Real Time Pathfinding with Heuristic Algorithms

Sai Durga Lakshmi SriLasya Addala
dept. of ECE
Stevens Institute of Technology
Hoboken, NJ, USA
saddala@stevens.edu

Priyanshu Singh Bisen
dept. of ECE
Stevens Institute of Technology
Hoboken, NJ, USA
psbisen@stevens.edu

Prateek Pravanjan
dept. of ECE
Stevens Institute of Technology
Hoboken, NJ, USA
ppravanj@stevens.edu

***Abstract*—Warehouse robots have been widely used by manufacturing companies to automate their delivery processes. The fundamental algorithm used in designing such systems is a pathfinding algorithm. While the traditional algorithms, A* and Djikstras, find the optimal path in a static environment, the real-world scenarios involve dynamic destination and obstacles, which reaches out for other algorithms like LRTA*, suited for the task. However, these algorithms show signs of failure with large-scale environments, and call for an approach with better amortized complexity. Our project proposes an algorithm LRTA* with Fibonacci heaps influenced by LRTA*(but with reduced time complexity for analogous operations with priority queues). We will also compare the proposed algorithm with traditional algorithms, LRTA* (standard) and D*, and report the results.**

I. Introduction

The increasing adoption of warehouse robotics in the manufacturing and logistics industries highlights the critical need for efficient path-finding algorithms. These robots are tasked with navigating complex environments to perform tasks such as inventory management, material handling, and order fulfillment. Traditionally, path-finding algorithms like A* and Dijkstra's algorithm have been employed to determine optimal paths in static and predictable environments. However, real-world scenarios often involve dynamic destinations and obstacles, which require adaptive algorithms that can handle changes effectively. One such algorithm is Learning Real-Time A* (LRTA*), which is designed to operate in dynamic environments by updating its heuristic estimates as the robot explores the environment. While LRTA* offers flexibility, it faces challenges when applied to large-scale environments due to increased computational overhead and limitations in its handling of priority queues. Similarly, algorithms like D* are designed for dynamic re-planning but also encounter efficiency bottlenecks in expansive settings. To address these challenges, this project proposes an enhanced path-finding algorithm: LRTA* with Fibonacci heaps. By integrating Fibonacci heaps into the structure of LRTA* ,the algorithm aims to reduce the time complexity of priority queue operations, a key factor in real-time path-finding performance. This modification allows for more scalability for large and dynamic environments. The proposed algorithm will be evaluated through a comparative analysis against traditional algorithms (A* and Dijkstra's), standard LRTA*, and D* in terms of efficiency, scalability, and adaptability in dynamic environments. The results of this study aim to demonstrate the potential of LRTA* with Fibonacci heaps as a viable solution for real-world warehouse robotic systems.

II. Motivation and Theoretical Foundation

A. Learning Real-Time A*

The LRTA* algorithm is a variation and an improvement over standard A* for dynamic or unknown environments. Unlike A*, which uses a fixed heuristic, LRTA* dynamically updates the heuristic values based on the observed costs during execution. It is widely used in fields like robotics, video games, and AI systems where decisions must be made without prior knowledge of the entire environment. LRTA* is a real-time heuristic search algorithm which updates the heuristic estimate of the current state in each iteration. The performance of the problem is improved after each successful trial by recording the heuristic estimates. Every state (s) has a heuristic estimate h(s) of the cost from state to goal [1]. It estimates the cost of reaching the goal from a state (s). The environment is represented as a graph or grid, where nodes are states and edges are transitions between states.

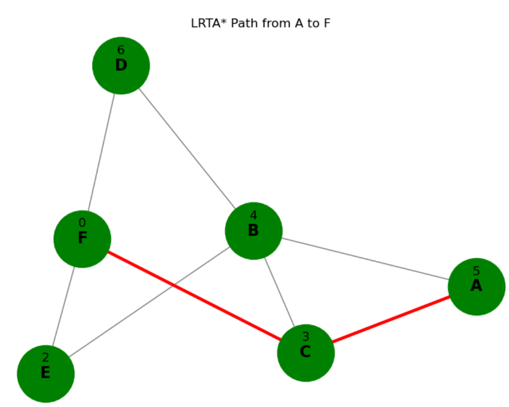

Fig. 1. LRTA* Representation

B. D*

The D* algorithm(dynamic A*) [2] is a search algorithm for optimal path finding in a changing environment such as weight updates, changing obstacles, that occur during the execution, widely used in robotics and path-finding applications. The

D* algorithm dynamically changes the weight values of the affected nodes in the graph by utilizing two properties:

- Process State - This property computes the optimal path cost to the target, by iteratively processing the states with the lowest past cost estimate (k(x)) and propagating the changes to their neighbours.
- Modify Cost - This property adjusts the weight between the states whenever there is any change in the graph. This ensures that the affected nodes are re-evaluated and updated accordingly.

Key Features:

- Ability to reuse previously computed results.
- Reduces computation time.
- Adaptability to changes in the graph.

C. LRTA*(w/ fibonacci heap)

Such a binomial heap implementation as the Fibonacci heap [3], which is the opposite of a binary tree, enhances the retrieval and insertion time complexities of the A* algorithm [4]. This is particularly important in optimizing heuristic path-finding approaches for large graphs. Although the increasing constant-time over larger graphs does incur an overhead, it becomes advantageous as the graph's complexity increases. This advantage is crucial when managing the open paths during path-finding.

REFERENCES

[1] C. Hernández and P. Meseguer, "Improving lrta*(k)," *IJCAI International Joint Conference on Artificial Intelligence*, pp. 2312–2317, 01 2007.

[2] A. T. Stentz, "The d* algorithm for real-time planning of optimal traverses," *Stentz-1994-13781*, no. CMU-RI-TR-94-37, October 1994.

[3] M. L. Fredman and R. E. Tarjan, "Fibonacci heaps and their uses in improved network optimization algorithms," *J. ACM*, vol. 34, p. 596–615, Jul. 1987.

[4] A. Patel, "Amit's a* pages: Pathfinding and differential game techniques," *Patel Astar Pages*.