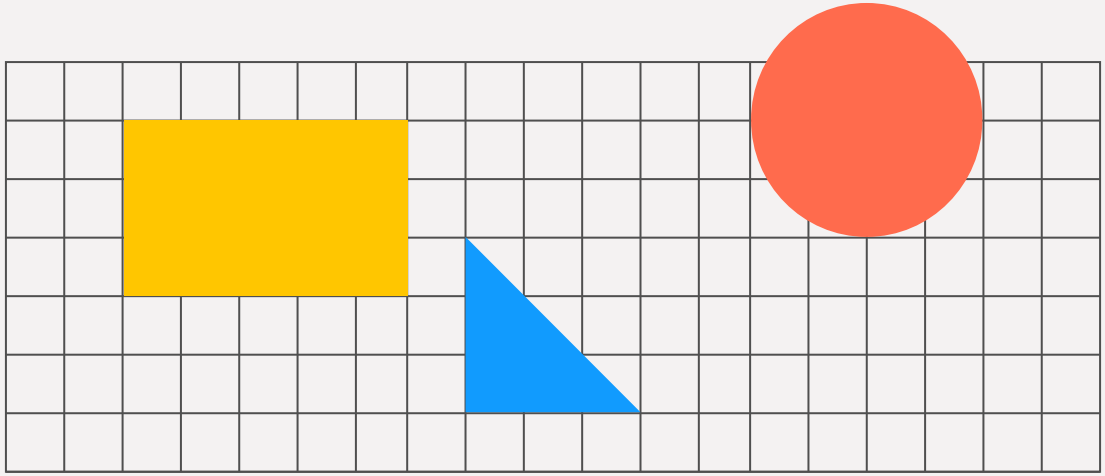
CPE 593-WS1 Data Structures and Algorithms

Real Time Pathfinding with Heuristic Algorithms



The Team

Lasya Addala

CWID: 20025775
MS in Applied AI

Priyanshu Singh Bisen

CWID: 20020360
MS in Applied AI

Prateek Pravanjan

CWID: 20021229
MS in Applied AI

Contents

Objective

The objective of this work is to design and evaluate real-time pathfinding algorithms that dynamically recompute paths in changing environments. The focus is on handling dynamic obstacles, including both consistently and randomly moving ones, using heuristic-based algorithms. Metrics like Time Per Planning (TPP) and score are used to evaluate performance. This system aims to improve real-time navigation in unpredictable and complex environments.

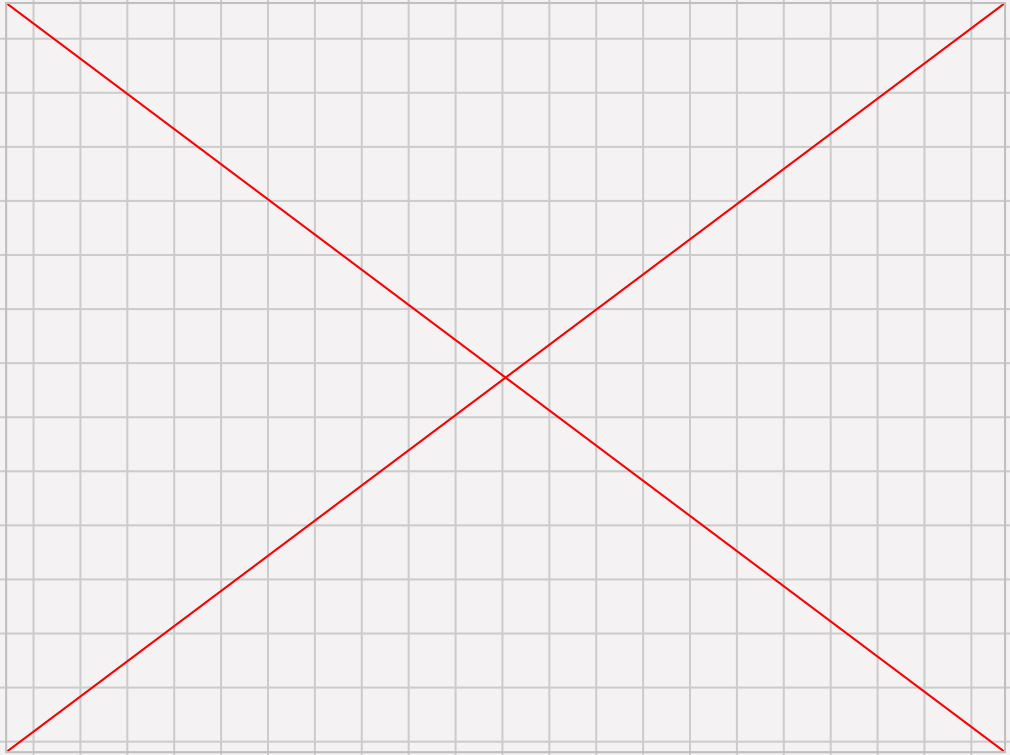
Algorithms

- A*
- Bi-Directional A*
- Dijkstra's
- Breadth-First Search
- Best-First Search
- Minimum Spanning Tree

Environment Generation

Environment Representation:
3D tensor: x, y coordinates, and state (obstacle, entity, start, or goal).
Obstacle Representation:
Tensor size: [O, 4, 2, 2] (O = number of obstacles).
Stores positional data of obstacle edges.
Velocity Representation:
Tensor size: [O, 1, 1, 2].
Enables dynamic updates using PyTorch broadcasting.

Maze



Environment Comparison

CMO

- Moves in a consistent direction
- Reverses at boundaries
- Highly predictable
- Easier to handle

RMO

- Moves with random directions
- Randomized within limits
- Less predictable
- Harder due to randomness

Dynamic Path Planning

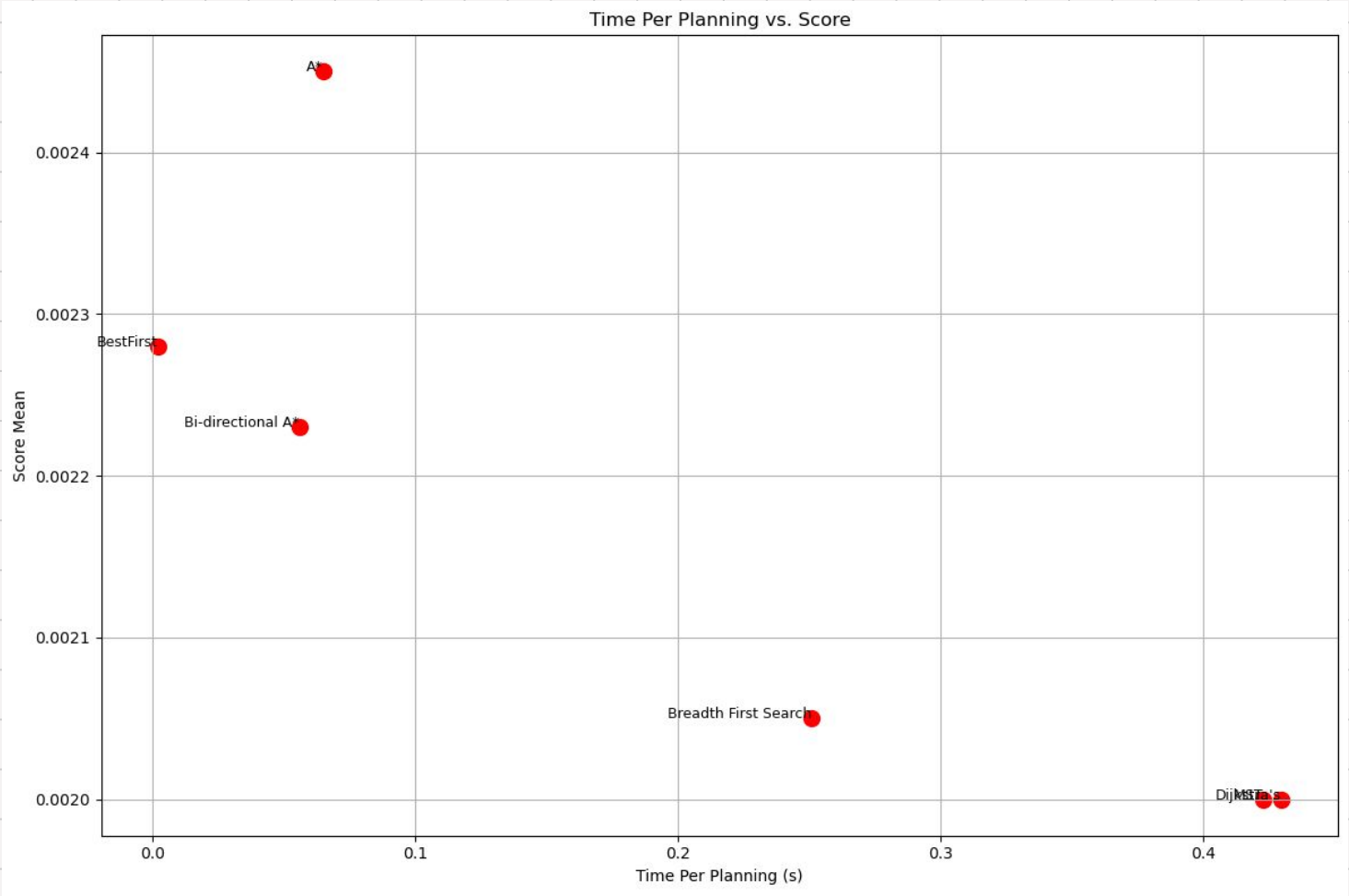
Dynamic Obstacles:
CMO: Consistently Moving Obstacles
RMO: Randomly Moving Obstacles
Agent Reactions:
Recomputes path upon detecting collisions or changes.
Avoids static and dynamic obstacles efficiently.
Visualization:
Pygame library for rendering paths and environment.

Evaluation Metrics

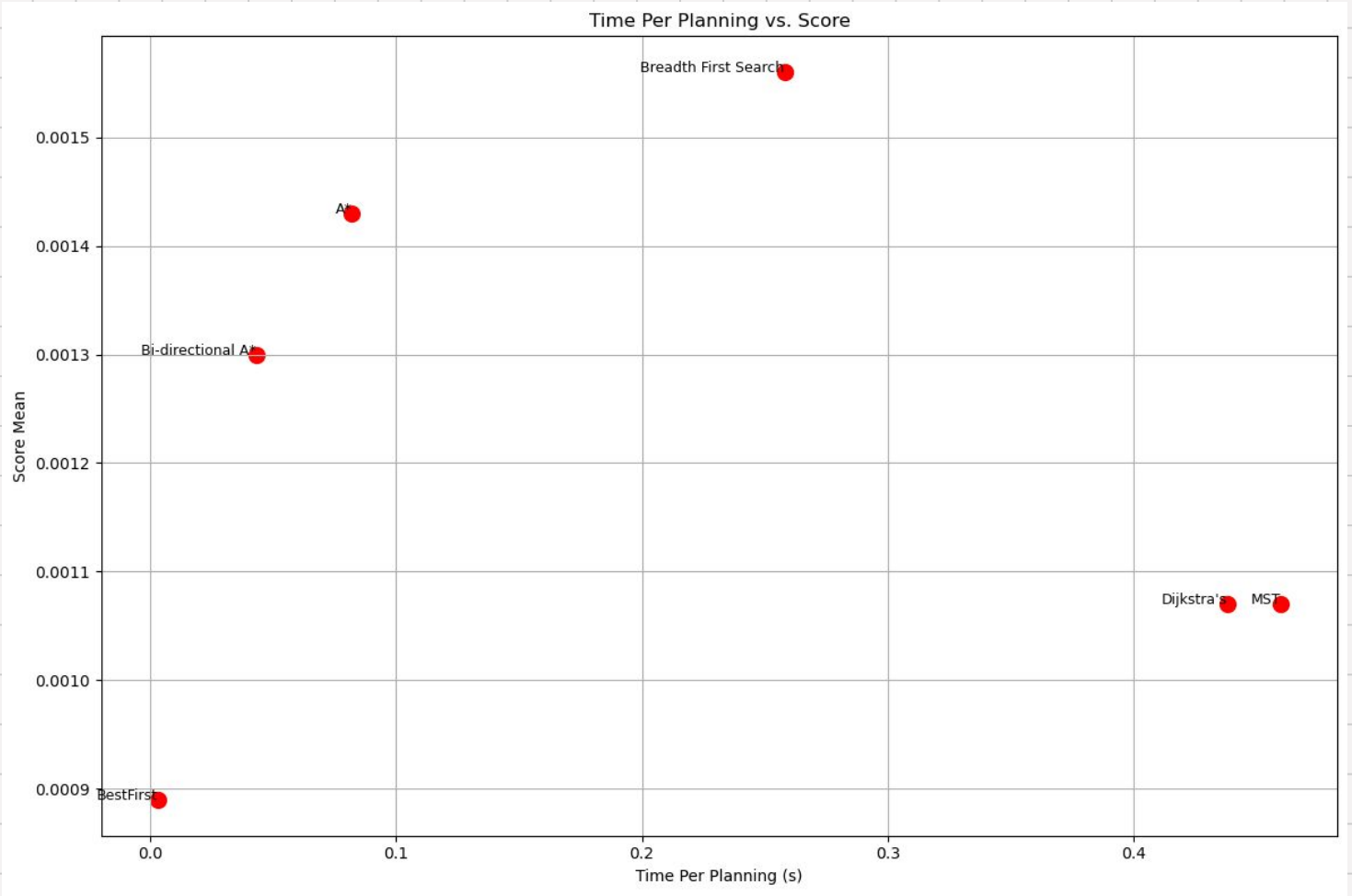
TPP (Time Per Planning):
Measures the average computation time required for the algorithm to find a path at each planning step. It is calculated incrementally to reflect real-time performance.

Score:
The primary performance metric that evaluates the effectiveness of the algorithm in finding a valid and efficient path and it is tied to travel distance.

TPP vs Score (CMO)



TPP vs Score (RMO)



Conclusion

- A* and Bidirectional A*, being an adaptive algorithm, showed to be most suitable for unpredictable environments despite having lower arrival rates and is preferred for predictable environment as well for being computationally efficient.
- Algorithms like Dijkstra and MST prioritize optimality at the cost of computational efficiency.
- Greedy approaches like Best First are fast but unreliable in dynamic settings.

Future Work

- Enhance learning mechanisms to predict obstacle behaviors.
- Incorporate additional evaluation metrics.

Thank you