8PUZZLE PROBLEM

Pietro Cicciari,
Mario Choto,
Israel Sanchez,
Raj Kumar,
Armon Lee,
Venkat Sairam Ravala

COMP 469 - ARTIFICIAL INTELLIGENCE

CONTENT

VARIOUS ALGORITHMS CAN SOLVE THIS PROBLEM:

The 8-puzzle problem involves arranging 8 tiles on a 3x3 grid so they are in numerical order.

- Breadth-First Search (BFS)
- Depth-First Search (DFS)
- A* Search Algorithm
- Greedy Best-First Search
- Iterative Deepening Search

FORMALIZING THE PROBLEM

THE 8-PUZZLE PROBLEM CAN BE DEFINED AS:

- Initial State: A starting configuration of the grid.
- Goal State: The target configuration (1 to 8, with empty space at the end).
- Actions: Move the empty space (up, down, left, right).
- Transition Model: Defines the result of a move.
- Action Cost: Uniform cost for each move (typically 1).

ALGORITHMS OVERVIEW

WE EXPLORED THE FOLLOWING ALGORITHMS:

1 BREADTH-FIRST SEARCH (BFS)

2 DEPTH-FIRST SEARCH(DFS)

3 A* SEARCH ALGORITHM

4 GREEDY BEST-FIRST SEARCH

5 ITERATIVE DEEPENING SEARCH

BREADTH-FIRST SEARCH (BFS)

1
EXPLORES ALL POSSIBLE MOVES LEVEL BY LEVEL.

2
GUARANTEED TO FIND THE SHORTEST PATH.

3
HIGH MEMORY USAGE DUE TO STORING ALL NODES.

4
TIME COMPLEXITY: O(B^D)

DEPTH-FIRST SEARCH (DFS)

1

EXPLORES AS FAR AS POSSIBLE ALONG ONE BRANCH BEFORE BACKTRACKING.

2

MAY NOT FIND THE SHORTEST PATH.

3

LOWER MEMORY USAGE COMPARED TO BFS.

4

TIME COMPLEXITY: O(B^M)

A* SEARCH ALGORITHM

1

USES HEURISTICS TO GUIDE THE SEARCH TOWARDS THE GOAL.

2

OFTEN FINDS THE OPTIMAL SOLUTION QUICKLY.

3

TIME COMPLEXITY: O(B^D) WITH HEURISTIC IMPROVEMENTS.

4

BEST BALANCE OF SPEED AND ACCURACY.

GREEDY BEST-FIRST SEARCH

1

EXPANDS THE NODE CLOSEST TO THE GOAL USING A HEURISTIC.

2

FAST BUT MAY NOT FIND THE OPTIMAL PATH.

3

USES HEURISTICS LIKE MANHATTAN DISTANCE.

4

CAN GET STUCK IN LOCAL MINIMA.

ITERATIVE DEEPENING SEARCH (IDS)

1

COMBINES DEPTH-FIRST AND BREADTH-FIRST SEARCH

2

MEMORY EFFICIENT AND GUARANTEES OPTIMAL SOLUTION.

3

REPEATS DEPTH-FIRST SEARCH WITH INCREASING DEPTH LIMITS.

4

EFFICIENTLY BALANCES DEPTH EXPLORATION WITH SOLUTION OPTIMALITY.

PERFORMANCE ANALYSIS

- A* consistently outperforms other algorithms in terms of speed and path optimality.
- BFS is thorough but slow and memory-intensive.
- DFS is fast but not always reliable for finding the optimal path.
- Greedy Best-First is fast but may miss the optimal path.
- Iterative Deepening combines BFS and DFS benefits.

CONCLUSION

- A* Search Algorithm is the most effective method for solving the 8-puzzle problem.
- Its use of heuristics makes it a powerful and efficient choice.
- Future improvements could focus on refining heuristics for even better performance.