Path Planning in a Square Room

1. Problem Definition

Room Representation The room is a 5×5 square grid, with:

- **Start (S)**: Bottom-left corner at $(0, 1)$.
- **Goal (G)**: Top-right corner at $(4, 4)$.
- **Obstacles (X)**: The center of the room, covering cells $(2, 2)$, $(2, 3)$, $(3, 2)$, and $(3, 3)$.

Constraints

- The robot cannot enter or stop at cells marked as obstacles.
- The robot must follow the shortest path from S to G.

2. Path Planning Algorithm

We use the **Breadth-First Search (BFS)** algorithm, which systematically explores all possible paths layer by layer and guarantees finding the shortest path in an unweighted grid.

Algorithm Steps

1. **Start at the initial position:** Place the robot at $S = (0, 1)$.
2. **Explore valid moves:** Move to neighboring cells (up, down, left, right) that:
   - Are within grid bounds.
   - Are not marked as obstacles.
   - Have not been visited already.
3. **Record the path:** Keep track of each cell's parent to reconstruct the path.
4. **Stop at the goal:** Once the robot reaches $G = (4, 4)$, terminate the search.
5. **Reconstruct the shortest path:** Backtrack from G to S using the parent relationships.

3. Manual Solution of the Problem

Grid Layout The initial grid:

$$\begin{array}{ccccc} 0 & S & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & X & X & 0 \\ 0 & 0 & X & X & 0 \\ 0 & 0 & 0 & 0 & G \end{array}$$

Shortest Path

1. Starting at $S = (0, 1)$, move step-by-step:

$$(0,1) \to (0,2) \to (0,3) \to (0,4) \\ \to (1,4) \to (2,4) \to (3,4) \to (4,4).$$

2. **Path Coordinates:** $(0,1) \to (0,2) \to (0,3) \to (0,4) \to (1,4) \to (2,4) \to (3,4) \to (4,4)$.
3. **Path Length:** 7 steps.

4. Visualize the Path

Grid with Path Mark the path with dots:

$$\begin{array}{ccccc} 0 & S & . & . & . \\ 0 & 0 & 0 & 0 & . \\ 0 & 0 & X & X & . \\ 0 & 0 & X & X & . \\ 0 & 0 & 0 & 0 & G \end{array}$$

Explanation The path avoids the obstacle at the center and moves directly to the goal using the shortest route.

5. Animation (Conceptual)

To animate the motion:

1. Highlight the robot's current position on the grid.
2. Move step-by-step along the path, updating the position and marking the visited cells.
3. Stop at the goal position, completing the animation.

Note

In a path planning sense, breadth first search will always find the path from the starting position to the goal as long as there is an actual path that can be found. Breadth first search is only optimal if the path cost is the same for each direction.

References

[1] https://www.daslhub.org/unlv/wiki/lib/exe/fetch.php?media=dylanw:bfsdfstutorial.pdf