Scholar Publishing Group
International Journal of Multimedia Computing
https://doi.org/10.38007/IJMC.2020.010102
ISSN 2789-7168 Vol. 1, Issue 1: 15-30

Motion Control and Path Planning and Tracking Algorithm of Football Robot

Aria Ghabussi*

Texas Tech University, USA

**corresponding author*

Keywords: Football Robot, Motion Control, Path Planning, Robot Motion, Algorithm Research

Abstract: In robotic football, the effectiveness of robot actions and behaviors is completely based on accurate prediction and effective planning of future walking paths and correct adjustment of the posture of the robot when it collides with the football. The purpose of path planning is mainly to plan a collision-free path that meets an evaluation index on a competition-filled field. As the basis for the realization of the basic action of a soccer robot, its pros and cons will directly affect the real-time and accuracy of the action. Therefore, every soccer robot researcher has made it a research focus. This paper mainly studies the motion control and path planning tracking algorithms of soccer robots. In the simulation experiment of this paper, at the end of the interpolation, each joint of the soccer robot has a sudden change in speed from 0.1 rad / s to 0.5 rad / s. Comparing the two speed planning interpolation methods, the speed of the forward segmented speed planning interpolation is more stable. Compared with the trapezoidal speed planning, the speeds jitter of each joint when the robot stops is reduced by more than 90%. This paper proposes the theory of computational verbs combined with the area search method to solve the problem of real-time image information acquisition of small football robotic systems, template matching and similarity solution for robot identification, and decomposing the computational verbs into action words and column verbs to reduce the system's calculation load. It improves the accuracy and stability of image information and lays a foundation for path planning and other research.

1. Introduction

Robot football is a very ingenious and attractive new form and new method for artificial intelligence and robotics to actively integrate into society. The social and economic benefits brought by robot football will come from the progress of multi-agent system research and the successful application in industrial, commercial and military fields. Secondly, robot football provides a vivid form of combining quality education and innovative education with cutting-edge science. Some foreign universities have already offered undergraduate courses in robot football, and the University of Science and Technology of China has also taken the lead in conducting teaching experiments in China. Practice has shown that robot football courses are a feasible way to combine quality education, innovation education and cutting-edge research.

The robot soccer game combines science and athletics, and has received more and more attention [1-2]. In order to organize an effective attack and then score a goal, or to occupy a favorable position in defense first, it is necessary to plan the robot's motion route [3-4]. Robot path planning is one of the basic contents of robot strategy research, and it is a key technology to realize robot intelligence. Its task is to find a collision-free path from the initial state (including position and attitude) to the target state (including position and attitude) in an environment with obstacles and according to certain evaluation criteria [5-6]. Path planning is mainly used in the underlying strategy of robots. As the basis for realizing the basic actions of soccer robots, its pros and cons will directly affect the real-time and accuracy of the actions. Therefore, every soccer robot researcher has made it a research focus.

Ping-Huan Kuo believes that obstacle avoidance is an important issue in robotics. Inspired by the collective behavior of birds, he designed particle swarm optimization (PSO) to solve obstacle avoidance problems. Some animals migrate to different places at certain times of the year and they are called migratory animals. The mover also represents the particles of the particle swarm algorithm, which is used to define the walking path at work. Immigrants consider not only collective behavior but also the geomagnetic field during natural migration. Therefore, in order to improve the performance and convergence speed of particle swarm optimization, he proposed the hybrid particle swarm optimization (H-PSO) algorithm using the concepts in mobile navigation methods. Combined the potential field navigation method with the designed fuzzy logic controller, good results have been obtained in simulation and experiments. Finally, taking the HuroCup obstacle course of the International Federation of Robotic Football Association (FIRA) as an example, the feasibility and practicability of the method were verified in real time. However, no matter whether the number of iterations or the number of populations is increased, the accuracy of his algorithm will not change much, that is, it is easy to fall into the position of a local optimal solution [7]. *Jie Ji* proposed a collision-free path planning and tracking framework for autonomous vehicles. In the path planning method, he superimposed the trigonometric function of the road and the exponential function of the obstacle to construct a three-dimensional virtual danger potential field, which generates an ideal collision avoidance trajectory when the vehicle is likely to collide with the obstacle. Secondly, in order to track the planned trajectory of the collision avoidance maneuver, the path tracking controller describes the tracking task as a multi-constrained model predictive control (MMPC) problem and calculates the front steering angle to prevent the vehicle from colliding with a moving obstacle vehicle. Simulink and CarSim simulations were performed in the presence of motion obstacles. His simulation results show that the proposed path planning method is effective for multiple driving scenarios, and the mmpc-based path tracking controller has good dynamic tracking performance and operability [8]. *Hossein Akbaripour* believes that an indispensable feature of modern intelligent robots is the ability to plan safe short-term movements when encountering obstacles in the work space, which is of great significance for the automatic picking, placing, welding, and painting of industrial robots. On the other hand, as the number of joints increases, the difficulty of collision-free motion planning for serial robots increases exponentially. Therefore, efficient sampling-based motion planning methods are widely used in most practical problems. Aiming at the problem of industrial robot motion planning, he proposed a new sampling-based method—Semi-Delayed Probabilistic Roadmap (SLPRM), which makes full use of the Basic Probabilistic Roadmap (PRM) and Delayed Probabilistic Roadmap (LPRM) The advantages. Unlike the exhaustive and zero-collision checking strategies implemented in PRM and LPRM, respectively, SLPRM collision checking only links m terminal robots (ie, backwards from the end effector) during the roadmap construction phase. As a result, compared with PRM, on the one hand, the roadmap construction time is reduced due to fewer conflict checks; on the other hand, due to the better quality of the initial roadmap, the query time is reduced compared to LPRM. A central decision of SLPRM is to correctly determine the value of m, which directly affects its speed. To this end, he proposed a parameter tuning method based on a combination of Shannon entropy and VIKOR method to determine the optimal values of m and all parameters of the algorithm. Through the simulation of an RV-E3J Mitsubishi industrial manipulator robot and the testing and implementation of actual working space scenarios, his results show that the average planning time of SLPRM is shorter than the planning time of PRM and LPRM. In order to make the algorithm resilient and robust to internal faults and environmental changes (such as position errors, joint failures, and obstacle displacements), he also proposed elasticity and robustness SLPRM, which can modify the program through centralized sampling and road map modification. Handle unexpected failures and changes [9].

This article will focus on the path planning of miniature soccer robots. In robotic football, the effectiveness of robot actions and behaviors is entirely based on accurate prediction and effective planning of future walking paths, and correct adjustment of the posture of the robot when it collides with the football. The purpose of path planning is mainly to plan a collision-free path that meets an evaluation index on a competition-filled field. As the basis for the realization of the basic action of a soccer robot, its pros and cons will directly affect the real-time and accuracy of the action. Therefore, every soccer robot researcher has made it a research focus.

2. Proposed Method

2.1 Path Planning

(1) Description of the path planning problem

The so-called robot soccer path planning problem is to find a path from the starting point to the target point in the robot's working environment that can avoid obstacles according to one or some optimization criteria (such as minimum work cost, shortest walking route, etc.). Optimal movement route [10-11]. When planning a path for a robot, we must consider not only the start and end positions of the path, the environmental conditions of the area where the path exists, but also the action intention of the robot [12]. In the offensive situation, in order to increase the offensive strength, as long as there is a chance to touch the ball, the robot is best to kick the ball to the opponent's half, so the end of the path is the position of the ball, which is a dynamic target that changes over time. ; On the other hand, in the defensive situation, if you want to intercept the ball from behind, and to prevent the emergence of the oolong ball, you should set the target point to a point on the ball’s future trajectory, and use the ball as an obstacle. , To avoid collision with it, to ensure the tightness and security of the defense. When planning a path, the following issues must be addressed first:

1) How the robot obtains the surrounding obstacle information and other related information from the environment;

2) How does the robot answer the current position on the map based on internal and external sensors?

3) How the robot determines the action strategy according to its position on the map and the

information in the current map;

4) How to generate appropriate driving signals to make the robot move on a predetermined trajectory;

2.2 Global Path Planning Method

The methods of global path planning are: topology method, view method and grid method. The topology method divides the planning space into subspaces with topological characteristics, and establishes a topological network. On the topological network, the topological path from the starting point to the target point is found, and the geometric path is finally obtained from the topological path [13-14]. The disadvantage is that the process of establishing a topological network is quite complicated, especially how to effectively modify the existing topological network and increase the graphics speed when adding obstacles are problems to be solved. The view method treats the robot as a point, and combines the robot, the target point, and the vertices of the polygon obstacle. It requires the robot and the obstacle vertices, the target point and the fixed points of the obstacle, and the obstacle fixed points the lines between fixed points cannot pass through obstacles, that is, straight lines are visible [15-16]. The problem of searching for the optimal path is transformed into the problem of the shortest distance from the starting point to the target point through these visible straight lines. Using optimization algorithms, you can remove unnecessary connections to simplify the view and reduce search time. This method can find the shortest path, but it is assumed that the size of the robot is negligible, so that the robot is too close to the obstacle and even contacts it when it passes the apex of the obstacle and the search time is long [17-18].

The grid method decomposes the working environment of the robot into a series of grid cells with binary information. The positions and sizes of obstacles in the work space are the same, and the positions and sizes of the obstacles do not change during the movement of the robot. Use grid pairs of the same size

The two-dimensional working space of the robot is divided, and the size of the grid is based on the size of the robot itself. If there is no obstacle in a grid range, this grid is called a free grid; otherwise, it is called an obstacle grid. Both free space and obstacles can be represented as an integration of grid blocks. There are two ways to identify grids: rectangular coordinate method and serial number method. Mostly use a quadtree or octree to represent the working environment, and complete the path search through an optimization algorithm. This method records environmental information in units of grids. The smaller the grid granularity, the more accurate the obstacles will be represented, but at the same time it will occupy a large amount of storage space, and the search range of the algorithm will increase exponentially. The granularity of the grid is too large,

The planned path will be very imprecise, so the determination of the size of the grid granularity is the main problem of the grid method [19-20].

(1) Simulated annealing algorithm

Simulated Annealing (SA) has been proved to be a very effective global optimization algorithm. It can search and optimize the objective function in the global scope, and this algorithm is versatile and easy to implement, but its convergence speed is very slow, especially when it runs on ordinary microcomputers, it takes a lot of machine time. Modified Simulated Annealing (MSA) not only retains the characteristics of global optimization of SA algorithm, but also improves its convergence speed. The MSA algorithm is based on the graph search technology, removing redundant points generated during the simulated annealing process and regenerating path points, thereby reducing unnecessary path twists and turns, and finally finding a collision-free optimal path for the robot quickly [21-22]. The MSA algorithm is to remove the path points generated in the annealing process and some path points that do not require annealing in advance, and then regenerate the path points. This can reduce the number of path points and shorten the time required for optimization, thereby speeding up the convergence speed. Because the algorithm is based on the SA algorithm, the path finally obtained is still the optimal path. In addition, the MSA algorithm removes some redundant points through filtering, and the optimal path obtained is more optimized than the path obtained by using the SA algorithm alone.

(2) Path planning method based on neural network

Based on the neural network path planning problem, in order to quantify the collision between the path and the obstacle, we must first build a hierarchical network model that is very suitable for optimal design according to the principle of artificial neural network, that is, between the path point and the obstacle Collision penalty function, which is also the most critical step in problem solving. Assume that the obstacles are polygonal and path planning is performed in a two-dimensional space. The network model is described as follows: This is a three-layer network model: the input layer, the middle layer, and the output layer. The excitation function of the neuron nodes in the middle layer and the output layer uses the S-shaped function [23-24].

From a mathematical perspective, the global path planning problem of a mobile robot can be expressed as solving an extreme value problem of an objective function: the objective function is the planned path; the constraint is to avoid collision with obstacles, and its mathematical model can be expressed as follows:

$$\min f(x) \quad X \in R^n \tag{1}$$

$$s.t. g_i(X) \le 0 \quad i = 1,2,3 \cdots p \tag{2}$$

Among them: $f(x)$ is the objective function, $g(x)$ is the constraint condition, and i represents the number of inequality constraints. Because the current algorithm of constrained optimization is difficult to judge the convergence rate compared to unconstrained optimization, and the research and progress of constrained optimization are far worse than unconstrained optimization. Therefore, for this constrained nonlinear optimization problem, the sequence unconstrained minimum The SUMT method is transformed into an unconstrained optimization problem, which is equivalent to optimizing an energy function E, which is composed of two parts: the path length function E_t and the collision penalty function E_c. Quantify them: define E_t as the sum of the squares of all path lengths, ie, for N path points, is defined as:

$$E_t = \sum_{i=1}^{N-1} L_i^2 = \sum_{i=1}^{N-1} \left[(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 \right] \tag{3}$$

Define E_c as the sum of the collision function energy of all path points

$$E_c = \sum_{i=1}^{N} \sum_{k=1}^{K} C_i^K \tag{4}$$

In the formula: N represents the number of path points, K represents the number of obstacles, and C_i^K represents the collision penalty function of the i-th passing point $P(x_i, y_i)$ for the k-th obstacle. Let the energy function of the entire path be E, μ_t, μ_c be weighting coefficients, then

$$E = \mu_1 E_1 + \mu_c E_c \tag{5}$$

Because the entire energy is a function of each way point, by moving each way point to make it move in the direction of reduced energy, the path with the smallest total energy can be finally obtained [25-26].

(3) Artificial potential field method

The artificial potential field method was proposed in 1986. The basic idea is to construct an artificial potential field that interacts with the gravitational field of the target pose and the repulsive field around the obstacle. In this constructed potential field, each point has its own magnitude and direction, and there are peaks and valleys in the potential field. In order to plan the path from the starting point to the target point, you can specify that the pose of the target point forms a gravitational field, the pose of the obstacle forms a repulsive field, and the vector superposition of the gravitational field and the repulsive field constitutes a path planning environment: artificial Potential field [27-28]. Construct a circle with the starting point as the center and r as the radius. The next path point of the robot is selected from this circle. In the potential field, the magnitude and direction of each point potential on this circle are different. Using the principle of maximum potential difference, we choose the point on the same circle that has the largest potential difference from the center of the circle as the next target point of the robot. Just as the steepest slope can be selected to get the maximum speed when the unpowered car goes downhill, this method follows the principle of minimum work cost and labor saving. The prominent advantage of this type of method is that the system's path generation and control directly implements a closed loop with the environment, which greatly enhances the system's adaptability and obstacle avoidance performance [29-30].

2.3 Motion Modeling

According to whether the speed and position of the particle system are subject to certain restrictions, there are non-free particle systems and free particle systems. These conditions that restrict the positions and velocities of the particle points of the system are called constraints, and the expressions that constraints can be parsed by mathematical equations are called constraint equations. Constraints are generally classified into geometric constraints and motion constraints based on constraints (whether they limit the position or velocity of the system's particles). In the constraint equation, only the coordinates of the points of the system are included, and the speed of each point of the system is not included. This kind of constraint is a geometric constraint; if the constraint equation includes the derivative of each coordinate of the system with time, it is a motion constraint. If the motion constraint equation can be integrated into a finite situation, there is no significant difference from the geometric constraint, which is called an integrable motion constraint, and the geometric constraint is generally called a complete constraint? It is impossible to integrate the motion constraint into a finite form, which is called Non-holonomic Constraint.

A system that is subject to non-integrity constraints is called a non-integrity system. Constraints cannot be equivalently expressed as generalized coordinate functions. Constraints contain derivatives of generalized coordinates with respect to time. The football robot studied in this paper is a typical non-integrity system, and its physical meaning is that the robot cannot move along the wheel axis. Non-integrity control systems do not have isolated equilibrium points, so there is no locally asymptotically stable equilibrium point, and only equilibrium manifolds containing far points exist.

As shown in Figure 1, when the control robot reaches the target point T, the distance error d and the angle error α are actually eliminated. The robot cart has two degrees of freedom, namely translation and rotation, and the motion control algorithm is to determine the linear velocity v and the angular velocity error ω through the distance error and the angular error, and then obtain the most accurate control input of the left and right wheel speeds.

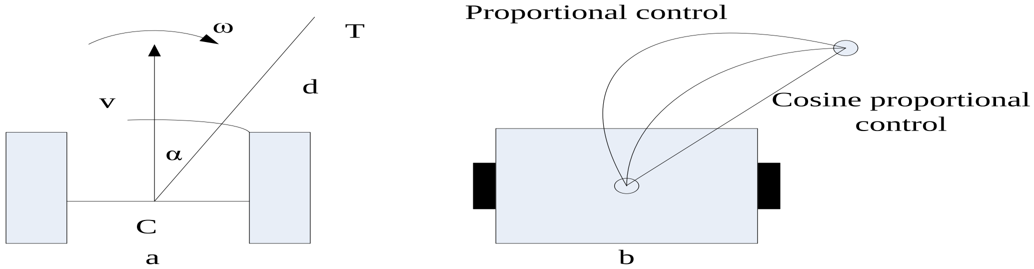

Figure 1: Schematic diagram of controlling the robot to the target point

Controlling the robot's movement to a given target point (that is, to a fixed point action) and controlling its movement toward a given target point (ie, a moving action) are two basic actions of a football robot, and also the basis of other action designs. The fixed-point control method is:

$$v_L = k_p d + k_a \alpha \tag{6}$$

$$v_R = k_p d - k_a \alpha \tag{7}$$

Among them, k_p, k_a are proportional gain and angular gain. The parameters in equations (1) and (2) are difficult to adjust and it is difficult to achieve a better control effect. According to the above formula, improved methods of segmented proportional control and proportional cosine control are proposed. As shown in Figure 1, b is a schematic diagram of the trajectory comparison between the proportional control method and the cosine control method. The algorithm for proportional cosine is:

$$v_L = k_p d \cos\alpha + k_a \alpha \tag{8}$$

$$v_R = k_p d \cos\alpha - k_a \alpha \tag{9}$$

The simulated trajectory of the robot robot from the initial point S to a given target point T is shown in Figure 2.

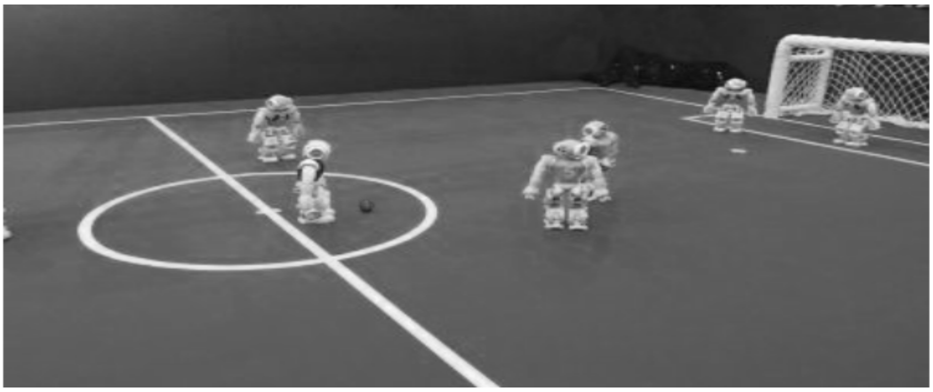

Figure 2: The trajectory of the robot reaching the target point

In fixed-point control, when the robot approaches the target point, the speed of the robot will become lower and lower until it stops. However, during the game, the target is constantly changing. In some cases, the target point will continue upward in the direction the robot needs to move. A distance makes it impossible for the robot to reach the target point, so the robot should maintain a certain speed and increase maneuverability. Of course, it can also be achieved through robot movements. This algorithm can keep the robot at a certain speed even when it is close to the target point. The algorithm is as follows:

$$v_L = k_p \cos\alpha + k_a \alpha \tag{10}$$

$$v_R = k_p \cos\alpha - k_a \alpha \tag{11}$$

In practical applications, the PID error calculation of the angle error and the distance error can be carried out into the above formula to enhance the dynamic characteristics of the system.

Each link of the robot can be regarded as a rigid body, and its motion can be either linear or rotational. To identify the dynamic parameters of the robot, we first need to establish a mathematical expression relationship between the mass center position, mass distribution, etc. of the connecting rod and the joint motion and joint torque according to Newton's equation and Euler's equation. For linear movement, according to Newton's kinematics equation, the driving force required for the linkage to move is a function of the expected acceleration of the linkage and its mass distribution. For rotational motion, according to Euler's equation, the relationship between the combined moment on the rotating rigid body and the rotational angular velocity, angular acceleration, and rigid body inertia tensor can be obtained. The acceleration at the center of mass of the connecting rod is v_c, the resultant force acting on the center of mass is F, and m represents the total mass of the rigid body. From Newton's second law, we can get the mathematical relationship between the three.

$$F = mv_c \tag{12}$$

3. Experiments

3.1 Experimental Environment

The main components of a robot motion control system based on kinematics and dynamics high-speed and high-precision optimization control algorithms can be divided into the following parts: robot, robot control cabinet and other parts. Among them, the robot part is the robot body. In this paper, a six-degree-of-freedom robot is selected for experimental verification. Robot control cabinet is composed of motion controller, servo driver and other parts.

3.2 Dynamic Obstacle Environment Simulation

The coordinates of the starting point of the soccer robot movement are (0, 0) and the coordinates of the target point are (100,100), which are represented by red filled areas in the figure. The circles moving on the way represent the trajectory of the mobile robot, and the rectangles represent the area where the obstacles are located. From the motion trajectory of the mobile robot, it can be seen that when an obstacle is detected on its motion trajectory, the rotation angle and step size are changed according to the designed fuzzy controller. It can be clearly seen that when the distance is close to the obstacle, the step size of the mobile robot becomes significantly smaller, and when it is far away from the obstacle, the step size of the mobile robot becomes significantly larger. In the process of moving toward the target point, the robot effectively avoided obstacles it encountered. When the situation of the space environment is changed, that is, after changing the position and size of the obstacle, the movement path of the mobile robot is also planned according to the aforementioned method, and the mobile robot can still effectively avoid the obstacle.

As shown in Table 1, five interpolation target points were taken to perform a continuous path at a set speed of 1000 mm / s, a set tangential acceleration of $10000 mm/s^2$, a set normal acceleration of $1000 mm/s^2$, and a bow height error limit of 0.01mm Interpolation is performed using the forward-looking speed control method and without the forward-looking control method, respectively, for comparison and simulation.

Table 1: Coordinates of the interpolation target

Teaching point	x/mm	y/mm	z/mm
1p	283.72	195.84	-50.40
3p	264.31	285.54	-54.50
7p	186.82	187.9	260.6
8p	256.53	-93.08	258.4
5p	385.91	-49.44	-54.54

4. Discussion

4.1 Analysis of NURBS Curve Speed Planning Method

As shown in Table 2, 13 teaching points $\{Q_k\}, k = 0,1,...,12$ were selected in the robot's motion space. The weight factors corresponding to the 13 points were all 1, and the 13 points corresponded to the pose of the robot in the base coordinate system.

Table 2: Experimental teaching point data

Teaching point	x/mm	y/mm	z/mm	α/rad	β/rad	γ/rad
P0	456.69	159.10	-58.97	0.75	-0.347	2.969
P1	470.73	158.29	-53.92	0.68	-0.361	2.687
P2	496.76	152.10	-42.57	0.71	-0.312	2.397
P3	536.76	134.10	-25.58	0.43	-0.377	2.269
P4	545.76	166.30	-16.57	0.59	-0.586	2.35
P5	559.36	199.30	-4.9781	0.324	-0.367	1.786
P6	519.36	235.61	-16.678	0.576	0.446	1.875
P7	489.79	280.61	-19.678	0.66	-0.309	1.615
P8	457.79	265.61	-37.678	0.522	0.484	1.786
P9	435.01	258.79	-48.371	-0.39	0.665	1.64
P10	455.01	198.96	-53.371	1.11	0.975	1.543
P11	457.99	175.23	-56.84	0.976	0.865	1.45
P12	456.69	159.10	-58.97	0.952	-0.54	1.969

As shown in Table 2 and Figure 3, the global interpolation method is used to generate a smooth NURBS curve, and then real-time forward segmented speed planning and interpolation are performed. The maximum interpolation speed selected in the $a_{c\max} = 400mm/s^2$ experiment is 100 mm / s. The tangential acceleration $a_{\tan} = 400mm/s^2$, the normal acceleration, and the bow height error limit are set to 0.01 mm. During the interpolation process, in each interpolation cycle, the planned curve parameter u and robot end speed v, and the encoder value of each joint motor are recorded in real time.

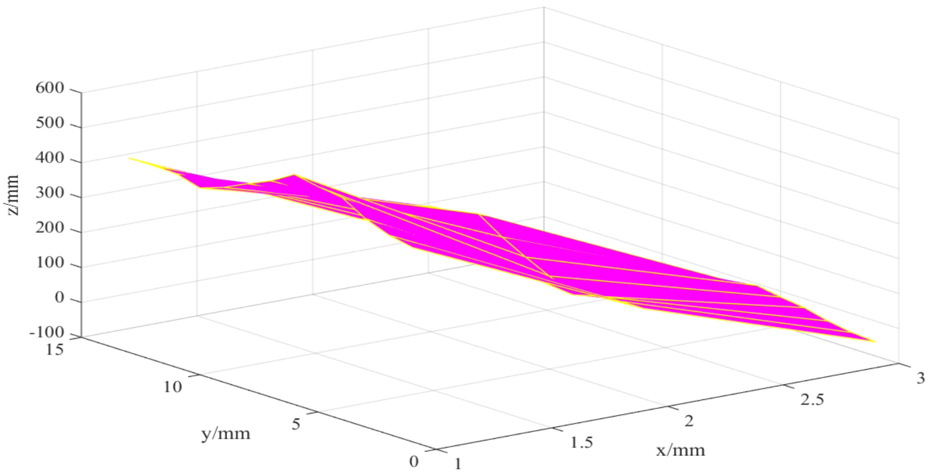

Figure 3: Teaching point data for experiments

The relationship between the interpolation parameter u and the interpolation speed v during interpolation is drawn in MATLAB. As shown in Figure 4, through the forward speed planning segmentation, the NURBS curve is divided into five interpolation segment intervals, each The speed transitions between sections are smooth and there is no sudden change in speed. In order to strengthen the comparison, a NURBS curve interpolation experiment based on the trapezoidal acceleration and deceleration planning at the end of the robot tool was also performed. When using trapezoidal acceleration and deceleration planning, because the geometric constraints of the curve are not taken into account, problems such as uneven speed curves and acceleration jumps due to accumulated vibration are prone to occur. At the same time, because of the accumulation of small errors of the parameter u, the problem of large position and speed inconsistency at the end of the interpolation is prone to occur. The common method is to force the end point of the actual interpolation to reach the set interpolation end point, which causes the end of the interpolation. Sudden changes in angular velocity and angular acceleration of each joint.

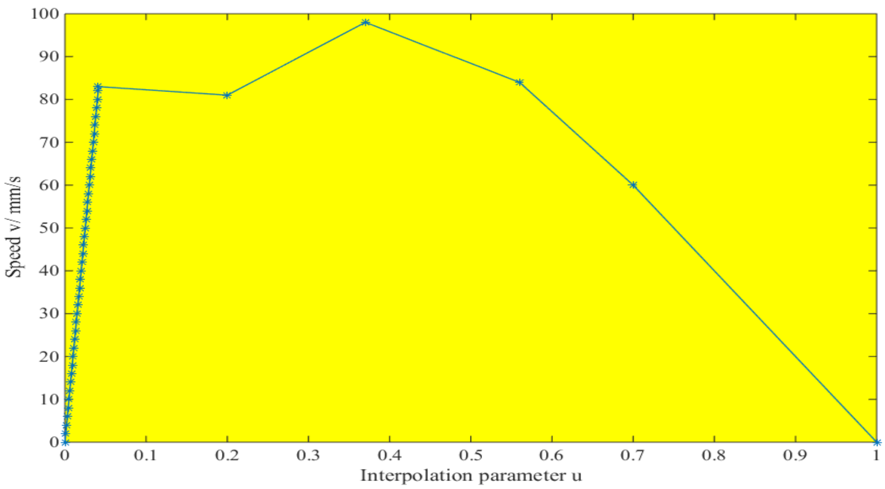

Figure 4: The relationship between the interpolation parameter u and the speed v

4.2 Analysis of Feed forward Torque Compensation Based on Dynamics Model

After identifying the basic dynamic parameters of the robot, a feedforward torque control experiment based on a dynamic model was started. The selected experimental trajectory is a “gate-shaped” trajectory commonly used in the 3C industry. The teaching point of the selected robot motion space is Table 3 shows.

Table 3: Teaching point pose for feedforward torque control experiment

Teaching point	x/mm	y/mm	z/mm	α/°	β/°	γ/°
3p	264.31	285.54	-54.50	73.09	-8.18	176.75
5p	385.91	-49.44	-54.54	18.59	-8.17	176.76
7p	186.82	187.93	260.63	79.76	-33.61	156.57
8p	256.53	-93.08	258.47	14.42	-33.15	156.93

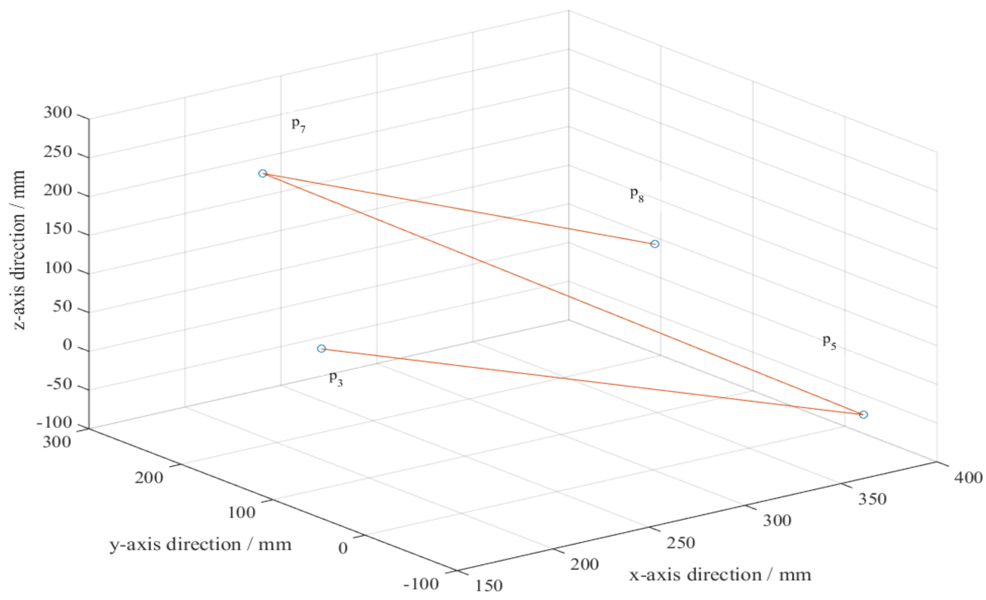

Figure 5: Teaching point pose for feedforward torque control experiment

When the robot's high-speed belt movement stops, due to the existence of the robot's body inertia, the joints of the robot will have a certain degree of jitter, which will cause the motor encoder values to fluctuate up and down. The fluctuation amount of the encoder value of each motor when the robot is stopped can also be used as an important indicator to judge the stability and accuracy of the robot's movement.

As shown in Table 3 and Figure 5, when the feedforward torque control is added, compared with the independent PD feedback control, when the robot is stopped (the interpolation point time series is 1075, each point corresponds to 1 ms), the robot's motor vibrations The situation (encoder values fluctuating up and down) has improved to some extent, and the jitters of the second, third and fifth joint motor encoders have been reduced by more than 20%. Because the main movement of the robot is the second, third, and fifth joints, the feedforward torque control plays a very important role in reducing the jitter of each joint when the robot stops, and improving the positioning accuracy of the robot. At the same time, within 200ms after the robot stops, the actual encoder value of each motor is close to the theoretical encoder value, that is, the feedforward torque control can meet the requirements of robot positioning accuracy.

4.3 Comparison of Interpolation Paths between Two Control Methods

Using forward-looking speed control method for interpolation and not using forward-looking control for interpolation, the interpolation points simulated in MATLAB and the path comparison chart are drawn as shown in Figure 6. It can be seen that after the forward speed control, Intermediate nodes p_7 and p_8 of the motion path are added with a spline curve. The coordinates of the two transition points added at p_7 are p_{71}(189.11, 190. 82, 251. 333) and p_{72} (189.228, 178. 229, 260.558). The coordinates of the two transition points added at 8p are p_{81}(254.13, -83.38, 258. 55) and p_{82}(260.33, -91.81, 249.31). After the transition curve is added, the transition of the continuous path is smoother, avoiding sharp corners between adjacent paths in the continuous path, and reducing the impact on the servo system when the robot passes through the sharp point.

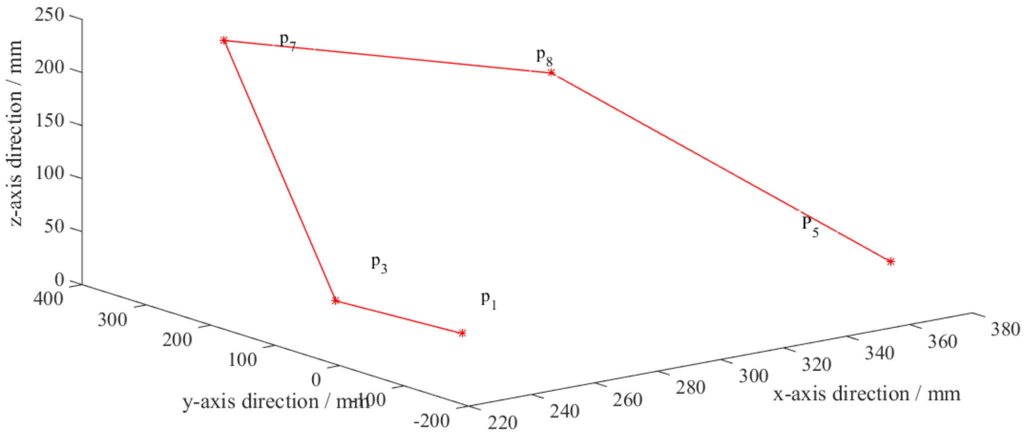

Figure 6: Schematic comparison of two control methods

4.4 Speed Curve Comparison of Two Control Methods

Using forward-looking speed control method, compared with not using forward-looking speed control, makes the robot run the same continuous path, the running time of the robot will be reduced to a certain extent, and the frequent start and stop of the robot can be avoided. Draw the time curve of the end tangential movement speed planned by two speed control methods in MATLAB. The comparison diagram is shown in Figure 7. After the forward speed control, there are 294 mm / s at the two intermediate nodes of the robot's motion path. And the transition speed of 246 mm / s, the speed change is more stable. At the same time, the interpolation time for the entire path of the look-ahead speed control is shortened by 30 ms. This improves the operating efficiency of the robot to a certain extent, and avoids frequent start and stop of the robot, which meets the requirements of high-speed and high-precision control of the robot.

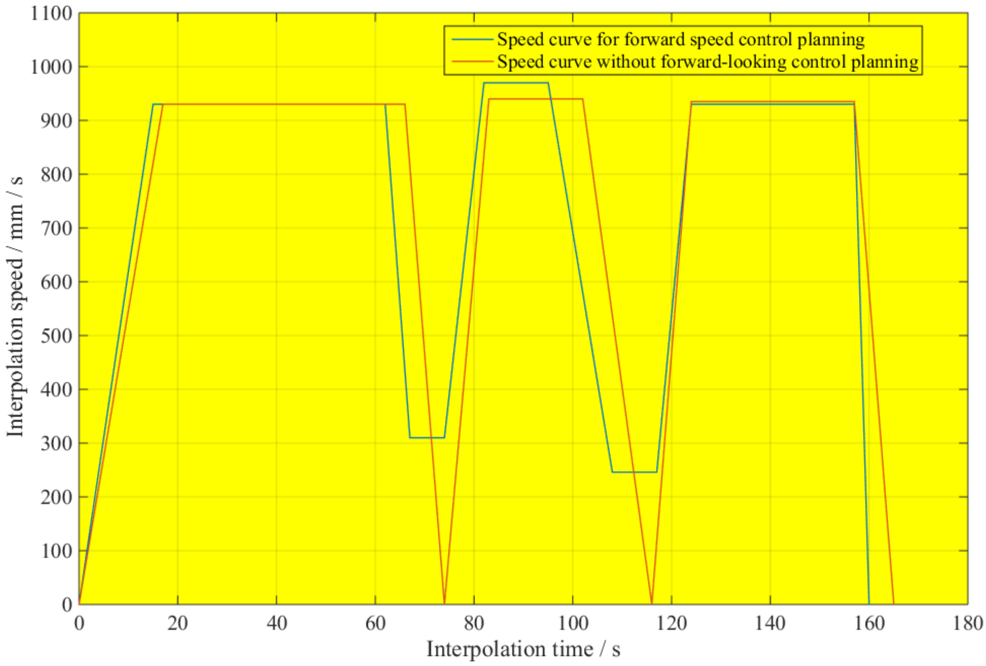

Figure 7: Schematic diagram of speed-time relationship between two control methods

5. Conclusions

This paper mainly studies the forward-looking speed control and interpolation method of robot continuous path. Under the constraints of continuous path geometry, add transition curves at the intersections of straight and straight paths, straight lines and arcs, and arcs and circular paths, and plan backwards and forwards for each segment of the continuous path and the maximum speed. A trapezoidal acceleration / deceleration model capable of adaptively adjusting acceleration and deceleration is established to ensure the accuracy of interpolation. This article proposes a new shooting action. In this method, the robot's target point is not only the ball, but also includes a series of target points determined through planning to guide the robot into the optimal shooting area and complete the final shooting action. First, according to the position of the opposing goalkeeper, determine a shooting point, and then create a straight line from the shooting point to the ball. At the position close to the ball, generate two circles tangent to the line at the same point, and then match the line to complete the shooting action the circle on the same side of the robot is equally divided, and the equal points are used as a series of target points. The robot moves along the target point to enter the best shooting area and can shoot.

In this paper, we study the Cartesian space interpolation method of spatial straight lines, spatial arcs and transition splines, and carry out simulation analysis. This paper studies the speed constraint model of the robot based on geometric constraints such as centripetal acceleration and bow height error of the arc segment. It also studies the speed transition model of continuous adjacent paths and the method of adding a cubic transition spline curve. Based on the above control methods, a forward-looking speed control method based on reverse speed planning is researched to achieve high-speed and high-precision control of the robot while ensuring that the robot's running speed at the end point reaches the set end speed. The simulation results show that when the forward speed control is performed, the path transition of the continuous path is smoother, the speed change is more stable, and the total interpolation time is shortened to a certain extent.

This paper studies the basic motion trajectory of football robots, path planning, basic behavior control, kick angle and other basic motion trajectories to help the overall decision-making process. In the environment with many obstacles, a multi-robot path planning algorithm based on Delaunay triangle mesh and the idea of optimizing the path by using key points and their poses are proposed, so that the small soccer robot motion planning can find a more optimized path It prevents the robot from entering the local minimum dead zone. This method has great advantages for the path planning of soccer robots in high-speed dynamic environments.

References

[1] Kaarthik Sundar, Sivakumar Rathinam. Algorithms for Heterogeneous, Multiple Depot, Multiple Unmanned Vehicle Path Planning Problems. Journal of Intelligent & Robotic Systems, 2016, 88(2-4):1-14. https://doi.org/10.1007/s10846-016-0458-5

[2] Hamed Fazlollahtabar, Samaneh Hassanli. Hybrid cost and time path planning for multiple autonomous guided vehicles. Applied Intelligence, 2017, 48(1):482-498. https://doi.org/10.1007/s10489-017-0997-x

[3] J.-W. Zhang, L. Liu, K. Chen. Omni-directional bipedal walking path planning. Acta Automatica Sinica, 2016, 42(2):189-201.

[4] Wei-Jen Chen, Bing-Gang Jhong, Mei-Yung Chen. Design of Path Planning and Obstacle Avoidance for a Wheeled Mobile Robot. International Journal of Fuzzy Systems, 2016, 18(6):1-12. https://doi.org/10.1007/s40815-016-0224-7

[5] Patrick Grosch, Federico Thomas. Geometric Path Planning without Maneuvers for Non-Holonomic Parallel Orienting Robots. IEEE Robotics & Automation Letters, 2016, 1(2):1066-1072. https://doi.org/10.1109/LRA.2016.2529688

[6] Joonwoo Lee. Heterogeneous-ants-based path planner for global path planning of mobile robot applications. International Journal of Control Automation & Systems, 2017, 15(5):1-16. https://doi.org/10.1007/s12555-016-0443-6

[7] Ping-Huan Kuo, Tzuu-Hseng S. Li, Guan-Yu Chen. A migrant-inspired path planning algorithm for obstacle run using particle swarm optimization, potential field navigation, and fuzzy logic controller. Knowledge Engineering Review, 2016, 32(158):1-17. https://doi.org/10.1017/S0269888916000151

[8] Jie Ji, Amir Khajepour, Wael William Melek. Path Planning and Tracking for Vehicle Collision Avoidance Based on Model Predictive Control With Multiconstraints. IEEE Transactions on Vehicular Technology, 2017, 66(2):952-964. https://doi.org/10.1109/TVT.2016.2555853

[9] Hossein Akbaripour, Ellips Masehian. Semi-lazy probabilistic roadmap: a parameter-tuned, resilient and robust path planning method for manipulator robots. International Journal of Advanced Manufacturing Technology, 2016, 89(5-8):1-30. https://doi.org/10.1007/s00170-016-9074-6

[10] Y. Chen, Y. Tan, L. Cheng. Path planning for a heterogeneous aerial-ground robot system with neighbourhood constraints. Robot, 2017, 39(1):1-7.

[11] Junghun Suh, Joonsig Gong, Songhwai Oh. Fast Sampling-Based Cost-Aware Path Planning With Nonmyopic Extensions Using Cross Entropy. IEEE Transactions on Robotics, 2017, 33(6):1313-1326. https://doi.org/10.1109/TRO.2017.2738664

[12] M. Zimmermann, C. König. Integration of a visibility graph based path planning method in the

ACT/FHS rotorcraft. Ceas Aeronautical Journal, 2016, 7(3):391-403. https://doi.org/10.1007/s13272-016-0197-0

[13] S. Zhang, X. Li, J. Zhang. UAV 3D real-time path planning based on dynamic step. Journal of Beijing University of Aeronautics & Astronautics, 2016, 42(12):2745-2754.

[14] Cheng Zhang. Path Planning for Robot based on Chaotic Artificial Potential Field Method. Science Technology & Engineering, 2018, 317(1):012056. https://doi.org/10.1088/1757-899X/317/1/012056

[15] B. Xu, L. Chen, M. Xu. Path Planning Algorithm for Plant Protection UAVs in Multiple Operation Areas. Transactions of the Chinese Society for Agricultural Machinery, 2017, 48(2):75-81.

[16] Victor Singh, Karen E. Willcox. Methodology for Path Planning with Dynamic Data-Driven Flight Capability Estimation. Aiaa Journal, 2017, 55(8):1-12. https://doi.org/10.2514/1.J055551

[17] Na Lin, Yanan Zheng, Jianming Li. Path Planning Method Based on Taxi Trajectory Data. Computer Applications & Software, 2016, 12(9):3395-3403. https://doi.org/10.12733/jics20105983

[18] Yang Zhang, Kai Tang. Automatic Sweep Scan Path Planning for Five-Axis Free-Form Surface Inspection Based on Hybrid Swept Area Potential Field. IEEE Transactions on Automation Science & Engineering, 2019, 16(1):261-277. https://doi.org/10.1109/TASE.2018.2827102

[19] Bijo Sebastian, Pinhas Ben-Tzvi. Physics Based Path Planning for Autonomous Tracked Vehicle in Challenging Terrain. Journal of Intelligent & Robotic Systems, 2019, 95(2):511-526. https://doi.org/10.1007/s10846-018-0851-3

[20] Shashikant Koul, Timothy K. Horiuchi. Waypoint Path Planning With Synaptic-Dependent Spike Latency. Circuits and Systems I: Regular Papers, IEEE Transactions on, 2019, 66(4):1544-1557. https://doi.org/10.1109/TCSI.2018.2882818

[21] Xiaoyu Tan, Pengqian Yu, Kah-Bin Lim. Robust path planning for flexible needle insertion using Markov decision processes. International Journal of Computer Assisted Radiology & Surgery, 2018, 13(2):1-13. https://doi.org/10.1007/s11548-018-1783-x

[22] Songqiao Tao, Juan Tan. Path Planning with Obstacle Avoidance Based on Normalized R -Functions. Journal of Robotics, 2018, 2018(1):1-10. https://doi.org/10.1155/2018/5868915

[23] E. Liu, X. Yao. AGV path planning based on improved genetic algorithm and implementation platform. Jisuanji Jicheng Zhizao Xitong/computer Integrated Manufacturing Systems Cims, 2017, 23(3):465-472.

[24] M. Zhang, W. Cai, L. Zhou. Obstacles Avoidance Based Quadratic Bezier Curve Path Planning for Wireless Sensor Networks. Chinese Journal of Sensors and Actuators, 2017, 30(10):1596-1601.

[25] L. An, T. Chen, A. Cheng. A Simulation on the Path Planning of Intelligent Vehicles Based on Artificial Potential Field Algorithm. Automotive Engineering, 2017, 39(12):1451-1456.

[26] Xiaochun Lu, Juntao Fei. Velocity Tracking Control of Wheeled Mobile Robots by Iterative Learning Control. International Journal of Advanced Robotic Systems, 2016, 13(3):1. https://doi.org/10.5772/63813

[27] Seokwon Yeom, Yong-Hyun Woo. Person-Specific Face Detection in a Scene with Optimum Composite Filtering and Colour-Shape Information. International Journal of Advanced Robotic Systems, 2013, 10(1):1. https://doi.org/10.5772/54239

[28] Han Wang, Hongjun Zhang, Kun Wang. Off-road Path Planning Based on Improved Ant

Colony Algorithm. Wireless Personal Communications, 2018, 102(2):1705-1721. https://doi.org/10.1007/s11277-017-5229-5

[29] Zhenhua Han, Shugui Liu, Xinghua Li. Path planning method for intelligent CMMs based on safety and the high-efficiency principle. International Journal of Advanced Manufacturing Technology, 2018, 95(9-10):1-10. https://doi.org/10.1007/s00170-017-1500-x

[30] Jungyun Bae, Woojin Chung. A Heuristic for Path Planning of Multiple Heterogeneous Automated Guided Vehicles. International Journal of Precision Engineering and Manufacturing, 2018, 19(12):1765-1771. https://doi.org/10.1007/s12541-018-0205-x