DOI: 10.54254/2755-2721/109/20241456

Analysis of the Applications of Algorithm and Automatic Pathfinding

Weikun He

University of Alberta, Edmonton, T6G 1K7, Canada

weikun1@ualberta.ca

Abstract. The current development of autonomous driving technology is very hot, which involves two fundamental aspects: one is the operating system as the foundation, and the other is the algorithm application. The theme of this review is to study the related algorithmic technologies and combine them with one of the key functions of autonomous driving: autonomous routing. The review discusses the application direction and environment of this function, and involves the use of algorithms in the backend. Autonomous routing is a key concept in multiple technical fields and plays an important role in helping entities effectively navigate complex environments. This review centers on the concept of autonomous routing and focuses on its application direction, usage environment, and supporting algorithms. The core research question is autonomous routing and its working principle. The review analyzes the main application scenarios of autonomous routing, such as autonomous driving and game development, and explores the algorithms commonly used in these scenarios. By conducting a comprehensive analysis of the main usage environments and algorithm structures, the review provides insights into the current state of autonomous routing technology. The research findings show that autonomous routing technology has been deeply embedded in multiple industries and is continuously expanding as the demand for technology grows. Furthermore, the review explores the potential future development of autonomous routing, anticipating that it will further develop in responding to various new challenges and opportunities.

Keywords: Autonomous navigation, algorithm, shortest path, autonomous driving.

1. Introduction

As an important research direction in the field of artificial intelligence and computer science, automatic pathfinding technology has been widely used in many industries in recent years. With the continuous advancement of technology, automatic wayfinding has become an integral part of autonomous driving, game development, robot navigation and other fields. The concept of automatic pathfinding is derived from graph theory and search algorithms to find the optimal path for moving entities in complex environments. This technology not only improves the automation level of the system, but also significantly enhances the performance and efficiency of various intelligent systems.

The research topic of this paper is to discuss the application direction of automatic pathfinding and its practical application in different environments, with emphasis on the analysis of various algorithms supporting these applications. In recent years, with the increasing demand for automation and intelligence, the research on automatic path-finding algorithms has been gradually deepened in academia

DOI: 10.54254/2755-2721/109/20241456

and industry, and many new algorithms and applications have emerged. However, there are still many challenges in this field, such as path planning in dynamic environments, real-time requirements, and multi-objective optimization.

In order to systematically answer the research question "what is automatic pathfinding and how it works", this paper reviews the application of automatic pathfinding technology in mainstream use environments through literature analysis, and discusses the structure and characteristics of different algorithms in detail. The research significance of this paper is that by summarizing the current situation and future development direction of automatic pathfinding, it can provide references for researchers and engineers in related fields to help them better understand and apply this technology. At the same time, this review also looks at the future challenges that automated pathfinding technology may face and its potential innovation opportunities.

2. Automatic path-finding related content introduction

2.1. The concept of automatic pathfinding

Automatic routing is the process of finding an optimal or feasible path for a moving entity in a complex environment. The concept originated in the field of artificial intelligence and robotics, but with the development of technology, automatic pathfinding has become a key technology in a variety of application scenarios. From navigation systems to game development, from robot control to logistics optimization, automated pathfinding technology plays an important role in various industries.

The essence of this feature is to solve the path planning problem, that is, how to move from the starting point to the end point in a known map or environment. This problem is often represented in computer science through graph theory, where the environment is modeled as a collection of nodes and edges, each representing a possible location, and edges representing a path from one node to another. The "good or bad" of a path is usually measured by the cost of the path, such as distance, time, or energy consumption.

2.2. The principle of automatic pathfinding

The principle of automatic pathfinding is based on graph theory and search algorithms. In order to implement automatic pathfinding, you first need to abstract the environment into a graph model, which can be a regular grid diagram, or a more complex polygonal grid or navigation grid. Then, a search algorithm is used to find a path from the starting point to the end point in this graph model.

Common automatic path-finding algorithms include depth-first search (DFS), breadth-first search (BFS), Dijkstra's algorithm, and heuristic search algorithms such as A*. These algorithms gradually expand the search space by evaluating the cost of the path, and finally finding an optimal or near-optimal path. Heuristic algorithms such as A* also combine heuristic functions to prioritize paths in the pathfinding process, making the search more efficient.

Automatic pathfinding is not limited to static environments. In many applications, such as robotic navigation or drone flight, the environment may be dynamic and the location of obstacles or targets may change over time. Therefore, automatic pathfinding algorithms need to have the ability to respond in real time and adjust dynamically to deal with uncertainties and changes in the environment.

An easy-to-understand example, in Figure 1, shows the shortest path for the cat to find the bone.

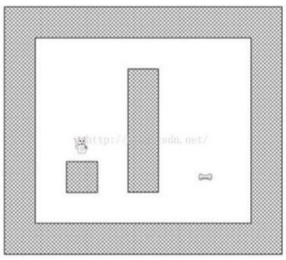

Figure 1. Map of the cat's search for bones [1].

According to common sense, the distance of the straight line is the shortest, but because of the existence of obstacles: walls, people need to use algorithms to find the shortest path around the road. The first is to simplify the search area so that the map is divided into smaller pieces. Then there is the operation of the algorithm; the A* algorithm works by two lists: one is the open list (open list), which holds all possible path blocks. The other is the closed list, which holds the blocks that have been explored. The algorithm adds the starting point to the closed list and the surrounding passable squares to the open list. Each square has a path cost F, which is the sum of the moving cost G from the starting point to the current square plus the estimated cost H from the current square to the end point. The algorithm selects the square with the smallest F-number, updates the F-number of its neighbors, and repeats the process until it finds an end point. In the process of exploration, the algorithm will constantly update the parent pointer of the block to make the path shortest. Eventually, the algorithm will find the shortest path from the end point back to the beginning. This process continues until the destination is found and added to the open list. The final path is determined by following the parent pointer from the end point back to the starting point [1].

2.3. Research on algorithm of automatic pathfinding

With the continuous progress of technology, the research direction of automatic path-finding algorithms is increasingly diversified. Over the past few decades, researchers have developed many different types of algorithms to meet the needs of different application scenarios. These algorithms have not only made breakthroughs in the efficiency and effect of path planning, but also made remarkable progress in dealing with complex environments, dynamic changes and multi-objective optimization.

2.3.1. Choice of algorithm. Selecting a suitable algorithm is one of the key problems in automatic routing research. Different application scenarios have different requirements for algorithms, so researchers need to consider a variety of factors when selecting algorithms, such as computational complexity, path quality, real-time and adaptability.

For scenarios with high real-time requirements, such as game development, pathfinding algorithms need to generate paths in a very short time, so approximate algorithms or heuristic algorithms are usually chosen, which can provide a better solution in a limited time. In scenarios with high safety requirements, such as automatic driving, the accuracy of the algorithm and the optimality of the path are particularly important. For example, the A* algorithm mentioned above is a typical example. In addition, Dijkstra is also a widely used algorithm in the field of game design [2].

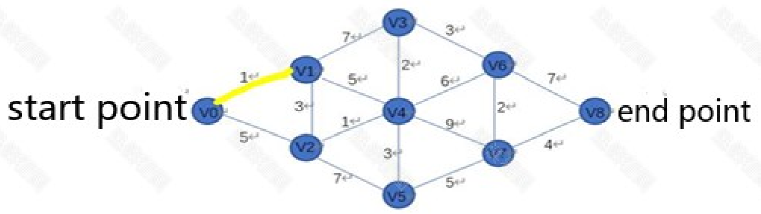

Figure 2. Dijkstra Algorithm diagram[2].

Dijkstra's algorithm is one of the earliest classical single-source shortest path algorithms, which is used to calculate the shortest path from one node to all other nodes. Its main feature is to start from the starting point and gradually expand outward until the end point is found (Figure 2). Unlike the A* algorithm, Dijkstra's algorithm does not find the shortest distance from V0 to V8 in one go, but rather finds the shortest distance of all vertices between V0 and V8 step by step, thus finding the shortest distance from V0 to V8.

Not only in games, but also in other fields, automatic pathfinding functions are also important, such as the demand for automatic pathfinding functions of obstacle avoidance food delivery cars when working [3]. Different from the above automatic pathfinding in the game, the obstacle avoidance car, in addition to the algorithm, has a set of underlying operating systems to implement the results given by the algorithm. In this article, the automatic obstacle avoidance design based on LiDAR is used to achieve the shortest path and obstacle avoidance functions. But the basic algorithm also requires A* to set an open list and a close list, representing the state to travel.

In addition, the MAFP algorithm also attracts attention in the algorithm selection of multi-agent automatic pathfinding [4]. This algorithm is designed to solve the problem of multi-agent automatic routing. The key use of this algorithm is to constrain multiple agents to travel along the planned path at the same time without conflict. MAPF algorithm is divided into centralized planning algorithm and distributed planning algorithm. The centralized planning algorithm is the most classic and most commonly used MAPF algorithm, which is mainly divided into four algorithms based on A* search, conflict search, cost growth tree and protocol. Distributed execution algorithms are divided into three types: expert speech, improved communication and task decomposition [5].

3. Environments using automatic pathfinding technology

Automatic pathfinding technology is widely used in multiple fields, each with different needs and challenges, and therefore different requirements for algorithms. By understanding the requirements in these application environments, people can better understand the advantages and disadvantages of different algorithms and choose the appropriate technical solution for the specific application.

3.1. Automotive driving application

In autonomous driving technology, an automatic pathfinding algorithm is an important part of the vehicle navigation system. In the process of driving, vehicles need to carry out path planning according to various factors such as real-time traffic information, road conditions, obstacle location and target location. The core goal of automatic driving is to ensure the safety of passengers and maximize the efficiency of driving. Therefore, the path planning algorithm should not only consider the optimality of the path, but also have efficient real-time response ability. Therefore, the global path search algorithm and driverless path planning based on A* algorithm are critical to the application of automotive driving [6]. So we can see the importance of the algorithm for the implementation of this function. In addition, the path planning can be roughly divided into the grid method, viewable method, free space method, ant colony algorithm and topology method. The approximate sequence of path planning is as follows: select the appropriate algorithm model to divide the workspace into small areas; then presuppose the obstacle information and search the area containing the obstacle; finally, the optimal path is calculated.

3.2. In-game usage

In video games, automatic pathfinding algorithms are mainly used to control the movement of game characters or non-player characters (NPCS) so that they can navigate reasonably in a virtual environment. Unlike autonomous driving, pathfinding algorithms in games focus more on computational speed and path feasibility, rather than absolute path optimality.

With the continuous optimization of the A* algorithm in the game automatic pathfinding, the game characters become more intelligent and flexible, increasing the fun of the game, so it is widely used in various games. Simple mini-games focus on character pathfinding, dodging, and stalking, especially when NPCS move around certain areas and react to the player's location. The algorithms to implement these behaviors are relatively simple and include random pathfinding, tracking, and dodging.

For large games, players are more demanding, and characters need to have more human and intelligent behaviors such as attacking, dodging, pathfinding, steering, and jumping, while also having virtual space awareness. These complex behaviors rely on more complex algorithms, covering aspects of character behavior, thinking, physical models, and perception. The A* algorithm determines the actions of the NPC through massive data search and instruction selection and passes them to the agent for execution [7].

Compared with the traditional path planning algorithm, A* algorithm has higher real-time performance, stronger flexibility, and is closer to the result of manual path selection. A* pathfinding algorithm does not find the optimal path, but only the relatively near path, because Zhao optimal path needs to find all feasible paths for comparison, which consumes too much performance, and the pathfinding effect only needs to be relatively near path [8].

In addition, the flow field pathfinding algorithm has excellent performance in realizing pathfinding functions in UE-based MOBA games [9]. This algorithm is a way finding method for efficient navigation in complex environments, especially for large groups or multi-unit navigation scenarios, such as real-time strategy games (RTS). It generates a Direction Field in advance so that any unit can quickly find the optimal path from the current position to the target point. Its working steps are mainly divided into the following three steps:

- (1) Target point generation field: First, a flow field is generated at the target point. Each cell of this flow field contains a vector, or direction, that points to the target point. This direction is calculated based on the shortest path from the current cell to the destination point.
- (2) Calculate the flow field: starting from the target point, through the way of backpropagation, gradually calculate the best direction of each cell, so that all cells on the entire map have a direction pointing to the target point.
- (3) Unit navigation: Once the flow field is established, each unit only needs to find the direction of the flow field in the cell where it is located and move along this direction. This method allows each unit to independently and quickly find a path from any location to the target point.

3.3. Other applications

In addition to autonomous driving and game development, automatic pathfinding technology is also widely used in other fields, such as robot navigation, logistics path optimization, drone flight, emergency rescue and so on. In these applications, the choice of algorithm usually depends on the needs of the specific task. In order to meet the needs of indoor work, such as the power distribution room robot, this product also realizes the function of automatic pathfinding. First of all, there is a background requirement of the work scene of the robot in the power distribution room, and the road stiffness planning of the algorithm is realized in the scope of the background map. It uses the SLAM algorithm to build an A map, stores the description information of obstacles in the workplace, and then uses the improved A* algorithm to plan the path for the robot in the power distribution room [10]. In order to solve the edge collision problem of the A* algorithm, the artificial potential field algorithm is very useful. When the robot in the power distribution room is near the obstacle, it can effectively guide the robot away from the obstacle by generating a large repulsive force cost, thus reducing the possibility of collision.

4. Conclusion

As the core tool to solve the problem of path planning in complex environments, automatic pathfinding technology has shown strong application potential and played an important role in many industries. Among them, the A* algorithm (A-STAR) is undoubtedly one of the most critical algorithms at present. With its balance between efficiency and optimality, the A* algorithm has become a mainstay in the field of path planning, not only providing efficient search performance, but also ensuring path optimality. Its flexibility allows it to adapt to many types of graph models, such as grids, navigation grids, and polygonal grids, which lays the foundation for a wide range of applications in game development, robot navigation, and map path planning.

The core goal of the A* algorithm is to bring practical convenience to people's lives through the calculation of the shortest path, and promote the application and development of related technologies in more emerging fields. In the field of automotive autonomous driving, the application prospect of the A* algorithm is particularly prominent. Its precise path planning capabilities not only provide theoretical support and practical basis for the development of autonomous driving technology, but also contribute to the overall optimization of intelligent transportation systems. In addition, the algorithm principle and application results of the A* algorithm have also opened up new directions for innovative research in other fields such as game design, virtual reality, and drone navigation.

With the continuous progress of technology, the application scenario of the A* algorithm will be further expanded, not only limited to the current field. In the future, automatic pathfinding technology is expected to provide new ideas and solutions for solving more complex and dynamic path planning problems through continuous optimization and iteration. In these extensive applications, the A* algorithm not only brings practical convenience to real life, but also provides a broad space for technological development and innovative exploration. Through this continued application expansion, the A* algorithm will play a more important role in future technological innovation, driving technological change and more possibilities.

References

[1] A* Automatic Path-finding Algorithm Detailed Explanation_A* Algorithm Four Neighborhoods-CSDN Blog O-K, A* Automatic Path-finding Algorithm Detailed Explanation, CSDN, 2018-04-22. https://blog.csdn.net/kevinwhh/article/details/80035274
[2] Yifu Li, Du Min, Danlei Du, et al. Application of A~* Algorithm in Automatic Path-finding in Games [J]. Journal of Hunan University of Science and Technology, 2021, 42(03): 48-52. DOI: 10.16336/j.cnki.cn43-1459/z.2021.03.014.
[3] Xinkai Yuan, Linchong Yan, Minghui Fang, et al. Automatic path-finding and obstacle-avoiding food delivery system based on A~* algorithm [J]. Science and Technology Innovation and Application, 2022, 12(14):35-38. DOI:10.19981/j.CN23-1581/G3.2022.14.009.
[4] Sigurdson Devon, Bulitko Vadim, Koenig Sven, Hernandez Carlos, Yeoh William. Automatic Algorithm Selection In Multi-agent Pathfinding. ArXiv. Sat, 15 Jun 2019. http://arxiv.org/abs/1906.03992
[5] Aiyu Xiaocaitao. A Review of Multi-agent Path Planning, CSDN, 2023-11-20
[6] Wenyue Li. Research on Automatic Path-finding and Dynamic Obstacle Avoidance of Driverless Vehicles [D]. Chang'an University, 2018.
[7] Jiang Kai. A brief discussion on the implementation and application of automatic pathfinding algorithm in games [J]. China New Communications, 2018, 20(02):235-236.
[8] Badao Xiaoming, Automatic pathfinding in AL games - detailed explanation of A* algorithm (C++ implementation). CSDN, February 13, 2022.
[9] Hu Da. Design and Implementation of MOBA Game Based on Unreal Engine 4 [D]. Shandong University, 2021. DOI:10.27272/d.cnki.gshdu.2021.004251.
[10] Jingpeng Du. Research and Application of Automatic Pathfinding and Precise Positioning of Distribution Room Robots [D]. North China University of Technology, 2024. DOI:10.26926/d.cnki.gbfgu.2024.000314.