Robotica (2004) volume 22, pp. 329–338. © 2004 Cambridge University Press
DOI: 10.1017/S0263574703005654 Printed in the United Kingdom

# Realization of a ball passing strategy for a robot soccer game: a case study of integrated planning and control

Jing-Sin Liu, Tzu-Chen Liang and Yi-An Lin

*Institute of Information Science, Academia Sinica, Nankang, Taipei 115, Taiwan (R.O.C.). Email: liu@iis.sinica.edu.tw*

(Received in Final Form: October 9, 2003)

## SUMMARY

Ball passing is an elementary and frequently employed human soccer skill. This paper examines the realization and visualization of ball passing, a low level move-to-ball behavior of a soccer robot, in a robot soccer game. A case study of three mechanically identical mobile robots with a formation ready to pass a ball cyclically in a zigzag pattern is examined. We build a control command driven mobile robot motion simulator with a controller and dynamics of mobile robots, not only nonholonomic kinematic constraints to simulate the motion of a soccer robot driven by wheels torques to generate wheels accelerations, to update the robot position and orientation at successive time instants. Kick motion follows a physical law, and a simplified collision check and response model is utilized for the efficient detection of the hitting a robot with the ball or other robots. The realization of specific ball passing strategy to drive each soccer robot in a position to receive a pass includes three levels of organization, coordination, and execution: careful integrated design of a dynamic formation and role change scheme, ball position estimation, and coordinated trajectory (i.e. path and velocity) planning and tracking control. Simulations are performed to illustrate the feasibility of the realization of ball passing among three robots, implemented by a software program for coordinated trajectory planning and tracking control in the developed simulator.

KEYWORDS: Robot soccer game; Ball passing; Multiple mobile robots; Simulator.

## 1. INTRODUCTION

In recent years, the robot soccer game has inspired very fruitful research issues such as multi-agent systems, autonomous navigation, multi sensor fusion, fast pattern recognition and vision-based real time intelligent control[1–7] and multi-robot cooperative teams.[8] It also has been proposed as a benchmark problem with sufficient complexity for developing and comparing new methods in the fields of artificial intelligence and multiple robotic systems.

For robot soccer games, there are two teams of wheeled mobile robots embedded with local on-board intelligence. A video camera captures the stadium image and the host computer extracts the locations of the home soccer robots, the opponent soccer robots and the ball. The communication between the soccer robot and host computer is via wireless communication. On the other hand, both major world robot soccer game leagues, Robocup and FIRA, hold simulation competitions, which assume the availability of locations of ball and robots. The objective of a simulation league is for persons who are interested in the software design to concentrate on the study of artificial intelligence or game strategy development.

A robot soccer game is a teamwork game in which multiple soccer robots in a team realize intended motions/actions with collaboration to score and defend.[8] However, communication between robots is limited and noisy, so a high level of strategy for the action decision of each robot based on insufficient information should be carefully planned. Some soccer skills,[1] like incessant ball passing movements, requiring not only fast and accurate information getting and transferring but also high performance of controls and trajectory planning, still seldom appear in physical robot soccer games.[2,3,8] The "Benchmark Test of Robot Soccer Ball Skills" proposed by FIRA (1999)[5] includes a millennium benchmark challenge, which is to control three robots in passing the ball around them, in turn, forming a circuit. Only two teams completed this exercise in FIRA Robot World Cup, 2000.

Robot simulation is a useful tool for verifying the performance of an overall system in a controllable, repeatable software environment. It affords greater flexibility to adapt to new situations and serves as a valuable resource for the development of real robot systems at low cost,[8] though the simulations may miss significant features to show the necessary accuracy of prediction. To obtain simulation results on real robots with acceptable performance is to a large extent founded on the accuracy of the simulations. An accurate simulation of the robot dynamical behavior is thus essential to describe the high-speed motion and for use in learning control program, as small differences between real and model robots are amplified through the robot learning program.[9] The model of robot behavior may be learned from the interactions with the environment by a simulated/physical approach using recorded data from real robot runs.[9] In the first part of the paper, we derive a dynamical model of a wheeled mobile robot, not only for use in simulators for visualization of motion in a timely manner, but for effective control. This expands the functions of SimuroSot,[10] which considers only the kinematics of motion. Ball motion is simplified as a pure slipping motion with friction. When the ball collides with the robot, a non-elastic collision model simulates the ball reaction. The simulator is aimed to serve as a platform for developing and verifying strategies, and trajectory planning and control methods for multiple mobile robots. On the strategy level,[1] conceptual strategies may be planned by techniques from decision theories, artificial intelligence. However, to make the strategy in effect, there should have methods to turn the strategy into collaborative actions that can be performed by the robots which require integrating position estimation, path planning and control.[11]

In the second part of the paper, we examine how to physically realize a specific ball passing strategy among three soccer robots to carry the ball in a specific direction without holding. This case study considers three mechanically identical robots to pass the ball alternatively, which is common in human soccer competitions and is one of basic soccer ball skills in team practicing. As the passing cycle starts, one robot goes to a position behind the ball to kick the ball toward a designed direction and the other two robots move to suitable locations to anticipate a possible pass, following a collision-free trajectory generated by a trajectory planning system. It is assumed that every player does not hold the ball. Specifically, three robots pass the ball with a fixed order and a direction intention and change their formation dynamically according to the ball locus. We also propose the realization of ball passing that subscribes to a basic cycle of prediction, observation, estimation, and coordinated trajectory planning and control, and then describe it by the simulator.

The rest of the paper is organized as follows: In the next section we present the details of a mobile robot motion simulator where the models of the motion of ball, collision and kick are described. In Section 3, the ball passing strategy among three robots and its realization are examined. Section 4 contains simulations performed by the simulator to demonstrate the feasibility of the realization of cyclic ball passing among three mobile robots. Finally, a conclusion is addressed in Section 5.

## 2. THE SIMULATOR: DETAILED DESCRIPTION

### *2.1. Dynamic Model of Mobile Robot*

The unicycle model of a mobile robot with a square shape is shown in Figure 1. The shape of robot is modeled as a rectangle with a center of mass at the center of a polygon. The pose of mobile robot is described by the coordinate $(x, y)$ of the mid point between the two driving wheels, and by the orientation angle $\theta$ with respect to a fixed frame. The dynamic equations of motion are described by[12]

$$\ddot{x} = -\sin(\theta)[\dot{x}\cos(\theta) + \dot{y}\sin(\theta)]\theta + \frac{\cos(\theta)}{mr}(\tau_R + \tau_L)$$

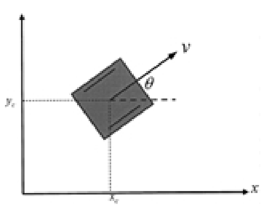



Fig. 1. The schematic representation of the mobile robot.

$$\ddot{y} = \cos(\theta)[\dot{x}\cos(\theta) + \dot{y}\sin(\theta)]\theta + \frac{\sin(\theta)}{mr}(\tau_R + \tau_L) \qquad (1)$$

$$\ddot{\theta} = \frac{l}{Ir}(\tau_R - \tau_L)$$

where $\tau_R$ and $\tau_L$ are driving torques of left and right wheels; $m$, $I$ are the robot mass, moment of inertia, respectively; $r$ is the wheel radius.

Under the hypothesis of "pure rolling" and "non slipping", of wheels motions, the kinematics of robot motion is described by,

$$\begin{aligned} \dot{x} &= v\cos(\theta) \\ \dot{y} &= v\sin(\theta) \\ \theta &= w \end{aligned} \qquad (2)$$

where $v$ is the linear velocity and $w$ is the angular velocity of robot.

**(i) Wheel torque-acceleration relation.** Most mobile robots are controlled by the velocities of their wheels, which are related to the radius of curvature of the car-like vehicles. Therefore, for mobile robot control the velocity command is more natural than torque command. No wheel velocity terms appear in Equation (1) and the constraint equation is implicit. Therefore, an explicit relationship between torque and velocity of wheels is derived and the kinematics would be incorporated into the relationship to assure satisfaction of the kinematic constraint.

Refer to Figure 2.[13] Let $l$ be the distance between the ground contact points of two driving wheels, $R$ be the turning radius of the midpoint point of robot, $v_L$ and $v_R$ denote the velocities of the left driving wheel and the right driving wheel, respectively. Then

$$\frac{1}{R} = \frac{2(v_R - v_L)}{l(v_R + v_L)} \qquad (3)$$

The velocity of a mobile robot can be computed by the average of two wheel velocities,

$$v = (v_R + v_L)/2 \qquad (4)$$

Differentiating the above equation yields the acceleration

$$\dot{v} = \frac{\dot{v}_R + \dot{v}_L}{2} = \frac{\tau_R + \tau_L}{mr} \qquad (5)$$

due to the robot velocity and wheel torques is always at the same direction. Hence,

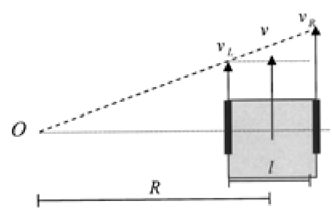



Fig. 2. The relationship between the robot linear velocity, angular velocity and two wheel velocities.

https://doi.org/10.1017/S0263574703005654 Published online by Cambridge University Press

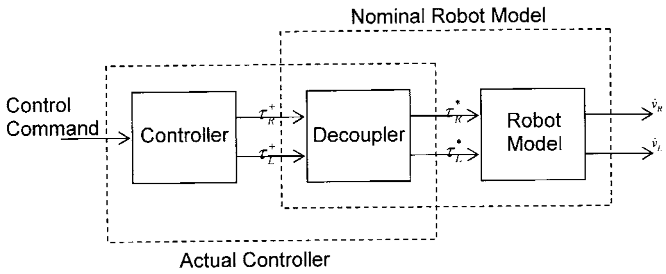



Fig. 3. The connection of decoupler, controller and robot model.

$$\dot{v}_R + \dot{v}_L = \frac{2}{mr}(\tau_R + \tau_L) \tag{6}$$

Furthermore, without loss of generality assume that $v_R > v_L$. From Figure 2, we can derive the relationship of a differential wheel velocity and the angular velocity of robot,

$$w = \theta = \frac{v}{R} = \frac{v_R - v_L}{l} \tag{7}$$

Differentiating the above equation yields

$$\ddot{\theta} = \frac{\dot{v}_R - \dot{v}_L}{l} = \frac{l}{Ir}(\tau_R - \tau_L) \tag{8}$$

where the last inequality follows from the third equation of (1). Thus,

$$\dot{v}_R - \dot{v}_L = \frac{l^2}{Ir}(\tau_R - \tau_L) \tag{9}$$

Solving Equations (6) and (9), the relationship between wheel acceleration and wheel torque can be derived as,

$$\begin{aligned} \dot{v}_R &= \frac{1}{mr}(\tau_R + \tau_L) + \frac{l^2}{2Ir}(\tau_R - \tau_L) \\ \dot{v}_L &= \frac{1}{mr}(\tau_R + \tau_L) - \frac{l^2}{2Ir}(\tau_R - \tau_L) \end{aligned} \tag{10}$$

This set of equations shows a coupled relation that one side wheel velocity is not solely dependent on its wheel torque. However, for a special case $I = ML^2/2$,

$$\begin{aligned} \dot{v}_R &= \frac{2}{mr}\tau_R \\ \dot{v}_L &= \frac{2}{mr}\tau_L \end{aligned} \tag{11}$$

where the torque-acceleration relation is decoupled. This is beneficial for wheel torque controller design.

**(ii) Torque-acceleration Decouple Method.** In general, it is not easy to physically construct a wheeled mobile robot whose moment of inertia is decoupled as (11). Therefore a torque-acceleration decouple method is suggested as follows: Figure 3 shows the block diagram of the torque-acceleration decouple method. The robot model represents the Equation (10), in which the relationship of torques and accelerations of two wheels is coupled. The objective of this method is to convert the output of controller, namely $\tau_L^*$ and $\tau_R^*$, to the nominal $\tau_L^+$ and $\tau_R^+$, whose desired relationship with acceleration of left wheel and right wheel is decoupled as (11). The "Nominal Robot Model" is the series connection of the robot model and a decoupler. As a result, the relationship between input torques $(\tau_R^+, \tau_L^+)$ and output accelerations is decoupled and can be described as,

$$\begin{aligned} \dot{v}_R &= \frac{2}{mr}\tau_R^+ \\ \dot{v}_L &= \frac{2}{mr}\tau_L^+ \end{aligned} \tag{12}$$

The input-output relation of (12) is decoupled (as (11)), suitable for designing control. The "Actual Controller" is the designed controller cascading with the decoupler.

The decoupler, which defines the algebraic relationship between $(\tau_R^+, \tau_L^+)$ and $(\tau^*_R, \tau^*_L)$, can be derived as follows: Substituting $(\tau_R, \tau_L)$ in (10) by $(\tau^*_R, \tau^*_L)$, and combining with (10) to eliminate $(\dot{v}_R, \dot{v}_L)$ yields

$$\begin{aligned} \frac{2}{mr}\tau_R^+ &= \frac{1}{mr}(\tau_R^* + \tau_L^*) + \frac{l^2}{2Ir}(\tau_R^* - \tau_L^*) \\ \frac{2}{mr}\tau_L^+ &= \frac{1}{mr}(\tau_R^* + \tau_L^*) + \frac{l^2}{2Ir}(\tau_R^* - \tau_L^*) \end{aligned} \tag{13}$$

Solving $(\tau_R^*, \tau_L^*)$ from the above two algebraic equations, we obtain the decoupler as the transformation

$$\begin{aligned} \tau_R^* &= \frac{\lambda_2 + \lambda_1}{2\lambda_2}\tau_R^+ + \frac{\lambda_2 - \lambda_1}{2\lambda_2}\tau_L^+ \\ \tau^*_L &= \frac{\lambda_2 + \lambda_1}{2\gamma_2}\tau_R^+ + \frac{\lambda_2 - \lambda_1}{2\lambda_2}\tau_L^+ \end{aligned} \tag{14}$$

where

$$\gamma_1 = \frac{1}{mr}, \ \gamma_2 = \frac{1}{2Ir}.$$

### 2.2. *Collision and Collision Response*

In physical robot soccer games, collision response between two mobile robots is a complex phenomenon involving impact effects in dynamics and cannot be perfectly simulated by a simple mathematical model. Since the simulator is developed for the purpose of strategy development and controller design, a complex yet accurate collision model does not quite fit the use for efficiency concern in the simulation of a dynamically changing environment due to interaction of multiple mobile robots and ball.

**2.2.1. Robot-robot Collision Check.** A simplified collision response model which can avoid the overlap of mobile

https://doi.org/10.1017/S0263574703005654 Published online by Cambridge University Press

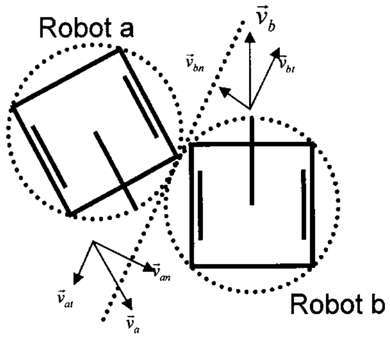



Fig. 4. Robot-robot collision check.

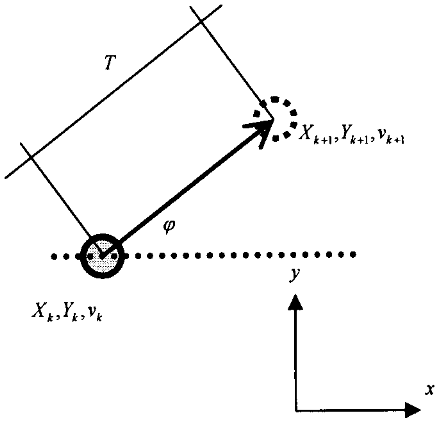



Fig. 5. Ball Motion.

robots when the robots collide is proposed. The robot is modeled as a square in the simulator. The robot-robot collision check is activated after every update of the robot location. We check their collision by representing the mobile robot by its enclosing a circle of radius $r$ for ease of collision check. As shown in Figure 4, two robots are colliding if their enclosing circles intersect, which can be judged by the criterion of the distance between two centers of circles less then $2r$. Robot velocity at collision point is decomposed into a normal and a tangential component. When a collision happens, to avoid overlapping, two robots can only move along the tangential direction of the collision surface (the robot front plane), i.e. the normal components of velocities of two colliding robots with respect to the collision plane are set to be zero.

### 2.2.2. Kick Model: Responding Motion After Kick

**A. Ball Motion.** In the simulator, the ball locus is a straight line and the motion is pure slipping subject to a friction force proportional to its velocity. Under this assumption, the discrete time ball motion (Cartesian position and velocity) can be modeled as,

$$\begin{aligned} v_{k+1} &= v_k + \alpha T \\ X_{k+1} &= X_k + v_k \cos(\varphi) T \\ Y_{k+1} &= Y_k + v_k \sin(\varphi) T \end{aligned} \quad (15)$$

where $\omega_k$, $\omega_{k+1}$ is the ball velocity at $t=kT$, $t=(k+1)T$, respectively. $\alpha$ is a negative constant that describes the deceleration caused by friction force. $\varphi$ is the ball velocity direction (Figure 5). It is assumed that the ball velocity direction $\varphi$ would not change unless a collision with robots happens.

**B. Kick Model.** A kick means a mobile robot collides with the ball. A point contact between robot and ball is assumed. In the simulator, the robot soccer player kicks the ball by its front surface, which is orthogonal to the velocity direction. When a collision of ball and robot occurs, the kick model is utilized to simulate the ball reaction. For the purpose of correctly assessing the physical consequence of a collision, it is assumed that compared to the ball mass, the robot is heavy in weight and would not change its velocity after kicking the ball. Due to the discrete-time nature of the simulator, we shall first solve the real kick time and position for reflection computation needed for describing ball motion after kicking.

Referring to Figure 6, suppose at $T_k=kT$ the ball and the robot are collision-free while at $T_{k+1}=(k+1)T$ the ball and the robot are overlapping, where $T=\Delta T_1+\Delta T_2$, $\Delta T_1$ is time-to-collision. This means that the time step is too large, and a smaller time step must be found. A kick must happen at a certain time instant between $t=kT$ and $(k+1)T$. An estimate time of $\Delta T_2$ is provided by the penetration distance at $T_{k+1}$ divided by the velocity of robot at $T_{k+1}$. The real kick position and time are then solved by simple geometric computations. By the principle of particle mechanics, the ball velocity vector after each kick by the robot can be computed by adding the robot velocity vector $\vec{v}_m$ right before collision and the zero-speed-collision reflection velocity vector $\vec{v}_{bo}$. The ball velocity after kick is the vector $\vec{v}_m+\vec{v}_{bo}$. Finally, the ball position at $(k+1)T$ is the vector of kick position plus ball velocity vector multiplying the travel time $\Delta T_2$.

### 2.3. Simulation Process

**(i) Discrete Time Simulation: Update Equations.** The dynamic model (1) of a wheeled mobile robot is a set of

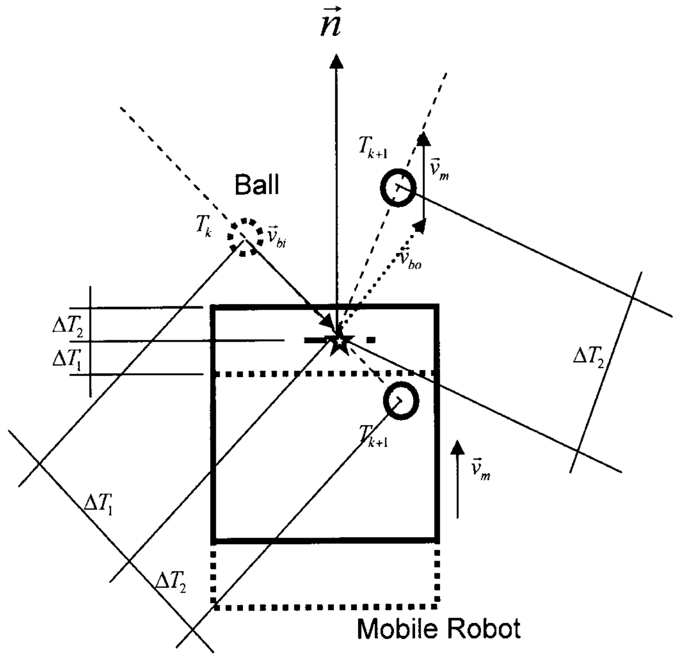



Fig. 6. Kick model.

https://doi.org/10.1017/S0263574703005654 Published online by Cambridge University Press

differential equations governing the dynamics of mobile robots. Physically, it depicts the continuous motion of a mobile robot. However, in computer simulation, we need a discrete time model to fit the computation characteristics of a digital computer. In this subsection, a discrete time model of motion of a mobile robot is developed for use in the simulator.

Now we can update the location of robot from the input torque by incorporating the nonholonomic constraints (2). From equation (13), for wheel input torque ($\tau_R$, $\tau_L$) at time instant $t=kT$, the wheel acceleration pair ($\dot{v}_R$, $\dot{v}_L$), denoted as ($\dot{v}_{Rk}$, $\dot{v}_{Lk}$), can be computed. Then at $t=(k+1)T$ wheel velocities are updated by

$$\begin{aligned} v_{Rk+1} &= v_{Rk} + \dot{v}_{Rk}T \\ v_{Lk+1} &= v_{Lk} + \dot{v}_{Lk}T \end{aligned} \tag{16}$$

Let ($x_k$, $y_k$, $\theta_k$) denote position and orientation of the robot at $t=kT$. By nonholonomic constraints (2), its position and orientation at $t=(k+1)T$ is updated as

$$\begin{aligned} x_{k+1} &= x_k + v_k \mathrm{T} \cos\left(\theta_k + \frac{\theta_k T}{2}\right) \\ y_{k+1} &= y_k + v_k \mathrm{T} \cos\left(\theta_k + \frac{\theta_k T}{2}\right) \\ \theta_{k+1} &= \theta_k + \theta_k T \end{aligned} \tag{17}$$

where $v_k=(v_{Rk}+v_{Lk})/2$, $\theta_k=(v_{Rk}-v_{Lk})/l$. The Equations (17) provide the positions and orientations of the mobile robot at successive time points.

**(ii) Update Cycle.** The whole simulation process of the simulator in one sampling interval is shown in the following recursive process

$t=kT$
Receive Control Commands
Compute Robots Motion
Compute Ball Motion
Check Collision and Kick
Compute Collision Response and Kick Reaction
Let $k=k+1$
Loop

## 3. BALL PASSING STRATEGY AND ITS REALIZATION

### *3.1. Formation and Role Change Scheme in Passing*

This section studies the realization of a ball passing skill, a low level move-to-ball behavior of soccer robot. Consider a passing movement among three mechanically identical soccer robots. It is assumed that relative locations of three robots are initially in a ready formation for passing. As the passing cycle starts, one robot goes to a position behind the ball and the other two robots move to suitable locations to anticipate a possible pass, following a collision-free trajectory generated by a path planning system. There are three roles to be assigned to each robot in a passing cycle (Figure 7): Passer, Previous Player and Next Player. Passer is the robot that runs to a desired position and prepares to kick the moving ball kicked by Previous Player, and Next Player is the robot that the Passer passes the ball to. Each robot acts as one of the three roles at a time and roles change after each kicking: Passer becomes the new Previous Player and Next Player becomes the new Passer. Besides, the Previous Player drives forward to hit the ball and becomes the new Next Player. It is not necessary that three robots have a nearly fixed formation, where each robot moves around a fixed location (as our early work did[14]), during the passing movements. The formation can be dynamically changed according to the main objective of passing a ball and the moving direction of the team. The ball locus and robot trajectories of a ball passing strategy to be investigated in this section are depicted in Figure 8. We design the ball locus to be of a zigzag type and each direction change point of the locus is where a robot kicks it. Three robots kick the ball in turn, and the relative positions of subsequent two kick points of the same robot are a to a′ and b to b′, etc. According to our strategy, the passing movement of each robot is a successive motion, i.e. after kicking the robot runs to the next predicted kick position. Though the situation changes dynamically and the ball motion prediction has

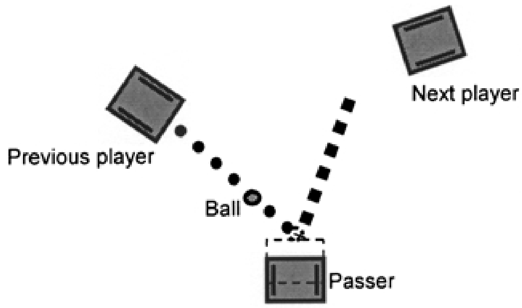



Fig. 7. The three roles of mobile robots in a passing cycle. At this moment, Robot A is the Passer, Robot B is the Next Player to whom the ball should be passed immediately and Robot C is a Previous Player.

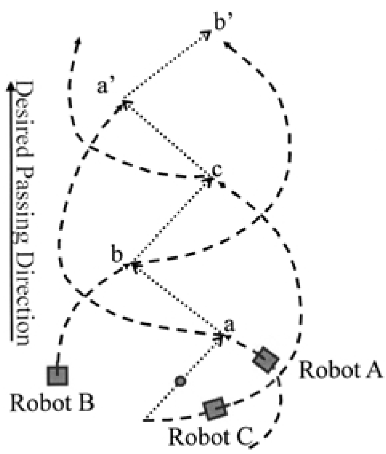



Fig. 8. The tactical plot of ball passing strategy performed by three robots (Robot A is Passer, Robot B is Next Player and Robot C is Previous Player). Dotted lines represent the ball locus. Dashed lines are trajectories of each robot. At each turning point (a, b, c, a′ and b′), a robot kicks the ball.

https://doi.org/10.1017/S0263574703005654 Published online by Cambridge University Press

much imprecision to cause the next ball pass miss, the information of a ball locus is becoming more accurate while the next kick is pending. There are two times for a soccer robot to adjust its trajectory by switching to arrive at the next kick position (the target position for passing) on time. The active motions for each of the robots are identical. For the whole motion, all three robots with different assigned roles plan their paths by a single trajectory planner, and then the ball passing strategy is achieved by one tracking controller.

The elementary skill for a robot to pass a ball includes the ability to go to a target position, the ability to kick the ball toward a desired direction suitable for another kick by another robot, and to add kinetic energy to the ball to accelerate it to resist the natural deceleration caused by friction. For kick position prediction, we assume that the ball velocity after each kick can be observed by a sensor and by prediction, so that the ball locus is exactly known at all time instants. The Passer chooses a point on ball locus and run there to kick the ball at the specific time instant. For a successful passing, the robot should hit the ball when the ball returns to it again. When the Passer kicks the ball, two things will happen. First, roles change as described above. As for the Previous Player and Next Player, because next kick positions are not known precisely at present, an approximate prediction is made as follows. As shown in Figure 9, the next two kick positions are predicted by the user-defined vectors $\vec{v}_1$ and $\vec{v}_2$. The choice of these two vectors is dependent on the designed zigzag locus of ball motion, which is a function of current kick direction and kick velocity. Second, the kick position is observed by sensor and two vectors are added to this position for setting the temporary goal position of the new Next Player and new Previous Player, who act the role of Previous Player and Passer, respectively, in a current passing cycle. Initially, the vector $\vec{v}_1$ suggests the approximate next kick position for Previous Player. The robot is driven there and the trajectory is switched as the robot's role changes to Next Player. A more accurate kick position prediction by $\vec{v}_2$ is added to a new kick position to plan a new trajectory. The robot is steered to track the new generated trajectory and hit the ball. Then when the role is a Passer again, a new passing cycle starts and now the robot obtains the exact ball information and tries to kick the ball.

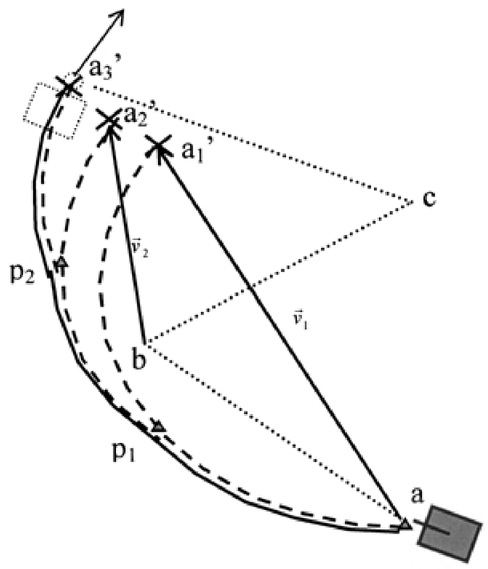



Fig. 9. This Figure shows the three kick-position corrections of a robot. Dotted lines represent the ball locus; dashed lines are three trajectories generated by kick prediction, and thick solid line is the actual trajectory that the robot desires to track. Vectors $\vec{v}_1$ and $\vec{v}_2$ are applied to generate the trajectory $a - a_1'$ and $p_1 - a_2'$ separately as robot's reference trajectory. Motion predictor tells the robot the exact kick position ($a_3'$), and trajectory $p_2 - a_3'$ is generated.

To summarize, for realization of a zigzag ball passing, each robot switches its trajectory twice between its two subsequent kicks by motion prediction. The switching of trajectory is based on the other two robots' kick positions and two prediction vectors. Though there is a chance that two robots collide, if the vectors $\vec{v}_1$ and $\vec{v}_2$ are chosen appropriately, the switch of trajectory would cause each robot motion for ball passing collision-free, and therefore the passing cycle will be fluent.

### *3.2. Trajectory Planning*

As described in Section 3.1, a successful passing cycle requires trajectory generator of mobile robots. As a passing cycle starts, the ball is the target of movement of mobile robots. The motion objective of three robots in a passing cycle is clear and the same: move to the next kick position to kick the ball to start a new passing cycle, while maintaining a formation suitable for passing. Thus, only one kind of the trajectory planning problem needs to be solved for three robots. For convenience of demonstrating the passing movements, let the time duration between two subsequent kicks be fixed. The trajectory planning problem can be decomposed into a geometric path planning problem and a velocity planning problem. In our ball passing problem setting, assume that the target location (next kick location) has been determined *a priori*. Then the path planning problem is to find a curve to connect the start-position, which is the current kick position, and the goal-location, which is the next kick position and orientation. It is noted that the selection of this curve takes into consideration the kick direction, i.e. the robot orientation at the kick position, for a successful ball passing. On the other hand, the velocity planning problem is to determine a velocity profile of a robot moving on the planned path.

**3.2.1. Circular Path Planning: Computation of Center and Radius.** Path planning problem for wheeled mobile robot has been widely analyzed in many studies.[15,16] They showed that a path synthesized by several basic curves, for example: straight lines or part of circles, is practical because the generation of a path is fast and the tracking of path is easy. For the path planning problem related to passing, it is necessary to generate a path between two kick positions (current and next). Here, the selection of circle as the planned path is enough to meet the need, due to its simplicity of computation in a dynamic environment. Referring to Figure 10, the current-point ($P_c$) and the temporary goal-point ($P_e$) of the robot are known. The orientation vector of robot at end-point ($\vec{v}_{eb}$), which is the

https://doi.org/10.1017/S0263574703005654 Published online by Cambridge University Press

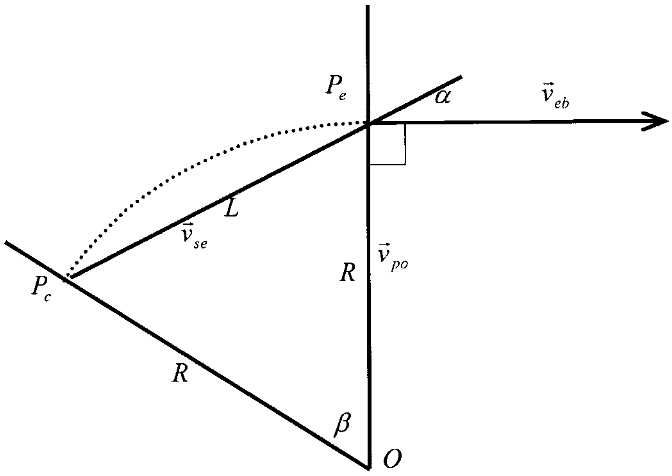



Fig. 10. The geometric diagram for the computation of circular path. $P_c$ is the current position of robot. The dotted line shows the planned path, which is an arc of the circle centered at $O$ with radius $R$. $P_e$ is the predicted kick position, which may be computed by $\vec{v}_{se}$ (either $\vec{v}_1$ or $\vec{v}_2$ in Figure 8). is the desired orientation for robot to kick the ball.

desired kick direction, is decided by strategy beforehand. Then the radius ($r$) of the circular path can be solved by geometric computation, as

$$R=\frac{L}{\sqrt{2(1-\cos(\beta))}}, \beta=2\alpha \tag{18}$$

where $L$ is the Euclidean distance between current and end positions (i.e. the length of the vector $\vec{v}_{se}$). Besides, geometric computation solves the location of the center of this circle ($O$). Arc $\widehat{P_cP_e}$ is the proposed path. Note that the mobile robot orientation at the start-point is not specified here, i.e. the starting configuration is not completely specified. It is only defined by the position with the direction being around a desired kicking direction. This is because as a new ball passing cycle starts, a robot switches to a new planned trajectory, the orientation error is small and can be easily regulated by tracking control. This is feasible for ball passing, as will be demonstrated by simulations in Section 4.

**3.2.2. Velocity Planning.** Along a selected arc path $s_r$, the velocity profile of the robot between two kicks is planned by meeting four boundary conditions imposed at start and end times,

$$s_r(0)=0, \dot{s}_r(0)=V_{robot}\equiv\dot{s}_0$$
$$s_r(t_f)=R\cdot\beta\equiv s_f, \dot{s}_r(t_f)=V_d\equiv\dot{s}_f \tag{19}$$

where $V_{robot}$ is the current velocity of robot; $V_d$ is the desired kick velocity. The instant that the robot starts to execute the path is set to be 0 for convenience, and the time duration allocated by strategy for the robot to traverse the path connecting two kicking positions is assumed fixed as $t_f$. We formulate the time trajectory as a 3rd-order polynomial,

$$s_r(t)=q_1t^3+q_2t^2+q_3t+q_4 \tag{20}$$

Then the velocity profile $\dot{s}_r(t)$ is

$$\dot{s}_r(t)=3q_1t^2+2q_2t+q_3 \tag{21}$$

Substituting the four boundary conditions (19) into the above two equations, the four coefficients of cubic polynomial trajectory (20) can be solved as,

$$q_1=\frac{\dot{s}_f't_f-2s_f'}{t_f^3}, q_2=\frac{3s_f'-\dot{s}_f't_f}{t_f^2}$$
$$q_3=\dot{s}_0, q_4=0 \tag{22}$$

where we define $s_f'=s_f-\dot{s}_0t_f$ and $\dot{s}_f'=\dot{s}_f-\dot{s}_0$.

*3.3. Tracking Control*

We have found a connected circular arc as the reference path of the robot's mass center, and a cubic polynomial velocity profile along the path. In this section we realize the tracking of planned trajectory of soccer robot in ball passing by designing a tracking controller so that the robot can follow the desired trajectory with a tolerable accuracy to successfully kick the ball.

Three errors of robot location can be defined for a robot to track a reference trajectory: $e_s$, $e_d$, and $e_\theta$. As depicted in Figure 11, at a certain instant $t'$, $e_d$ is the signed distance between the robot and the reference path, and distance between the projection, $s(t')$, and $s_r(t')$ is $e_s$. $e_\theta$ is the orientation error (heading misalignment) of the mobile robot to the tangent direction of the reference path at position $s(t')$.

We define

$$\tilde{\tau}(t)=\ddot{s}_r(t)-k_2\dot{e}_s(t)-k_1e_s(t) \tag{23}$$

where the position and velocity tracking errors are defined as $e_s(t)=s_r(t)-s(t)$ and $\dot{e}_s(t)=\dot{s}_r(t)-\dot{s}(t)$. Furthermore, let

$$\tau_\Delta(t)=\frac{R}{l}k_2\dot{s}_r(t)+k_3e_d(t)+k_4e_\theta(t) \tag{24}$$

The first term in the RHS of Equation (24) is the compensation of differential velocity derived from dynam-

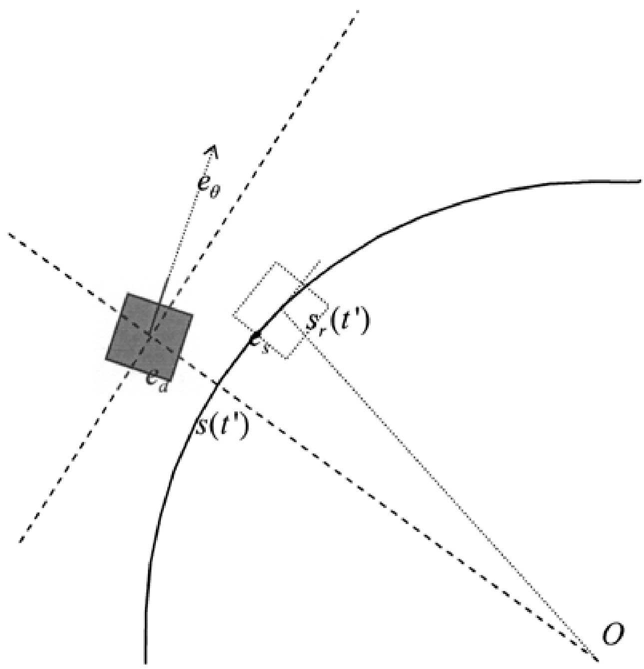



Fig. 11. Three location tracking errors at a time instant $t'$. The solid line is the planned circular arc with center O.

https://doi.org/10.1017/S0263574703005654 Published online by Cambridge University Press

ics of motion (1); the second and third terms are feedbacks to let the robot ride along the reference path with acceptable accuracy. The suggested tracking controllers for left and right wheels are finally designed as

$$\tau_R(t) = \frac{1}{2}(\tilde{\tau}(t) - \tau_\Delta(t))$$

$$\tau_L(t) = \frac{1}{2}(\tilde{\tau}(t) - \tau_\Delta(t)) \qquad (25)$$

The performance of the tracking controllers are validated through simulations in the following section.

## 4. SIMULATIONS

The following simulations are performed by the simulator introduced in Section 2. For this simulation, the simulation tick time is set as $T=0.02$ sec and is equal to the sampling time. All computations about strategy, including ball motion prediction and robot trajectory planning, should be completed in a time duration less than $T$. This imposes a demand of the computational speed in practical realization of the ball passing strategy among multiple mobile robots. The control command at time $kT$ can be computed by all motion data accessed at time $t=(k-1)T$. In the simulation, three mechanically identical mobile robots perform the proposed cyclic ball passing strategy. The robot mass, moment of inertia and length are $m=1$kg, $I=0.02$kg $\cdot$ m$^2$, and $l=0.08$m, respectively. The wheel radius is $r=0.04$m. Considering hardware limits, it is assumed that maximum linear velocity is 1.5m/s, and maximum torque is 0.4N $\cdot$ m for each mobile robot. For a passing movement, no robot holds the ball. The robot kicks the ball always by its front surface, whose orientation ready for kick is orthogonal to the moving direction of robot. The desired kick velocity is 0.7m/s, and the kick direction, is set as $+/-0.25\pi$ angle apart from the desired passing direction (see Figure 8). The two vectors $\vec{v}_1$ and $\vec{v}_2$ for prediction of the kick (as shown in Figure 9) are represented in terms of polar coordinates (2.34, 0) and (3.66, $\pm$0.145) with respect to the passing direction. In this way, as the passing direction rotates, these two vectors and desired kick direction also change accordingly. As a result, an intention of change of ball passing direction is realized without utilizing any more commands.

The initial relative locations of a ball and three mobile robots are shown in Figure 12. For the starting cycle of ball passing robot "John" (robot 1) is the first Passer, "Mary" (robot 2) is the Next Player, and "Rosa" (robot 3), the Previous Player will kick the ball after Mary. Then the ball returns to robot John again and another passing cycle starts. The time duration between subsequent two kicks of the trip to the ball is set to be 1.6 s as the temporal constraint of passing movement. It is noted that a successful ball passing cycle depends on factors such as the relative speed of the soccer robot and ball, the kicking direction etc. Here we show some simulations of successful ball passing. In the first simulation, after first five kicks, the desired passing direction is designed to rotate clock-wise by an angle 0.2 rad. after each kick. In the second simulation, the robots are designed to follow an S-shape course for ball passing among them fluently. Figure 13 depicts the simulation results. Both simulations demonstrate the formation change of the robot team according to the objective of passing a ball

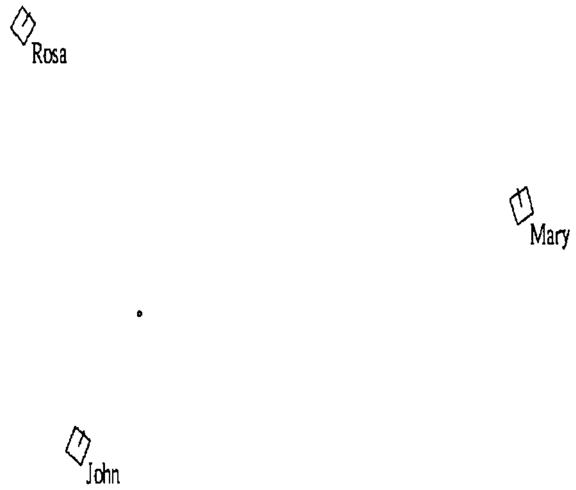
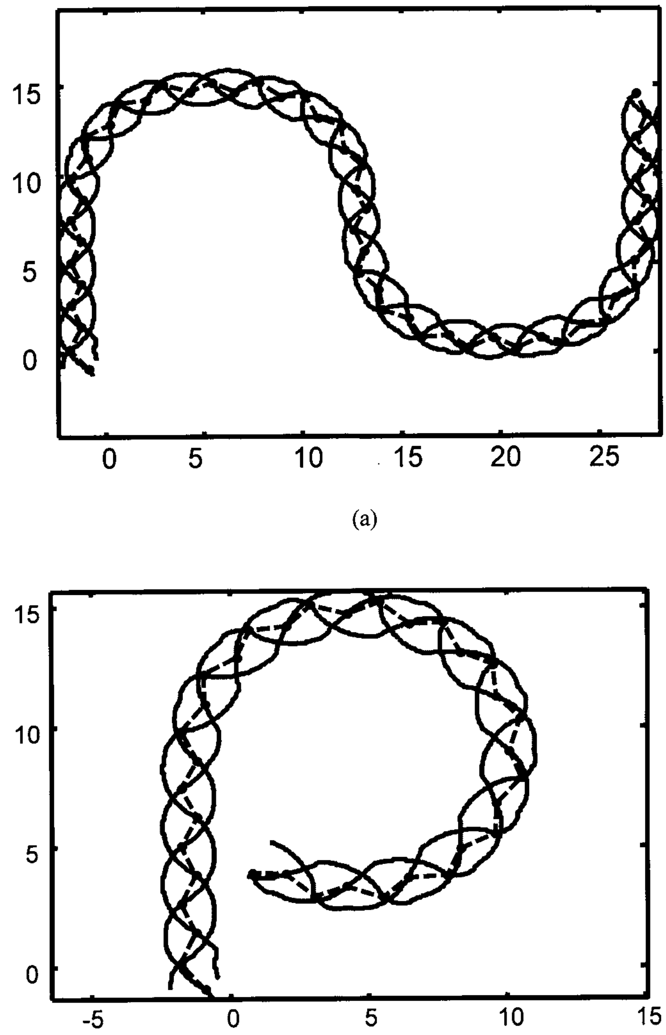



Fig. 12. The initial positions of three robots and the ball.




(a)

(b)

Fig. 13. Traces of 36 kicks for (a)simulation 1 and (b) simulation 2. Solid lines are trajectories of three robots; dashed line is the ball locus, and dots are the positions where a robot kicks the ball.

https://doi.org/10.1017/S0263574703005654 Published online by Cambridge University Press

in turn. Three trajectories of robot motion and the locus of ball are plotted in the $x-y$ plane. The velocity, position and orientation tracking errors of robot "John" in the first simulation are shown in Figure 14. Since the width of robot front plane to hit the ball is 80 mm, small tracking errors do not cause a deviation in the kicking direction with respect to the desired value to miss a ball pass during ball passing movements. Figure 14 depicts a part of robot John's trajectory in the second simulation

A path planning method described in Section 3.2.1 generates these circles, and a partial arc of each circle compose part of a reference trajectory of robot in each passing cycle. As the robot switches the prediction of its moving trajectory twice between its two subsequent kicks, in Figure 15 three circles form a group. It is to be noted that the efficiency of the controller for ball passing, in terms of the distance traveled by the individual robot [11, pp. 337–338], is not the same for an individual robot in a team. In Figure 16, we show a snapshot of the simulator for this simulation.

The simulations show that a soccer robot moves along a preplanned arc to kick the ball. After each kicking, the robot modifies its trajectory to goal position by obtaining the information of the kicking position of the other two robots. Finally, the planned path for each robot is a continuous curve composed of three connected circular arcs, and each of the robots is accurately steered along the path to a location to receive the pass and hit the ball from a specified direction to perform a successful ball passing cycle: pass a ball to the teammate immediately without holding.

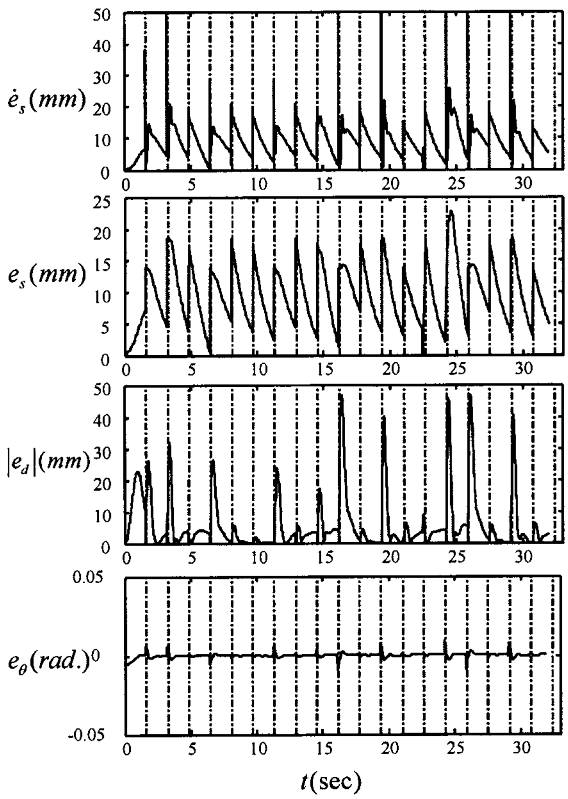



Fig. 14. The error-time plots of robot "John" show the evolution of $\dot{e}_s$, $e_s$, $|e_d|$, $e_\theta$ in first 33 s of simulation.

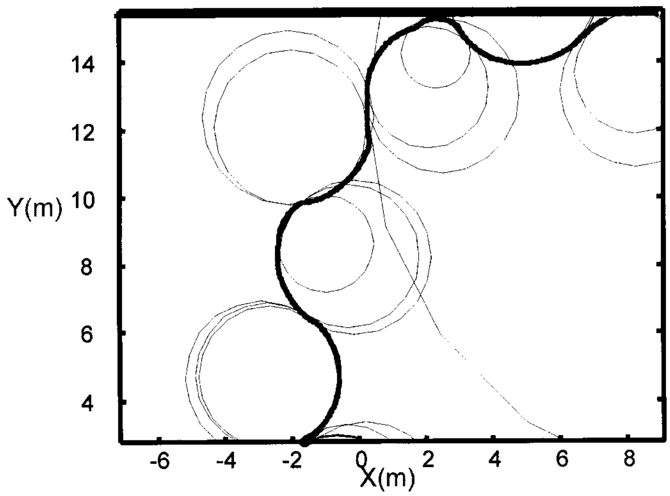



Fig. 15. The planned circles and the robot trajectory. The thick solid line is part of robot John's trajectory. The circles are planned path.

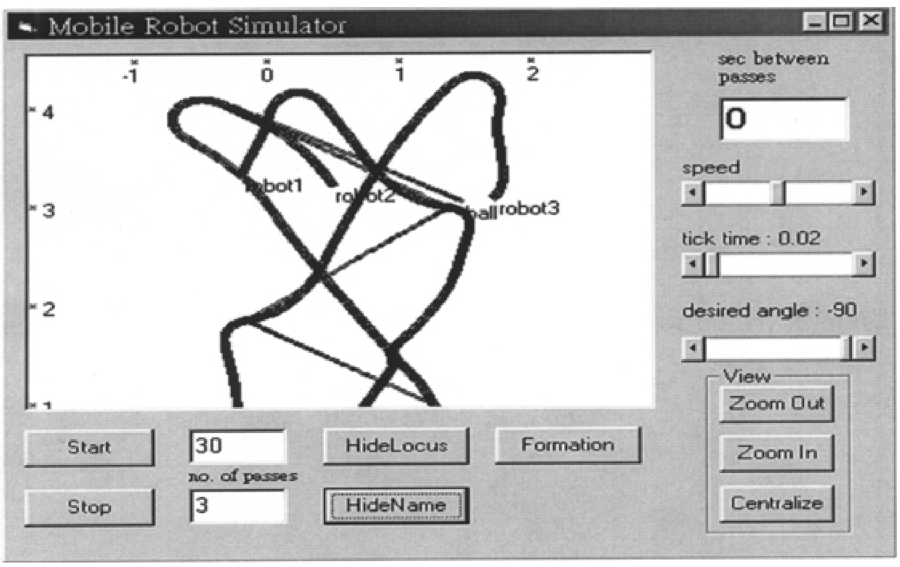



Fig. 16. Snapshot of the simulator running the ball passing strategy among three robots.

https://doi.org/10.1017/S0263574703005654 Published online by Cambridge University Press

## 5. CONCLUSION

In this paper, we built a computer simulator for multiple mobile robots and ball, which incorporates not only kinematics, but also dynamics and control of mobile robots. The simulator serves as a platform for soccer strategy development, path planning and effective tracking controller design of mobile robots, so that the methodology developed in the simulator can be transferred to real physical soccer robots for validation. This expands the functions available in the FIRA's Simurosot, which uses only the kinematics of motion for simulation.

We examined a case study of cyclic ball passing strategy for three mobile robots to pass a ball in a zigzag pattern in turn at a specific direction without holding. For the realization of soccer robots to pass a ball in turn like human players do, a coordinated trajectory planning and formation control problem was formulated and solved for multiple mobile robots. The realization of successful passing cycles is accomplished by a design of coordinated trajectory planning and tracking control with appropriate role assignments and enables dynamic formation change for achieving accurate ball passing. The trajectory of a soccer robot, constituted by circular arcs with a velocity profile as a cubic polynomial, is on-line adjusted by obtaining more accurate information of ball locus after each kick. The simulations demonstrate the practicability and highlight the ingredients of the whole design required for the three levels of hierarchical intelligent control: organization, coordination, and execution.[11] For a larger number of robots in a team, the strategy must be more complicated and its realization is more difficult. Future work is the extension of the experience of collaborative design in this case study to a broader range of strategies and multiple robot systems.

### Acknowledgment

This research was supported by National Science Council of R.O.C. under contract NSC 91–2212-E-001–001.

### References

1. P. Stone, *Layered learning in multiagent systems*: A winning approach to robotic soccer (MIT Press, Cambridge, Massachusetts, 2000).
2. P. Stone, T. Balch and G. Kraetzschmar (eds.), *Robocup 2000: Robot Soccer World Cup IV* (Springer-Verlag, Berlin Heidelberg, 2001).
3. T. Balch and L.E. Parker (ed.), *Robot Teams* (A.K. Peters, Massachusetts, 2002).
4. J. Johnson, Josep Lluis de la Rosa et al. *Benchmark tests of robot soccer ball control skills* (The Federation of International Robot-soccer Association, FIRA, 1999).
5. B.-J. Lee and G.-T. Park (1999). "A robot in intelligent environment: soccer robot", *Procceedings of the 1999 IEEE/ASME International Conference on Advanced Intelligent Mechatronics* (1999), pp. 73–78.
6. B.-J. Lee, S.O. Lee and G.-T. Park, "Trajectory generation and motion tracking control for the robot soccer game", *1999 IEEE/RSJ International Conference on Intelligent Robots and Systems* (1999), pp. 1149–1154.
7. C. Messom, "Robot soccer: sensing, planning, strategy and control, a distributed real time intelligent system approach", *3rd Int. Conf. Artificial Life and Robotics* (1998).
8. S. Buck, M. Beetz and T. Schmitt, "M-ROSE: a multi robot simulation environment for learning cooperative behavior", *Distributed Autonomous Robotic Systems* (Springer-Verlag, Tokyo, Japan, 2002), **Vol. 5**, pp. 197–214.
9. T. Lee, U. Nehmzow, and R. Hubbold, "Computer simulation of learning experiments with autonomous mobile robots", *Proc. TIMR'99 Towards Intelligent Robots* (1999).
10. H. Bingrong, G. Quansheng et al., "Robot soccer simulation competition platform base on multi-agent", *FIRA-KAIST Cup Workshop* (2000).
11. T.L. Dean and M. P. Wellman, *Planning and Control* (Morgan Kaufmann Publishers, San Mateo, CA, 1991).
12. M.L. Corradini and G. Orlando, "Robust tracking control of mobile robots in the presence of uncertainties in the dynamical model", *Journal of Robotic Systems*, **18**, No. 6, 317–323 (2001).
13. K. Komoriya and K. Tanie, "Trajectory design and control of a wheel-type mobile robot using B-spline curve", *IEEE/RSJ International Workshop on Intelligent Robots and Systems'89* (Tsukuba, Japan, 1989), pp. 398–405.
14. T.-C. Liang and J.-S. Liu, "Motion controller realizing cyclic ball passing strategy among multiple mobile robots in robot soccer games", *2002 IEEE International Conference on Robotics and Automation* (Washington DC, 2002), pp. 2587–2592.
15. P. Soueres and J.P. Laumond, "Shortest paths synthesis for a car-like robot", *IEEE Trans. on Automatic Control*, **41**, No. 5, 672–688 (1996).
16. Y. Kanayama and S. Yuta, "Vehicle path specification by a sequence of straight lines", *IEEE Journal of Robotics and Automation*, **4**, No. 3, 265–276 (1988).