**Difference Between Different Heuristic Based**

**Informed Search Algorithms**

Prepared By

19L-1236

CS-6A

Farhan Ali

National University of Computer & Emerging Sciences

**Abstract**

The objective of this report is to differentiate three different heuristic based informed search algorithms with the help of the diagrams. In this analysis, I have tested all of these three algorithms (A8, RTA\*, LRTA\*) on the same environment and found out that A\* and LRTA\* has successfully found the shortest path quickly which proves their property of optimality and completeness. Unlike the previous two algorithms, RTA\* has opted the path where it has to visit the minimum number of nodes and reach the goal node as soon as possible. It is important to identify the difference between the two requirements. Former is to find the shortest path between two nodes and the latter is to reach the goal as soon as possible.

**A\* Algorithm**

A\* is a best-first search algorithm where the merit of a node, f(n), is the sum of the actual cost in reaching that node from the initial state, g(n), and the estimated cost of reaching the goal state from that node, h(n). A\* has the property that it will always find an optimal solution to a problem if the heuristic function never overestimates the actual solution cost. Its major drawback is that it requires exponential space in practice.

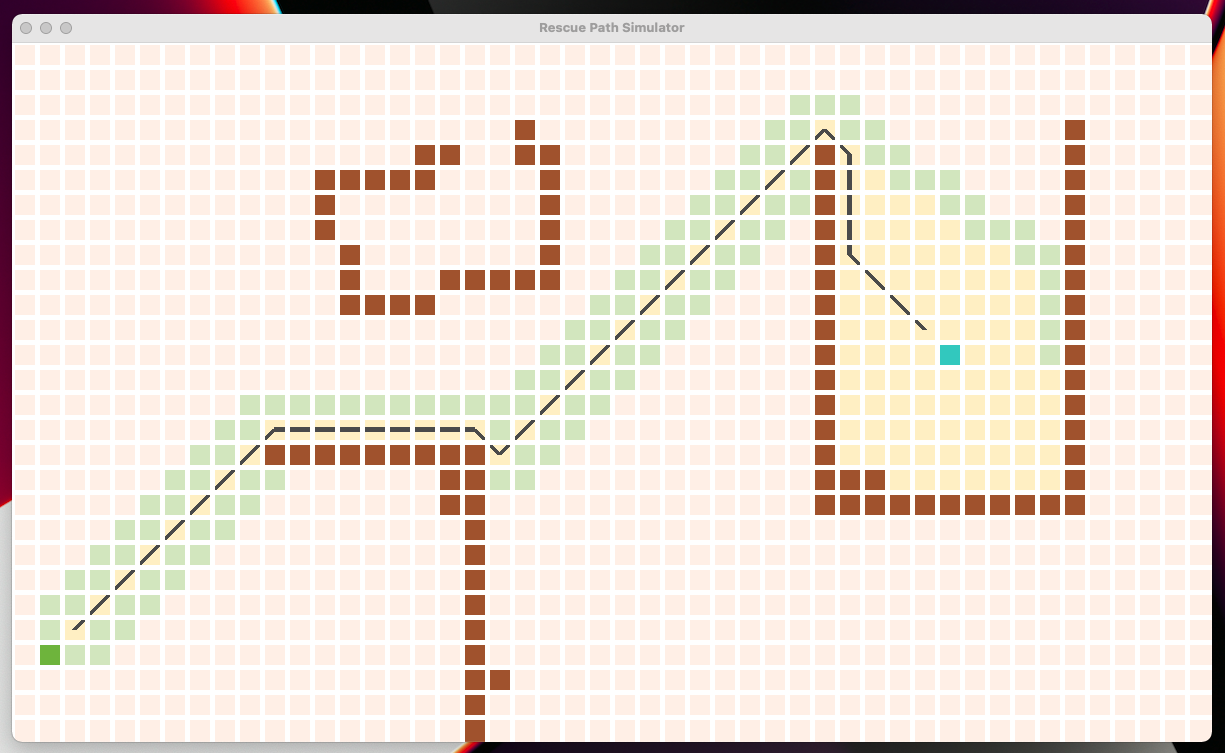


Fig. 1: A\* Algorithm Simulation

**Real-Time A\* Algorithm**

In RTA\*, the merit of a node **n** is **f(n)** = **g(n)** + **h(n),** as in **A\*.** However, unlike A\*, the interpretation of **g(n)** in RTA\* is the actual distance of a node **n** from the current state-of the problem solver, rather than from the original initial state. The key difference between RTA\* and A\* is that in RTA\*, the merit of every node is measured relative to the current position of the problem solver, and the initial state is irrelevant. RTA\* is simply a best-first search given this slightly different cost function.

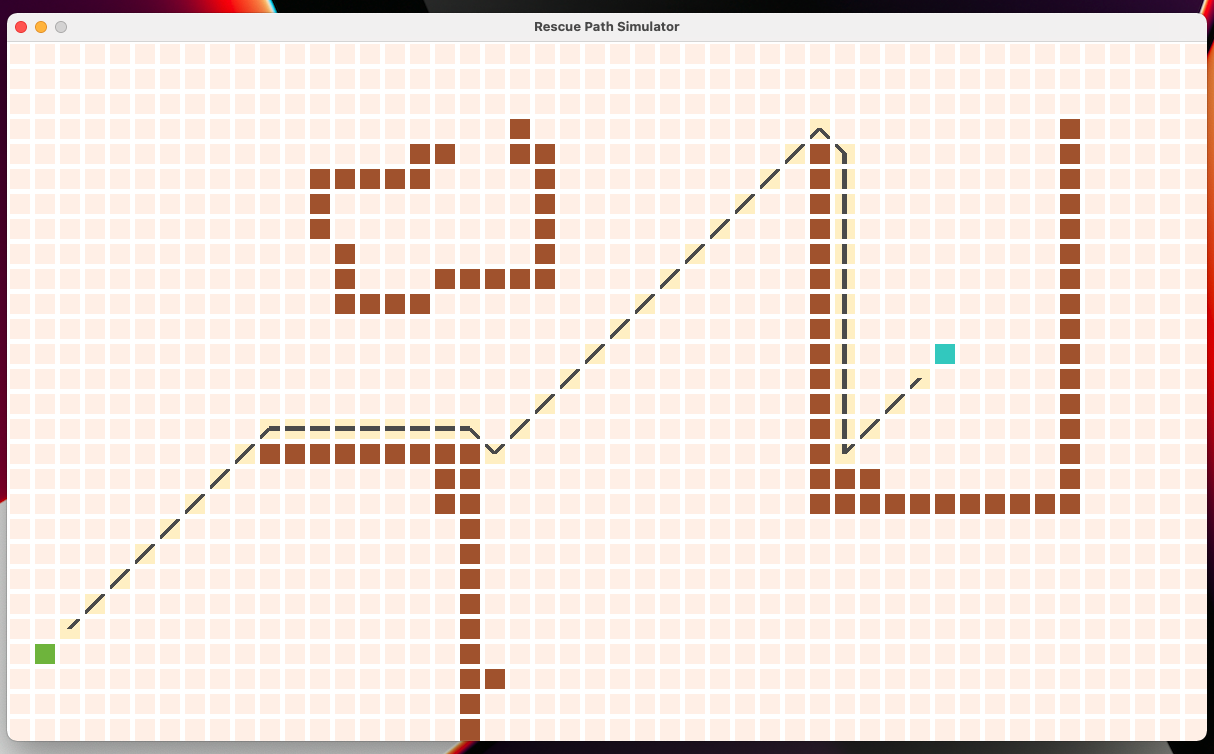


Fig. 2: Real-Time A\* Algorithm Simulation

**Learning Real-Time A\* Algorithm**

LRTA\* retains the completeness and optimality property of RTA\* by repeatedly refining the suboptimal path. It does not, however, always make locally optimal decisions but repeated trials cause the heuristic values to converge to their exact values.

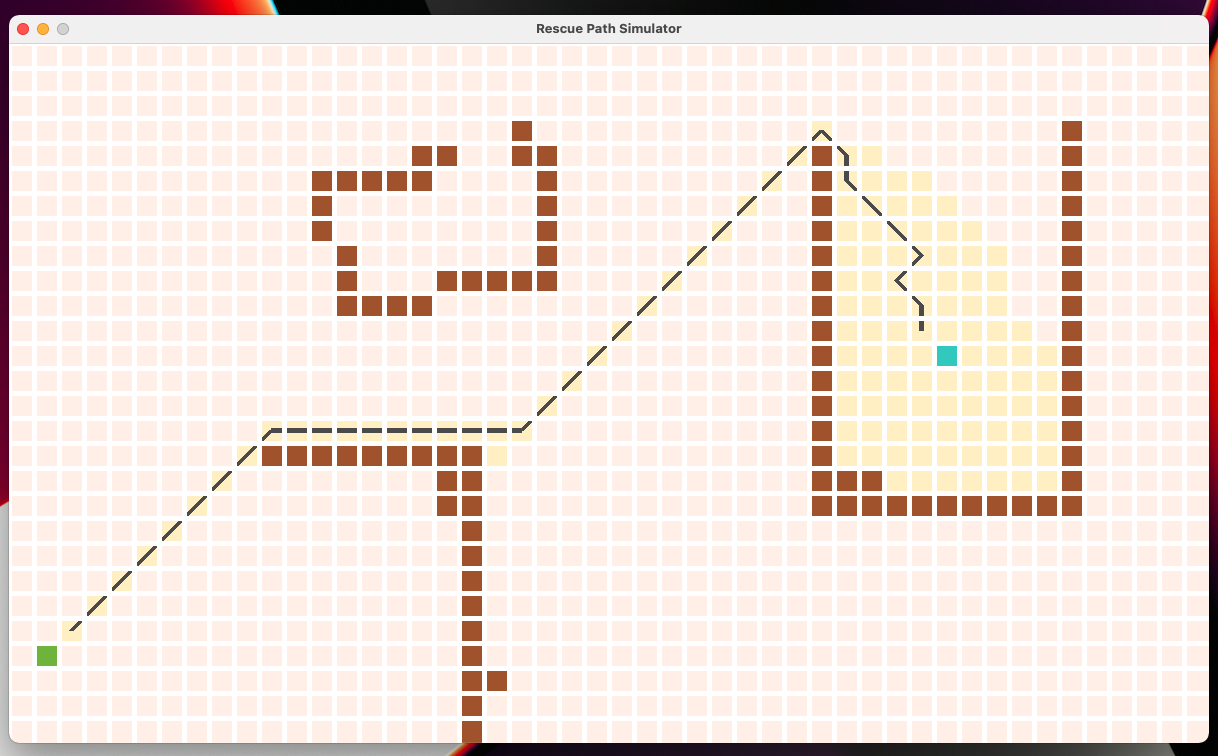


Fig. 3: Learning Real-Time A\* Algorithm Simulation

**Index**

Light Green Blocks - Represents the opened node when the algorithm stopped.

Light Yellow Blocks - Represents the visited node when the algorithm stopped.

Cyan Blocks - Represents the start node.

Green Blocks - Represents the end node.

Brown Blocks - Represents obstacles.

Black Line - Represents the path between start and end nodes.

**Conclusion**

Existing heuristic search algorithms like A\* cannot be used at large-scale real-time applications, due to their computational cost and the fact that they cannot commit to an action before its ultimate outcome is known. Real-Time-A\* efficiently solves the problem of when to abandon the current path in favor of a more promising one, and is guaranteed to eventually find a solution. In addition, RTA\* makes locally optimal decisions on a tree. Finally, Learning RTA\* is a slight modification of RTA\* which preserves its completeness properties while learning exact heuristic values over repeated problem solving trials. These algorithms effectively solve larger single-agent problems than have previously been solvable using heuristic evaluation functions.

**References**

[1] [AStar Comparison](https://github.com/maynull/AStarComparisons) - GitHub

[2] [About Risk Game - AI Agent](https://github.com/Elzawawy/risky) - GitHub

[3] [Search-Based Path Planning](https://github.com/zhm-real/PathPlanning) - GitHub

[4] [A\* Pathfinding](https://youtu.be/-L-WgKMFuhE) - Youtube

[5] [Pathfinding Visualizer](https://youtu.be/msttfIHHkak) - Youtube

[6] [Coding Challenge 51.1: A\* Pathfinding Algorithm](https://youtu.be/aKYlikFAV4k) - Youtube

[7] [Real-Time Search](http://turing.cs.pub.ro/blia_2003/Real-time_search_1.htm) (B. Roberts).