CHAPTER TWO

TRAVELING SALESMAN PROBLEM (TSP)

2.5 Solving TSP

The TSP asks for an optimal tour through a specified set of cities. To solve a particular instance of the problem we must find a shortest tour and verify that no better tour exists. We have two types of TSP, 1st, the route A-B is same to B-A, 2nd; the route A-B is different from B-A.

In this section we describe some of the techniques employed in the Concorde code for the TSP, focusing on the difficult verification task.

Now what can we say about solution methods for the TSP? It is of course easy to develop methods that have a guarantee that is proportional to (n-1)! since the number of n-city tours is only (n-1)!/2. A much better result was obtained in 1962 by M. Held and R. Karp, who found an algorithm and a guarantee that is proportional to $n^2 * 2^n$. For any large value of n the Held-Karp guarantee is much less than (n-1)! (for n≥10).

2.5.1 Complete Enumeration Method (CEM)

This method summarized by testing all possible of TSP routs ((n-1)!).

Example (2.1): Find the minimum route for the following table using CEM, where n=4:

	A	B	C	D
A	0	2	3	7
B	2	0	5	4
C	3	5	0	6
D	7	4	6	0

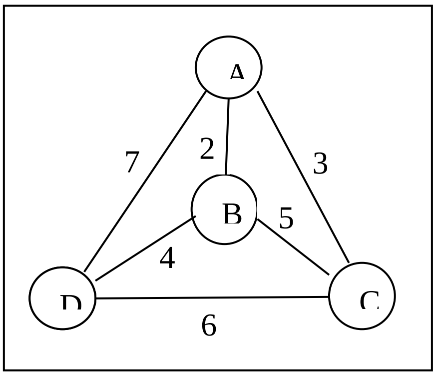
1. ABCDA, Z=20.
2. ABDCA, Z=15.
3. ACBDA, Z=19.

4. ACDBA, Z=15.
5. ADBCA, Z=19
6. ADCBA, Z=20.

The optimal solution is at the route ABDCA and ACDBA. Note that the route 1 and 6 are same, and 3 and 5 are same.

Hamiltonian Cycles

Is graph of the n-cities when the route A-B is same as B-A. for example(2.1) we have only (3) routs:



1. ABCDA, Z=20.
2. ABDCA, Z=15.
3. ACBDA, Z=19.

In general

No. of nodes	**No. of ways Creating cycles**
3	1
4	3
5	12
6	60
7	360
⋮	⋮
n	(n-1)!/2

Note: In routs which A-B different from B-A, the no. of cycles is (n-1)!.

2.5.2 Minimizing Distance Method (MDM)

Example (2.2): (www.neuroptics.com)

Lets have the following TSP

	A	B	C	D	E
A	--	7	6	8	4
B	7	--	8	5	6
C	6	8	--	9	7
D	8	5	9	--	8
E	4	6	7	8	--

1^{st} step: we have to minimize row and column.

	A	B	C	D	E
A	--	3	2	4	0
B	2	--	3	0	1
C	0	2	--	3	1
D	3	0	4	--	3
E	0	2	3	4	--

⇒

	A	B	C	D	E
A	--	3	0	4	0
B	2	--	1	0	1
C	0	2	--	3	1
D	3	0	2	--	3
E	0	2	1	4	--

2^{nd} step: calculate penalties of all 0's (least no. in row+least no. in column). All rows and columns have at least one zero.

	A	B	C	D	E
A	--	3	0^{0+1}	4	0^{0+1}
B	2	--	1	0^{3+1}	1
C	0^{1+0}	2	--	3	1
D	3	0^{2+2}	2	--	3
E	0^{1+0}	2	1	4	--

Choose max Penalty if Equal choose arbitrarily ⇒

	A	B	C	D	E
A	--	3	0^1	4	0^1
B	2	--	2	0^4	1
C	0^1	2	--	3	1
D	3	0^4	2	--	3
E	0^1	2	1	4	--

Max penalty is 4.

Cross row of B and column of D. we have **B→D** route. So we have the following reduced matrix.

	A	B	C	E
A	--	3	0	0
C	0	2	--	1
D	3	0	2	3

⇒

	A	B	C	E
A	--	3	0	0
C	0	2	--	1
D	3	--	2	3

E	0	2	1	--	E	0	2	1	--

Since we obtained the route **B→D** then we change the cell D-B to (--).

Now we will reduced the matrix by taking row and column min.

	A	B	C	E
A	--	3	0	0
C	0	2	--	1
D	1	--	0	1
E	0	2	1	--

$\Rightarrow$

	A	B	C	E
A	--	1	0^0	0^1
C	0^0	0^0	--	1
D	1	--	0^1	1
E	0^0	0^0	1	--

Max. penalty is 1. Choose **A→E** arbitrarily.

Now reduce matrix again:

	A	B	C
C	0	0	--
D	1	--	0
E	0	0	1

$\Rightarrow$

	A	B	C
C	0^1	0^0	--
D	1	--	0^2
E	--	0^1	1

We have at least one zero in each row and column. Max. penalty is 2. We have **D→C**, reduced matrix.

	A	B
C	0^0	0^0
E	--	0^0

Here we cant choose **E→A** (then can't choose **C→B**), so we have to choose **E→B**.

Lastly we have **C→A**.

So we have the following routs:

B→D, A→E, D→C, E→B and **C→A**.

Then we obtain the following complete route:

A→E→B→D→C→A, with total minimum distance:

Z=4+6+5+9+6 = 30.

Exercises (2.1):

1.

	A	B	C	D	E
A	--	3	6	2	3
B	3	--	5	2	3
C	6	5	--	6	4
D	2	2	6	--	6
E	3	3	4	6	--

2.

	A	B	C	D	E	F
A	--	20	23	27	29	34
B	21	--	19	26	31	24
C	26	28	--	15	36	26
D	25	16	25	--	23	18
E	23	40	13	31	--	10
F	27	18	12	35	16	--