International Core Journal of Engineering
ISSN: 2414-1895
Volume 6 Issue 5, 2020
DOI: 10.6919/ICJE.202005_6(5).0031

Obstacle Avoidance Trajectory Planning of Loading Robot Based on Improved RRT Algorithm

Chuan Liao[1, a], Yinghua Liao[1, b] and Jun Xie[1, c]

[1]College of Mechanical Engineering, Sichuan University of Science &Engineering, Yibin 644000, China.

[a]635993390@qq.com, [b]191151820 @ qq.com, [c]1225066706 @ qq.com

Abstract

The feeding path of the feeding robot on the production line is directly related to the production efficiency of the production line and the service life of the equipment. Aiming at the problem of artificially adjusting the feeding path of the feeding robot, such as unevenness and poor quality, an improved RRT algorithm is proposed to search for a smooth path and avoid obstacles. High-quality feeding path. First use AABB and K-DOPs bounding box technology to simplify the actual production line environment, then use the improved RRT algorithm to simplify the environment search path, and finally use Mathematica software to complete the obstacle avoidance path simulation. The simulation results show that the loading robot successfully bypasses the obstacle from the starting point to reach the target point, which illustrates the feasibility of the obstacle avoidance path search of the improved RRT algorithm.

Keywords

Robot, RRT algorithm, Obstacle avoidance path, Mathematica.

1. Introduction

In the late 1940s, since the world's first "autonomous" robot was introduced at the Oak Ridge National Laboratory in the United States, the development and application of industrial robot technology has taken the lead in developed countries[1]. The development of robots to today has gone through three generations.

Generally speaking, obstacle avoidance path planning must first meet obstacle avoidance requirements, and secondly, search for an optimal path that meets the requirements[2]. At present, the development of path search algorithms is very mature. Even the algorithms based on C-configuration space for searching through graphs include A* and improved RRT algorithms. A* is a classic algorithm for searching the shortest path to avoid obstacles. The evaluation function has the greatest probability to find the shortest path, but the heuristic search traverses the entire search environment, which makes the search process cumbersome and computationally intensive, especially when the graph (configuration space) is very large, the search program may also be stuck. Aiming at this problem, the improved RRT algorithm uses the thinking of constructing a spatial tree to perform a full-graph search in an incremental manner, which greatly improves the search efficiency.

2. Collision detection algorithm

2.1 AABB bounding box

Collision detection is also known as interference detection[3]. As the name suggests, it is to detect whether an object has collided or interfered with other objects during the movement or assembly of the three-dimensional space.In fact, because the robot's loading environment is relatively complicated,

ISSN: 2414-1895 DOI: 10.6919/ICJE.202005_6(5).0031

both obstacles such as machine tools and machine doors must be considered, as well as obstacles such as silos. These obstacles are geometrically irregular, which makes the robot's collision detection calculation amount huge when loading. Affects detection efficiency. In response to these problems, some scholars have proposed the concept of a bounding box, which is to simplify the irregular obstacles to be detected into a regular but larger model.

AABB bounding box is also called axial bounding box. It is the smallest cuboid that contains objects in three dimensions. As shown in Figure 1, its sides and faces are parallel to three given coordinate systems. Since the AABB bounding box is a cuboid in 3D space (a matrix in 2D space), constructing an AABB bounding box requires 6 scalars, and storage requires 6 4-byte floating-point numbers. Common expressions are As follows:

Width-minimum expression:

$$R=\left\{(x,y,z)\middle| x_0 \le x \le x_0+d_x, y_0 \le y \le y_0+d_y, z_0 \le z \le z_0+d_z\right\} \tag{1}$$

Maximum-minimum expression:

$$R=\left\{(x,y,z)\middle| x_{\min} \le x \le x_{\max}, y_{\min} \le y \le y_{\max}, z_{\min} \le z \le z_{\max}\right\} \tag{2}$$

Center value-radius expression:

$$R=\left\{(x,y,z)\middle| |x_c-x| \le r_x, |y_c-y| \le r_y, |z_c-z| \le r_z\right\} \tag{3}$$

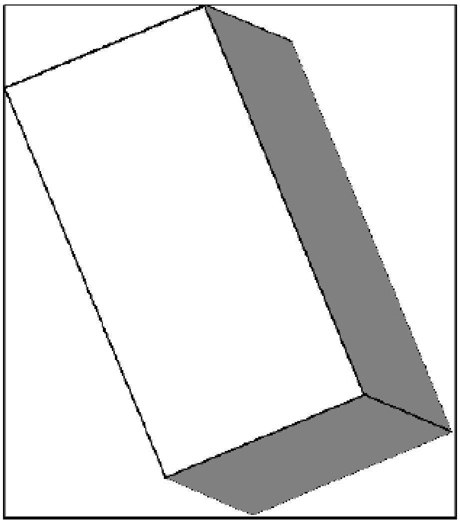

Figure 1. AABB bounding box

The AABB bounding box has the characteristics of simplifying the obstacle geometry. Therefore, when the bounding box intersects, according to the above expressions, it is possible to quickly obtain whether their projections on the coordinate axes coincide[4]. This bounding box method simplifies the three-dimensional space It is a one-dimensional problem, and it goes through 6 operations at most.

2.2 K-DOPs bounding boxes

The K-DOPs bounding box structure is relatively simple[5]. It is determined by the direction vector of each fixed direction of the containing object and the maximum value of each vertex of the contained object. Assuming X is a non-empty point set, the K-DOPs bounding box is expressed in the D direction as follows:

$$FDH_D(X)=\left(H_d(X)\right)_{d\in D} \tag{4}$$

Compared with traditional AABB boxes, K-DOPs have a better compact type, as shown in Figure 2, and this compact type also increases with the increase of K value, so it is easier and simpler to use AABB bounding boxes than obstacles. Convenient and fast, K-DOPs for robots are more accurate and highly accurate.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202005_6(5).0031

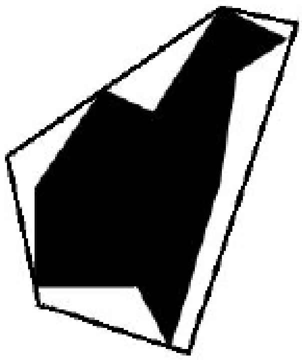

Figure 2. K-DOPs bounding box

3. Improved RRT algorithm

The classic RRT algorithm has many advantages such as rapid expansion, but also has the disadvantages of low quality search paths and strong randomness[6]. In order to enhance the purpose of search and improve the quality of the path, the improved RRT algorithm can quickly find the initial path by re-selecting the parent node and biasing the target guidance method, and continuously optimize the path as the number of sampling points increases.

The obvious difference between the improved RRT algorithm and the classic RRT algorithm is that the target bias of finding a parent node for a child node and adding a probability P is again. Finding a parent node for a child node again means that when a new child node is created, a circle with the child node as the center and a specified distance as the radius will be added, as shown in Figure 3. All of the circles can be used as parent node alternatives, instead of choosing the closest one as the parent node, and finally the parent node is optimized by the evaluation function. The evaluation criterion of the evaluation function is the overall distance from the initial point to the target point. Increasing the target bias of a probability P refers to pointing at a probability P (0 <p <1) to determine whether the tree is growing towards the target or randomly growing during each growth of the random tree, which accelerates the random tree's direction to a certain extent. The growth of the target point improves the quality of the path.

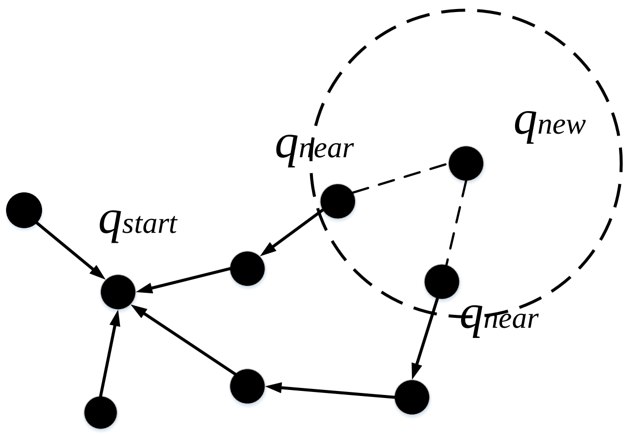

Figure 3. K-DOPs bounding box

4. Simulation results

Collision detection is a complex geometric problem, especially in terms of simulation effect expression. Similarly, this paper uses Mathematica software with strong image interaction capabilities to simulate collision detection. The algorithm principle of collision detection has been described in detail, and the K-DOPs bounding box for robots and the AABB bounding box for obstacles are simplified. The simplified environment is imported into Mathematica software.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202005_6(5).0031

As shown in Figure 4. The size of the map is: $4100 \times 3800 \times 2300\ mm^3$,Obstacle No. 1 in the figure is the earth, Obstacle No. 2 is the workpiece placement device waiting for the robot to grab, Obstacles No. 3-6 are CNC machine tools, and Obstacle No. 7 is other obstacles outside the loading area .

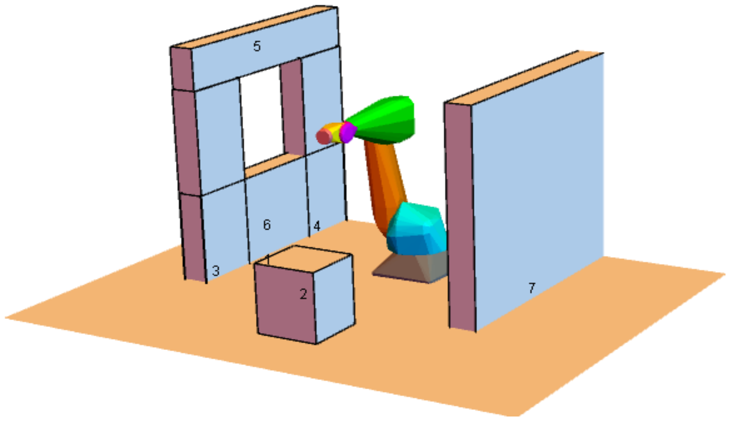

Figure 4. Simplified environment based on bounding box technology

In order to observe the collision detection effect more intuitively, only the robot is set to perform collision detection on obstacle 2. When the robot is not in contact with the obstacle, the obstacle will be grayed out and "OK" will be displayed on the map. When the robot is in contact with the obstacle , The obstacle color will turn red and "Collision" will appear on the map. The implementation code of Mathematica's collision detection algorithm is shown in Figure 5.

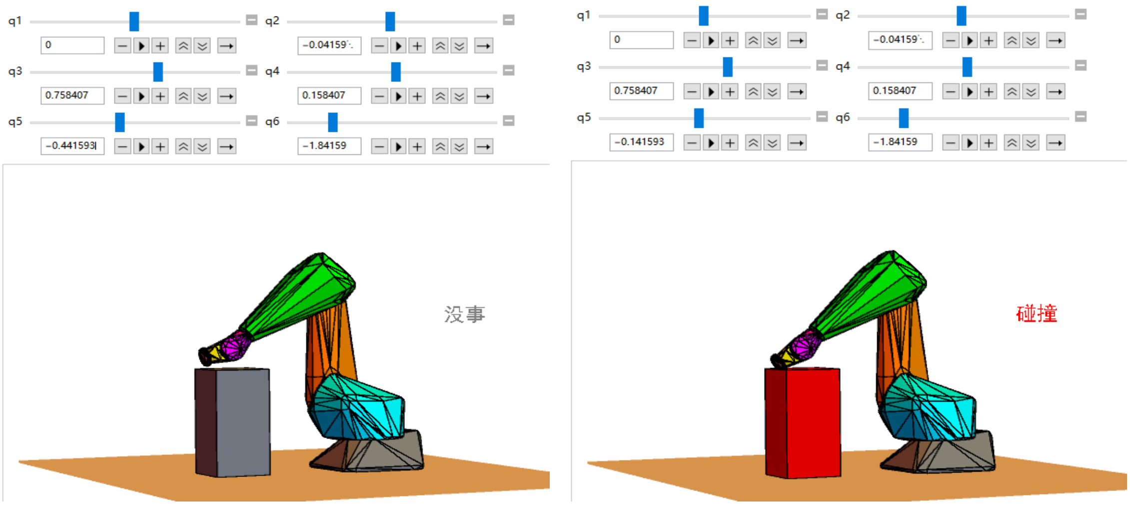

Figure 5. Impact checking

Finally, the simulation of the robot's obstacle avoidance path is shown in Figure 6.

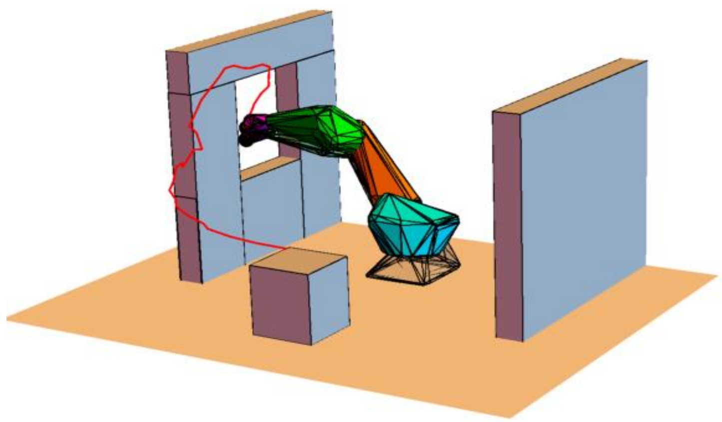

Figure 6. Robot obstacle avoidance path

ISSN: 2414-1895 DOI: 10.6919/ICJE.202005_6(5).0031

It can be seen from Figure 6 that the robot successfully avoided obstacles from the starting point to reach the end point, and left a red trajectory, which is the obstacle avoidance path searched by the improved RRT algorithm combined with the collision detection algorithm.

5. Conclusion

An improved RRT algorithm is proposed, which finds a smooth obstacle avoidance path through data iteration. Use Mathematica software to complete the obstacle avoidance path simulation. The simulation results show that the loading robot successfully bypasses the obstacle from the starting point to the target point, which illustrates the feasibility of the improved RRT algorithm for obstacle avoidance path search.

Acknowledgments

Graduate Innovation Fund(y2018033).

References

[1] Ha Yi, Wu Xi, Cao Ruichao.Application Research of Industrial Robots in Intelligent Manufacturing [J].Hebei Agricultural Machinery, 2019 (12): 62.

[2] Zhang Yihui, Wang Changning, Sun Ling. Research on robot path planning and obstacle avoidance based on A ~ * algorithm [J]. Microcomputer Applications, 2020, 36 (02): 120-123.

[3] Li Yuhu, Wang Zongyan. Improvement of collision algorithm based on hybrid hierarchical bounding box [J]. Journal of East China Jiaotong University, 2019, 36 (06): 112-118.

[4] Lu Jiang, Qian Deying, Zhou Weizhong, Zhou Mi, Xu Rentong, Chen Ning. Assembly collision detection method based on observation coordinates and mixed bounding boxes [J]. Ship Engineering, 2019, 41 (09): 12-16 + 51.

[5] Yu Ruiyun, Zhao Jinlong, Yu Long, Zhang Qianni. Collision detection algorithm combining axis-aligned bounding box and space division [J].Journal of Image and Graphics, 2018,23 (12): 1925-1937.

[6] J. Liu, E.L. Chen and Z.T. He: Journal of Shi Jia Zhuang Railway Institute (Natural Science), Vol. 22 (2009) No. 4, p.40-42.