SNS COLLEGE OF TECHNOLOGY, COIMBATORE –35

Branch and Bound: Assignment Problem

Branch and Bound

- Branch-and-bound technique is an improvement of backtracking technique
- Branch-and-bound strategy is applicable to optimization problems.
- For each node (partial solution) of a state-space tree, computes a bound on the value of the objective function for all descendants of the node (i.e., extensions of the partial solution)

Branch and Bound

- Uses the bound for:
  - eliminating certain nodes as “nonpromising” to prune the tree - if a node’s bound is not better than the best solution seen so far
  - guiding the search through state-space tree

Assignment Problem

- Assignment problem is to assign *n* people to *n* jobs so that the total cost of the assignment is as small as possible.
- An instance of the assignment problem is specified by an *n*-by-*n* cost matrix *C*.

Example:

	Job 1	Job 2	Job 3	Job 4
Person *a*	9	2	7	8
Person *b*	6	4	3	7
Person *c*	5	8	1	8
Person *d*	7	6	9	4

37

Assignment Problem

- Select one element in each row of the cost matrix *C* so that
  - no two selected elements are in the same column and
  - their sum is minimized.

Assignment Problem

- Lower bound: The cost of any solution, including an optimal one, cannot be smaller than the sum of the smallest elements in each of the matrix's rows
- Thus, any solution to the given instance will have total cost at least: $2 + 3 + 1 + 4 = 10$.
- We choose the node with the smallest lower-bound value as the most promising node.

Assignment Problem

Example: Levels 0, 1 of the State-Space Tree

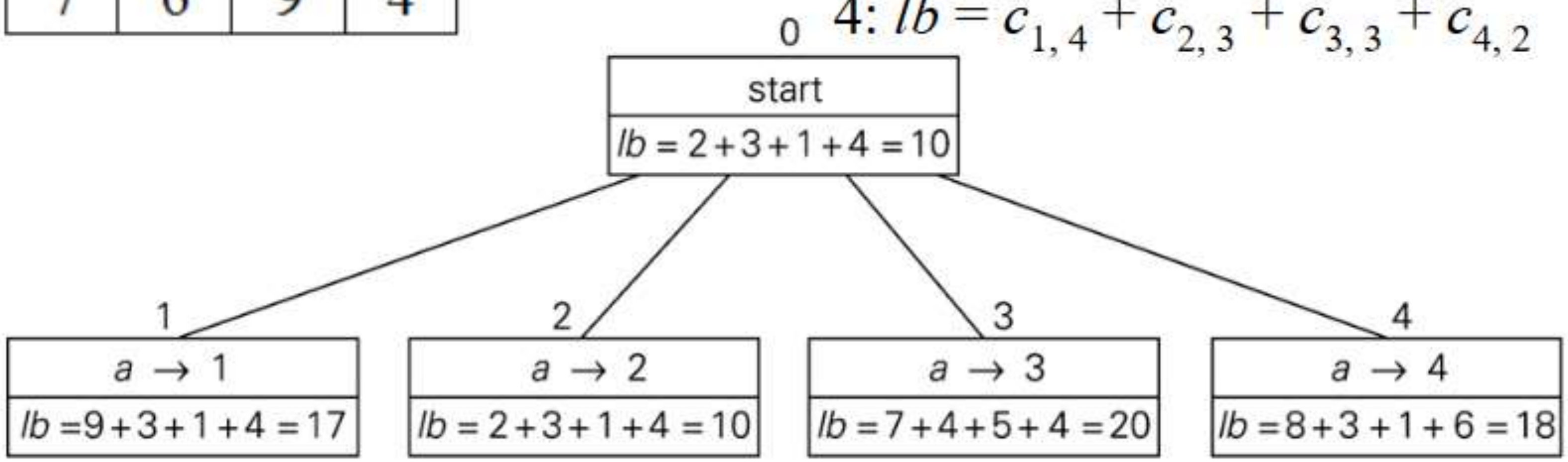
9	2	7	8
6	4	3	7
5	8	1	8
7	6	9	4

1: $lb = c_{1,1} + c_{2,3} + c_{3,3} + c_{4,4}$

2: $lb = c_{1,2} + c_{2,3} + c_{3,3} + c_{4,4}$

3: $lb = c_{1,3} + c_{2,2} + c_{3,1} + c_{4,4}$

4: $lb = c_{1,4} + c_{2,3} + c_{3,3} + c_{4,2}$



The most promising of them is node 2 because it has the smallest lower-bound value.

Assignment Problem

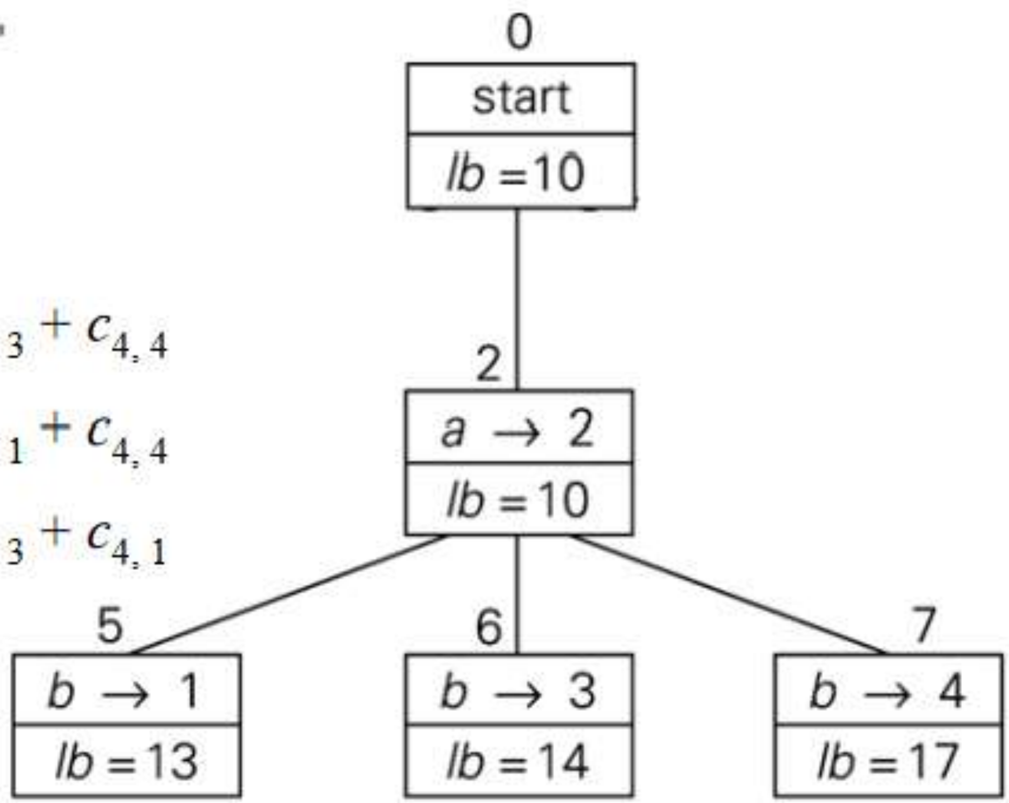
Example: Levels 0, 1, 2 of the State-Space Tree

9	2	7	8
6	4	3	7
5	8	1	8
7	6	9	4

5: $lb = c_{1,2} + c_{2,1} + c_{3,3} + c_{4,4}$

6: $lb = c_{1,2} + c_{2,3} + c_{3,1} + c_{4,4}$

7: $lb = c_{1,2} + c_{2,4} + c_{3,3} + c_{4,1}$



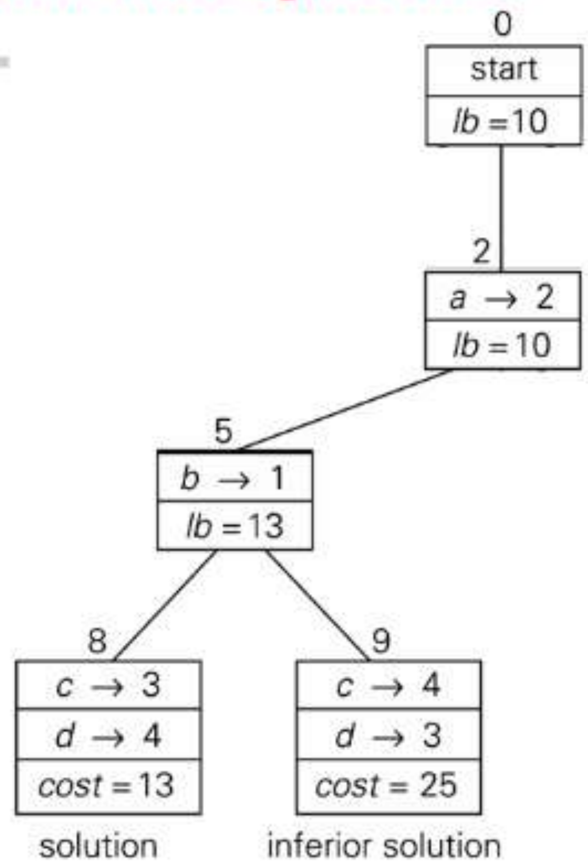
Node 5: $lb = 2 + 6 + 1 + 4 = 13$. Node 6: $lb = 2 + 3 + 5 + 4 = 14$. Node 7: $lb = 2 + 7 + 1 + 7 = 17$.

41

Assignment Problem

Example: Levels 0, 1, 2, 3 of the State-Space Tree

9	2	7	8
6	4	3	7
5	8	1	8
7	6	9	4



8: $cost = c_{1,2} + c_{2,1} + c_{3,3} + c_{4,4}$

$= 2 + 6 + 1 + 4 = 13$

9: $cost = c_{1,2} + c_{2,1} + c_{3,4} + c_{4,3}$

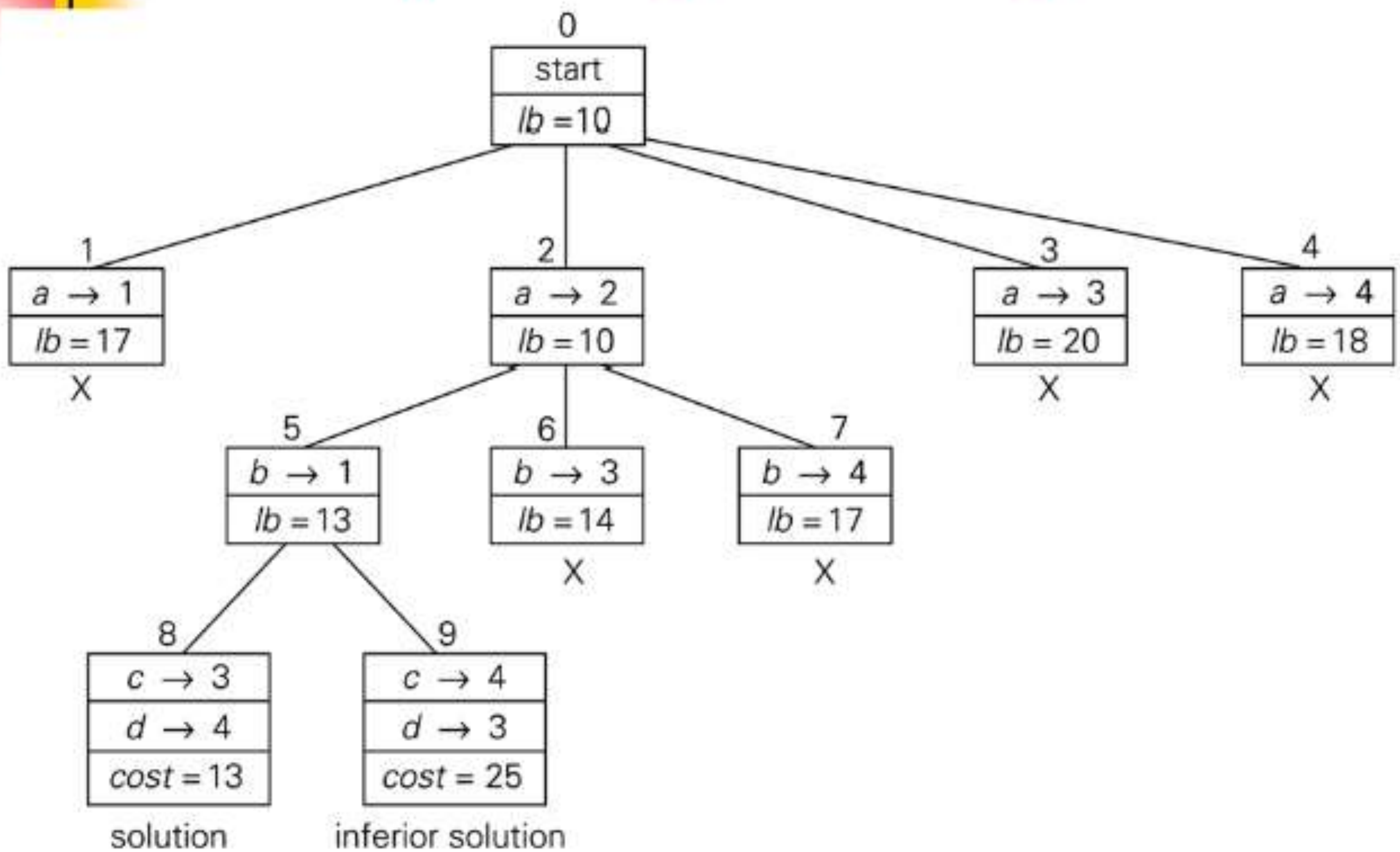
$= 2 + 6 + 8 + 9 = 25$

The optimal solution is $\{a \rightarrow 2, b \rightarrow 1, c \rightarrow 3, d \rightarrow 4\}$ (or (2, 1, 3, 4)) with the total cost of 13.

42

Assignment Problem

Example: Complete State-Space Tree



Assignment Problem

Example: Complete State-Space Tree

Node 0: $lb = c_{1,2} + c_{2,3} + c_{3,3} + c_{4,4} = 2 + 3 + 1 + 4 = 10$

Node 1: $lb = c_{1,1} + c_{2,3} + c_{3,3} + c_{4,4} = 9 + 3 + 1 + 4 = 17$
Node 2: $lb = c_{1,2} + c_{2,3} + c_{3,3} + c_{4,4} = 2 + 3 + 1 + 4 = 10$
Node 3: $lb = c_{1,3} + c_{2,2} + c_{3,1} + c_{4,4} = 7 + 4 + 5 + 4 = 20$
Node 4: $lb = c_{1,4} + c_{2,3} + c_{3,3} + c_{4,2} = 8 + 3 + 1 + 6 = 18$

Node 5: $lb = c_{1,2} + c_{2,1} + c_{3,3} + c_{4,4} = 2 + 6 + 1 + 4 = 13$
Node 6: $lb = c_{1,2} + c_{2,3} + c_{3,1} + c_{4,4} = 2 + 3 + 5 + 4 = 14$
Node 7: $lb = c_{1,2} + c_{2,4} + c_{3,3} + c_{4,1} = 2 + 7 + 1 + 7 = 17$
Node 8: $cost = c_{1,2} + c_{2,1} + c_{3,3} + c_{4,4} = 2 + 6 + 1 + 4 = 13$
Node 9: $cost = c_{1,2} + c_{2,1} + c_{3,4} + c_{4,3} = 2 + 6 + 8 + 9 = 25$