2nd International Conference on Autonomous Robots and Agents
December 13-15, 2004 Palmerston North, New Zealand

Dynamic Model and Shooting Algorithm on Simurosot

Yang TW*, Chan-Tan YW, Lee HA, C Teoh EL, Jiang H, Sng HL

Centre for Advanced Robotics & Intelligent Control
School of Electrical Engineering, Singapore Polytechnic, Singapore, 139651
yangtianwu@sp.edu.sg

Abstract

An accurate dynamic model of the ball and the soccer robot based on the latest platform for Simurosot (5v5) is presented in this paper. The platform allows the strategy development to be carried out easily. Based on this model, a shooting algorithm is proposed. For shooting a fast-moving ball, the shooting angle and the path of the attacker robot is determined dynamically. Simulation results show good performance, especially for shooting the fast-moving ball.

Keywords: robot soccer, dynamic model, shot, Simurosot

1 Introduction

FIRA Simurosot is a standard software platform for simulated robot soccer game. The platform provides the vision data with full consideration of the practical limitation of the real-time vision and the wireless communication system. Therefore, effort is only needed to focus on the development of the algorithm for path planning, obstacle avoidance and strategy designing. Nevertheless, the algorithm developed can be easily adapted in a real robot system.

The robot soccer game presents a dynamic environment [1]. This makes it essential to have an accurate dynamic and collision model for the soccer robot and the ball. The kinematics and sliding mode tracking of wheeled-mobile robot are discussed in [1]~[3]. Based on their principles and our experiences, an accurate dynamic model was established for the Simurosot(5v5)[4].

The soccer robot's shooting action is the key to winning a game. In [1], a method for obstacle avoidance and shooting action is proposed. It basically introduces the method to get round to shoot one stationary ball into the goal, without considering the movements of the ball and the goalie. In a game, the ball is seldom stationary. As such the soccer robot often misses the ball or fails to shoot the ball into the goal. It is a challenge to shoot a fast-moving ball into the goal successfully. An adaptive shooting algorithm is proposed in this paper, which focuses on improving the success rate of shooting a fast-moving ball into the goal. The proposed algorithm first predicts the ball position and the desired shooting angle. It then plans the shooting path dynamically. Finally it calculates the velocities of both the wheels.

2 Dynamic Model

2.1 Dynamic Model for Soccer Robot

The kinematics of wheeled mobile robot is given by Equation 1 from [3].

$$\begin{pmatrix} \dot{x}_t \\ \dot{y}_t \\ \dot{\theta}_t \end{pmatrix} = \begin{pmatrix} v_t \cos\theta_t \\ v_t \sin\theta_t \\ w_t \end{pmatrix} \quad (1)$$

where x_t and y_t are position variables, θ_t is the heading direction angle, v_t is the forward linear velocity, and ω_t is the angular velocity of wheeled robot.

In Simurosot, the size of each soccer robot is limited to 7.5cm cubic. The server sends the vision information to each team every time cycle, i.e. the position and the rotation of every soccer robot and the ball. Each team would then send back its strategy, i.e. the left and right wheel velocities V_l and V_r for every soccer robot. If the velocities of both wheels are constant, the soccer robot will eventually arrive at a stable linear velocity v_m and angular velocity ω_m. v_m and ω_m can be determined by

$$w_m = k_w \times (V_r - V_l) \quad (2)$$

$$v_m = k_l \times (V_r + V_l) \times k_v(V_r, V_l) \quad (3)$$

where V_r and V_l can range from –125 to 125. k_w, k_l and k_v are coefficients. If the length unit, angular unit and time unit are inch, degree and time circle respectively, k_w is about 0.268 and k_l 0.0076. The coefficient k_v is no more than 1, it is a function of V_r and V_1. When V_r and V_1 are equal, k_v is 1. The bigger $|V_r - V_l|$ is, the smaller is k_v. Their relationship is

* Mr Yang Tianwu is currently with Singapore Polytechnic. He is also a PhD candidate of Southwest Jiaotong University, Chengdu, China

shown in Figure1 and a multi-valued function is used to describe the curve of k_v.

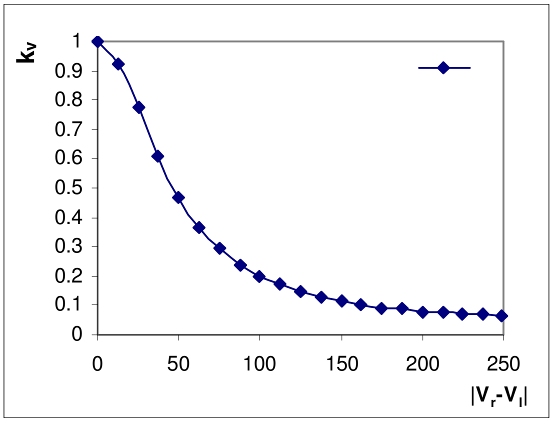


Figure 1: The curve of coefficient, k_v

In Simurosot, the linear and angular accelerations of the soccer robot are proportional to v_m-v_t and ω_m-ω_t, where v_t and ω_t are instant linear and angular velocities at time t. v_t and ω_t can be calculated by

$$w_t = w_m - (1-k_{am})^{t-t'} \times (w_m - w_{t'}) \quad (4)$$

$$v_t = v_m - (1-k_{al}(V_r, V_l))^{t-t'} \times (v_m - v_{t'}) \quad (5)$$

where $v_{t'}$ and $\omega_{t'}$ are the known linear and angular velocities at time *t'*. The velocities of both wheels remain constant from time *t'* to time *t*. The coefficient k_{am} is about 0.268. The coefficient k_{al} is a function of V_r and V_l. It increases with $|V_r - V_l|$. When V_r equals to the V_l, the soccer robot moves in a straight line and k_{al} has a minimum of about 0.059. Its value is no more than 0.268. A multi-valued function is used to describe the curve of k_{al}.

The position and rotation of robot soccer can be easily predicted from equations (1), (4) and (5). This information is very useful for designing a dynamic strategy.

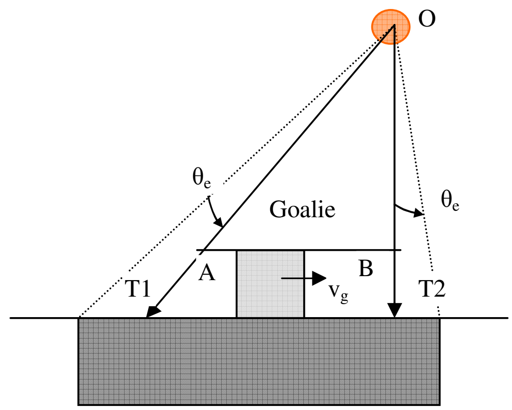
2.2 Dynamic Model for the Ball

In Simurosot, an orange golf ball having a diameter of 42.7mm is used. With small field friction and air resistance, the motion of the ball can be considered linear with a constant speed in a short time, i.e. the displacement in the previous time cycle can be used to estimate the displacement of the current time cycle.

If the ball collides with the wall, the velocity in the tangential direction changes a little, practically keeping its original velocity. But the velocity in the normal direction will change to the opposite direction. The velocity will be stable after 3-5 time cycles and its value is about 0.25~0.3 times the velocity before the collision.

It is very difficult to analyze the collision between the soccer robot and the ball, due to the difficulty in the data collection. But a rule about a moving soccer robot colliding with the ball can be described approximately. In general, the tangential velocity of the ball stays almost constant while the normal velocity of the ball changes to the same as the soccer robot after 3~5 time cycles. After collision, the velocity of the soccer robot will decrease, but the decrease is so little that it can be ignored.

Figure 2: The Shooting Point



3 Adaptive Shooting Algorithm

3.1 The Shooting Point

It is not very ideal in a real match that the robot always shoots the ball at the center of the goal, since the goalie will try to block off the ball. Changing the shooting angle by analyzing the goalie motion can efficiently improve the success rate of scoring a goal. As shown in Figure 2, assuming that the angle θ_e is the shooting angle error of robot, it is better to shoot the ball at T_1 or T_2 points. The choice of the shooting point is based on the predicted time the goalie can arrive at A and B. If the goalie can arrive at A faster than it can at B, T_2 point will be chosen, otherwise T_1 point will be chosen.

3.2 The Shooting Angle

While the ball is moving, the shooting angle of the robot will not be the angle from the predicted ball position to the shooting point, since the tangential velocity of the ball will remain the same after the shooting. As shown in Figure.3, the velocity of the ball should be decomposed in the normal and the tangential direction before the shooting. The tangential velocity remains constant while the normal velocity will change to that of the velocity of the robot, and the resultant velocity will be the velocity of the ball after shooting. The direction of the resultant velocity ought to be the direction from the ball to the shooting point, thus the following function must be satisfied.

$$\frac{y_T - y_O}{x_T - x_O} = \frac{v_b \times \sin(\theta_b - \theta_r) \times \cos\theta_r + v_r \times \sin\theta_r}{-v_b \times \sin(\theta_b - \theta_r) \times \sin\theta_r + v_r \times \cos\theta_r} \quad (6)$$

In equation (6), v_b and v_r are linear velocities of the ball and the robot respectively at the shooting time. Since the motion of the ball can be considered linear with constant speed, the velocity of the ball remains the same in the shooting action. For estimating the ideal shooting angle, i.e. the direction of v_r, the value of v_r has to be predicted first. Based on equation 6, the ideal shooting angle θ_r is calculated.

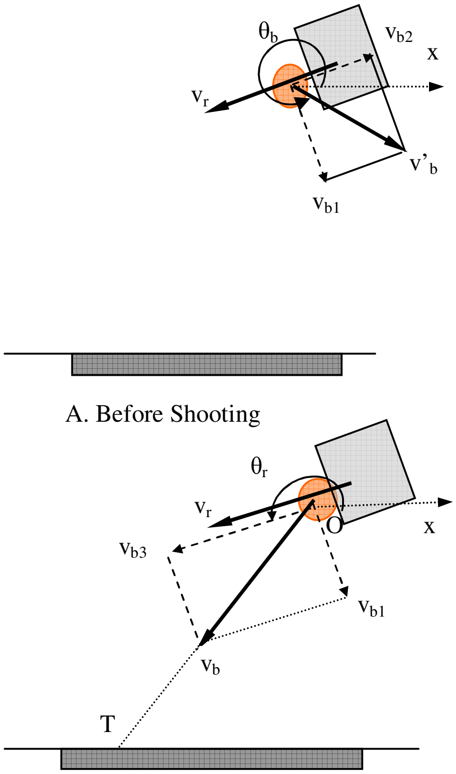


Figure 3 The shooting angle

3.3 Path Planning

Since a lot of factors need be taken into consideration, such as the current velocity of the robot, the shooting angle and the turning angle, etc., the path planning for the shooting action is a complicated problem. In a real match, the ball is always moving; this complicates the problem even further. A simple method of planning a shooting action is to move the robot behind the ball, and then circle it to shoot the ball [1]. This method can be realized easily, but is not very efficient and often misses the shooting opportunity. To increase the shooting opportunity and the success rate of scoring a goal, it is necessary to plan the shooting path dynamically.

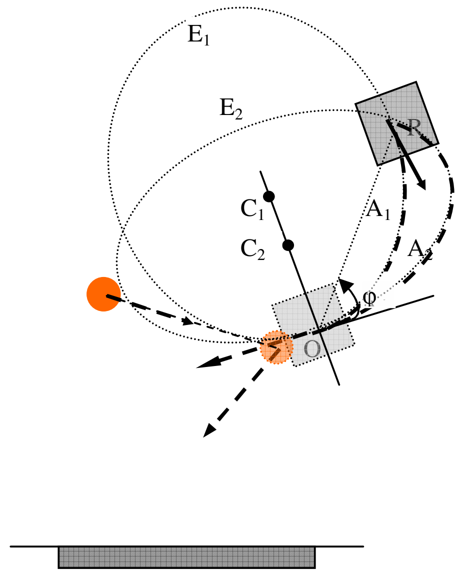


Figure 4 The Planning Path

As shown in Figure 4, OC_1 line is perpendicular to the shooting direction. If a point on the OC_1 line is selected as the centre, an ellipse passing through points O and R can be drawn. The elliptical arc can then be a shooting path, such as elliptical arcs A_1 and A_2 in Figure.4. While the robot moves along the elliptical arc to shoot the ball, the shooting angle is just ideal and the robot can move at a high speed at the same time. This will make it harder for the goalie to block off the ball, thus increase the success rate of scoring a goal. It is difficult to control the robot to move along the planning path precisely, especially when the robot is moving fast and the turning angle needed is big. To find a suitable elliptical arc is beneficial. The nearer the centre of ellipse is to point O, the smaller is the turning angle required near point O, and the easier it is to control the robot, but the path to shoot the ball is longer. The selection of the elliptical arc should take into consideration the dynamic environment and the obstacle avoidance.

3.4 The Algorithm

Based on the above discussion, an adaptive shooting algorithm is proposed. Taking into account that all parameters are dynamic, the algorithm adopts the method of "progressively approaching" to realize the shooting action precisely. The detailed algorithm is summarized as follow.

Step 1: Initialize

If (the first time cycle of shooting action)

Let the predicted linear velocity of shooting v_s=1.5 ;

Let the minimum time cycles needed before shooting t_n=1.

Else let t_n=t_n-4; If t_n is less than 0, let t_n=0.

Step 2: Estimate the time cycles needed from now to the moment the robot shoots the ball.

S2-1: predict the position of the ball based on t_n;

S2-2: Use equation (6) to predict the expected shooting angle.

S2-3: Plan the path based on the shooting angle.

S2-4: Calculate the length of the elliptical arc of the path

S2-5: Decide on the average velocity of both wheels. For shooting the ball fast, always set the velocity of one wheel to maximum, i.e. 125 or –125. The velocity of another wheel is decided based on the turning angle needed.

S2-6: Use equation (3) to calculate v_m based on the velocities of two wheels.

S2-7: Predict the minimum time cycles t_n' needed before shooting the ball.

S2-8: If t_n'- t_n is more than the given error , go to step **S2-1.**

Step 3: Let t_n= (t_n+ t_n')/2; Estimate the position of the ball; Calculate the new v_s; Plan the path.

Step 4: Calculate the velocities of both wheels at the current time cycle and send them to the server.

4 Experiments

4.1 The Precision of Dynamic Model

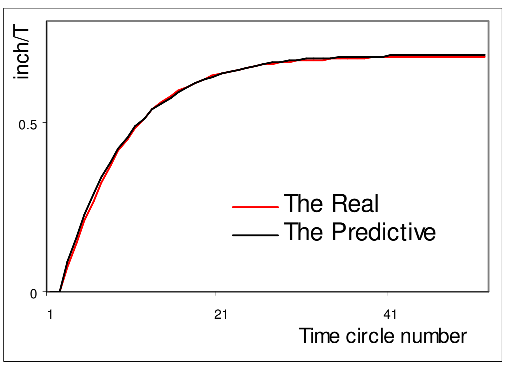


Figure 5 The predictive precision of linear velocity

For simulating the real environment, Simurosot platform adds random interference to the vision information. Without any interference, the presented dynamic model is very precise. The error of instant angular velocity is within 0.05 degree and instant linear velocity within 0.05 inch. The corresponding accumulated error is within 5 degrees and 1-2 inches. Figure 5 shows the predicted and the actual linear velocity where V_r and V_l is 100 and 60 respectively.

4.2 The Precision of Shooting

Table 1 lists the angle errors of shooting a stationary ball. The turning angle needed for shooting a ball is very important. The angle error of shooting increases rapidly with the increase in the turning angle needed. The distance from the robot to the ball also affects the angle error. The effect is obvious when the distance is less than 20 inches, but little when it is more than 30 inches.

Table 2 shows the angle errors and the percentage the robot misses the moving ball. From the table, the velocity of the ball slightly affects the precision of shooting, but the percentage the robot miss the ball was greatly affected.

Table 1 The of shooting a immobile ball

Distance of |RO| / Angle φ	10 Inch	20 Inch	30 Inch	40 Inch
20°	3.6°	2.2°	2.1°	2.3°
40°	5.6°	3.2°	2.7°	2.8°
60°	7.2°	5.1°	4.4°	4.1°
80°	8.5°	6.0°	5.1°	4.4°

Table 2 The angular error of shooting a moving ball

Velocity of the ball	Angle error of shooting a goal	Missing ratio
0.15 inch/T	4.6°	2%
0.3 inch/T	5.3°	4%
0.45 inch/T	5.4°	8%
0.6 inch/T	5.6°	12%

5 Conclusion

Based on the kinematics of wheeled robot and much experiment on the Simurosot, a precise dynamic model is presented, which is fundamental to strategy design. Using this dynamic model, an adaptive shooting algorithm is proposed. It analyses the motion of the ball and the soccer robot, the shooting point and the shooting angle dynamically, this greatly improves the success rate of scoring a goal. The results show that the robot is able to shoot the fast-moving ball with high precision. Since the algorithm aimed at adapting to the dynamic environment, the precision of shooting is slightly affected by the velocity of the ball and the average angle error of shooting is only about 5 degrees. The concept can be easily applied in other similar systems.

6 References

[1] Ching-Chang Weng, Ming-Fong Chou, Chin-Po Hwang,et.. "A method for obstacle avoiding and shooting action of robot soccer", *Proceedings of international conference on Robotic & Automation Seoul Koreal 2001,* pp3779-3782.

[2] Peter J.Thomas, Russel J.Stonier, "Using Evolved Paths for Control in Robot Soccer" In *Proceeding of international symposium on computational intelligence in robotics and Automatio, Japan 2003*, pp735-740.

[3] Dongkyoung Chwa, "Sliding-Mode Tracking Control of Nonholonomic Wheeled Mobile Robots in Polar Coordinates", *IEEE Transactions on Control systems technology*, July 2004, Vol. 12, pp637-644.

[4] Federation of International Robot-soccer Association, "Home Page", http://www.fira.net, visited on 30/7/2004.