



33304

myCOMPANION

Assignment - 13

* Title :- Travelling salesman Problem (Branch and Bound)

* Problem statement :- Write a program to solve the travelling salesman problem and to print the path using BB.

* Objective :- To understand and implement least cost branch and bound algorithm for solving travelling salesman problem and study BB strategy.

* Theory :-

1. Travelling Salesman Problem →

Let $G = (V, E)$ be a directed graph defining an instance of TSP. Let c_{ij} be the edge $\langle i, j \rangle$, $|V| = n$. We may assume tour starts and ends at 1. So solution space $S = \{1, \pi, 1\}$ is permutation $(2, 3, \dots, n)$. S may be organized into state space tree.

• Least cost Branch and Bound →

In order to use LCBB to search TSP tree, we need to define cost function $c(\cdot)$ and two other functions $\hat{c}(\cdot)$ and $U(\cdot)$ such that $\hat{c}(R) < c(R) < U(R)$, is such that solution node with least $c(\cdot)$ corresponds to G .

With every node in TSP state space tree we may associate a reduced cost matrix. A be reduced matrix for R node. If R is not a leaf then reduced cost matrix for R may be obtained as follows:

1. Change all entries in row i and column j of A

33304

to ∞ . This prevents the use of any more edges from i or j .

2. Set $A(j, i)$ to ∞ . This prevents the use of edge $\langle j, i \rangle$.

3. Reduce all rows and columns in the resulting matrix except for rows and columns containing only ∞ .

• Algorithm for TSP using BB \rightarrow

1. Find the no. of cities and adj. matrix.
2. Initial reduction matrix to original.
3. Find reduced cost and reduce the matrix.
4. Expand on this node and arrive at the leaf.
5. If cost of the leaf is least of all other explorable nodes then terminate.
6. Else, explore the smallest node.

• Example \rightarrow

∞	26	30	10	11
15	∞	16	4	2
3	5	∞	2	4
19	6	18	∞	3
16	4	7	16	∞

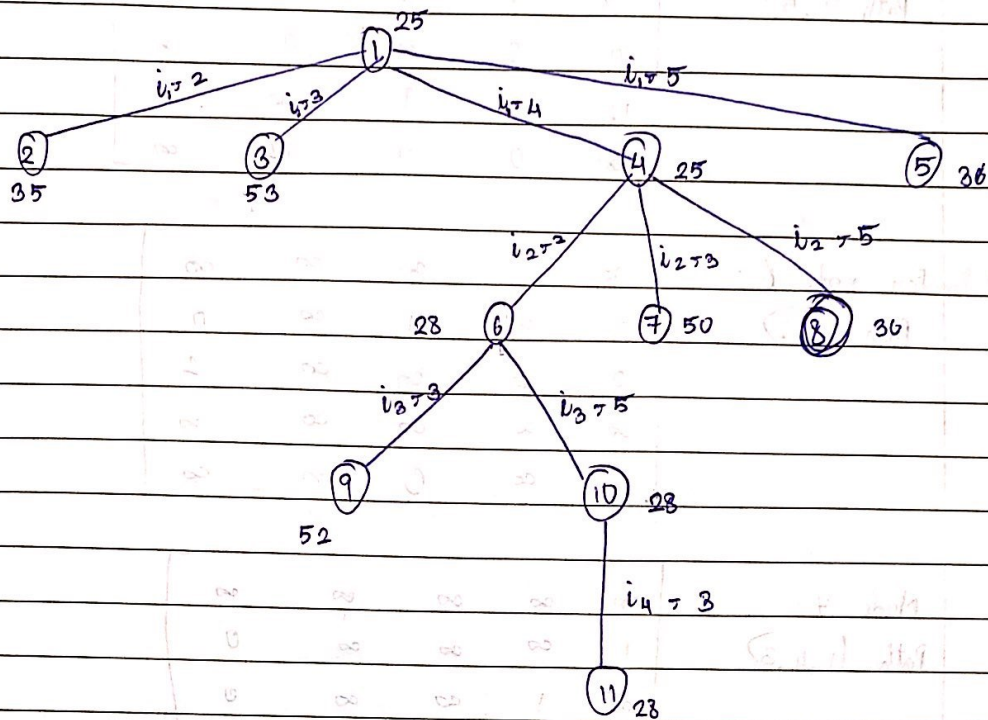
 \rightarrow

∞	10	17	0	1
12	∞	11	2	0
0	3	∞	0	2
15	3	12	∞	0
11	0	0	12	∞

Cost Matrix

Reduced cost Matrix
($L=25$)

State space tree generated by LCBP



Answer node

For node 2:

Path (1, 2)

∞	∞	∞	∞	∞
∞	∞	11	2	0
0	∞	∞	0	2
15	∞	12	∞	0
11	∞	0	12	∞

For node 3:

Path (1, 3)

∞	∞	∞	∞	∞
1	∞	∞	2	0
∞	∞	∞	0	2
4	3	∞	∞	0
0	0	∞	12	∞

For node 4:

Path (1, 4)

∞	∞	∞	∞	∞
12	∞	11	∞	0
0	3	∞	∞	2
∞	3	12	∞	0
11	0	0	∞	∞

33304

myCOMPANION

For node 5:

Path (1, 5)

∞	∞	0	∞	∞
10	∞	9	0	∞
0	3	∞	0	∞
12	0	9	∞	∞
∞	0	0	12	∞

For node 6:

Path (4, 2)

∞	∞	∞	∞	∞
∞	∞	11	∞	0
0	∞	∞	∞	2
∞	∞	∞	∞	∞
11	∞	0	∞	∞

Node 7:

Path (1, 4, 3)

∞	∞	∞	∞	∞
1	∞	∞	∞	0
∞	1	∞	∞	∞
∞	∞	∞	∞	∞
0	0	∞	∞	∞

Node 8:

Path (1, 4, 5)

∞	∞	∞	∞	∞
1	∞	0	∞	∞
0	3	∞	∞	∞
∞	∞	∞	∞	∞
∞	0	0	∞	∞

Node 9:

Path (1, 4, 2, 3)

∞	∞	∞	∞	∞
∞	∞	∞	∞	∞
∞	∞	∞	∞	0
∞	∞	∞	∞	∞
0	∞	∞	∞	∞

Node 10:

∞	∞	∞	∞	∞
∞	∞	∞	∞	∞
0	∞	∞	∞	∞
∞	∞	∞	∞	∞
∞	∞	∞	∞	∞



83304

myCOMPANION

Input : Cost matrix TSP graph

Output : Reduced matrix obtained by applying LCB.

* Conclusion :- The least cost branch and bound strategy for TSP is studied and implemented