Mobile Robot Path Planning in 2D Space: A Survey

Sitong Zhang[1,†], Tianyi Zhao[2,*,†]

[1] Yinbin road, Zhongshan distinct, Dalian, Liaoning, China

[2] Tianjinmeidi, Lubei distinct, Tangshan, Hubei, China

* Corresponding Author Email: guanghua.ren@gecacademy.cn

13842809922@139.com

[†]These authors contributed equally.

Abstract. Robot path planning is task of navigating a mobile robot around a space in which lie a number of obstacles that have to be avoided. Path-planning can be static or dynamic.It is also an important primitive for autonomous mobile robots that lets robots find the shortest or otherwise optimal path between two points. Here we deal with static path-planning. We proposed 8 methods for path planning on this topic.

Keywords: Mobile Robot, Path Planning, Primitive, Navigating.

1. Introduction

With the rapid development of science and technology in the world, the word "robot" has been welcomed by many people. Today's robot is autonomous and has the ability of self-learning, reasoning, decision-making and planning. This robot is called artificial intelligence , which has changed our life. [1]In 2015, AlphaGo, a robot proficient in chess, defeated the famous Korean chess player and the Chinese world champion Jacko in 2016 [2][3]. A lot of work will be replaced by artificial intelligence, which is more efficient and can work 24 hours. For example, cleaners are replaced by automatic cleaners, and waiters are replaced by automatic delivery robots. Intelligent mobile robot is a comprehensive system integrating environmental perception, dynamic decision-making and planning, behavior control and execution. It integrates the research results of sensor technology, information processing, electronic engineering, computer engineering, automatic control engineering and artificial intelligence. It represents the highest achievement of mechanics and is one of the most active fields in the development of science and technology. With the continuous improvement of robot performance, the application range of mobile robot is greatly expanded. It is not only widely used in industry [4], agriculture [5], medical treatment [6] and service industries [7], but also in harmful and dangerous occasions such as urban security [8], national defense [9] and space exploration [10]. Therefore, mobile robot technology has attracted extensive attention all over the world.

The research of mobile robot began in the late 1960s. Nils Nilsson and Charles Rosen of Stanford Research Institute (SRI) developed an autonomous mobile robot called Shakey from 1966 to 1972. [11] Using artificial intelligence technology to study the autonomous reasoning, planning and control of robot system in complex environment. According to the robot's recognition of the surrounding environment information, the degree of mastering the information and the recognition of different kinds of obstacles, the robot path planning can be divided into four categories. [12]Firstly, the robot plans the path of the mobile robot according to the position of the static obstacles in the known and familiar environment. Secondly, the robot plans the path of mobile robot according to the position of static obstacles in unknown and unfamiliar environment. Thirdly, in the known and familiar environment, the path of mobile robot is planned according to the running state of dynamic obstacles. Fourthly, the path of mobile robot is planned according to the running state of dynamic obstacles in an unknown and unfamiliar environment. According to the robot's ability to master the surrounding environment, the path planning technology can be divided. The first kind is to plan the path of the mobile robot based on the verification of the surrounding environment information, and the planned

path is the global path; The second is to plan the path of the robot based on the sensor information, and the planned path is a local path. The path planning method of mobile robot can be divided into two types: traditional method and intelligent method. Our main topic is the path planning of robot in two-dimensional space, so we can take the automatic distribution robot as an example



Figure 1. This is a transfer robot that initializes and records the path to each workbench.

The performance indexes of robot path planning include real-time, security and accessibility. In the dynamic environment, due to the continuous change of environmental information, if the real-time performance of mobile robot is poor and lags the dynamic environment, it may lead to obstacle avoidance failure. Security and accessibility are also important. A method with poor performance index will be eliminated even if it can make the mobile robot go out of the ideal trajectory. Although some methods have no advanced theory, they have their own space because of their simple calculation, good real-time performance, and good security. How to improve the performance index is an important research direction of various algorithms. The term "algorithm" means "a process or set of rules to be followed in calculation or other problem-solving operations". [13] Therefore, an algorithm is a set of rules or instructions that gradually define how to perform work to achieve the desired results.

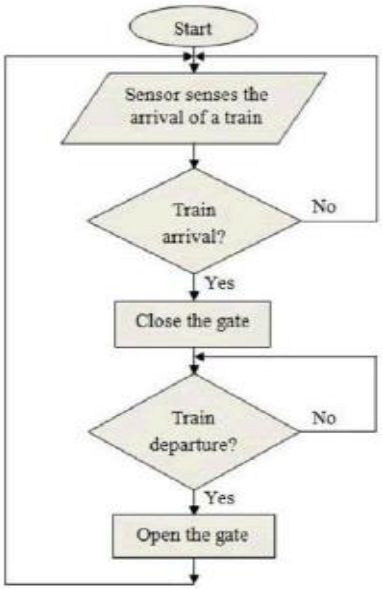


Figure 2. This is a path planning for a transportation robot

This is a simple algorithm. When the sensor senses the arrival of the train, it needs to judge whether the train arrives. When it arrives, it closes the gate. When it does not arrive, it returns to the starting process. Then when the train leaves, the gate will open, otherwise it will close. Therefore, the algorithm starts from sensor sensing and is a cyclic process. In this topic we will investigate the motion planning for the robots. Motion planning consists of path planning (space) and trajectory planning (time) [14][15]. The sequence of points or curves connecting the starting point and ending point are called path, and the strategy forming the path is called path planning. So there are 2 kinds of algorithms that we are going to investigate which is the path planning algorithms and the trajectory planning algorithms. In my opinion, the differences of the two kind of algorithms is the goal of path planning is to make the distance between the path and the obstacle as far as possible and the length of the path as short as possible but the main purpose of trajectory planning is to make the running time or energy of the robot as short as possible in the joint space movement. We will investigate 8 algorithms. 6 path planning algorithms and 2 trajectory planning algorithms. For the path planning algorithm. There are 2 graph-based algorithms which is Dijkstra algorithm [16] and A* algorithm [17], and 2 sampling algorithms which is RRT [18] and PRM [19], and there are also choose 2 intelligent planning algorithms which is the Genetic algorithm [20] and Ant colony algorithm [21]. For the trajectory algorithm there are 2 algorithms, which is the Polynomial curves [22] and the Beziers curves [23]. We will introduce my algorithms in background part.

2. Background

Let’s briefly explain the word "algorithm". The word "algorithm" means "a process or a set of rules to be followed in calculation or other problem-solving operations. In the global path planning algorithm, it can be roughly divided into three categories: traditional algorithm (Dijkstra algorithm, A* algorithm, etc.), intelligent algorithm (PSO algorithm, genetic algorithm, reinforcement learning, etc.), traditional and intelligent algorithms. There are many kinds of intelligent algorithms, but the traditional algorithms are more basic. Therefore, based on the principle of from simple to deep, we first learn the traditional algorithms. In the traditional path planning algorithms, the implementation principles and application ranges of various algorithms vary greatly, but the following five algorithms can be regarded as one class (Dijkstra, A*, D*, LPA*, D* Lite), the basic principles of each algorithm are described below, and the search principles and application scenarios are compared and distinguished.

Dijkstra algorithm, also known as Dijkstra algorithm, was proposed by E.W. Dijkstra in 1959. The algorithm adopts a greedy mode, which solves the shortest path problem from a single node to another node in a directed graph. Its main feature is that the next node selected in each iteration is the nearest child node of the current node, that is, the journey of each iteration is the shortest. In order to ensure the shortest path finally found, the shortest path from the starting node to all traversed points should be updated in each iteration.

A* algorithm is a heuristic search algorithm. Heuristic search is to establish heuristic search rules in the search process, so as to measure the distance relationship between the real-time search location and the target location, make the search direction first towards the location of the target point, and finally achieve the effect of improving the search efficiency.

Based on the A* algorithm, Anthony Stentz proposed the dynamic A* algorithm, that is, the D* algorithm, in 1994. D* algorithm is a reverse incremental search algorithm, that is, the algorithm gradually searches from the target point to the starting point; Incremental search, that is, the algorithm calculates the distance measurement information of each node in the search process. In the dynamic environment, if there are obstacles that cannot continue to search along the pre path, the algorithm will re plan the path at the current state point according to the distance measurement information of each point previously obtained, without re planning from the target point.

In 2001, life planning A* algorithm jointly proposed by Sven Koenig et al. is an incremental heuristic search algorithm based on A* algorithm. Implementation principle of LPA* algorithm: the

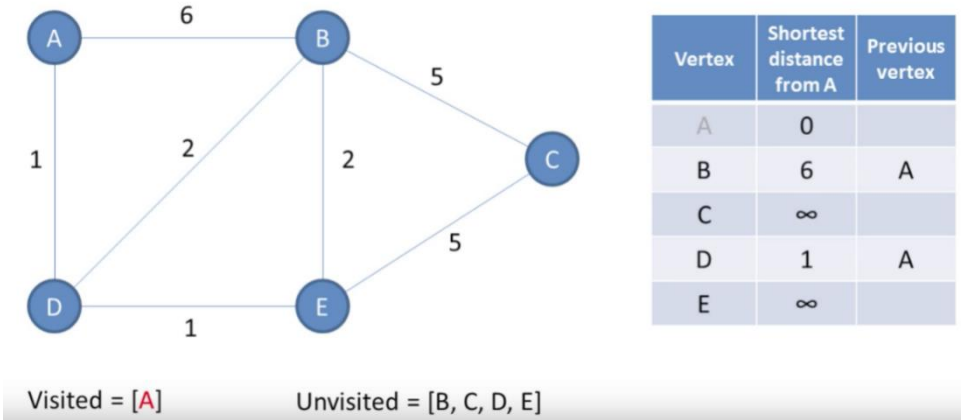
starting point of search is the set starting point (forward search), and the size of key value is taken as the principle of search progress. When the target point is the next search point, the planning is completed. The key value contains the heuristic function term as the heuristic principle to affect the search direction. When in a dynamic environment, LPA* can adapt to the changes of obstacles in the environment without recalculating the whole environment. The method is to use the g value obtained from the previous search twice during the current search in order to re-plan the path.

Researchers have proposed many planning methods, such as artificial potential field method, element decomposition method, random road map (PRM) method, fast search tree (RRT) method and so on. The traditional artificial potential field and element decomposition method need to accurately model the obstacles in space. When the obstacles in the environment are complex, it will lead to a large amount of calculation of the planning algorithm. PRM method based on random sampling technology can effectively solve the path planning problem in high-dimensional space and complex constraints' (rapid exploring random tree) generates a tree on the initial configuration by using random samples in the search space. As you plot each sample, try to establish a connection between it and the most recent state in the tree. If the connection is feasible (passed entirely through free space and subject to any constraints), it will cause a new state to be added to the tree. In the case of uniform sampling in the search space, the expansion probability of the existing state is proportional to the size of its Voronoi region. Since the largest Voronoi region belongs to the state on the search boundary, this means that the tree preferentially expands to the large region not searched. The connection length between tree and new state is often limited by growth factors.

3. Mobile Robot 2D Path Planning Algorithms

In the previous section, we briefly introduced the algorithm that we will investigate. In this section, we will introduce them more deeply.

3.1. Dijkstra algorithm



Vertex	Shortest distance from A	Previous vertex
A	0	
B	6	A
C	∞	
D	1	A
E	∞	

Figure 3. This is the image of Dijkstra algorithm, which is a common algorithm for finding the shortest path in graph theory proposed by Dijkstra, a Dutch computer scientist, in 1959. The algorithm can find the shortest path from a point in the graph to any vertex.

Dijkstra algorithm is a common algorithm to find the shortest path in graph theory proposed by Dutch computer scientist Dijkstra in 1959. This algorithm can find the shortest path from a point in graph to any other vertex. Dijkstra algorithm divides network nodes into three parts: unlabeled nodes, temporarily labeled nodes and permanently labeled nodes. In the network diagram, all nodes are initialized as unmarked nodes. In the search process, the peripatetic nodes and nodes connected in any path are temporary marked nodes. In each cycle, the node with the shortest path length from the origin point is searched from temporarily marked nodes as permanently marked nodes. The algorithm does not end until the target node is found or all nodes become permanent marker nodes. Dijkstra algorithm is one of the prominent algorithms to find the shortest path from the source node to a

destination node. Dijkstra's algorithm is to solve the problem of single source shortest path, that is, the shortest path from a source point to other vertices; There are two key points in Dijkstra's algorithm that should be paid attention to. There are two sets are identified: all vertex sets V and selected vertex sets S. The algorithm first finds the point closest to the source point in the current unselected point. Then it updates the distance between the unselected point and the source point.

3.2. A* algorithm

A* algorithm is a very common path search and graph traversal algorithm. It has better performance and accuracy. The A* algorithm was first published in 1968 by Peter Hart, Nils Nilsson and Bertram Raphael of the Stanford Research Institute. It can be thought of as an extension of Dijkstra's algorithm. Because of the guidance of heuristic functions, it is a function that estimates the distance between the current state and the destination state for path decisions.

In other words, the IQ of the function directly determines the speed and accuracy of finding the path. If we apply this algorithm to the role of the computer in the game, then this function will directly determine the difficulty of the game. When we adjust the difficulty of the game, we are actually choosing a more perfect or stupid heuristic function. A* algorithms usually have better performance.

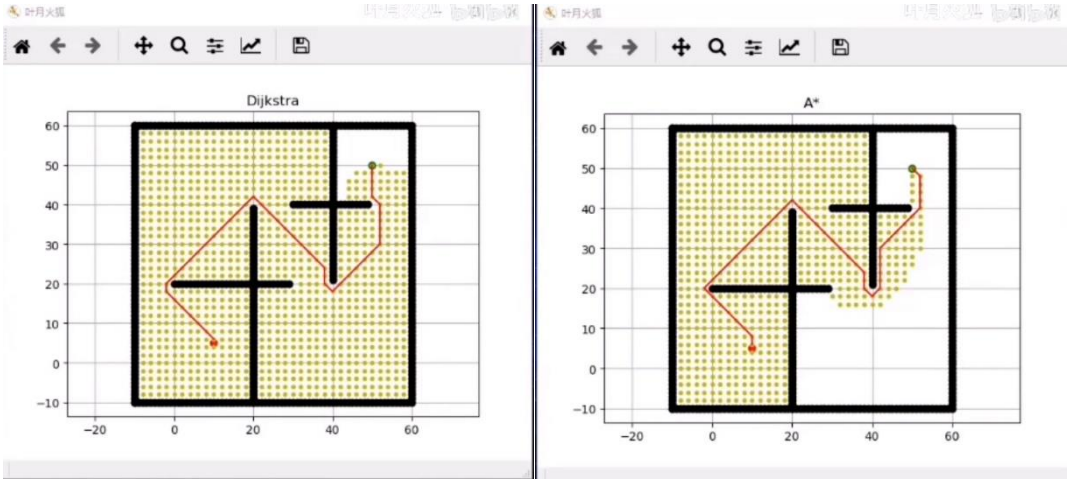


Figure 4. Here is the comparison of the Dijkstra algorithm and A* algorithm.

Here is the comparison of the Dijkstra algorithm and A* algorithm. From the graph we can see the points of the A* algorithm is less than the points of the Dijkstra algorithm, so that means the speed of A* algorithm is faster than the Dijkstra algorithm. The efficiency of the A* is greater than the Dijkstra algorithm.

3.3. RRT algorithm

Rapid-exploring Random Tree (RRT) algorithm, proposed by Lavalle, is a random sampling algorithm with incremental growth, which is used to solve high-dimensional space problems with algebraic constraints (obstacles) and differential constraints (non-integrity and dynamic environment). The advantage of RRT algorithm is that there is no need to model the system or geometrically divide the search area [24]. It has a high coverage in the search space and a wide range of search, so it can explore the unknown area as much as possible., but at the same time, there is the problem that the algorithm calculation cost is too high. Researchers have proposed various variations of RRT to solve these problems

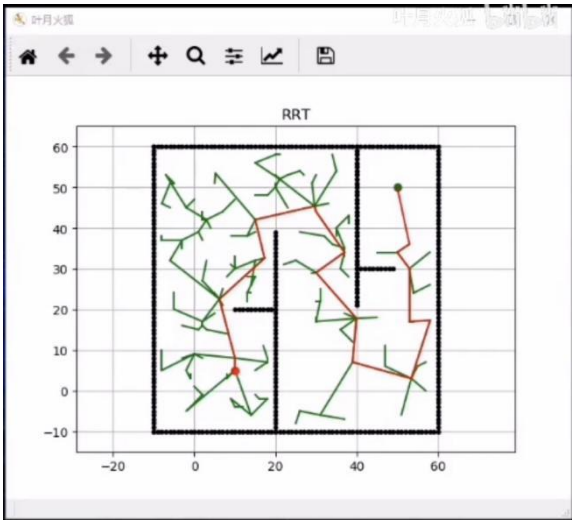


Figure 5. This figure shows how RRT algorithm works. Such search especially brings great advantages to path planning of high-dimensional space.

This is how RRT algorithm working for, compared with other path planning algorithms, such as raster method, A*, D* algorithm, RRT algorithm is characterized by random search of space. Such search especially brings great advantages to path planning of high-dimensional space. However, such search also brings A major disadvantage of RRT algorithm: the algorithm is not high in operation efficiency.

3.4. PRM algorithm

PRM is a method based on graph search. It converts continuous space into discrete space and uses search algorithms such as A* to find paths on the road map to improve search efficiency. This method can find a solution with a relatively small number of random sample points, sufficient to cover most of the feasible space for most problems, and the probability of finding a path is 1 (as the number of samples increases, the P (finding a path) exponent tends to 1). Obviously, when the sampling points are too few or the distribution is unreasonable, the PRM algorithm is not complete, but with the increase of the points used, it can also be complete. So PRM is probabilistically complete and not optimal. It is the same kind of the algorithm within RRT, but RRT is more accurate and spend more time than PRM.

3.5. Genetic Algorithm

Genetic algorithm (GA) was first proposed by John Holland of the United States in the 1970s. The algorithm is designed and proposed according to the evolution law of organisms in nature. By means of mathematics and computer simulation, the algorithm transforms the problem-solving process into a process similar to the crossover and mutation of chromosome genes in biological evolution. When solving complex combinatorial optimization problems, compared with some conventional optimization algorithms, it can usually obtain better optimization results faster. Genetic algorithm has been widely used in combinatorial optimization, machine learning, signal processing, adaptive control, and artificial life [25].

3.5.1. Basic framework

Because genetic algorithm cannot directly deal with the parameters of problem space, the problem to be solved must be expressed as chromosome or individual of genetic space by coding. This transformation operation is called coding or representation. The following three specifications are often used to evaluate the coding strategy.

1)Completeness: all points (candidate solutions) in the problem space can be represented as points (chromosomes) in GA space.

2)Soundness: chromosomes in GA space can correspond to candidate solutions in all problem spaces [26].

3)Non-redundancy: one-to-one correspondence between chromosomes and candidate solutions.

3.5.2. Fitness function

In evolutionary theory, fitness refers to the adaptability of an individual to the environment and the ability of the individual to reproduce. The fitness function of the genetic algorithm is also called the evaluation function. It is an index used to judge the quality of individuals in the population. It is evaluated according to the objective function of the problem.

In the process of searching evolution, a genetic algorithm generally does not need other external information, and only uses the evaluation function to evaluate the advantages and disadvantages of individuals or solutions, which can be used as the basis of genetic operation in the future. It can be seen that in many cases, it is necessary to map the objective function into a fitness function in the form of maximum value and non-negative function value.

The design of fitness function mainly meets the following conditions:

1) Single value, continuous, non-negative, maximization.

2) Reasonable and consistent.

3) Small amount of calculation

4) Strong versatility.

In the specific application, the design of fitness function should be combined with the requirements of solving the problem itself. The design of fitness function directly affects the performance of genetic algorithm.

3.5.3 Operation process editing voice

The basic operation process of genetic algorithm is as follows:

1) Initialization: set the evolutionary algebra counter t = 0, set the maximum evolutionary algebra T, and randomly generate m individuals as the initial population P (0) [27].

2) Individual evaluation: calculate the fitness of each individual in population P (T) [28].

3) Crossover operation: the crossover operator is applied to the population. Crossover operator plays a central role in genetic algorithm [28].

4) Judgment of termination condition: if t = t, the individual with the maximum fitness obtained in the evolution process is taken as the output of the optimal solution to terminate the calculation [29].

3.6. Ant Colony Optimization

Ant colony algorithm is an intelligent optimization algorithm, which solves complex problems through ant colony optimization. ACO has better advantages in discrete optimization problems.2 ant colony algorithm is a probabilistic algorithm used to find the optimal path. It was proposed by Marco Dorigo in his doctoral thesis in 1992.

3.6.1 Algorithm background

The behavior of a single ant is extremely simple. The number of behaviors is less than 10, but the ant colony composed of thousands of ants can have great wisdom, which is inseparable from their way of information transmission - pheromone. During walking, ants release a substance called pheromone to identify their walking path. In the process of looking for food, choose the direction of walking according to the concentration of pheromone, and finally reach the place where the food is located. Pheromones will gradually volatilize over time. At the beginning, because there is no pheromone on the ground, the walking path of ants is random. In the process of walking, ants will constantly release pheromones to identify their walking path. Over time, several ants found food, and there were several paths from the cave to the food. Because the behavior trajectory of ants is randomly distributed, the number of ants on the short path is more than that on the long path in unit time, and the higher the pheromone concentration left by ants. This provides a strong direction for the ants behind, and more and more ants gather on the shortest path.

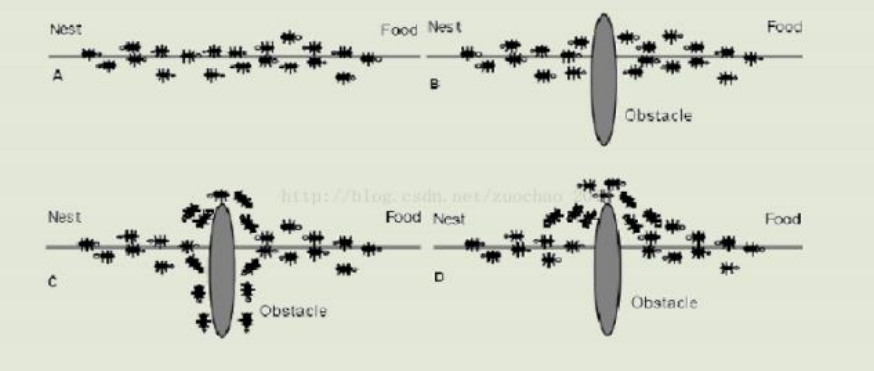


Figure 6. Picture shows that when thousands of ants walk, they release a pheromone substance to identify their path

1)Highly structured organization - although the individual behavior of ants is extremely simple, the ant colony composed of individuals constitutes a highly structured social organization. The members of ant society have division of labor, mutual communication, and information transmission.

2)Natural optimization - during the foraging process, the ant colony can always find the shortest path from the ant nest to the food source without any hint; When there are obstacles on the route, it can quickly find a new optimal path.

3)Positive information feedback - when ants are looking for food, they release pheromones (pheromones) on their path. Ants basically have no vision, but they can detect the trajectory of pheromones emitted by the same species in a small range, so as to decide where to go, and tend to move in the direction of high pheromone intensity.

4)Autocatalytic behavior - the more ants walk along a path, the more pheromones are left (Part evaporates over time), and the higher the probability of ants choosing the path later.

3.7. Polynomial Curve Fitting

Polynomial regression algorithm belongs to supervised regression learning algorithm. The regression algorithm establishes the regression model between variables and obtains the correlation between variables and dependent variables through the learning (training) process. Regression analysis can be used for prediction models or classification models. Common regression algorithms include linear regression, non-linear regression, logistic regression, polynomial regression, ridge regression, Lasso regression and elastic net regression. Among them, linear regression, non-linear regression, and logical regression are the most used.

In many cases, the linear model cannot fit the target data curve well, so it is necessary to introduce the nonlinear regression model. There are many strategies for nonlinear regression. The first strategy is to transform nonlinear regression into linear regression; The second strategy is to transform nonlinear regression into polynomial regression. In polynomial regression, adding a higher power of the feature (such as square term or cubic term) is also equivalent to increasing the degree of freedom of the model to capture the nonlinear changes in the data. In many cases, the linear model cannot fit the target data curve well, so it is necessary to introduce the nonlinear regression model. There are many strategies for nonlinear regression. The first strategy is to transform nonlinear regression into linear regression; The second strategy is to transform nonlinear regression into polynomial regression. In polynomial regression, adding a higher power of the feature (such as square term or cubic term) is also equivalent to increasing the degree of freedom of the model to capture the nonlinear changes in the data.

The main idea of polynomial regression is to fit the equation of polynomial regression through historical data and use the equation of polynomial regression to predict new data. To obtain a polynomial regression model that fits perfectly with the target data set, the essence is to solve the weight of each characteristic independent variable θ. Linear regression first constructs an optimization function of convex function (such as the minimum square sum of the difference between

the given function value and the predicted value of the model) and uses the least square method and gradient descent method to calculate the final fitting parameters.

The core steps of logistic regression classification algorithm are as follows:

Construct the predict function and use the polynomial regression equation.

The loss function is constructed, and the sum of squares of the difference between the given function value and the predicted value of the model is the smallest.

The least square method and gradient descent method are used to calculate the final fitting parameters; Iteratively optimize the final fitting parameters.

The core advantages of the logistic regression classification algorithm for outputting the final fitting parameters are as follows:

Computational scalability: computational complexity is controllable.

Parameter dependence: few adjustable parameters.

Universality: applicable to continuous and discrete data sets, focusing on over fitting.

Anti-noise ability: it is sensitive to missing data and abnormal data and needs special attention.

3.8. Bezier curve

Bessel curve was widely published by French engineer Pierre Bézier in 1962. He used Bessel curve to design the main body of the car. Bessel curve was originally developed by Paul de Casteljau in 1959 by using de Casteljau algorithm to obtain Bessel curve by stable numerical method.

Bezier curve is a kind of spline. In short, Bezier curve is to draw a smooth curve through several specified points. Through the later explanation, we will find that the Bezier curve is a series (say that the advanced point is called a series, and the straight white point is a pile of numbers added together).

The curve is divided into primary, secondary, tertiary, or multiple Bezier curves. The reason for this division is to better understand the connotation.

A Bezier curve is actually a straight line connecting two points. Quadratic Bezier curve is a parabola between two points. A control point is used to control the shape of the parabola. Cubic Bezier curve needs a starting point, an ending point and two control points to control the shape of the curve.

3.8.1. Characteristics of Bezier curve

In Bezier curve, only the starting point and endpoint are on the curve, and other points are the control points for adjusting the shape and end of the curve. The Bezier curve passes through the start point and end point and is tangent to the polyline of the start point and endpoint. In the process of self-vehicle path planning, the self-vehicle start point attitude and target point attitude can be determined according to the tangent direction of the start point and endpoint of the curve. At least third-order Bezier curve (four control points) is required to generate a path with continuous curvature. If the arcs at both ends are required to be spliced together and still have continuous curvature, the curvature of the two arcs at the connection must be equal. If a control point in the Bezier curve changes, the shape of the curve will change accordingly.

3.8.2. Application of Bezier curve

For local path planning in urban environment, for example, Bessel curve can fit straight and curve, and two Bessel curves can be selected for fitting where the curvature changes greatly. For the motion planning of unmanned vehicles, the curvature of the target trajectory is continuous, and the curvature of the trajectory does not exceed the limit of the curvature of the vehicle's drivable trajectory.

4. Conclusions

In this paper we briefly introduce 8 algorithms then we introduced each algorithm more deeply. In my opinion, Path planning under 2D map is a key issue in robot applications. Most related algorithms rely on point-by-point traversal. This causes them usually cannot find the strict shortest path, and their time cost increases dramatically as the map scale increases. In my opinion, these methods have

many limitations, and the current robot planning problem is more and more close to reality, such as the above method can dynamic obstacle avoidance, is not actually true dynamic obstacle avoidance, more is dependent on the prior environmental information for decision making, rather than to the DWA, according to the environmental change in real time. The other point is that you actually have to think more about the behavior of other robots than the environment, and making decisions based on behavior predictions is much harder.

References

[1] Russell, S., & Norvig, P. (2002). Artificial intelligence: a modern approach.

[2] Chouard, T. (2016). The Go Files: AI computer wraps up 4-1 victory against human champion. Nature News.

[3] Chao, X., Kou, G., Li, T., & Peng, Y. (2018). Jie Ke versus AlphaGo: A ranking approach using decision making method for large-scale data with incomplete information. European Journal of Operational Research, 265(1), 239-247.

[4] Vysocky, A. L. E. S., & Novak, P. (2016). Human-Robot collaboration in industry. MM Science Journal, 9(2), 903-906.

[5] Garcia-Alegre, M., Ribeiro, A., García-Pérez, L., Martínez, R., Guinea, D., & Pozo-Ruz, A. (2001, June). Autonomous robot in agriculture tasks. In 3ECPA-3 European Conf. On Precision Agriculture, France (pp. 25-30).

[6] Ohio State University. (2020, September 21). Soft robots, origami combine for potential way to deliver medical treatments: Magnetic fields allow for wireless, faster, less invasive delivery, study finds. *ScienceDaily*. Retrieved October 23, 2021

[7] Lechevalier, S., Nishimura, J., & Storz, C. (2014). Diversity in patterns of industry evolution: How an intrapreneurial regime contributed to the emergence of the service robot industry. Research Policy, 43(10), 1716-1729.

[8] Schlenoff, C., & Messina, E. (2005, November). A robot ontology for urban search and rescue. In Proceedings of the 2005 ACM workshop on Research in knowledge representation for autonomous systems (pp. 27-34).

[9] MAGNUSON, S. (2014). Slow Pace of Robot Acquisition Programs Frustrates End Users. National Defense, 99(731), 28-31.

[10] Fong, T., & Nourbakhsh, I. (2005). Interaction challenges in human-robot space exploration. Interactions, 12(2), 42-45.

[11] Nilsson, N. J. (Ed.). (1984). Shakey the robot.

[12] Mac, T. T., Copot, C., Tran, D. T., & De Keyser, R. (2016). Heuristic approaches in robot path planning: A survey. Robotics and Autonomous Systems, 86, 13-28.

[13] Siddiqui, A. S., & Barton, G. J. (1995). Continuous and discontinuous domains: an algorithm for the automatic generation of reliable protein domain definitions. Protein Science, 4(5), 872-884.

[14] Ge, S. S., & Cui, Y. J. (2000). New potential functions for mobile robot path planning. IEEE Transactions on robotics and automation, 16(5), 615-620.

[15] Tian, L., & Collins, C. (2004). An effective robot trajectory planning method using a genetic algorithm. Mechatronics, 14(5), 455-470.

[16] Deng, Y., Chen, Y., Zhang, Y., & Mahadevan, S. (2012). Fuzzy Dijkstra algorithm for shortest path problem under uncertain environment. Applied Soft Computing, 12(3), 1231-1237.

[17] Yao, J., Lin, C., Xie, X., Wang, A. J., & Hung, C. C. (2010, April). Path planning for virtual human motion using improved A* star algorithm. In 2010 Seventh international conference on information technology: new generations (pp. 1154-1158). IEEE.

[18] Naderi, K., Rajamäki, J., & Hämäläinen, P. (2015, November). RT-RRT* a real-time path planning algorithm based on RRT. In Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games (pp. 113-118).

[19] Bohlin, R., & Kavraki, L. E. (2000, April). Path planning using lazy PRM. In Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065) (Vol. 1, pp. 521-528). IEEE.

[20] Mirjalili, S. (2019). Genetic algorithm. In Evolutionary algorithms and neural networks (pp. 43-55). Springer, Cham.

[21] Dorigo, M., Birattari, M., & Stutzle, T. (2006). Ant colony optimization. IEEE computational intelligence magazine, 1(4), 28-39.

[22] Cleland, A. C. (1994). Polynomial curve-fits for refrigerant thermodynamic properties: extension to include R134a. International journal of refrigeration, 17(4), 245-249.

[23] Choi, J. W., Curry, R., & Elkaim, G. (2008, October). Path planning based on bézier curve for autonomous ground vehicles. In Advances in Electrical and Electronics Engineering-IAENG Special Edition of the World Congress on Engineering and Computer Science 2008 (pp. 158-166). IEEE.

[24] Wang tiefang. Computer genetics grid trust model based on family genes [M]. Beijing: Intellectual Property Press, 2016:93-94

[25] Tao Jinghui, Wang Shengxu, Xu Biao. Optimization strategy of intelligent management of container port [M]: China Fortune press, 2016:61-63

[26] Wang Yanhui, Wang Jingmin. Power system security risk assessment and emergency management [M]: China Quality Inspection press, 2017:92-93

[27] Li Deyi, Yu Jian, Chinese society of artificial intelligence. Introduction to artificial intelligence in the new generation information technology series of China Association for science and technology [M]. Beijing: China Science and Technology Press, 2018:73-82