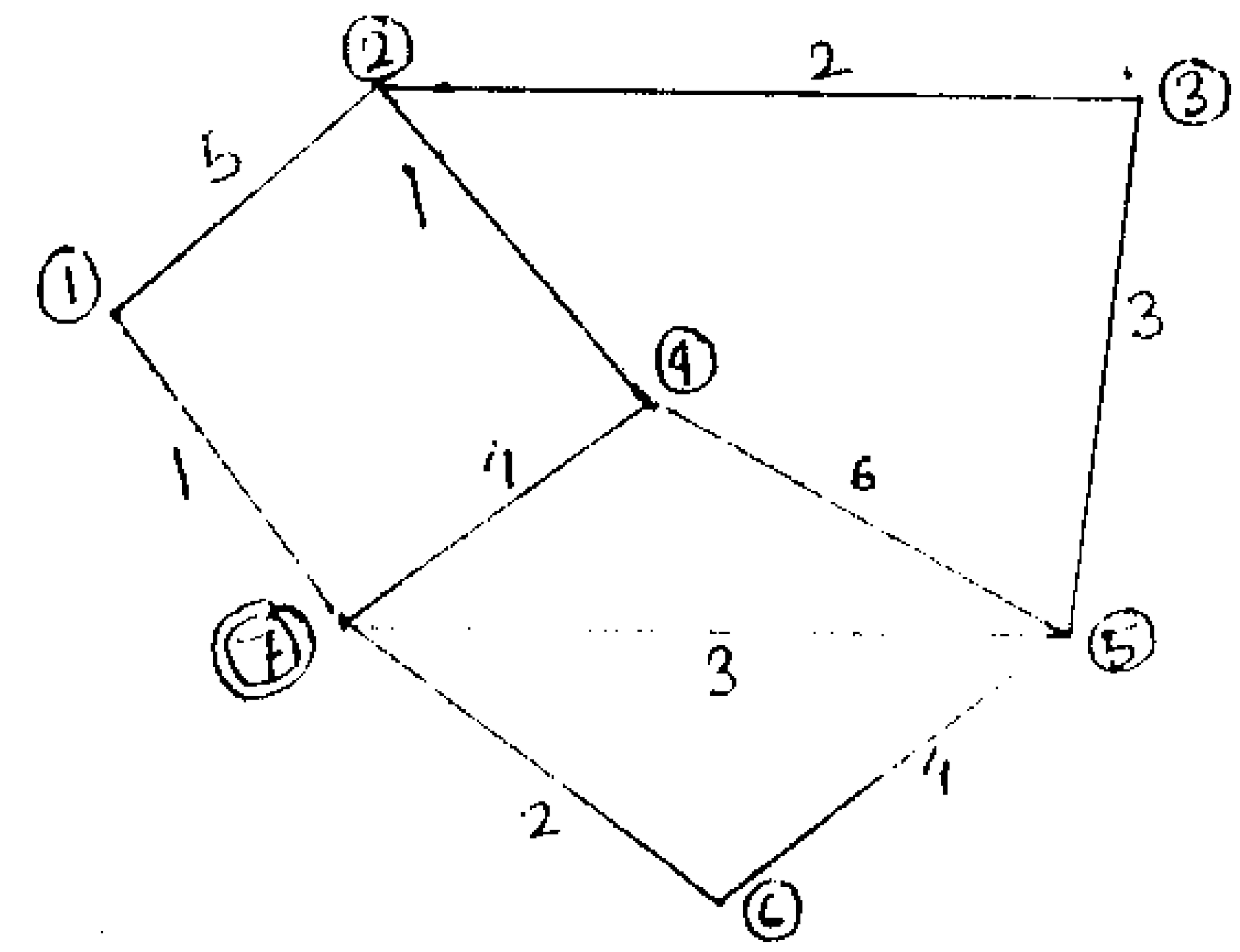


Fig - 9.1 Input Graph.

Suppose node 7 is not Dumping node. So all pair of shortest path among dumping nodes {1 2 3 4 5 6 } is given below :

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

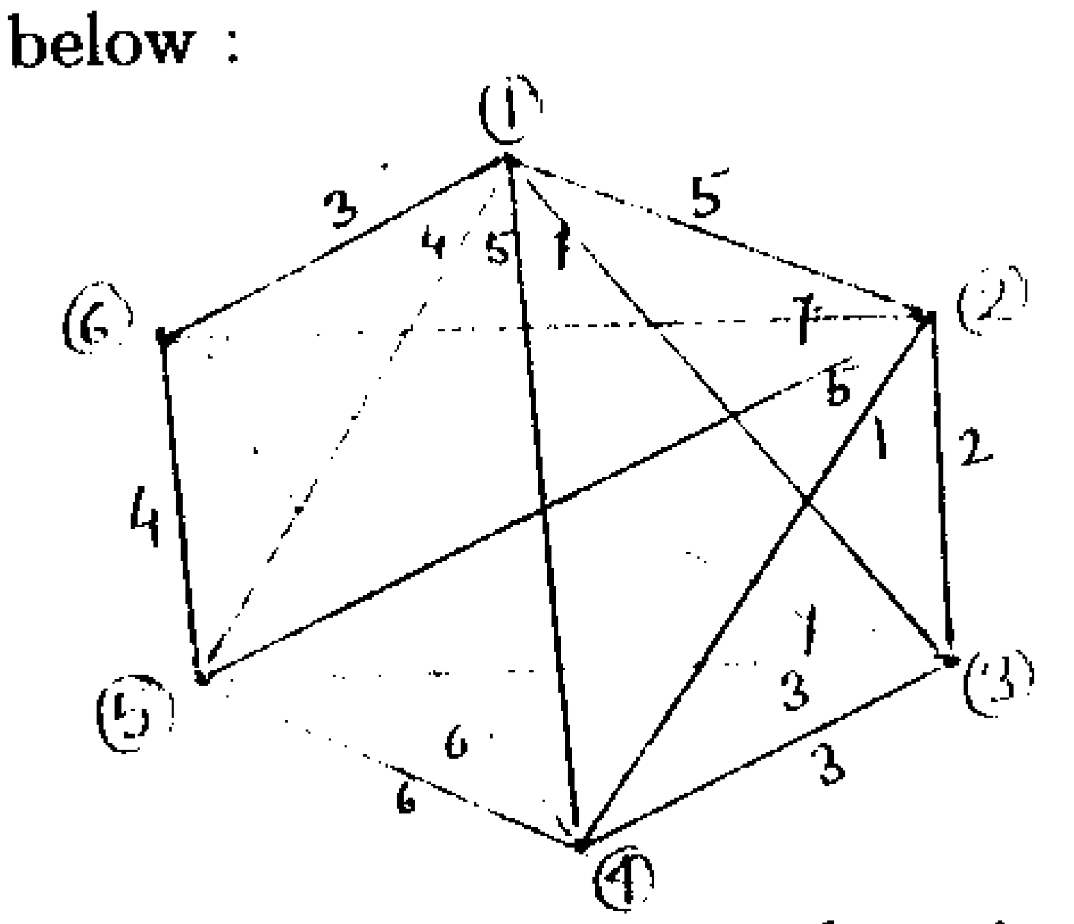


Fig- 9.2 shortest path among dumping node.

Minimum spanning tree of figure - 9.2 is shown below :

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 4 & 3 \\ 0 & 0 & 2 & 1 & 0 & 0 \\ 0 & 2 & 0 & 0 & 3 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 4 & 0 & 3 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

weight_matrix of MST

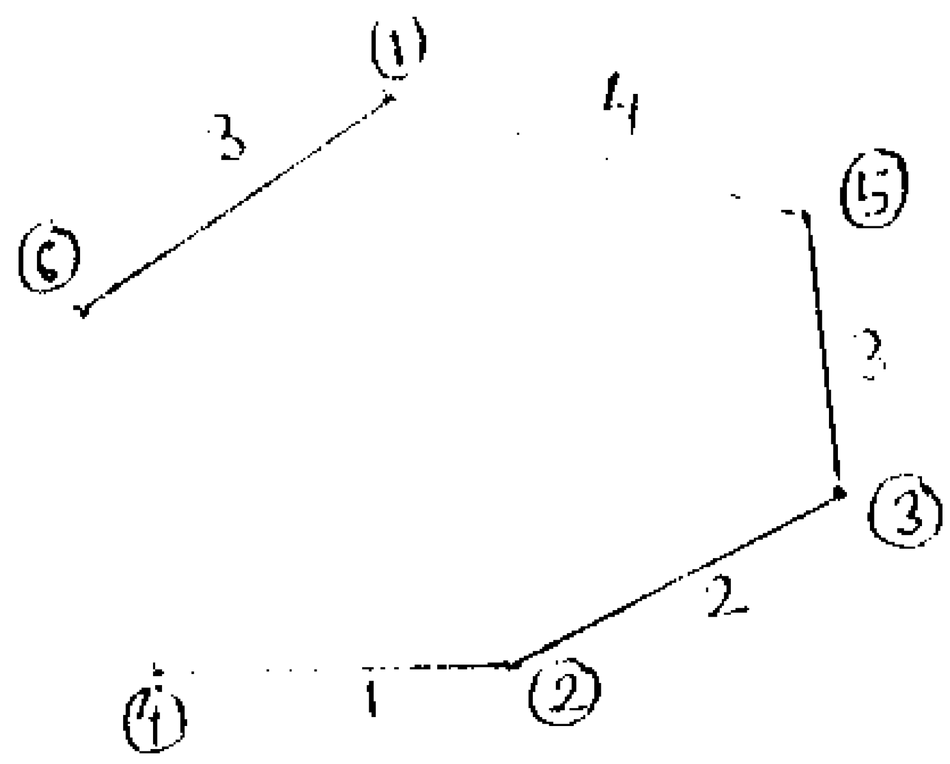


Fig - 9.3 MST.

Apply Step - 3 of our algorithm to the graph in Fig- 9.3 we get the diagram shown below :

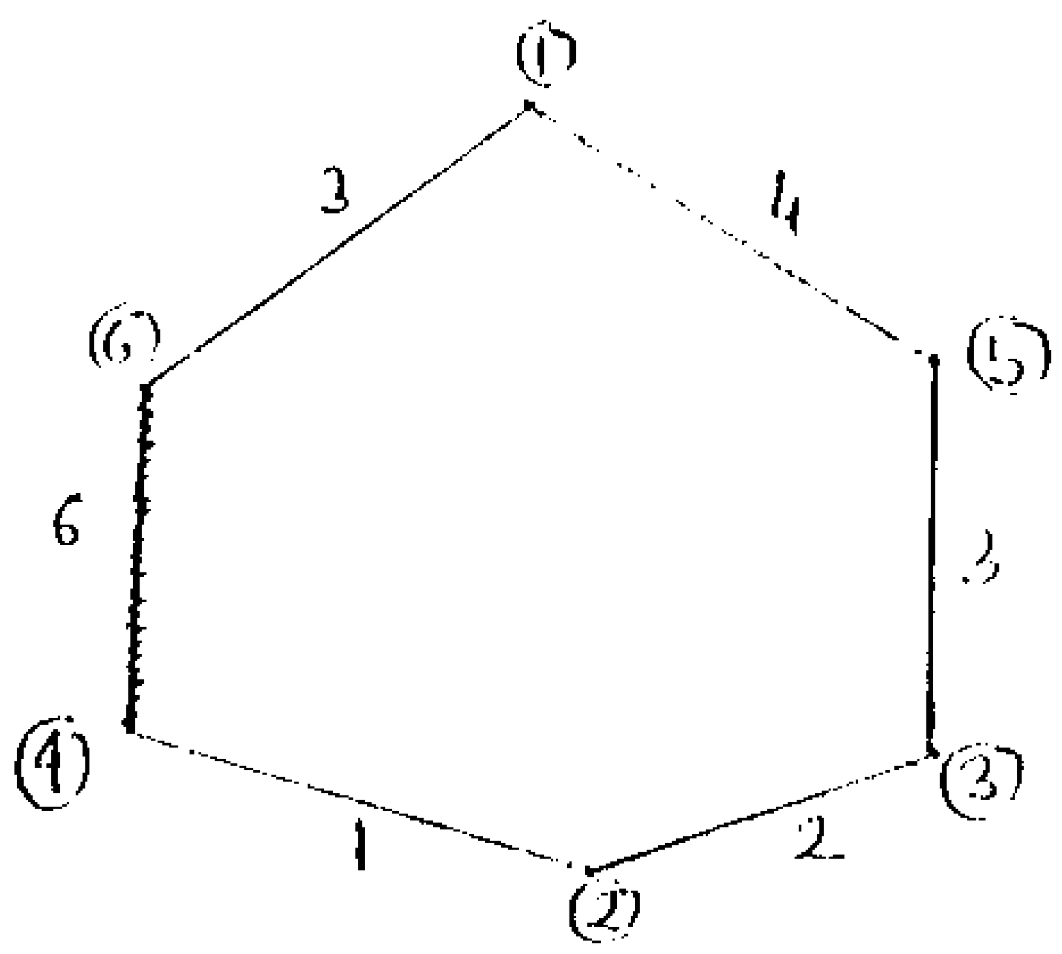


Fig - 9.4.

Eulerian tour of the Fig- 9.4 is shown below :

{ 1 5 3 2 4 6 1 }

Embeded hamiltonian circuit is given below :

{ 1 5 3 2 4 6 1 }

So total cost of the traversal { 1 6 4 2 3 5 } found by our algorithm = 15.

Note that optimal cost = 13 (order - 1 6 5 3 2 4).

example - 3

$$\begin{bmatrix} 0 & 5 & 3 & 1 & \infty & \infty & \infty \\ 5 & 0 & 4 & \infty & \infty & 2 & \infty \\ 3 & 4 & 0 & 2 & \infty & \infty & \infty \\ 1 & \infty & 2 & 0 & 3 & \infty & \infty \\ \infty & \infty & \infty & 3 & 0 & 1 & 4 \\ \infty & 2 & \infty & \infty & 1 & 0 & 3 \\ \infty & \infty & \infty & \infty & 4 & 3 & 0 \end{bmatrix}$$

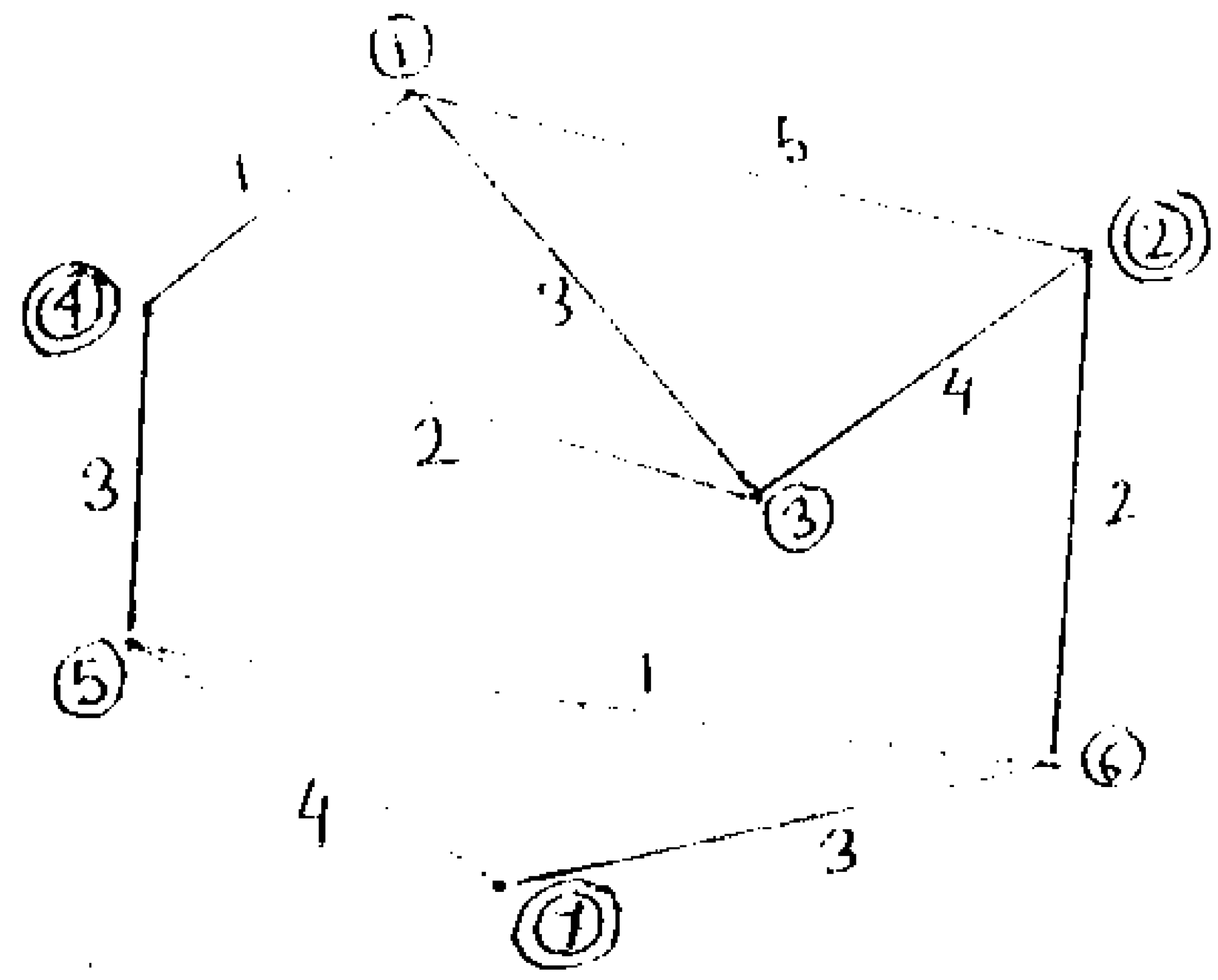


Fig - 9.1 Input Graph.

Suppose nodes 2,4, 7 are not Dumping node. So all pair of shortest path among dumping nodes { 1 3 5 6 } is given below :

$$\begin{bmatrix} 0 & 3 & 4 & 5 \\ 3 & 0 & 5 & 6 \\ 4 & 5 & 0 & 1 \\ 5 & 6 & 1 & 0 \end{bmatrix}$$

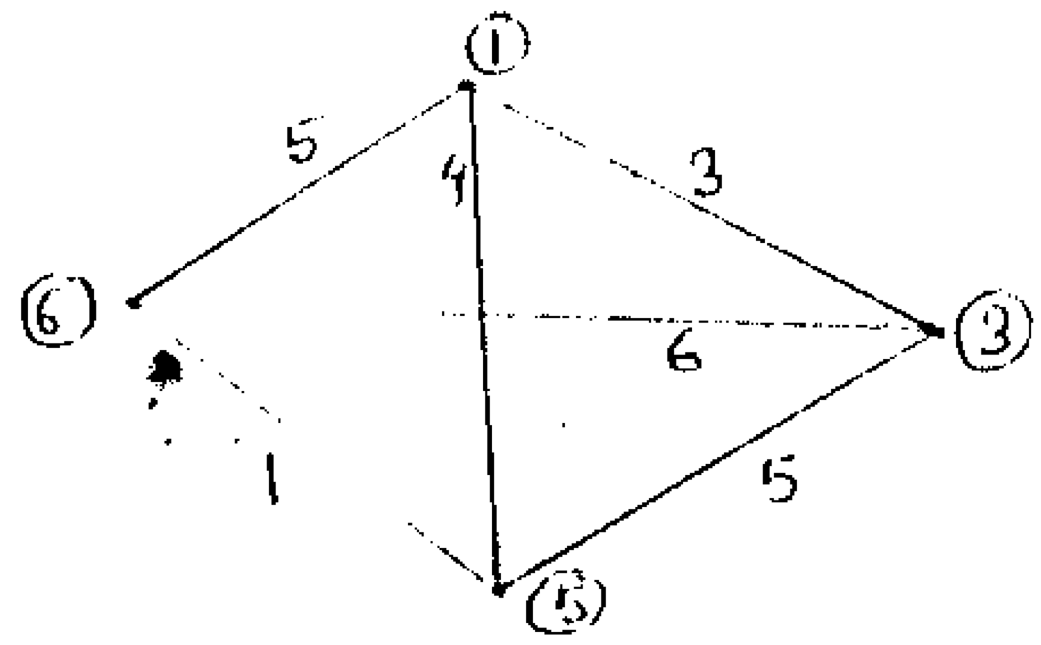
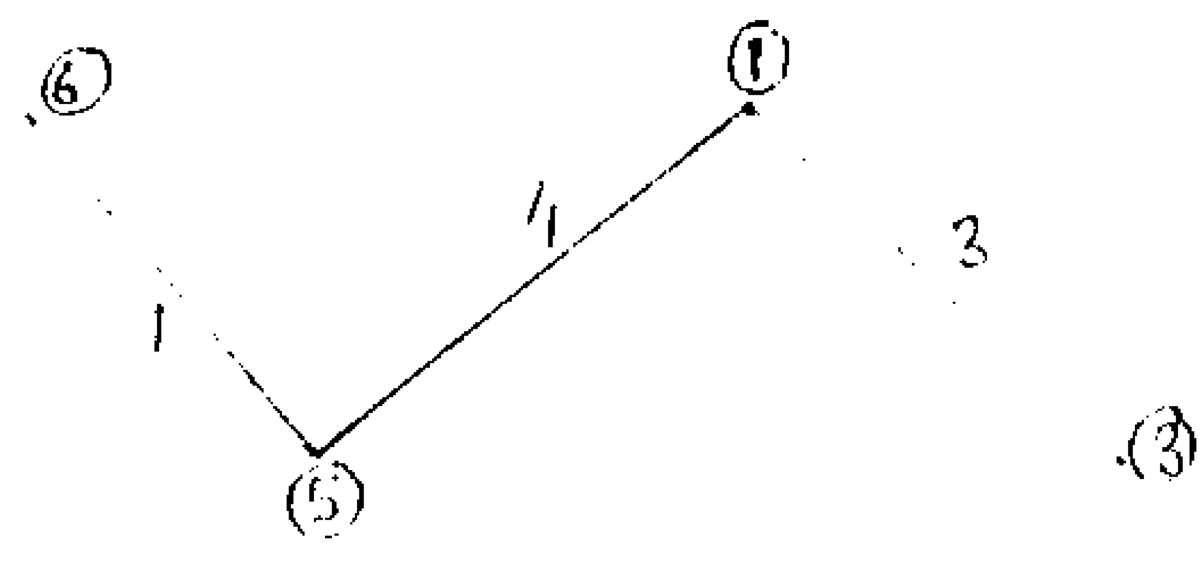


Fig- 9.2 shortest path among dumping node.

Minimum spanning tree of figure - 9.2 is shown below :

$$\begin{bmatrix} 0 & 3 & 4 & 0 \\ 3 & 0 & 0 & 0 \\ 4 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$



weight_matrix of MST

Fig - 9.3 MST.

Apply Step - 3 of our algorithm to the graph in Fig- 9.3 we get the diagram shown below :

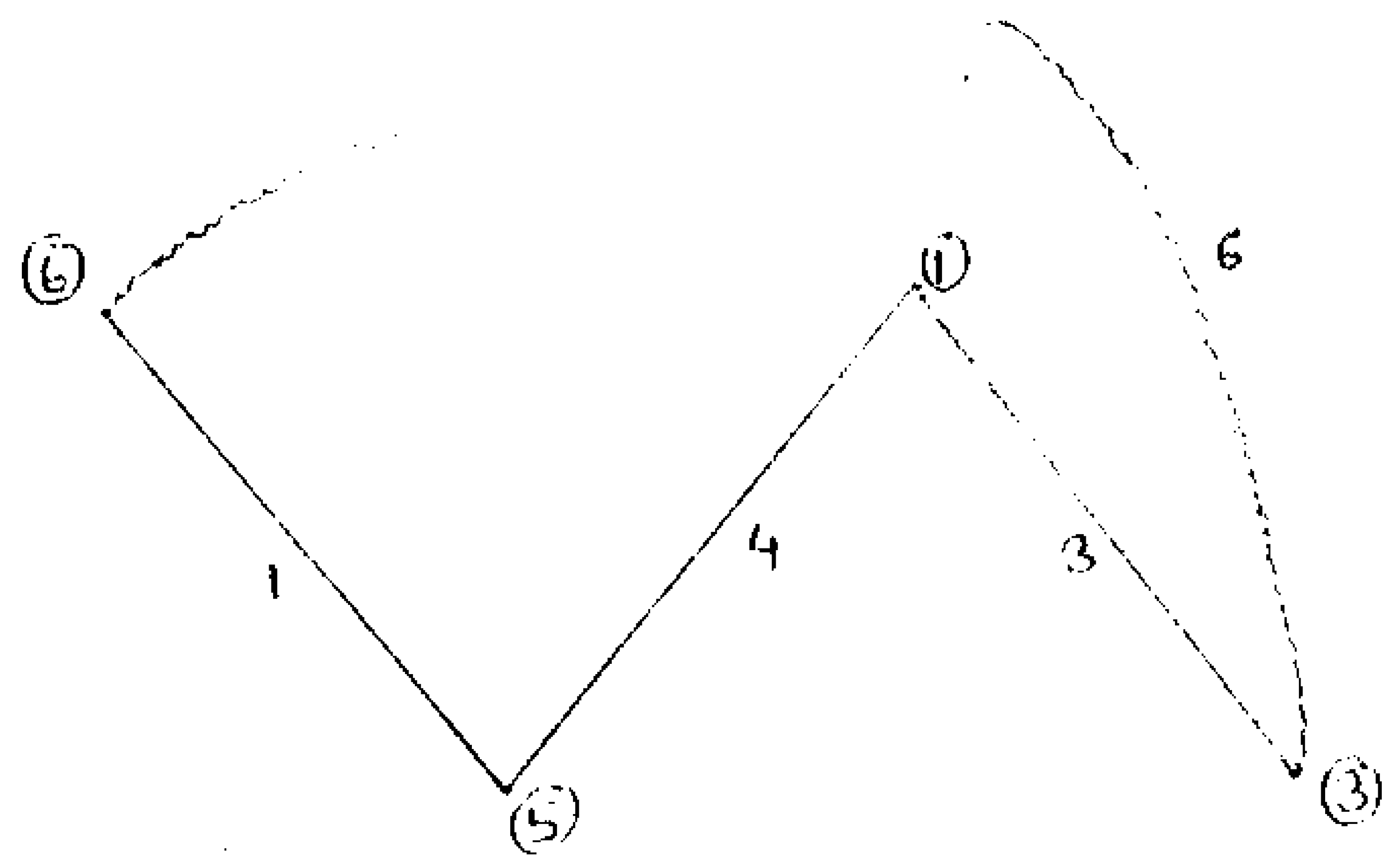


Fig - 9.4.

Eulerian tour of the Fig- 9.4 is shown below :

{ 1 3 6 5 1 }

Embeded hamiltonian circuit is given below :

{ 1 3 6 5 1 }

So total cost of the traversal { 1 3 6 5 } found by our algorithm = 10.

Note that optimal cost = 9 (order - 1 3 5 6)

10 References.

[1] Achutan et el, Routing of Newspaper from vehicle - preprint.

[2] Edmonds, J. and Jonson, E.L.(1973). Matching, Euler tours and chinese postman. math. programming, 5, 88 - 124.

[3] A. Bondy & U.S.R. Murty, graph theory with applications.

[4] Christos H. papadimitrou & Kenneth steiglitz, Combinatorial optimization : Algorithms and Complexity, Prentice-Hall 1982.

[5] George L. Nemhauser & Laurence A. Wolsey, Integer and Combinatorial optimization, John Wisley & Sons, 1988.

[6] Necos christofides, worst-case analysis of a new heuristic for the travelling salesman problem.