Collision-free Path Planning Based on Clustering

Lantao Liu, Xinghua Zhang, Hong Huang and Xuemei Ren
Department of Automatic Control，Beijing Institute of Technology，
Beijing, 100081, China

Abstract

The goal of collision-free path planning is to find a continuous path from a starting position to goal position without any collisions. The collision-free path planning is a critical skill in mobile robot applications, and by now it is still a tough problem because mobile robots would easily run into the situation of planning for the current path while neglecting the global goal. In this paper, an approach of path planning based on clustering is presented, which combines the methods of both local collision-free path planning and global path intelligent search. This method firstly set a visual field for robot to imitate human being's driving. Inside of the given visual field, the Maximal-Minimal Distance clustering rule is utilized to tackle the problem of current local obstacle avoidance. Then the strategy of searching an optimal path through a serial of navigating points is introduced with the consideration of a nearest path while searching global goal position. The simulation results prove the effectiveness of the proposed method.

Key words

path planning; clustering; collision-free;

1 Introduction

The goal of collision-free path planning is to find a continuous path from a starting position to goal position, while avoiding obstacles. Collision-free path planning is an important feature of autonomous mobile robot guidance during outdoor missions. The algorithm is especially complex when mobile robot is situated in nonstationary obstacle environment [1, 2]. In such dynamic and real-time environment, collision-free path planning is problematic because the path needs to be continually recomputed as new information becomes available, and that brings forward high level requirements in terms of time and space resource consuming on self hardware condition. Therefore, an effective collision-free path planning method is of importance for those mobile robots under missions in complex environment. Due to a wide array of applications that arises out of various solutions for the robot obstacle avoidance problem, quite an extensive amount of literature has been devoted to this topic. One popular approach for collision-free path planning is APF (Artificial Potential Field) firstly proposed by Professor Khatib [3, 4], subsequently other scholars ameliorated the APF method. APF has limitation because it may get into local minimum point. Carnegie Mellon University has presented a nice method based on grids and it can be well utilized in static environment [5, 6]. Other frequently researched methods consist of the theory of Genetic Algorithm and Neural Network [7, 8, 9]. These categories have advantages in the performance of adapting ability, but path planning has high space and time requirements. Methods based on geometry [10, 11] are usually succinct and clear, but they may lack intelligence in aspect of optimal path searching. In this paper, we propose a new path planning scheme based on clustering to settle the problem between complexity of intelligence and succinctness required by hardware, as well as the conflict between local collision avoidance and global path intelligent search.

Model simplifications with robot soccer simulation platform are introduced to test the effect of the proposed approach. Stochastic models based on the current position of opponent robots are used to describe the obstacles' possible future trajectories. The utility of this clustering method tested on the simulator shows fine results.

2 Problem Descriptions

Collision-free path planning for a robot involves in traveling from one location to another in an obstacle environment without the occurrence of unwanted collisions. In static environment, equipped robot is able to capture all still obstacles' information and store it in its database, and the procession and analysis of the captured data may educe an optimal path for local even global map. However, in the mission of multi-robots' collaboration or antagonism, it is impossible to get obstacles' invariable information. In such situation a global path planning is unpractical especially if the obstacles are running with high speed and the moving rule can not be predicted. The path planning task would easily get into a blind and passive state, that is, mobile robot is able to passively avoid the obstacles at foot, but is unable to predict the moving direction in the future.

Robot soccer is a good testing platform of path planning in both static and dynamic environment, and by now it has become a hot game with characteristic of extensive antagonism. In Figure 1, the vehement competition scene reveals us a good and new type entertainment, at the same time it also brings forth the big challenge on effective path planning methods. In the following sections we will present an approach whereby mobile robot has the ability to estimate and control its general moving direction while avoiding current local obstacles.

Figure 1: Scene of robot soccer

3 Path Planning Strategy

3.1 Movement model and Visual Field

In any two dimensional ground, the position of robot can be defined as $P=[x_c,\ y_c,\ \theta]$, where $x_c(t)$, $y_c(t)$ are the coordinate values on given $x-y$ plane, and $\theta(t)$ is the angle between robot's current direction and x axis. The movement model is described as:

$$\begin{bmatrix}\dot{x}_c(t)\\ \dot{y}_c(t)\\ \dot{\theta}(t)\end{bmatrix}=\begin{bmatrix}\cos\theta & 0\\ \sin\theta & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}v_c(t)\\ \omega(t)\end{bmatrix} \tag{1}$$

And the discrete equation of this movement model is given by:

$$\begin{cases}x(t)=x(t-1)+v_c(t-1)\bullet\cos\theta(t-1)\bullet T_s\\ y(t)=y(t-1)+v_c(t-1)\bullet\sin\theta(t-1)\bullet T_s\\ \theta(t)=\theta(t-1)+\omega(t-1)\bullet T_s\end{cases} \tag{2}$$

where v, w and T_s are robot's linear velocity, angular velocity, and sampling period, respectively.

While mobile robot is avoiding obstacles in a two dimensional plane, it is attempting to find a nearest and safest route leading to destination through obstacles. Because that the position and moving rule of obstacles at a distance can not be detected and predicted, what we need do first is to analyze the current situation. Whereas the current local path planning should not be blind, that requires us to consider the final goal position to meet the basic demand of a nearest length. We will discuss it later for the issue of global path estimation and selection. For current obstacle avoidance in local area, suppose we are driving a car running on a vast dark square at night. The headlights only illuminate a limited area in front of the car, and we see everything clearly in the illuminated area, but see nothing outside of it. The driver on the one side watches forward carefully and avoids the flow of other vehicles, on the other side keeps selecting the correct way leading to the destination. By all appearances the route will be shortest as long as the moving direction is correct and optimal.

To imitate the way people drive vehicles, we set visual field (VF) for robot too, which corresponds to the area illuminated by headlights. It means that only those obstacles inside of VF is visible and need to be computed. It is expected that as more obstacles as possible can be seen within this limited VF , so we define the VF in shape of a rectangular area whose length is about twice longer than width, as it is shown in Figure 2. Therefore, to plan the current local collision-free path, only a limited amount of obstacles inside of this rectangular VF need to be analyzed, which greatly decreases the complexity of information computation. The appropriate direction of VF's orientation also ensures a correct route by actively searching for future ways.

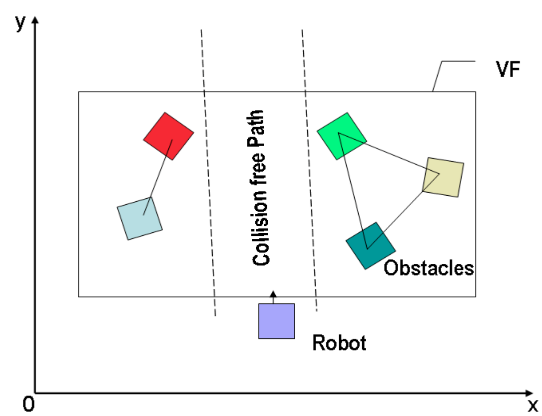


Figure 2: VF of robot and collision-free path

In a given VF, the amount of obstacles, m, can be obtained, and the information of obstacles including the current position (x, y) and moving direction ϕ can also be detected. Suppose the robot has n wheels to be manipulated, then the control of robot is a mapping expression F from the obstacle information matrix to the manipulation matrix of robot.

$$I_{m\times 3} = \begin{Bmatrix} x_1, & y_1, & \phi_1 \\ x_2, & y_2, & \phi_2 \\ \cdots & \cdots & \cdots \\ x_m, & y_m, & \phi_m \end{Bmatrix}, \quad O_n = \left[V_1,\ V_2, \text{..£}V_n\right] \qquad (3)$$

$$F: \ I_{m\times 3} \rightarrow O_n$$

where $I_{m\times 3}$ is obstacles' information matrix, and O_n is robot's manipulation matrix.

3.2 Local area collision-free path planning with clustering

The collision-free path planning in current local area can be regarded as a classification course of clustering. The Maximal-Minimal Distance (MMD) rule is used to classify obstacles.

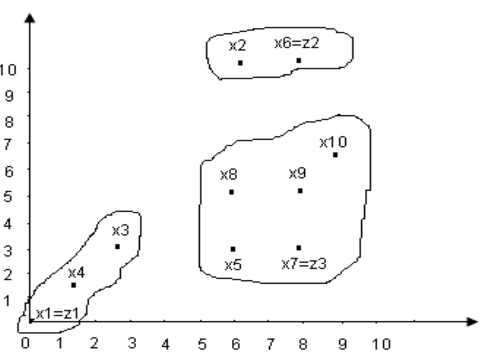


Figure 3: MMD rule sketch map

The procedure of MMD clustering rule is:

Step1:

Assume parameter θ, and $0 < \theta < 1$, randomly set one sample as the cluster center Z_1, e.g. $Z_1 = x_1$ located at original in Figure 3;

Step2:

Search for new cluster center:

Calculate the distance values between all other samples and Z_1, and set the one with the longest distance away from Z_1 as the second cluster Z_2, which is described as $D_{k1} = \max_i \{D_{i1}\}$; In Figure 3, $Z_2 = x_6$;

Calculate distance values D_{i1}, D_{i2} between all left samples and cluster center Z_1, Z_2, if

$$D_l = \max\{\min(D_{i1,} D_{i2}), i = 1,2,...,n\}$$

and $D_l > \theta D_{12}$, where D_{12} is the distance between Z_1 and Z_2, then set x_l as the third cluster center Z_3; In Figure 3, $Z_3 = x_7$;

If Z_3 exists, continue the calculation as follows,

$$D_J = \max\{\min(D_{i1}, D_{i2}, D_{i3}), i = 1,2,...,n\}$$

If $D_J > \theta D_{12}$, set a fourth cluster center…calculate in this way until the maximal minimal distance value is not larger than θD_{12}, then finish the calculation of cluster center search. In Figure 3, there exists only three cluster centers, $Z_1 = x_1$, $Z_2 = x_6$, $Z_3 = x_7$;

Step3:

Gather all those left samples to the nearest cluster centers utilizing Nearest Neighbor clustering rule. In Figure 3, $\{x_1, x_3, x_4\} \in Z_1$; $\{x_2, x_6\} \in Z_2$; $\{x_5, x_7, x_8, x_9, x_{10}\} \in Z_3$

Estimate whether the clustering result is satisfactory utilizing the selected evaluation criterion, if not, we should find a new θ or starting cluster center Z_1, and repeat the calculating procedure begin with step 2.

Obviously, the selections of parameter θ and starting cluster center Z_1 have great influence on the final clustering result. However, clustering can also have a fast convergence rate if some pre-knowledge is available, we will discuss it mainly in Section 3.3. To get ready for it, a positive value T is introduced to be as a distance parameter between two samples, that is, T is the minimal slot width for robot's safe pass. In order to get as more clusters, positive parameter θ should be set a value far less than 1 and satisfy $\theta D_{12} \geq T$.

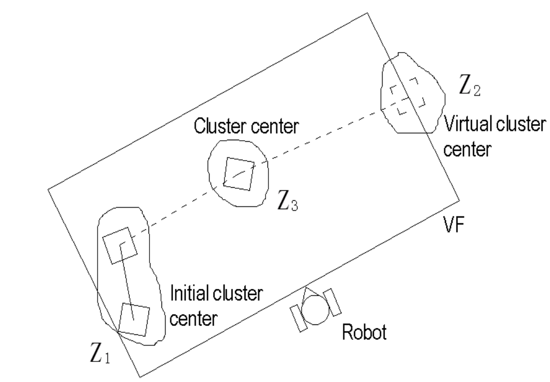
It is necessary to add other constraints when utilizing MMD rule to classify obstacles. To ensure the effectiveness of clustering, during the course of searching the second cluster center Z_2, D_{12} is restricted by:

$$\begin{cases} D_{21} = \max\{D_{i1}, i = 1,2,...,n\} \\ D_{21} > \lambda T \end{cases} \qquad (4)$$

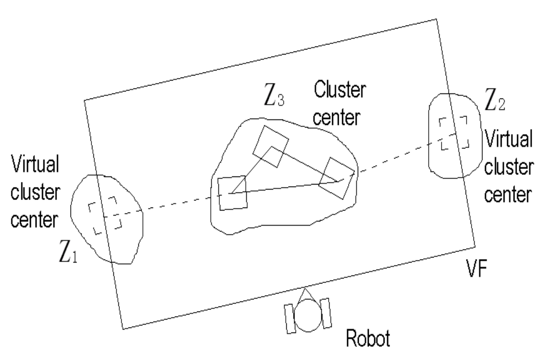
where $\lambda > 1$ is a security coefficient. Otherwise Z_2 and all other samples should be gathered in one cluster. This improvement combines the advantages of Nearest Neighbor clustering rule and MMD rule. The samples are classified generally depending on their distributing density, aiming at securing the slots between neighbor-clusters not to be unsafely narrow, which guarantees the robot body to safely get across.

3.3 Selection of initial cluster center and introduction of virtual samples

The goal of current local obstacle-avoidance path planning in VF is to search a shortest route so as to cross the obstacle groups. It is expected that the cluster result takes on a state of sparsely arraying in a line in front of robot. The restriction of VF's width and distance threshold value T helps to form such expected state. What we need do next is attempt to distribute the clusters onto the periphery of VF in the given situation, namely near the left and right bound of VF. The reason for this operation is to help robot easily estimate and select a satisfactory path among all cluster groups in next step. It is a characteristic of MMD rule to obtain a peripheral clustering state as wanted.



(a) Initial cluster center can be found on VF bounds



(b) Virtual cluster centers substitute real ones
Figure 4: Selection of initial cluster center and introduction of virtual cluster center

The analysis above shows that initial clustering center should be those samples that lie on or near the left or right bound of VF, so as to guarantee the first cluster center to lie on the periphery of whole sample-area. However, sometimes an initial cluster center could not be found because that no sample exists near VF bounds. In order to better implement this path planning approach, we introduce some virtual samples to substitute the unexisted ones, as shown in Figure 4. If no cluster center could be found near the left or right bound of VF, the virtual sample must be added, and the virtual sample should be located on a point which has shortest distance with the nearest cluster center. The introduction of virtual sample not only promotes the exertion of MMD rule, but also prepares for the generation of local obstacle avoidance navigating points in next step.

3.4 Generation of obstacle-avoidance navigating points

We designed the VF in shape of a rectangle and define the length about twice longer than the width, thus all the clusters inside of VF generally can be arrayed in a line in left-right direction. From left to right, calculate the distance matrix $D_{k(k+1)}$ of samples separately from two neighbor clusters,

$$D_{k(k+1)} = \{D_{k(k+1)}(x_{ki}, x_{(k+1)j})\},$$
$$k = 1,2,...,(l-1);\ i = 1,2,...m;\ j = 1,2,...n$$

The slot between two neighbor clusters is a to-be-considered pathway for mobile robot, and the distance $d_{k(k+1)} = \min\{D_{k(k+1)}\}$ is its width. Hence l clusters can generate $l-1$ pathways. Our current task is to estimate and select an available path to avoid obstacles at foot, to achieve it, we set the middle point of pathway width section as current obstacle-avoidance navigating point P_k,

$$P_k = 1/2\begin{bmatrix} x_k + x_{k+1} \\ y_k + y_{k+1} \end{bmatrix},\quad k = 1,2,...,(l-1) \qquad (5)$$

In Figure 5, P_1, P_2, P_3 are current obstacle-avoidance navigating points, and C_1 - C_4 are the real or virtual clusters, and the dotted line shows the direction of final goal position. The obstacle-avoidance navigating points are used to lead collision free ways for mobile robot.

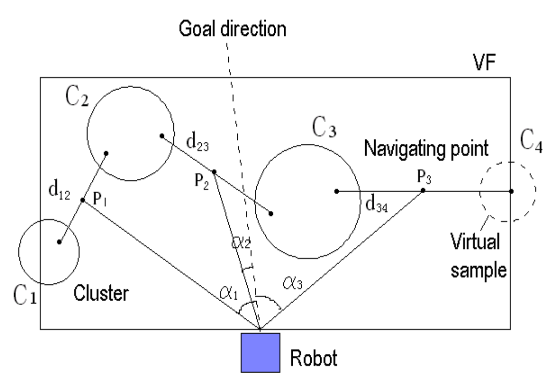


Figure 5: Obstacle-avoidance navigating points

3.5 Estimation and selection of optimal path

The collision-free path of robot is the slot between groups of clustering. Security and shortness of the selected path are two main factors which must be greatly emphasized.

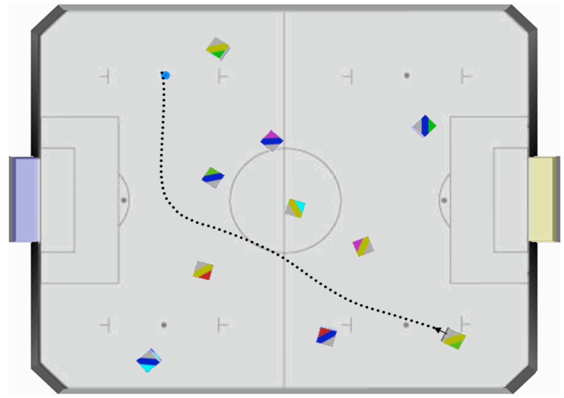
Among all the paths created following the means presented above, most of them are available for robot's passing. Whereas it should always be kept in mind that the final path ought to be integratively optimal, which means the direction of robot must generally lead towards the final goal point. The performance of robot's trajectory mainly lies on two parameters, the distance d between two nearest samples from two different clusters, as well as the angle α between local path navigating direction and final goal direction. The definition of d and α is illustrated in Figure 5. Obviously, the smaller the value of α, the more accurate the robot's orientation, and the bigger the value of d, the more secure the robot's path. Here we use parameter J for evaluation.

$$J = \frac{1}{\alpha}\ln(d - T + 1) \tag{6}$$

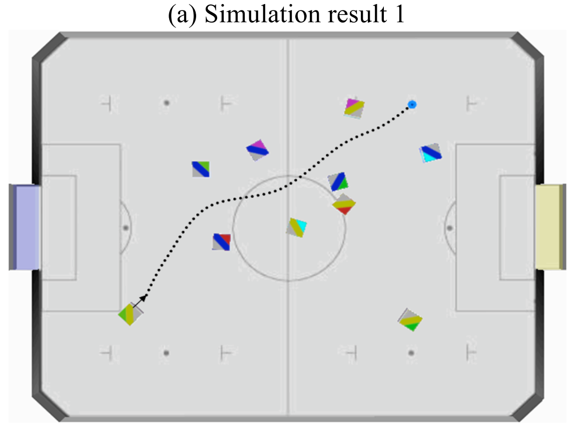
where T is the threshold mentioned before for a safe pass, and α is a radian value. J must be larger than zero and should be selected with its largest value. As it is illustrated in Figure 5, P_2 is the optimal navigating point, and the path which leads towards P_2 will not only avoid collisions, but also show best direction in view of final goal point in global environment, thus resolves the conflict of collision-free path planning between local consideration and global consideration.

4 Simulation

We utilize FIRA (Federation of International Robot-soccer Association) robot soccer standard simulation platform, Robot Soccer v1.5a, to test the approach presented in this paper. This simulation platform was based on Korean Yujin Robot model, and was developed by Australian RSS research group. It provides accurate coordinate values for each robot in a two dimensional playground. Choose one robot to execute collision-free algorithm, e.g., the yellow robot with arrow shown in Fig. 6, and appoint all other home side and opponent side robots as obstacles. Fix the ball at one point in a corner to be as the final goal. Program for obstacle robots to make them run randomly, and let the arrow marked robot to carry out the task of moving towards the ball. Figure 6 shows a collision-free path generated by the arrow marked robot with new algorithm presented in this paper. Extensive amount of simulations achieved a high rate of success.



(a) Simulation result 1



(b) Simulation result 2

Figure 6: Collision-free path planning in simulation

5 Conclusion

Collision-free path planning for a robot involves in traveling from one location to another in an obstacle environment without the occurrence of unwanted collisions. Aimed at solving the problem that mobile robot blindly avoids obstacles in dynamic environment, we proposed a collision-free path planning approach based on clustering. This approach firstly set a visual field for robot to imitate human being's driving. Inside of the given visual field, the Maximal-Minimal Distance rule is utilized to tackle the problem of current local obstacle avoidance. Then the method of searching an optimal path through a serial of navigating points is introduced with the consideration of global goal position. Finally, the results from the computer simulation confirm the effectiveness of the proposed path planning approach.

Acknowledgment

This program has been supported by National Science Foundation in China (No.60474033).

References

[1] Allan R. Willms and Simon X. Yang, An Efficient Dynamic System for Real-time Robot-path Planning, *IEEE Transactions on systems, man and cybernetics*. Vol. 36, Aug, 2006

[2] C. Seshadri and A. Ghosh, Optimum path planning for robot manipulators amid static and dynamic obstacles, *IEEE Trans. Syst., Man, Cybern.*, Vol. 23, pp. 576–584, 1993.

[3] O Khatib, Real-time Obstacle Avoidance for Manipulators and Mobile Robots in Proc. *IEEE Int. Conf. On Robotics and Automation*, pp. 500-505, 1985,

[4] M. Khatib and R. Chatila, An extended potential field approach for mobile robot sensor-based motion. *In Proc. of the Intelligent Autonomous Systems*, IAS-4, IOS Press, pp. 490-496, 1995

[5] Warren C W, A technique for Autonomous underwater Vehicle Route Planning. *In IEEE J oceanic Eng.*, Vol. 15, pp. 199-204, 1990

[6] Elfes, A., Using occupancy grids for mobile robot perception and navigation. *Computer magazine*, pp. 46-57, 1989

[7] S. X. Yang and C. Luo, A neural network approach to complete coverage path planning, *IEEE Trans. Syst.,Man, Cybern. B, Cybern.*, Vol. 34, pp. 718–725, 2004.

[8] S. X. Yang and M. Meng, An efficient neural network method for realtime motion planning with safety consideration, *Robot. Auton. Syst.*, Vol. 32, pp. 115–128, 2000.

[9]Xinying Xu, Jun Xie, Keming Xie, Path Planning and Obstacle-Avoidance for Soccer Robot Based on Artificial Potential Field and Genetic Algorithm. *Proceedings of the 6th World Congress on Intelligent Control and Automation*, Dalian, China, pp. 3494-3498, 2006

[10] Hong Huang, Xuemei Ren, Lantao Liu. A new method of local obstacle avoidance based on geometry. *Journal of Harbin Institute of Technology*, Vol. 38, pp. 1205-1207, 2006

[11] Li He, Chen Zhao, Junqi Zang. Path planning method based on geometry. *Journal of Harbin Institute of Technology*, Vol. 37, pp. 947-949, 2005