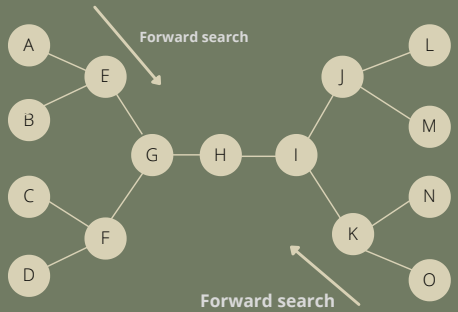
Bi-Directional Search

Bidirectional Search is Graph Search Algorithm where two graph traversals (BFS) take place at the same time and is used to find the shortest distance between a fixed start vertex and end vertex.



- Take A is the initial node and O is the goal node, and H is the intersection node.
- We will start searching simultaneously from start to goal node and backward from goal to start node.
- Whenever the forward search and backward search intersect at one node, then the searching stops.

Time Complexity: O(bd).
Space Complexity: O(bd).

Two main types of bidirectional searches are as follows:

- Front to back or BFEA
- Front to Front or BFFA

Advantages

- Faster results as it drastically reduces the time taken by the search by having simultaneous searches.
- It also saves resources for users as it requires less memory capacity to store all the searches.

Disadvantages

- The fundamental issue is that the user should be aware of the goal state to use bidirectional search.
- The algorithm must be robust enough to understand the intersection when the search should come to an end or else there's a possibility of an infinite loop.
- It is also not possible to search backwards through all states.

References

- https://www.sciencedirect.com/topics/computer-science/bidirectional-search
- https://www.javatpoint.com/ai-uninformed-search-algorithms
- https://www.geeksforgeeks.org/bidirectional-search/

Team

- SHRIYA PINGULKAR
  16010420078
  BATCH 3
- DITI DIVEKAR
  16010420091
  BATCH 3