Uninformed Search

Asmaa Elbadrawy
PhD, Lecturer
IFT Program, ASU

Arizona State University

Uninformed Search

An uninformed search method has no knowledge of how close each state is to the goal state.

Path Finding with Uninformed Search

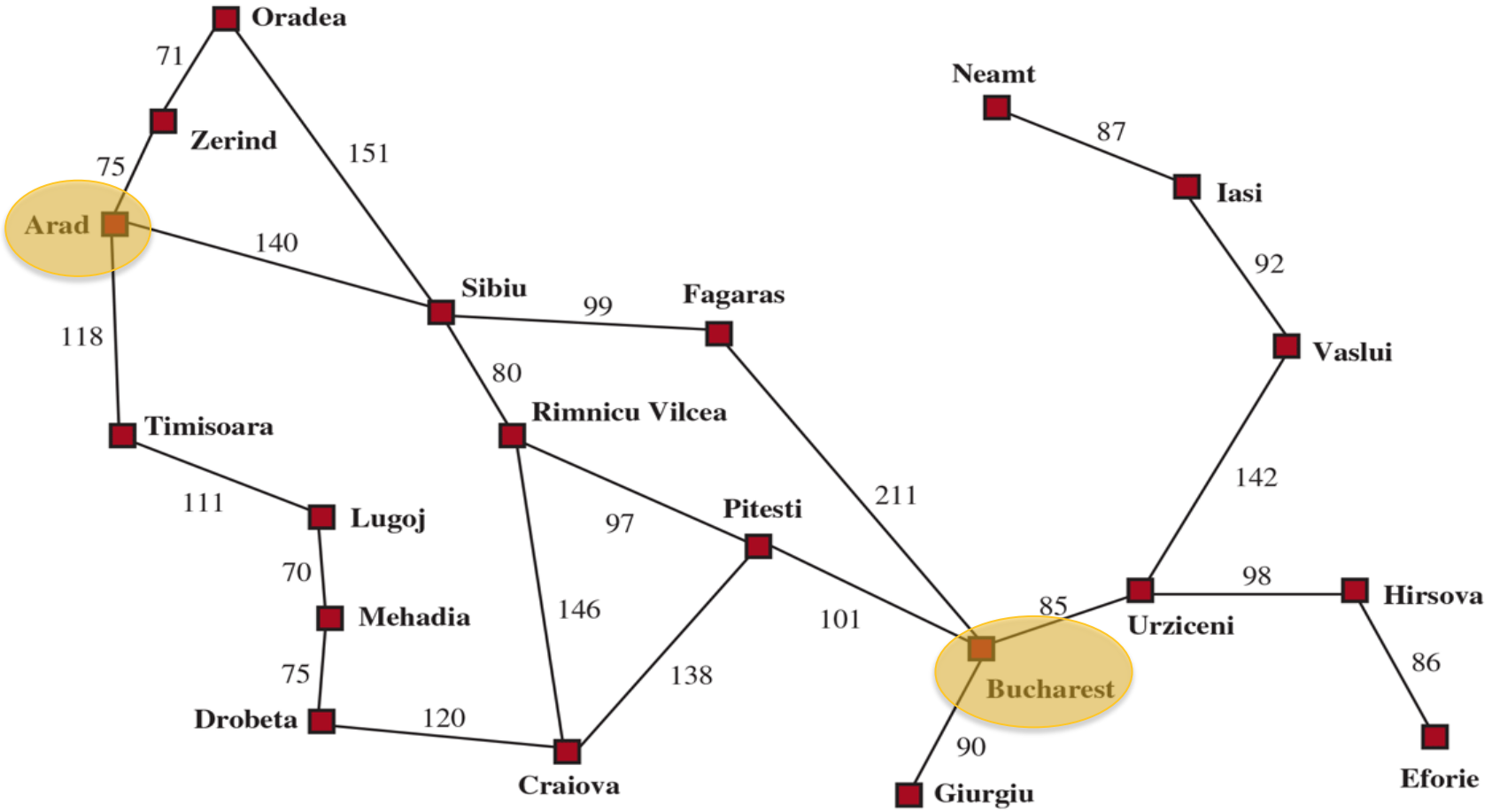


Fig 3.1, Russell & Norvig's Textbook

Algorithm has no clue whether going through Sibiu or Zerind is closer to Bucharest.

Uninformed Search Strategies

- Breadth-First
- Dijkstra’s Uniform Cost
- Depth First
- Depth-Limited
- Iterated Deepening

Breadth-First Search

Expand root, then all its child nodes, and proceed one level at a time.

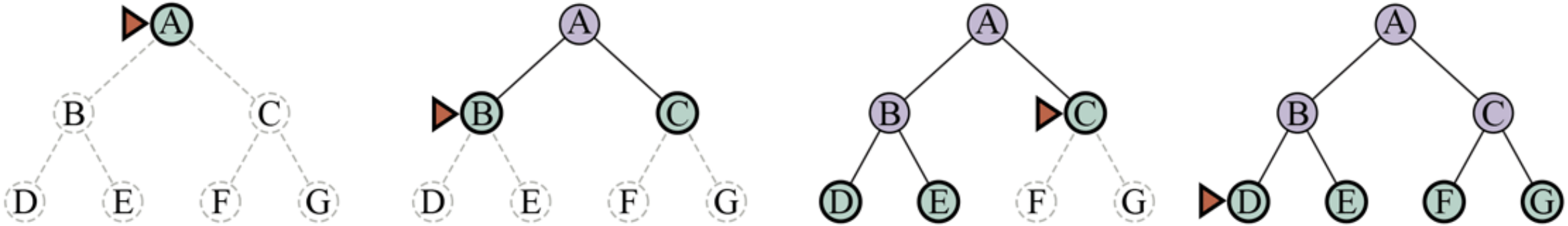


Fig 3.8, Russell & Norvig's Textbook

Always finds the solution with the minimum number of steps from root.

Breadth-First Search

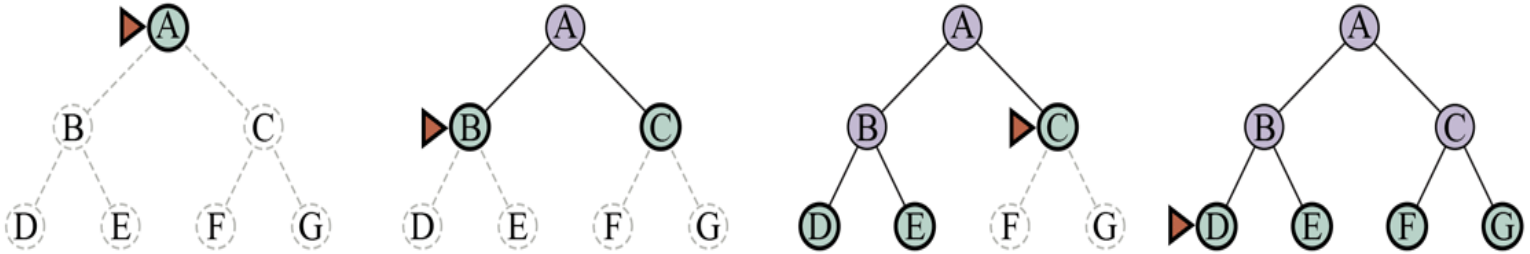


Fig 3.8, Russell & Norvig's Textbook

Problem with memory and time complexity.

If each node has 2 children:
Level 1 has 1 node
Level 2 has 2 nodes
Level 3 has 4 nodes
Level 4 has 8 nodes
Level 5 has 16 nodes

➔

If each node has b children:
Level 1 has b^1 node
Level 2 has b^2 nodes
Level 3 has b^3 nodes
Level 4 has b^4 nodes
Level 5 has b^5 nodes

Breadth-First Search

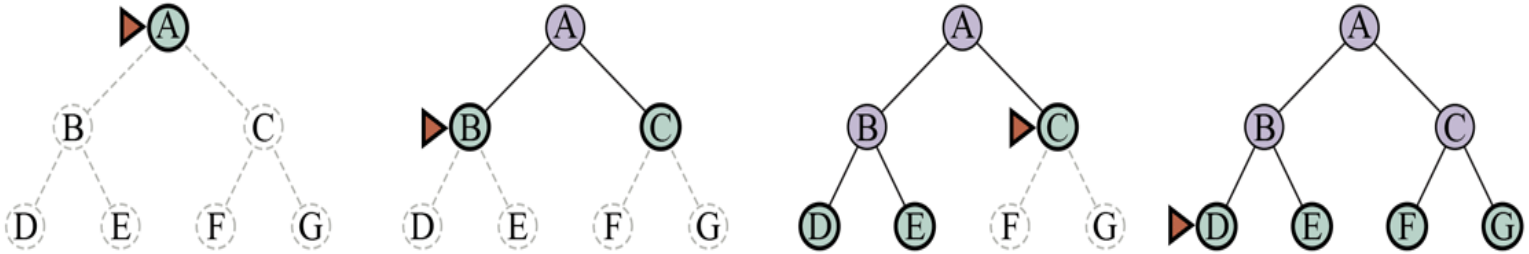


Fig 3.8, Russell & Norvig's Textbook

Problem with memory and time complexity.

Level 15 has b^{15} nodes, for b=10 ➔ total number of nodes = 10^{15}

If each node requires 1Kb of memory ➔ total required memory = 10^{15}Kb ~ 1 million terabytes!

If we can process 1 million nodes/sec, we need 10^{12} seconds to process all nodes, that's (10^{12}/60 /60 /24 /365) ~ 31.7 years!!

An Exponential-complexity search problems cannot be solved using uninformed search methods except for the smallest instances.

Dijkstra's Algorithm (Uniform Cost Search)

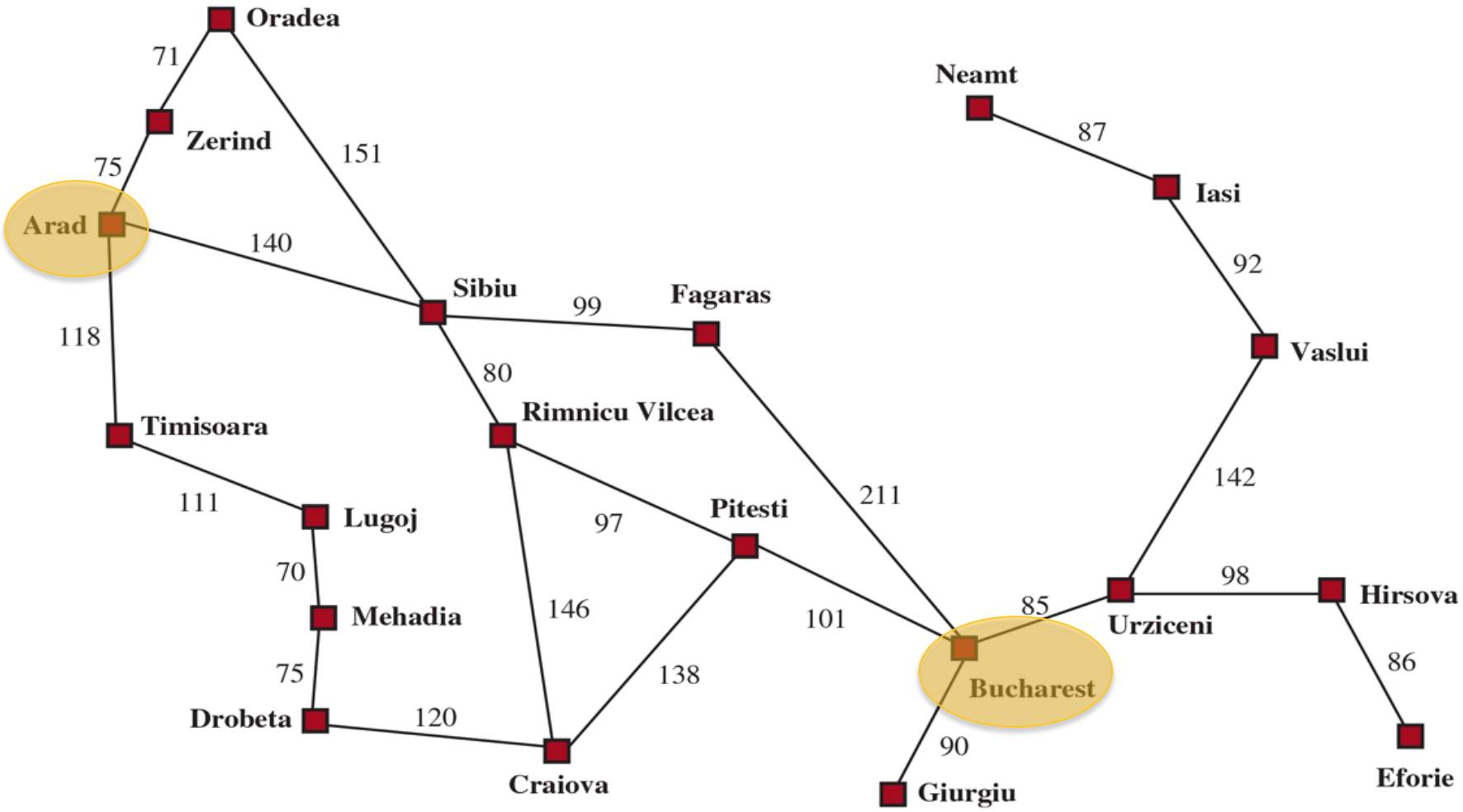


Fig 3.1, Russell & Norvig's Textbook

Each path has a different cost. Dijkstra's expands the minimum cost nodes. This way it reaches the optimal solution without having to expand all nodes at each level.

Complete and Cost-Optimal

Depth-First Search

Solves the memory problem of Breadth-First Search.

Expand one path to the end, then moves to the next path and so on.

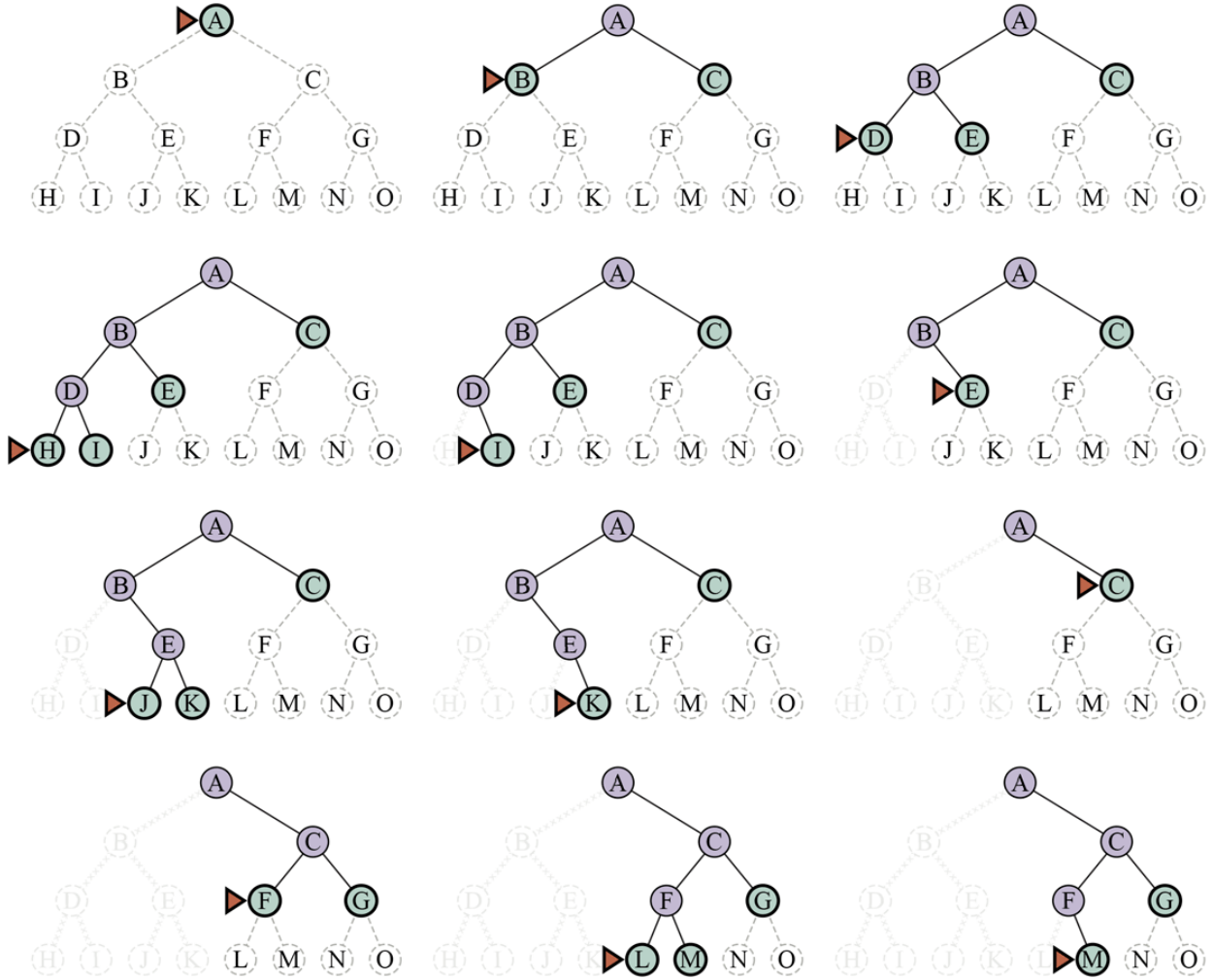


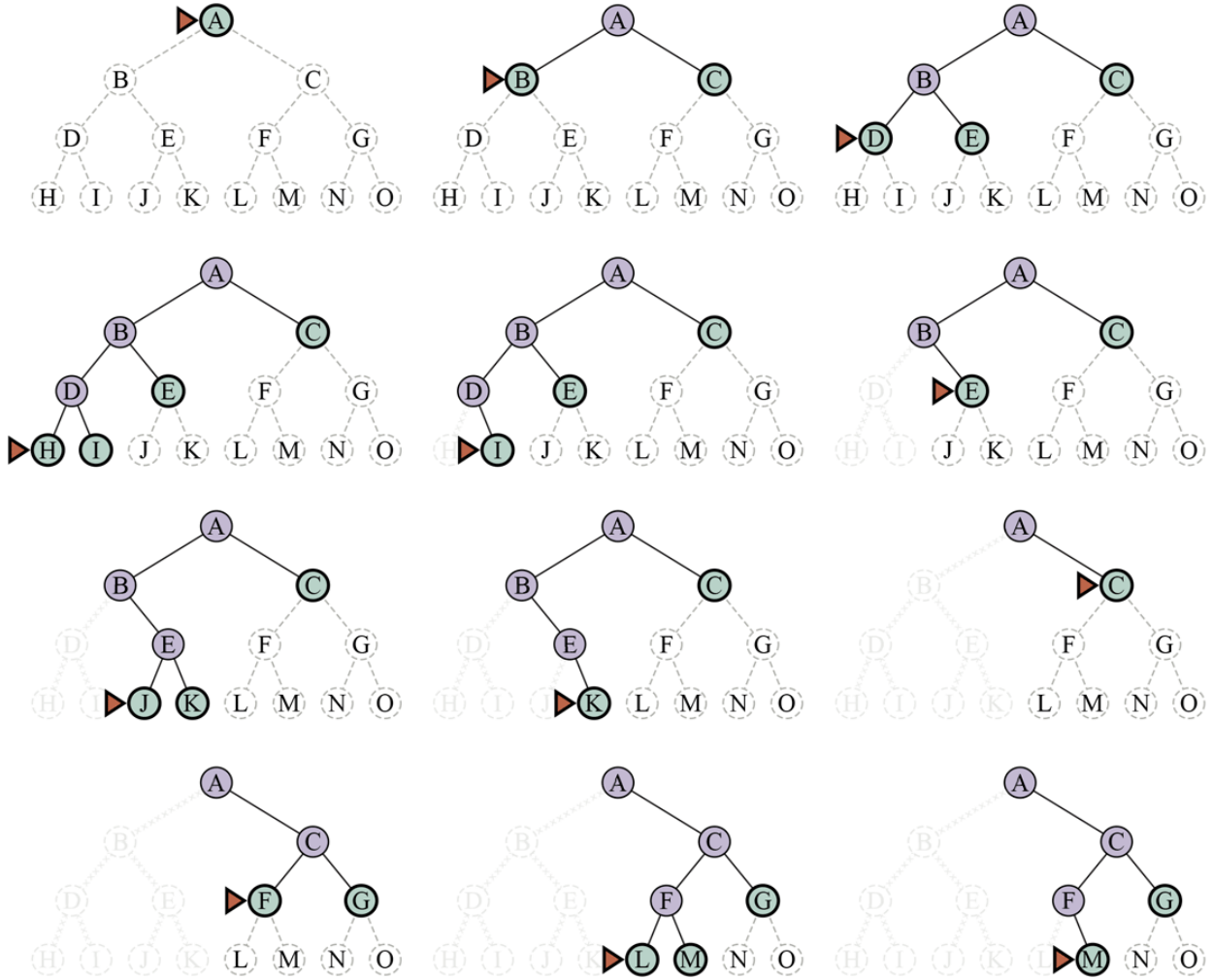
Fig 3.11, Russell & Norvig's Textbook

Not cost-optimal. Returns the first solution, which may not be the cheapest.

Must check for cycles to avoid infinite looping!

Depth-Limited Search

Avoids infinite loops by putting a cutoff on the maximum depth the algorithm should reach.



Not cost-optimal. Returns the first solution, which may not be the cheapest.

Fig 3.11, Russell & Norvig's Textbook

What if the solution exists at a depth greater than the maximum depth limit?

Iterative Deepening Search

Overcomes the limits of the depth-limited search.

Keeps increasing the depth limit one step at a time until a solution is found.

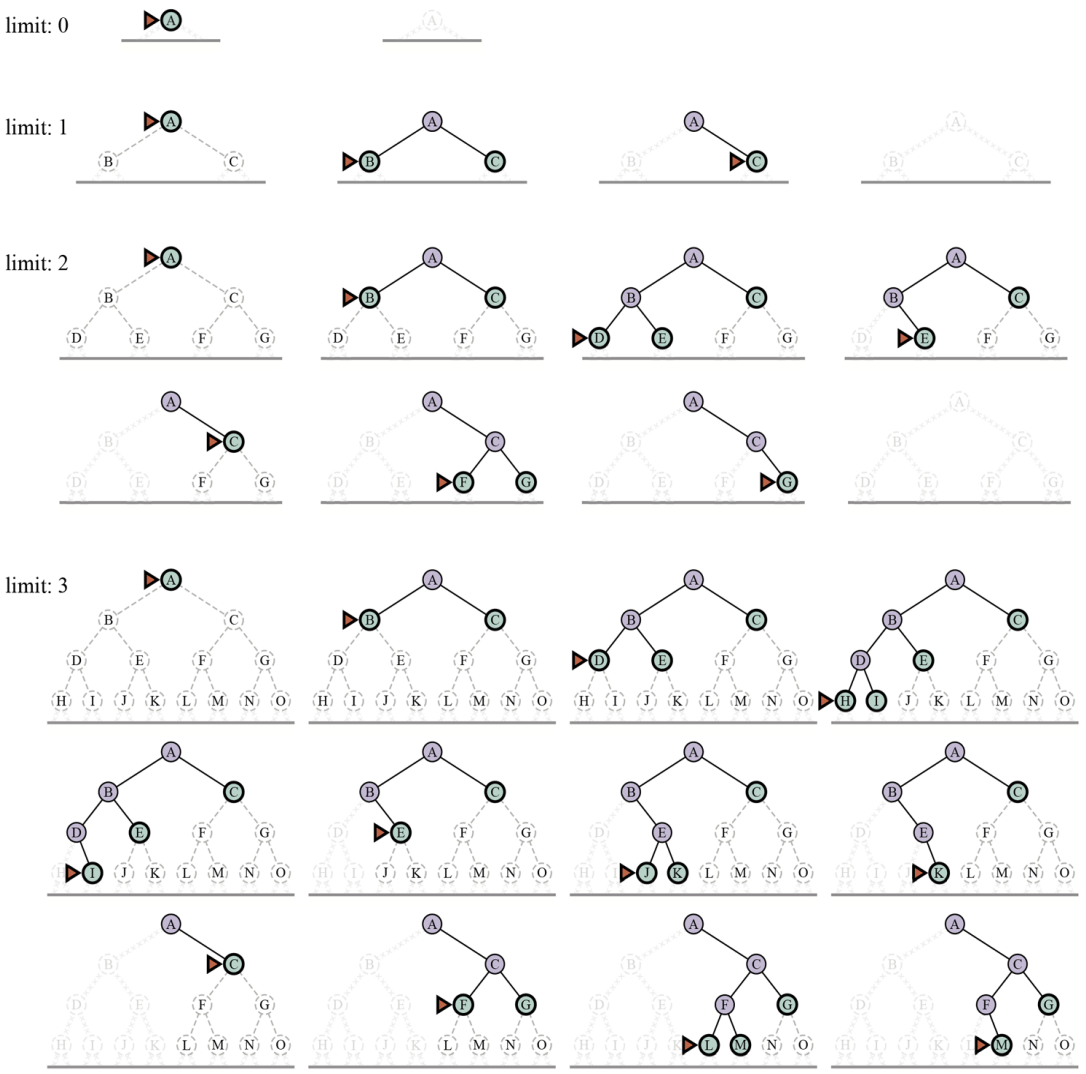
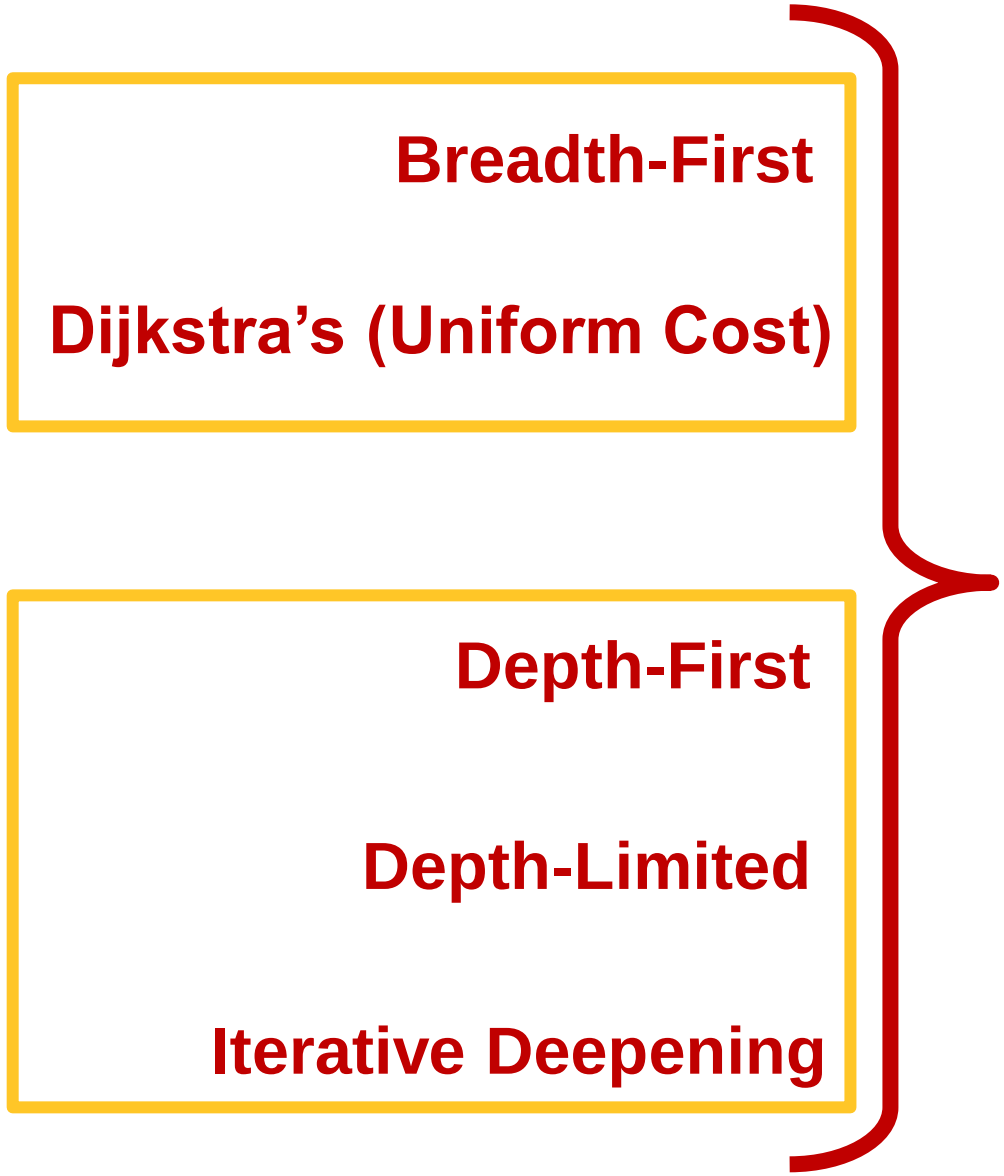


Fig 3.13, Russell & Norvig's Textbook

Not cost-optimal. Returns the first solution, which may not be the cheapest.



Uninformed Search Methods