

1. A (A-star) Search Algorithm*

Overview:

Graph traversal and path-finding algorithm for efficiently finding shortest paths. Uses a combination of **$g(n)$** (the actual cost to reach the current node) and **$h(n)$** (the estimated cost from the current node to the goal) to make decisions. Emphasis of this algorithm is on the heuristic, which makes or breaks the effectivity.

Pros:

- **Efficiency:** A* is known for being faster than other algorithms, especially when a good heuristic is available. It balances between DFS's deep search and BFS's broad search, giving both completeness and optimality.
- **Customizable Heuristic:** The heuristic can incorporate climber height, wall angle, hold type/size, and distance between holds.
- **Guaranteed Optimal Path:** Assuming an admissible heuristic

Cons:

- **High Memory Usage:** A* keeps all generated nodes in memory, which can end up using a good amount of memory.
- **Dependent on Heuristic Quality:** If heuristic is poor, the solution will be too.

Suitability for Bouldering:

In my opinion, A* is the best algorithm to employ here, since we can design a heuristic that reflects the varied challenges of climbing. For example, it can account for the effort required to reach certain holds, the difficulty of the holds, and how wall angle impacts movement. We can avoid exploring every path by following the lowest estimated cost to goal.

Ranking: 1st

2. Dijkstra's Algorithm

Overview:

Dijkstra's algorithm finds the shortest path from a starting node to all other nodes in a graph. It explores all possible paths uniformly, making it similar to A* without the heuristic. It systematically expands the nearest node that hasn't been visited yet.

Pros:

- **Simple and Reliable:** Dijkstra's algorithm is easy to implement and always finds the shortest path in terms of edge weight. It's especially good when edge costs (i.e., distance or effort) are uniform or when no domain-specific heuristic is available.
- **Guaranteed Optimal Path:** Since it evaluates every possible path uniformly, it is guaranteed to find the optimal path.

Cons:

- **Inefficient for Large Graphs:** Dijkstra's algorithm explores every node without guidance from a heuristic, making it slower and more computationally expensive than A* for larger or more complex problems.

Suitability for Bouldering:

If we were to make edge costs take into account the many difficulty multipliers in climbing, this algorithm could work well. The issue is in systematically exploring every single path, which will end up being much more expensive than using the heuristic guided search in A*.

Ranking: 3rd

3. Genetic Algorithm (GA)

Overview:

Genetic algorithms (GAs) are a type of evolutionary algorithm that solve problems through a process mimicking natural selection. The algorithm evolves the population over several

generations, selecting the fittest paths and applying crossover and mutation to generate better paths.

Pros:

- **Good for Non-linear Problems:** GAs are effective at solving complex, non-linear problems that are difficult to solve using traditional algorithms. They can handle large search spaces and don't require a well-defined heuristic.
- **Adaptability:** GAs can be adapted to incorporate multiple factors like hold type, wall angle, and distance as part of the fitness function.

Cons:

- **Computationally Expensive:** GAs require multiple generations and a large population of solutions to converge, which can be slow for large or complex problems.
- **No Guarantee of Optimal Solution:** GAs often find good solutions but are not guaranteed to find the best possible solution, especially within a limited number of generations.

Suitability for Bouldering:

GAs are likely best suited for climbing routes with non-linear paths (down-climbing, jumps, traverses, which are mostly out-of-scope if we are limiting to difficulty V3 problems and below). However, for indoor bouldering, where the problem can be represented as a graph, GAs may be overkill. They might find interesting solutions, but they are computationally expensive and may not always find the optimal solution. While they offer flexibility in dealing with complex climbing factors, their lack of guaranteed optimality and higher computational cost makes them less ideal than A*.

Ranking: 2nd

Conclusion and Ranking for Indoor Bouldering Path-Finding

- **1st - A*:** A* search stands out due to its balance of efficiency, flexibility, and guaranteed optimality assuming a well-defined heuristic. By customizing the

heuristic to include factors like climber height, hold type/size, and wall angle, A* can provide highly accurate and practical path recommendations for climbers.

- **2nd - Genetic Algorithm (GA):** GAs offer flexibility and can explore a wider range of potential solutions, but don't guarantee optimality and can be very expensive.
- **3rd - Dijkstra's Algorithm:** Dijkstra's algorithm guarantees the shortest path but lacks the efficiency and adaptability needed for indoor bouldering, where the difficulty of holds and wall angles vary significantly.

Climbing holds



WHEN YOU BECOME FAMILIAR WITH DIFFERENT TYPES OF HOLDS YOU'LL BE ABLE TO PREDICT HOW TO USE THEM JUST BY LOOKING AT THEM. PLUS, LEARNING THE LINGO WILL MAKE SHARING SEQUENCES OF MOVES WITH YOUR FRIENDS MUCH EASIER.

JUG

A climber's best friend. Big and deep, it's the kind of hold you can get your fingers securely behind. When using a jug, keep your arms straight to avoid them getting tired too quickly.



POCKET

A hold with a hole in it. For three finger hand pockets use all fingers apart from your little one (unless you can fit your little finger in too).



SLOPER

A rounded hand hold with no lip or edge of any kind. The secret to slopers is to get as much surface area of the hand on the climbing hold as possible. Slopers work best when your body is below the hold.



PINCH

The kind of hold that you'll grip by squeezing with your fingers and thumb placed on opposing sides. Pinches come in all different sizes and shapes. Some are slopy, some are in-cut, and some are easy like jugs.



UNDERCUT

A hand hold oriented upside down, which you hold with your palm facing upwards. It can be difficult to get a good grip when you are below these holds but it becomes much easier as you climb higher.



CRIMP

Imagine the top of a door frame. A crimp is a small hold with a positive edge for your fingers. Crimps require finger strength, which can take a while to gain. When you first start climbing it is good to build finger strength very slowly and gently to avoid injury. Try not to over use crimps.





Question: If a climb spans across multiple faces is it feasible to have multiple pictures we stitch together? video?