A Pathfinding in GridWorld: An AI Approach

Incorporating advanced AI technology in real-world

2148143

December 8, 2023

Artificial Intelligence - COMS30072

University of Bristol

Introduction:

In Gridworld, careful route planning is required to discover paths as efficiently as possible by guiding agents. Just as search and rescue personnel must devise effective search methods in time to find missing persons. This report will examine the solutions used in the first three parts and discuss how they can be extended by incorporating new AI techniques. The goal is to make these search strategies capable of solving real-world problems.

Strengths and Weaknesses:

Part 1: Single Pathfinding

The combination of the A* algorithm and the breadth-first search (BFS) strategy is relatively effective for pathfinding problems with known and unknown goals. For the former, the A* algorithm uses heuristic functions to efficiently determine the shortest path. This is particularly useful for applications such as navigation systems that require fast navigation. For the latter, the A* algorithm has the advantage of combining the evaluation of heuristics to efficiently guide to the destination. However, the A* algorithm relies on the location information of the target. Therefore, its effectiveness is limited when the target location is unknown. This limitation is similar to the problem faced by urban navigation systems under dynamically changing road conditions. Therefore, the use of BFS can compensate for this deficiency. It can effectively determine the exact location of an unknown target. However, this combined strategy may face the challenge of computational complexity when dealing with large search spaces.

Part 2: Multiple Pathfinding

In stochastic environments with multiple goals, it is important to plan the shortest path to each goal. Breadth-first search (BFS) ensures that all targets are thoroughly searched and the shortest path from the nearest target location is prioritized, but this approach is inefficient in large search spaces, especially in large urban environments. For example, BFS can be effective in automated logistics centers or large warehouses where data is relatively small and concentrated. However, in complex environments such as urban traffic, BFS may lead to a significant increase in search time due to the large search area.

Part 3: Multi-Agent Maze Navigation and Coordination

The use of depth-first search (DFS) combined with right-handedness laws is an effective way to deal with single-agent and multi-agent pathfinding problems in mazes. The advantage of this algorithm is the ability to completely cover the maze through depth search and ensure that the exit from the maze is found. In a search and rescue mission at a disaster site, this approach is particularly helpful in conducting a comprehensive search to ensure that no possible survivors are missed. However, this method has its limitations in terms of efficiency. In large mazes or extremely complex mazes, DFS may result in a large number of unnecessary paths being explored, resulting in high backtracking costs, which may lead to inefficient searches in time-critical rescue missions.

New Technological Perspectives:

Genetic Algorithms

Genetic algorithms are a very effective way to address the limitations of the first and second parts. Genetic algorithms can not only effectively deal with search problems in large search spaces. It is mainly used to continuously improve management strategies for dynamic traffic roads. This is achieved by modeling the biological evolutionary process. For example, some road contingencies (e.g., traffic jams, traffic accidents, etc.) can be considered and route planning can be dynamically adjusted using genetic algorithms. As shown in the study by Ma (2022), this algorithm is very effective in dealing with the dynamic complexity of urban traffic. By responding to changes in traffic conditions in real-time, genetic algorithms can both reduce traffic congestion and increase road utilization, thereby improving overall traffic efficiency.

Design detail of Genetic Algorithm

In the design of genetic algorithms, this can be achieved mainly through coding, fitness functions, selection mechanisms, crossover, and mutation. Firstly, each possible route is encoded as a chromosome, with each gene representing a goal in the road network (Ma, F. 2022). Next, a fitness function is used to assess the validity of a path from one node to another (based on the condition of the roads in the path). Finally, a selection mechanism such as roulette is used to select the best individual based on the fitness value. A higher fitness value indicates a better path. In addition, crossover and mutation processes can be used. Once two optimal paths are selected, the crossover can be used to find new optimal paths. All these design details ensure that the algorithm can effectively explore and optimize urban traffic paths.

Particle Swarm Optimization

Particle Swarm Optimisation (PSO) provides an effective solution to the maze search problem when dealing with the limitations of Depth-First Search (DFS) and right-hand laws. Particle swarm algorithms are particularly suitable for searching uncertain roads. By simulating the characteristics of natural bird flocks, not only adaptation between multiple agents but also path optimization can be achieved. For example, in a SAR task, each particle represents a possible path for a SAR operator. Particle swarm optimization allows these SAR operators to dynamically adjust their paths within the search area to cope with unexpected events such as possible changes in the location of a distressed person. According to Geng (2021), when performing well in dynamic and complex search environments, search efficiency can be optimized by swarm characteristics.

Design detail of Particle Swarm Optimization

When designing Particle Swarm Optimisation (PSO) for maze exploration or rescue operations, each particle must first be evaluated as a fitness function, and, as with genetic algorithms, the higher the fitness, the better the path. Then the coverage of the search and rescue area and the arrival speed are calculated and the paths of the particles are dynamically adjusted by simulating the behaviour of the flock. Finally, the position is updated according to the historical optimal solutions of individuals and flocks until the most efficient search and rescue path is found.

Reference

Ma, F. (2022). Multimedia urban road path optimization based on genetic algorithm. Computational Intelligence and Neuroscience. From https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9078773

Geng, N. et al. (2021) Particle swarm optimization algorithm for the optimization of rescue task allocation with uncertain time constraints. From https://link.springer.com/article/10.1007/s40747-020-00252-2