Design and Analysis of Algorithms

Branch and Bound

Reetinder Sidhu

Department of Computer Science and Engineering

Design and Analysis of Algorithms

Branch and Bound

Reetinder Sidhu

Department of Computer Science and Engineering

Branch and Bound

Unit 5: Limitations of Algorithmic Power and Coping with the Limitations

- Dynamic Programming
  - Computing a Binomial Coefficient
  - The Knapsack Problem
  - Memory Functions
  - Warshall's and Floyd's Algorithms
  - Optimal Binary Search Trees
- Limitations of Algorithmic Power
  - Lower-Bound Arguments
  - Decision Trees
  - P, NP, and NP-Complete, NP-Hard Problems
- Coping with the Limitations
  - Backtracking
  - **Branch and Bound**

Concepts covered

- Backtracking
  - General Approach
  - Knapsack Problem
  - Assignment Problem
  - Travelling Salesman Problem

Branch and Bound

Introduction

- An enhancement of backtracking
- 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 (extensions of the partial solution)
- Uses the bound for:
  - ruling out 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

Branch and Bound
Example: Assignment Problem

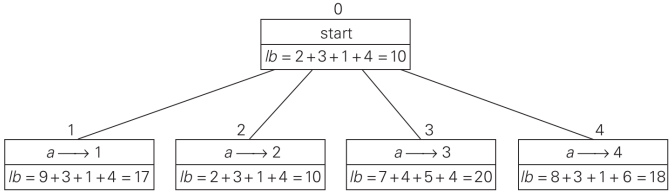
- Select one element in each row of the cost matrix C so that:
  - no two selected elements are in the same column
  - the sum is minimized
- Example

	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

- Lower bound (sum of smallest elements in each row): $2 + 3 + 1 + 4 = 10$
- Best-first branch-and-bound variation: Generate all the children of the most promising node

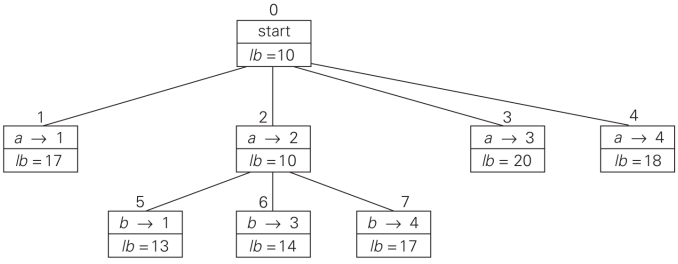
Branch and Bound

Example: First two levels of the state-space tree

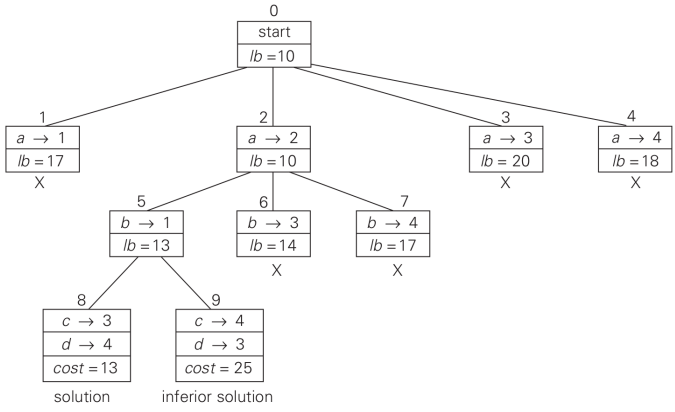


Branch and Bound

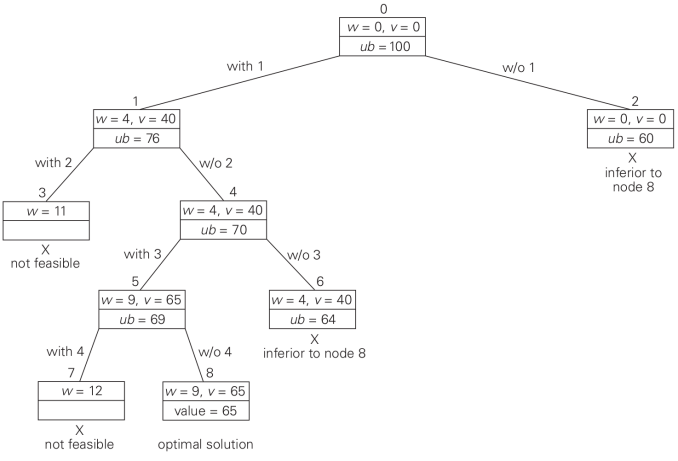
Example: First three levels of the state-space tree



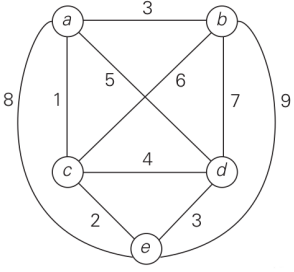
Branch and Bound
Example: Complete state-space tree

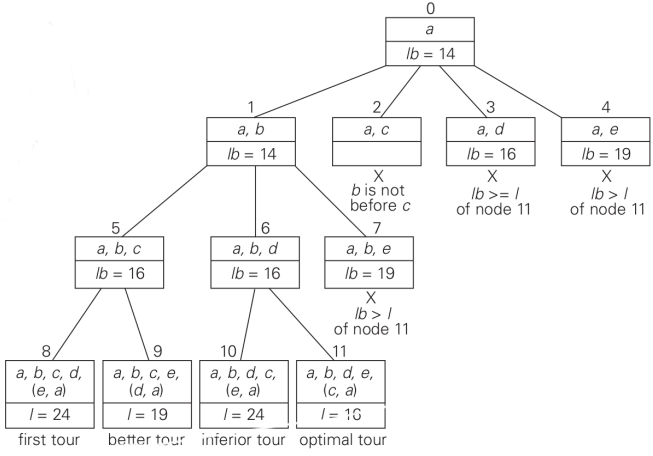


Branch and Bound
Example: Knapsack Problem



Branch and Bound
Example: Traveling Salesman Problem





Think About It

- What data structure would you use to keep track of live nodes in a best-first branch-and-bound algorithm?
- Solve the assignment problem by the best-first branch-and-bound algorithm with the bounding function based on matrix columns rather than rows