

## A Bezier curve based path planning in a multi-agent robot soccer system without violating the acceleration limits

K.G. Jolly<sup>b,\*</sup>, R. Sreerama Kumar<sup>a</sup>, R. Vijayakumar<sup>a</sup>

<sup>a</sup> National Institute of Technology Calicut, Calicut – 673601, Kerala, India<sup>1</sup>

<sup>b</sup> NSS College of Engineering Palakkad, Palakkad – 678008, Kerala, India

### ARTICLE INFO

#### Article history:

Received 14 March 2007

Received in revised form

29 February 2008

Accepted 26 March 2008

Available online 6 April 2008

#### Keywords:

Bezier curves

Path planning

Acceleration limits

Velocity profile

Obstacle avoidance

### ABSTRACT

This paper proposes an efficient, Bezier curve based approach for the path planning of a mobile robot in a multi-agent robot soccer system. The boundary conditions required for defining the Bezier curve are compatible with the estimated initial state of the robot and the ball. The velocity of the robot along the path is varied continuously to its maximum allowable levels by keeping its acceleration within the safe limits. An obstacle avoidance scheme is incorporated for dealing with the stationary and moving obstacles. When the robot is approaching a moving obstacle in the field, it is decelerated and deviated to another Bezier path leading to the estimated target position. The radius of curvature of the path at its end points is determined from the known terminal velocity constraint of the robot.

© 2008 Elsevier B.V. All rights reserved.

## 1. Introduction

Robot soccer is a dynamic and competitive system in which each robot has to move along a prescribed path for fulfilling its goal oriented tasks. Because of the competitive nature of the system, path planning of soccer robots plays a key role in winning the game. A robot soccer system consists of a number of moving objects consisting of opponent robots and the teammates. The walls of the playground are taken as the stationary obstacle. Due to the inaccuracies and uncertainties associated with the pose estimation and also due to the dynamic nature of the environment, robot path planning is a challenging task. Since the system configurations are changing from time to time, a static path planning scheme is not advisable. Different approaches such as potential field methods [26,27], limit cycle navigation methods [11], probabilistic roadmap techniques [18, 21,29], graph generation schemes [36] and artificial coordinating field methods [28] have been proposed in the recent past for the path planning of mobile robots. Various types of geometric curves are adapted for generating a robot path [7,12,13,15,23]. Problems

associated with the path planning of the mobile robots working in indoor [11,18] and outdoor environments [36] are available. Path planning of the robot working under static [5,27], dynamic [6, 9] or partially known [17,33] environmental conditions are also available. The collision avoidance schemes and planning of the collision free path were studied by a number of researchers [4, 7,19,20,28,30,34]. The omni-directional robots [14,18] are found to have superior manoeuvring capability compared to the more common nonholonomic car-like vehicles. Artificial neural network (ANN) [25], genetic algorithm [35] and fuzzy logic [3] based techniques are also attempted for path planning of the mobile robots.

The paper is organized as follows: The path planning problem of a robot in a robot soccer environment is discussed in Section 2. Section 3 presents an overview of the Bezier curve synthesis and analysis with the determination of the acceleration components. Limitations in the velocity and accelerations of the robot along the curved path are given in Section 4; Section 5 describes the proposed curve optimization techniques. Criteria for avoiding the static and dynamic obstacles are presented in Section 6. Discussion based on the simulation results and the comparisons with an established work are included in the last section.

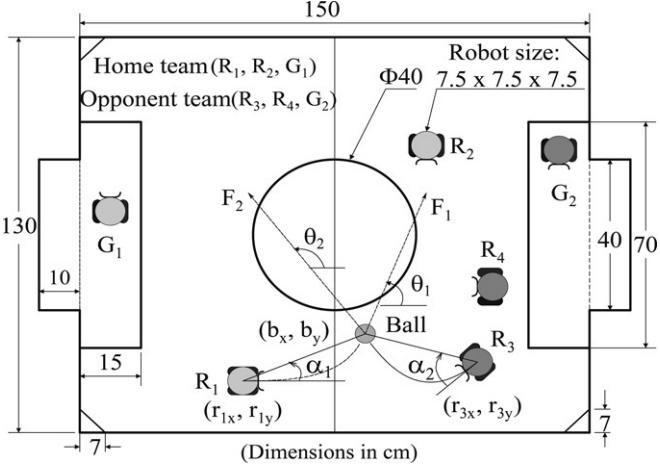
## 2. Problem formulation

This paper proposes a path planning scheme for a mobile robot in MiroSot small league system using Bezier curves. As shown in Fig. 1, MiroSot small league is a robot soccer game played between

\* Corresponding author at: NSS College of Engineering Palakkad, Palakkad – 678008, Kerala, India. Tel.: +91 491 2524921 (residence), 2555255 (office); fax: +91 491 2555900.

E-mail addresses: kg\_jolly@hotmail.com (K.G. Jolly), sreeram@nitc.ac.in (R. Sreerama Kumar), vijay@nitc.ac.in (R. Vijayakumar).

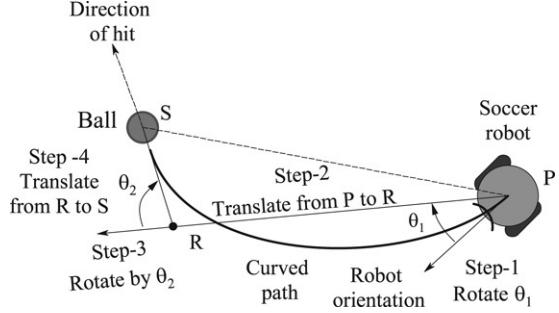
<sup>1</sup> www.nitc.ac.in.



**Fig. 1.** Schematic representation of a MiroSot small league system.

two teams each consisting of three players and an orange coloured golf ball. One of the three robots is the goalkeeper. At any instant during the game, the robots of a team have an arbitrary position and orientation with respect to the ball. Similarly to the case of human players in which the one who is nearer to the ball moves to the ball, the robot which is nearer to the ball should approach the ball spontaneously. For the decision making and coordination among the robots of the team, an ANN based method has been developed [10]. In Fig. 1, the position and orientation of the nearest robots of the teams are shown along with the location of the ball. Orientation of the robots is represented relative to the line joining the robot centre and the ball centre. Probable paths for the nearest robots to approach the ball position are also indicated in the figure. As shown in the figure, the geometry of the paths depends on the parameters such as position and orientation of the robot ( $r_{1x}, r_{1y}, \alpha_1$ ), the position of the ball ( $b_x, b_y$ ), and the action of the robot on the ball ( $F_1, \theta_1$ ). The geometry of the path is also influenced by the presence of any other moving or stationary robots in the region. The robot which reaches near the ball first, kicks the ball to the desired location depending upon an action selection mechanism. The action selection mechanism [8,16,22,31] is based on the game strategy which establishes the cooperation among the members of the team. Fig. 1 shows that robot  $R_1$  of the home team tries to pass the ball to its team-mate  $R_2$  positioned near the opponents' goal area. At the same time the nearest robot  $R_3$  of the opposing team tries to kick the ball to the opposite side by advancing towards the goal area of the home team. The required accuracy of shooting action is attained by means of a path planning mechanism. Since the robot soccer environment consists of other moving robots, the path planning must be collision free. In a dynamic environment, collision chances are reduced by making the path planning dynamic in nature. A dynamic path planner modifies the planned path at each sampling time if an obstacle comes closer to the robot. In order to develop an effective path planning scheme in a robot soccer environment it is necessary to make certain assumptions as follows:

- The robot soccer system consists of remote brainless robots and they are subjected to nonholonomic constraints.
- The pose of the various robots in the environment is available through a pose estimation algorithm within an allowable tolerance.
- The robots have longitudinal and transverse axes of symmetry and the centre of gravity is located at the intersection of the planes containing these axes of symmetry.
- The axes of the driving wheels are located along the plane of transverse axis of symmetry.



**Fig. 2.** Analysis of the motion of the robot from P to S.

The motion of robot  $R_1$  or  $R_3$  along the curved path towards the ball, shown in Fig. 1, can be decomposed into four discrete steps as shown in Fig. 2. In the first step, the robot has to rotate so that its angular orientation is directed to the point  $R$  behind the ball along the direction of hit. In the second step, the robot has to translate towards the point  $R$ . The orientation of the robot is rearranged along the direction of hit through a rotation in the third step. In the fourth step, the robot moves along the direction of hit and acquires sufficient velocity to kick the ball towards the destination. In this analysis, it is assumed that the position and orientation of the robots in a multi-agent robot soccer system at any sampling time are estimated with reasonable accuracy by processing the image of the play field captured through a video camera [32]. The motion of the robot implemented through such discrete translation and rotation is discontinuous and time consuming. The path of the robot is hence modified to a curve starting from the point  $P$  along the direction  $\theta_1$  and ends up at point  $S$  in the direction  $\theta_2$ . Along this curved path a robot experiences a continuous motion through the simultaneous rotation and translation. The path of the robot obtained in this manner is a plane curve and since rotations and translations are combined, the travel time is reduced compared to that for the discrete motions. Various geometric curves like cubic splines [12], polynomials [15], blends of polynomials and straight lines [13], have been attempted for making such a continuous path for soccer robots.

Once the initial state of the nearest robot  $R_1$  or  $R_3$  and the position of the ball of a multi-agent robot soccer system are estimated, the force of hit and the direction of hit can be decided through an action selection module [31]. Using these initial conditions, various geometric curves can be considered for the path planning. However due to the compatibility of the estimated position and orientation of the nearest robot, position of the ball, the force of hit and direction of hit, a Bezier curve [1,2] based path planning scheme is a suitable solution. The accuracy of the Bezier path is subjected to the accuracy of the pose estimation algorithm and no other assumptions or approximations are required for the Bezier curve based path planning. Bezier curve has a more convenient obstacle avoidance characteristic than the interpolation techniques. This is because a Bezier curve passes through the first and last control points. Therefore the path planner has the full control over the shape of the curve in a predictable way by changing a few simple parameters. However, most of the curve-defining techniques presently used in robot path planning involve the interpolation of a given set of points and the curve so produced passes through all the control points.

### 3. Path planning using Bezier curves

Bezier curve is a space curve, which is credited to Pierre Bézier of the French car firm Regie Renault. Unlike other type of curves like polynomials or cubic splines, Bezier curve does not pass through all the data points used to define it. The points that are

used to define a Bezier curve are called control points. A polygon that can be drawn through these control points is known as Bezier polygon. Bezier curve is contained within the convex hull of the defining polygon. The turning points are the points where the slope of the curve changes its sign. Bezier curves have fewer turning points so that it is smoother than cubic splines. The first and the last points on the curve are coincident with the first and last control points. The tangent vectors at the end points of the Bezier curve are directed along the first and last span of the polygon. The radius of curvature of the Bezier curve varies smoothly from the starting point to the end point because of its continuous higher order derivatives. Paths of the various robots in a multi-agent robot soccer system for any field configurations can be generated using this curve. A parametric Bezier curve is defined as

$$P(u) = \sum_{i=0}^n B_{n,i}(u), \quad 0 \leq u \leq 1 \quad (1)$$

where  $u$  is a parameter,  $n$  is the degree of Bernstein basis or blending function,  $i$  is the summation index and  $B_i$  represents the  $i$ th vertex of the Bezier polygon. For defining an  $n$ th degree Bezier curve a polygon with  $(n+1)$  vertices are required. The  $i$ th Bernstein function in Eq. (1) is given as

$$J_{n,i}(u) = {}^n C_i u^i (1-u)^{n-i} \quad (2)$$

where  $u^i (1-u)^{n-i}$  is the  $n$ th degree blending function and the coefficient  ${}^n C_i = \frac{n!}{(n-i)!i!}$ . A third degree Bezier curve can be defined through four control points  $(A_0, B_0)$ ,  $(A_1, B_1)$ ,  $(A_2, B_2)$  and  $(A_3, B_3)$  as

$$\begin{aligned} x(u) &= \sum_{i=0}^3 A_i J_{3,i}(u) \\ &= A_0(1-u)^3 + 3A_1u(1-u)^2 + 3A_2u^2(1-u) + A_3u^3 \end{aligned} \quad (3a)$$

$$\begin{aligned} y(u) &= \sum_{i=0}^3 B_i J_{3,i}(u) \\ &= B_0(1-u)^3 + 3B_1u(1-u)^2 + 3B_2u^2(1-u) + B_3u^3. \end{aligned} \quad (3b)$$

Eqs. (3a) and (3b) are expanded and rearranged to the form of a third order polynomials in  $u$  as

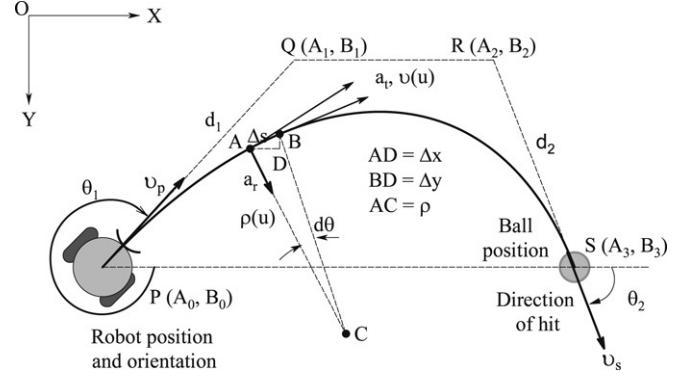
$$x(u) = a_0 + a_1u + a_2u^2 + a_3u^3 \quad (4a)$$

$$y(u) = b_0 + b_1u + b_2u^2 + b_3u^3 \quad (4b)$$

where  $a_0, a_1, a_2, a_3$  and  $b_0, b_1, b_2, b_3$  are the coefficients evaluated in terms of the control points  $(A_0, B_0)$ ,  $(A_1, B_1)$ ,  $(A_2, B_2)$  and  $(A_3, B_3)$ . The curvature  $K(u)$  at any point on the Bezier curve is expressed in terms of the first and second derivatives  $x'(u)$ ,  $x''(u)$ ,  $y'(u)$  and  $y''(u)$  with respect to the parameter  $u$  as

$$K(u) = \frac{1}{\rho(u)} = \frac{x'(u)y''(u) - y'(u)x''(u)}{\left(x'(u)^2 + y'(u)^2\right)^{3/2}}. \quad (5)$$

The Bezier paths of the robots  $R_1$  and  $R_3$  of a multi-agent robot soccer system is shown in Fig. 1. In this figure, the estimated position of the robot is taken as the first control point and the estimated ball position is taken as the fourth control point. The second control point  $Q$  is located using the estimated angular orientation of the robot  $\theta_1$  and the length of the first side of the defining Bezier polygon  $d_1$  as shown in Fig. 3. The third control point  $R$  is calculated by knowing the selected action of the robot on the ball and the length of the third side of the Bezier polygon  $d_2$  as shown in Fig. 3. The various actions of a robot such as dribbling, passing, kicking and shooting the ball depend upon the force of hit and direction of hit. Immediately after the selection of the nearest robot [10], the appropriate action of that robot is also selected by an action selection mechanism [16,31] and it



**Fig. 3.** Path of the nearest robot in a MAS described through a Bezier curve.

will be made available for the path planning module. Through the decision making module, action selection module and path planning module, coordination, cooperation, and intelligence is imparted to a multi-agent robot soccer system. The required final velocity of the robot depends on the force of hit. The required final velocity and the estimated initial velocity of the robot are used to optimize Bezier polygon sides  $d_1$  and  $d_2$ . Thus a smooth curve can be defined using the control point  $P$ ,  $Q$ ,  $R$  and  $S$  as shown in Fig. 3. Note that the Bezier curve does not pass through the intermediate points  $Q$  and  $R$  and at the same time it passes through the terminal points  $P$  and  $S$ . The Bezier curve is optimized by optimizing the sides  $d_1$  and  $d_2$  of the polygon with respect to the terminal velocities. Similarly to other curve fitting techniques, the accuracy of the Bezier path depends on the accuracy of the pose estimation algorithm.

For an estimated robot position  $P$  and ball position  $S$ , a number of Bezier paths can be easily obtained by changing the lengths of the polygon sides  $d_1$  and  $d_2$ . This promising characteristic of the Bezier curve is exploited for collision avoidance. The length of the Bezier path varies directly with  $d_1$  and  $d_2$  so that the path length is increased by increasing the lengths of the polygon sides  $d_1$  and  $d_2$ . The radius of curvature at the initial point  $P$  and the target point  $S$  is also increased by increasing  $d_1$  and  $d_2$ . Thus the lengths of the polygon sides  $d_1$  and  $d_2$  are the useful parameters for optimizing the Bezier path with respect to different objectives and this flexibility is not directly available with cubic splines. In Fig. 3, OXY is the reference frame and the orientation angles are measured clockwise as shown. The estimated initial velocity of the robot at  $P$  is  $v_p$  and the required final velocity of the robot decided by the action selection mechanism is  $v_s$ . The coordinates of the control points  $Q$  and  $R$  are calculated as

$$A_1 = A_0 + d_1 \cos \theta_1 \quad (6a)$$

$$B_1 = B_0 + d_1 \sin \theta_1 \quad (6b)$$

$$A_2 = A_3 + d_2 \cos(180 + \theta_2) \quad (7a)$$

$$B_2 = B_3 + d_2 \sin(180 + \theta_2). \quad (7b)$$

The parametric form of the Bezier path defined through the four control points  $P$ ,  $Q$ ,  $R$  and  $S$  is given in Eqs. (3) and (4). While moving along the Bezier path for the position of the robot at  $A$ , let  $\rho$  is the radius of curvature,  $a_t$  the tangential acceleration and  $a_r$  the radial acceleration. After an infinitesimal time  $\Delta t$ , the robot is moved to the neighbouring point  $B$  and  $\Delta s$  is the corresponding displacement along the curved path. The average velocity of the robot during this period is represented by  $v(u)$  so that the infinitesimal displacement  $\Delta s$  of the robot is expressed as

$$\Delta s(u) = v(u)\Delta t. \quad (8)$$

The average velocity  $v(u)$  of the robot is expressed as

$$v(u) = \sqrt{\left(\frac{\Delta x(u)}{\Delta t}\right)^2 + \left(\frac{\Delta y(u)}{\Delta t}\right)^2} \quad (9a)$$

$$v(u) = \frac{1}{\Delta t} \sqrt{(\Delta x(u))^2 + (\Delta y(u))^2}. \quad (9b)$$

As  $\Delta t \rightarrow 0$ , the point B is closer to point A, so that  $\Delta x(u) \rightarrow dx(u)$ ,  $\Delta y(u) \rightarrow dy(u)$  and  $\Delta s(u) \rightarrow ds(u)$ , the average velocity of the robot  $v(u)$  approaches its instantaneous velocity at point A. Substituting Eq. (9b) in Eq. (8), multiplying and dividing by  $du$ , the displacement  $ds$  is obtained as

$$ds(u) = \sqrt{\left(\frac{dx(u)}{du}\right)^2 + \left(\frac{dy(u)}{du}\right)^2} du. \quad (10)$$

For a differential displacement  $ds(u)$ , the travel time of the robot from position A to position B along the path is obtained from Eq. (9b) as

$$dt = \frac{ds(u)}{v(u)}. \quad (11)$$

The time required for the robot to travel from its initial position P to the target position S along the Bezier curve is calculated by integrating Eq. (11). For the integration of Eq. (11), velocity profile of the robot is to be expressed as a function of parameter  $u$ . The travelling time of the robot from point P to point S is expressed as

$$t = \int_0^1 \frac{ds(u)}{v(u)}. \quad (12)$$

For determining the minimum travel time of the robot along the Bezier curve, Eq. (12) is to be integrated with maximum possible instantaneous velocity. For performing the integration the maximum possible instantaneous velocity is to be expressed as a function of the parameter  $u$ . The velocity of the robot along the path is limited by the radial and tangential acceleration limits of the robot. These acceleration components are calculated as

$$a_r = \frac{v^2(u)}{\rho} \quad (13)$$

$$a_t = \frac{dv(u)}{dt}. \quad (14)$$

The angular orientation of the robot at any point along the Bezier path is calculated from the slope of the curve at that point. Thus the change in angular orientation  $d\theta$  and the orientation angular velocity  $\omega$  are calculated as

$$d\theta(u) = \frac{y'(u)}{x'(u)} \quad (15)$$

$$\omega(u) = \frac{d\theta(u)}{dt}. \quad (16)$$

$v(u)$  and  $\omega(u)$  are used as the reference velocities for tracking the robot along the planned path.

#### 4. Limits on velocity and acceleration

As shown in Fig. 4, the schematic representation of a nonholonomic soccer robot, the body of the soccer robot is symmetric and the centre of mass C coincides with its geometric centre. The two driving wheels are powered by separate DC motors and configured in such a way that the wheel axis passes through the mass centre C [24]. In some cases, a passive self-adjusting wheel is fitted at the front end of the robot for keeping the stability of the system during the motion. A MiroSot robot is cubical in shape and its overall size is limited to 7.5 cm × 7.5 cm × 7.5 cm, without

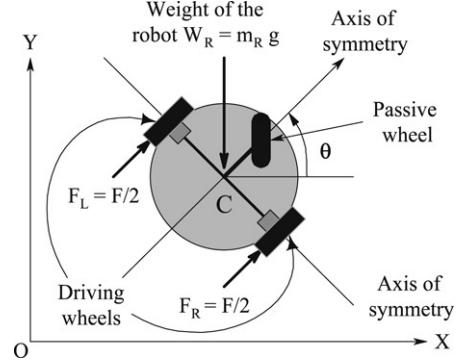


Fig. 4. Schematic diagram of a nonholonomic mobile robot.

exceeding a weight of 600 g. The angular orientation of a robot changes continuously during its motion along a curved trajectory. During the motion, the tangential acceleration of the robot is responsible for the change in its velocity. The radial acceleration of the robot is due to the curvature of the path, and as the curvature of the path increases, its radial acceleration also increases. Thus for keeping the total acceleration of the robot within the limits it becomes necessary to keep the tangential acceleration smaller. The maximum possible acceleration of a mobile robot is limited by the coefficient of friction between the wheel and the ground. Fig. 4 shows the forces acting on the wheels of the mobile robot in the direction of motion. The weight of the robot is acting downward along the mass centre C, due to which the wheels are pressed on to the ground, and consequently they receive an equivalent normal reaction. Because of the coefficient of friction, these normal reactions introduce horizontal frictional forces  $F_R$  on the right wheel and  $F_L$  on the left wheel as shown in the figure, and which are expressed as

$$F_R = F_L = \frac{\mu m_R g}{2} \quad (17a)$$

where  $m_R$  is the total mass of the robot and  $\mu$  is the coefficient of friction between the wheel and the ground. The effective frictional force acting on a mobile robot in the direction of motion is expressed as

$$F = F_R + F_L = \mu m_R g. \quad (17b)$$

The equations of motion of the robot becomes

$$F = \mu m_R g = m_R a_{Max}.$$

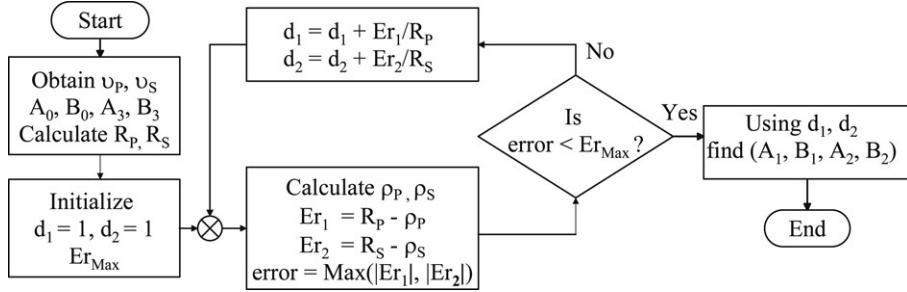
The maximum acceleration of the robot is limited to

$$a_{Max} = \mu g. \quad (17c)$$

Eq. (17c) gives the maximum possible acceleration of the soccer robot. The centre of gravity of the robot is located at some distance above the ground level. Therefore during the period of deceleration, moment of the inertial force causes the passive wheel to press more on to the ground, and consequently actual contact force between the driving wheels and the ground is lowered. This causes a further reduction in the maximum possible acceleration of the robot. Hence the limit on the tangential acceleration is lower than the limit on the radial acceleration. The maximum acceleration is the resultant of the maximum tangential and radial accelerations, given by

$$a_{Max} = \left(a_{rMax}^2 + a_{tMax}^2\right)^{1/2} = \mu g \quad (18)$$

where  $a_{tMax}$  and  $a_{rMax}$  are the maximum allowable acceleration components of the robot. The distance  $\Delta s$  between the points A and B and the travel time  $\Delta t$  are obtained respectively through the



**Fig. 5.** Flowchart of the iterative algorithm that determines  $d_1$  and  $d_2$ .

numerical integration of Eqs. (8) and (12) using Simpson's rule. For integrating Eq. (12), the maximum possible velocity profile  $v(u)$  of the robot along the path is to be evaluated as a function of the parameter  $u$ . Since the limit on the tangential acceleration is lower than the limit on radial acceleration, the relation between the tangential and radial accelerations [12] is represented as

$$\frac{a_t^2}{a_{t\text{Max}}^2} + \frac{a_r^2}{a_{r\text{Max}}^2} = 1. \quad (19)$$

For a soccer robot, the reasonable values for the limits on tangential and radial accelerations are selected from Reference [12] as  $a_{t\text{Max}} = 2.0 \text{ m/s}^2$  and  $a_{r\text{Max}} = 4.0 \text{ m/s}^2$ . If the acceleration of the robot is greater than the limits, continuous slipping or skidding of the robot occurs. Consequently actual velocity of the robot differs significantly from the reference velocities. The maximum possible velocity of the robot along a curved path is limited by the curvature of the path as indicated by Eq. (13). The velocity of the robot is to be restricted for keeping the accelerations within limits while moving it along a curved path. At each and every sampling point on the curved path, it is possible to determine the maximum velocity of the robot without exceeding the acceleration limits. The velocity of the robot so calculated from the starting point to the end point along the curved path is known as the maximum allowable velocity profile of the robot. The maximum velocity profile depends on the curvature of the path. If the radial acceleration of a point on the path is lower than the radial acceleration limit, then the corresponding velocity of the robot is realizable. In order to satisfy the radial acceleration limits these calculations are started from any one of the points such as initial point, end point, or turning points. Corresponding to these values of radial acceleration, the maximum possible tangential acceleration is determined using Eq. (19), with which the velocity at the next sampling point can be evaluated assuming that the robot is accelerated uniformly between the points, and this procedure is continued till the end point.

## 5. Determination of optimal path

During the tracking of a robot along the planned path, it is necessary that the velocity at each and every point is maximized without exceeding the acceleration limit so as to avoid sliding or skidding of the robot. Marko Lepeti et al. [12] have proposed a path planning scheme based on the definition of a spline curve through a number of control points. Various paths can be defined between the given initial and final positions by varying the number of control points used for the interpolation. The path has been optimized with respect to the travel time by varying the number of control points used for the interpolation. The analysis of the results obtained indicates that the path is optimum when the number of control points used to define the curve is four. No considerable decrease in the travelling time is observed when the number of control points is increased beyond four. The Bezier curve based path planning scheme proposed in this paper also uses four control

points for the curve design. Moreover the Bezier curve represented in Eq. (4) is similar to a cubic polynomial. Considering these points in view of Lepeti's work, a Bezier curve defined using four control points provides the optimal path.

Fig. 3 shows a Bezier curve in which the first and last control points are indicated as the robot position P, and the ball position S. At the estimated robot position P, orientation angle and estimated initial velocity of the robot are shown. Similarly at the estimated ball position S, the direction of hit and the velocity of hit are shown. Let  $R_p$  be the minimum radius of curvature required at P, so that the robot can start to move with its initial velocity without exceeding the radial acceleration limit. In a similar manner  $R_s$  is the minimum radius of curvature required at S, so that the robot can attain the velocity of hit without violating the radial acceleration limit. The radii of curvatures  $R_p$  and  $R_s$  are evaluated through Eq. (13) using the estimated terminal velocities. The required lengths of the terminal radii of curvature  $R_p$  and  $R_s$  depend on the length of the first side  $d_1$  and the third side  $d_2$  of the Bezier polygon. The length of the polygons  $d_1$  and  $d_2$  can be optimized using several criteria such as minimum path length, obstacle free path, satisfying the terminal velocities etc. In this work  $d_1$  and  $d_2$  are optimized by satisfying the terminal velocity constraint of the robot. Equations for the actual terminal radius of curvature  $\rho_p$  at the initial position P and  $\rho_s$  at the target position S are derived in terms of  $d_1$  and  $d_2$  using Eqs. (3) and (5) as

$$\rho_p = \frac{3d_1^2}{(h_1 + d_2 g_1)} \quad (20a)$$

$$\rho_s = \frac{3d_2^2}{(h_2 + d_1 g_2)} \quad (20b)$$

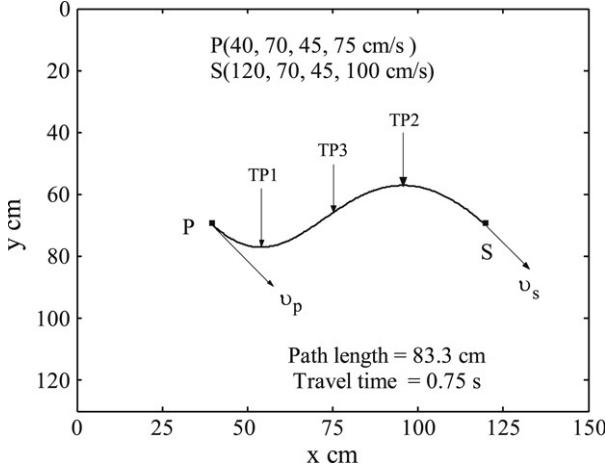
where

$$g_0 = A_3 - A_0, h_0 = B_3 - B_0$$

$$g_1 = 2 \sin(\theta_2 - \theta_1), \quad h_1 = 2(h_0 \cos \theta_1 - g_0 \sin \theta_1)$$

$$g_2 = 2 \sin(\theta_2 - \theta_1), \quad h_2 = 2(h_0 \cos \theta_2 - g_0 \sin \theta_2).$$

The algorithm for finding the optimum values of  $d_1$  and  $d_2$  by iteratively solving Eqs. (20a) and (20b) is shown in Fig. 5. According to this iterative algorithm,  $d_1$  and  $d_2$  are initialized with a small number, say one. The actual radii of curvatures are calculated as  $\rho_p$  and  $\rho_s$  and the corresponding differences  $R_p - \rho_p$  and  $R_s - \rho_s$  between the actual and required curvatures are taken as errors. In each iteration,  $d_1$  and  $d_2$  are updated with a value proportional to these errors until  $\rho_p$  and  $\rho_s$  are sufficiently close to  $R_p$  and  $R_s$ . Once the convergence is achieved the velocity constraints at P and S are satisfied and the corresponding values of  $d_1$  and  $d_2$  provide the optimum length of the first side  $d_1$  and the third side  $d_2$  of the Bezier polygon. When compared to all the other Bezier curves that can be generated through the estimated robot position P and ball position S, the one generated with optimum  $d_1$  and  $d_2$  satisfies the terminal velocity constraints and give minimum path length. Now the other control points at Q and R are evaluated using Eqs. (6)

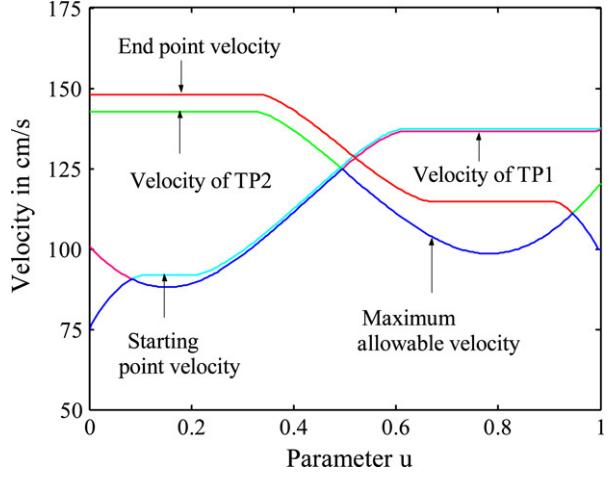


**Fig. 6.** An optimum Bezier path connecting points P and S.

and (7). Fig. 6 shows the optimum Bezier curve generated through this algorithm. The travel time of the robot along the Bezier path is determined by the numerical integration of Eq. (12) for which the maximum allowable velocity of the robot is to be expressed as a function of the parameter u. The maximum allowable velocity profile provides the maximum possible velocity of the robot along the Bezier path without violating the acceleration limits. The various velocity profiles obtained for the Bezier path shown in Fig. 6 are shown in Fig. 7. The sequence of major computations for the determination of the maximum possible velocity profile along the optimum Bezier path is as follows:

- Define a Bezier path through the given control points and calculate a finite number of closely spaced points on the curve using Eq. (4).
- Calculate the radius of curvature and slope of the curve at these points using Eqs. (5) and (15).
- Locate the turning points on the curve using Eq. (15) and applying the fact that the radius of curvature attains minimum values at the turning points.
- Determine the maximum allowable velocity of the robot at the turning points considering the limit on radial acceleration using Eq. (13).
- Velocity of the robot along the curve is changed due to tangential acceleration. Calculate velocity profiles corresponding to the initial velocity, end velocity, and velocities at the turning point considering the uniform acceleration of the robot.
- Calculate the initial velocity profile in the forward direction from the starting point P along the Bezier curve.
- Calculate the end velocity profile in the backward direction from the end point S along the Bezier curve.
- For each turning point, the first portion of the turning point velocity profile is calculated in the backward direction from the turning point to the initial point P, and its second portion in the forward direction from the turning point to the end point S.
- At each point on the path determine the minimum velocity from the various velocity profiles calculated. Thus minimum velocity determined along the path from P to S gives the maximum velocity profile without violating the acceleration limits.

The various velocity profiles obtained for the Bezier path can be visible in Fig. 7. Because of the large radius of curvature of the path at TP3, the maximum allowable velocity profile is higher than that of the maximum velocity of the robot and hence the velocity profile corresponding to the turning point-3 (TP3) is not included in Fig. 7. The maximum allowable velocity profile without violating the acceleration limits is shown in this figure. The flat portions of the individual velocity profiles indicate the regions where the acceleration limits are violated. The maximum allowable velocity profile does not have any flat portions because it does not violate the acceleration limit.



**Fig. 7.** Velocity profiles of the Bezier curve from P to S.

## 6. Obstacle avoidance

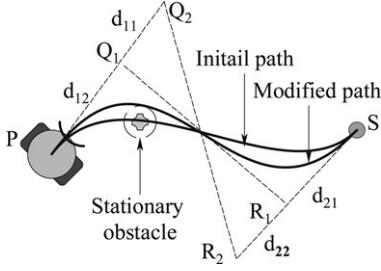
In a robot soccer system, the position and angular orientation of the robots and the ball are changing time to time. Due to the competitive nature of the environment, the robot of the home team tries to pass the ball to its team-mates or kick it into the opponents' goal post. At the same time, the players of the opposing team try to block the ball or intercept it according to their game strategy. Due to this competitive nature of the robot soccer environment, the probability of collision chances is very high. These collisions are between the robots of the same team, between the robots of the opposite teams, or between the robots and the field boundary. These collisions are to be minimized through the dynamic path planning scheme. In a dynamic path planning scheme, a robot does not stick to an initially planned path. Instead, the robot detects the collision chances in advance and makes necessary changes in its path. A certain level of intelligence may help the robots to foresee collision chances. Such capabilities are to be imparted to the path planning module to assist the robots in moving along a collision free path.

Different schemes proposed for avoiding collision with the stationary obstacles in the field are shown in Figs. 8a and 8b. Fig. 8a describes the scheme for avoiding collision with a stationary robot in the field. At the time of planning a path, if it is found to intersect with a stationary robot in the field, the path is re-planned immediately by increasing  $d_{11}$  to  $d_{12}$  and  $d_{21}$  to  $d_{22}$ . The new Bezier path has a larger terminal radius of curvature and thus it avoids the stationary obstacle that has come across the initial path. Fig. 8b shows the scheme for avoiding the collision of a robot with the field boundary. In this scheme, while moving along the path, robot is stopped at a safe distance from the field boundary. Then the body of the robot is rotated about the wheel through a suitable angle  $\theta$  which changes its angular orientation away from the field boundary. The angle  $\theta$  is calculated by defining a safety margin  $r_m$  between the Bezier path and the field boundary as

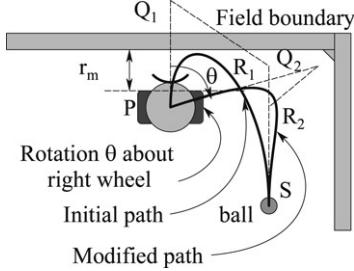
$$r_m = r_R + \delta r \quad (21a)$$

where  $r_R$  is the minimum radius that can encircle the robot and  $\delta$  is a safety factor. The newly generated Bezier path with a modified angular orientation of the robot avoids the collision with the field boundary.

In the dynamic obstacle avoidance scheme, whenever a robot approaches a moving obstacle, it starts to decelerate. This is essential for keeping the radial acceleration of the robot within the limit while taking a deviation to a modified path near the obstacle. The robot continues to travel along the same path till the distance



**Fig. 8a.** Avoiding collision with a static object by changing the polygon sides  $d_{11}$  and  $d_{21}$ .



**Fig. 8b.** Avoiding collision with field boundary through a rotation.

between the obstacle and the robot is reduced to a safety margin  $r_o$  defined by

$$r_o = r_{ob} + r_R + \delta \quad (21b)$$

where  $r_{ob}$  is the minimum radius that can encircle the obstacle. Thereafter the robot takes a slow deviation through the boundary of the safety margin and approaches the target in a different path. The various steps involved in this path planning technique are shown in Fig. 9. It is imagined that the obstacle is encircled by an area known as the safety margin. The collision between the robots occurs if the centre of the robot happens to cross this safety margin. As shown in Fig. 9, near the point A, the robot takes a deviation for avoiding the collision with an obstacle which is currently located at O. The first phase of the deviated path is described by a Bezier path using the polygon ABDC. The side AB of the polygon ABDC is taken along the tangent to the initial path at A and the side CD is taken along a line parallel to the tangent at A and passing through the point C. The point C is the intersecting point between the circumference of the safety margin and the normal drawn to the tangent at A through the centre of the obstacle. Point C is evaluated using the slope  $\theta$  of the initial path at A as

$$x_C = x_0 - r_o \sin \theta \quad (22a)$$

$$y_C = y_0 + r_o \cos \theta. \quad (22b)$$

The path planning algorithm decides automatically whether to take a left turn or a right turn depending upon the slope of the path at A. The direction in which the obstacles tend to cross the initial path is also a criterion for deciding the turning direction. In this work, the turning direction is decided by the slope of the initial path at A and Eqs. (22) are defined for the right turn. The equations of the coordinate of the point C for the left turn can similarly be calculated as

$$x_C = x_0 + r_o \sin \theta \quad (23a)$$

$$y_C = y_0 - r_o \cos \theta. \quad (23b)$$

The coordinates of the points B and D of the Bezier polygon ABDC for the first phase of the deviated path is evaluated by considering the polygon sides  $d_{11}$  and  $d_{12}$ . The polygon sides  $d_{11}$  and  $d_{12}$  are calculated through the continuity in the velocity of the

robot at A and a maximum radial acceleration at C. Hence the points B and D are evaluated using the values of  $d_{11}$  and  $d_{12}$  as

$$x_B = x_A + d_{11} \cos \theta \quad (24a)$$

$$y_B = y_A + d_{11} \sin \theta \quad (24b)$$

$$x_D = x_C - d_{12} \cos \theta \quad (25a)$$

$$y_D = y_C - d_{12} \sin \theta. \quad (25b)$$

The second phase of the deviated path is defined using the Bezier polygon CEFS. Point E is on the same line CD located by considering the continuity in the velocity of the robot at point C. The point C is common for the first and the second phases of the deviated path. Hence the polygon sides  $d_{21}$  and  $d_{22}$  are evaluated through the continuity in the velocity at C and the required terminal velocity at S. The point E is evaluated using the polygon side  $d_{21}$  as

$$x_E = x_C + d_{21} \cos \theta \quad (26a)$$

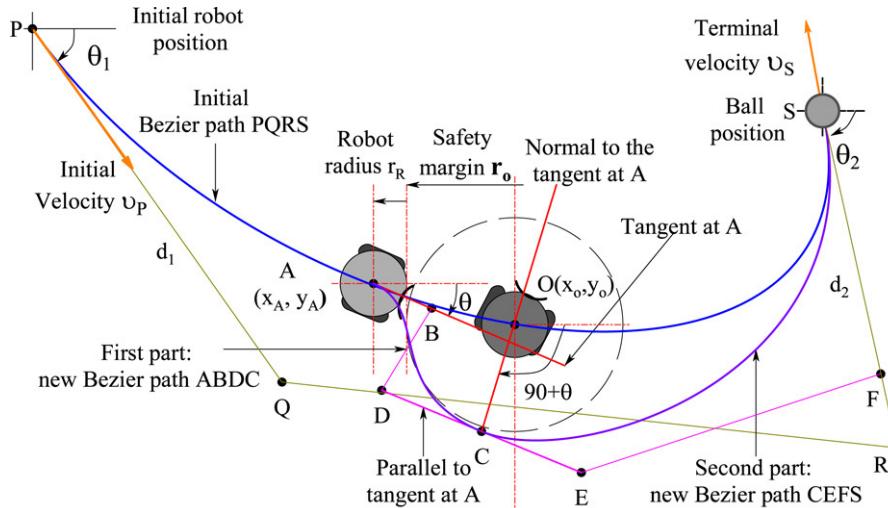
$$y_E = y_C + d_{21} \sin \theta. \quad (26b)$$

The side  $d_{22}$  locates the point F using Eqs. (7) along the side RS of the initial Bezier polygon. The same procedure is repeated if the robot again encounters another obstacle. Fig. 10 shows the modified Bezier path generated using the proposed approach for avoiding an obstacle. The maximum velocity profile for this path is calculated and is shown in Fig. 11.

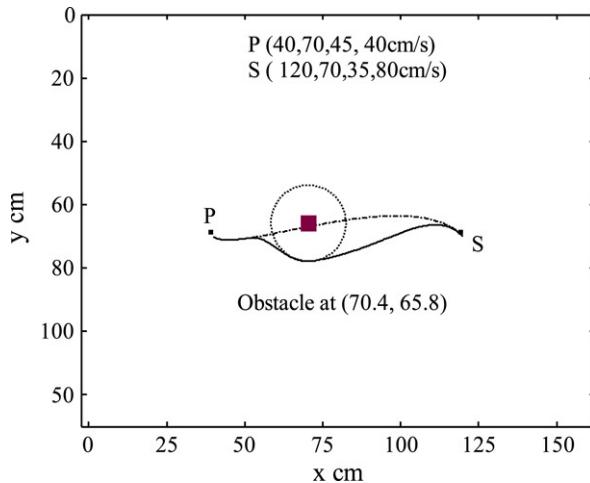
## 7. Results and discussion

According to the results of Marko Lepetic et al. [12], Bezier curves defined with four control points provide optimal paths. For testing the effectiveness of the proposed path planning technique, the path planning problem solved by Lepetic using cubic spline is taken and it is solved using Bezier curve and is shown in Fig. 12. Fig. 12 shows that the initial position of the robot is  $(-50, 100)$  with an orientation angle of  $225^\circ$  (in the counter-clockwise direction) and has an initial velocity of  $100 \text{ cm/s}$ . The ball is located at  $(0, 0)$  and the hitting angle is  $180^\circ$  with a terminal velocity of  $100 \text{ cm/s}$ . The maximum velocity of the robot is limited to  $150 \text{ cm/s}$ . The first and third sides of the Bezier polygon are optimized to  $d_1 = 62.84 \text{ cm}$  and  $d_2 = 27.79 \text{ cm}$ . The integrated length of the optimal Bezier path is obtained as  $131.9 \text{ cm}$ , and the maximum allowable velocity of the robot along the path is only  $146 \text{ cm/s}$ . While moving along this optimal path the minimum travel time of the robot is obtained as  $1.4 \text{ s}$ .

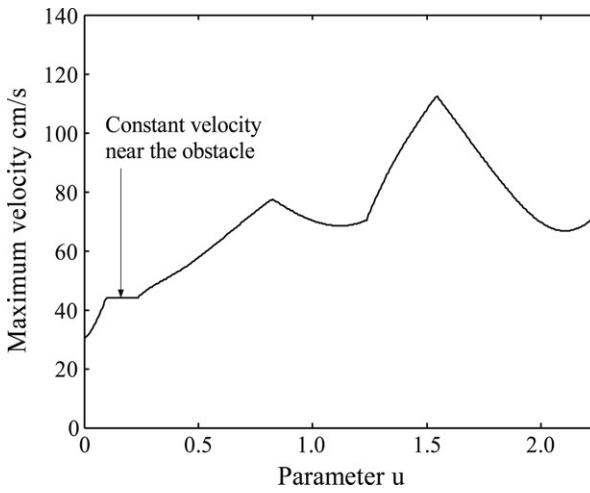
Fig. 13 shows the variation of the curvature of the Bezier path in which all the turning points are indicated. The curvature of the path corresponding to the turning point TP2 is very low and hence the corresponding velocity profile is not shown in Fig. 14. However, velocity profiles of all the other turning points are included. In Fig. 14, the actual end point velocity of the robot is lower than the required value. This is due to the presence of the turning point TP3. Fig. 13 shows that the curvature of the turning point TP3 is high and therefore its maximum allowable velocity is lower than the required terminal velocity. The flat portions on the velocity profiles indicate the violation of the acceleration limits. The velocity profile corresponding to the maximum allowable velocity does not have any flat portion so that the acceleration limits are not violated. Even though the end velocity is low, the maximum allowable velocity profile shows that the average velocity of the robot is high so that it has a better travel time. In order to increase the terminal velocity, the position of the turning point TP3 is to be moved to the left, by increasing the polygon side  $d_2$ . An increase in  $d_2$  is found to increase the length of the path and hence the travel time. The maximum allowable velocity profile of the Bezier curve shown in Fig. 14 has a smooth variation, when compared with the velocity profile for the cubic spline [12].



**Fig. 9.** Robot avoiding collision with a moving object.

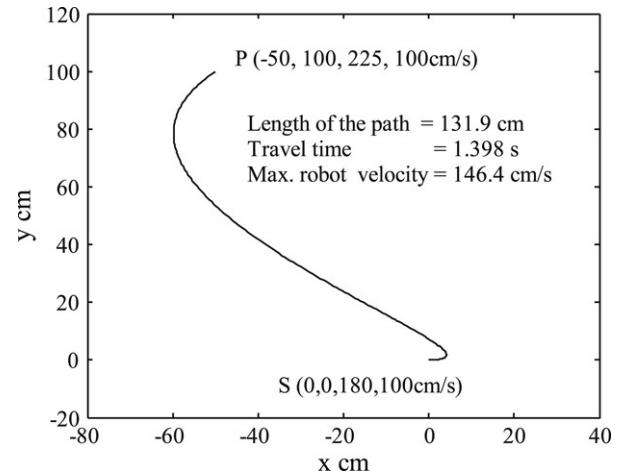


**Fig. 10.** Robot avoiding the obstacle by deviating from the initial path.

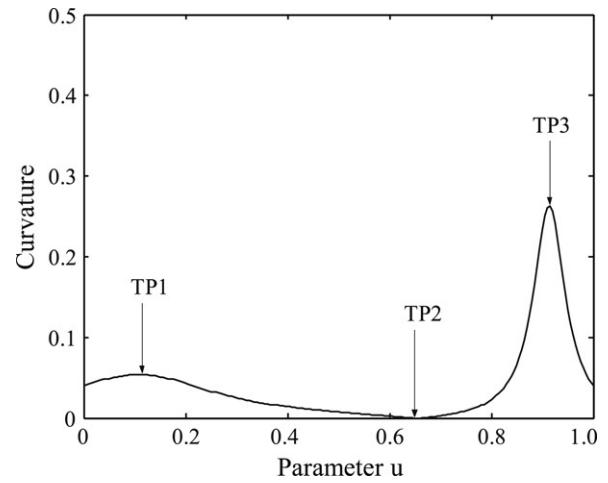


**Fig. 11.** Maximum allowable velocity profile for the deviated path.

Tangential and radial accelerations of the robot along the path are shown in Fig. 15 and it indicates that the acceleration limits are not violated. A sudden variation in the tangential acceleration can be observed near the turning points for which radius of curvature is high. For obstacle avoidance, the robot deviates from the initial

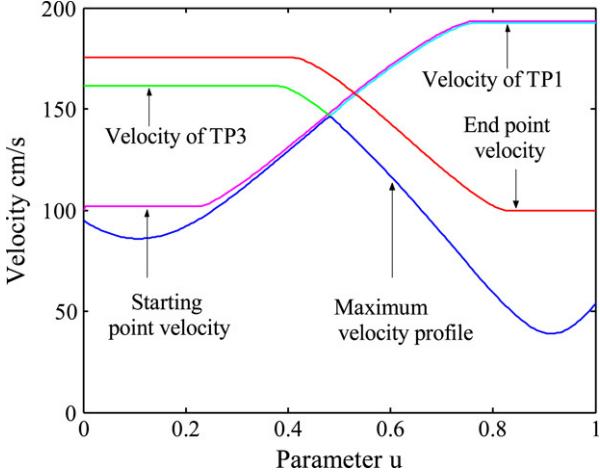


**Fig. 12.** Equivalent Bezier curve for the cubic spline referred in [12].

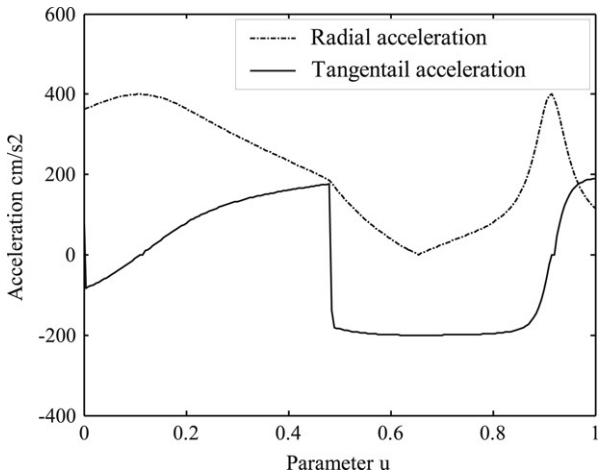


**Fig. 13.** Variation of curvature for the Bezier curve shown in Fig. 11.

path and as a result, the length of the path is increased. During the simulation, static obstacle avoidance techniques are observed to be working very efficiently. For avoiding the collision, the robot changes its angular orientation to a collision free direction and plans a new path to the destination. The rotation of the robot about the wheel for changing the direction increases the travel time.



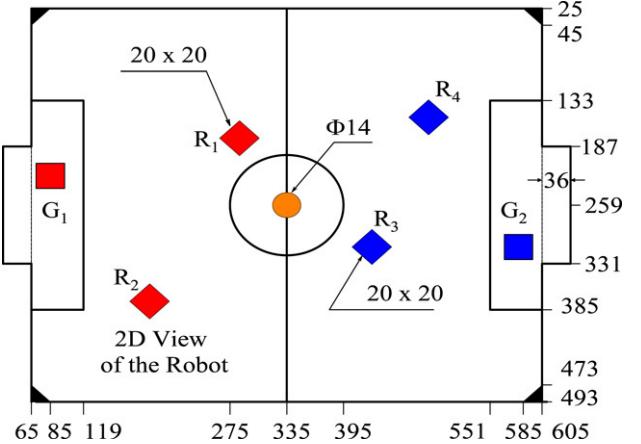
**Fig. 14.** Maximum allowable velocity along Bezier curve in Fig. 11.



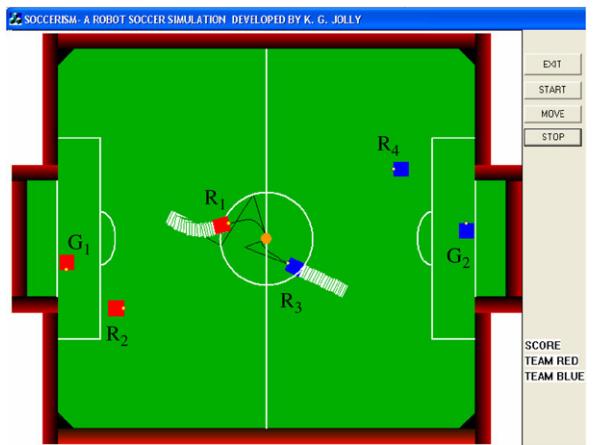
**Fig. 15.** Maximum allowable acceleration along Bezier curve in Fig. 11.

However these are the unavoidable situations in the presence of an obstacle.

For establishing the validity of the Bezier curve based path planning scheme, the curve generated through the proposed techniques is compared with that of the Marko Lepetic et al. [12]. Marko Lepetic et al. optimized the cubic spline trajectories by varying the number of control points used for defining the path. Their optimization approach which is based on the Nelder–Mead simplex direct search method has shown that a cubic spline with four control points yields optimal paths. In their investigations, the robot takes 1.6 s to reach the ball position by travelling along the optimal cubic spline interpolated with four control points. However, for the optimal Bezier path, the travelling time of the robot is only 1.4 s as shown in Fig. 12. The length of the optimal Bezier path is only 131.9 cm, and this information is not given for the optimal cubic spline path [12]. The method of locating the intermediate control points is also not discussed in [12]. In the case of Bezier paths intermediate control points can be located without any ambiguity. A parametric Bezier curve generated using four control points yields third order polynomials and provides smoother paths, because it has less number of turning points. A third order Bezier curve may have one, two, or three turning points depending upon the angles  $\theta_1$  and  $\theta_2$ . Simulation results indicate that the Bezier curves are efficient and effective for the path planning of the soccer robots in most of the cases. Obstacle avoidance strategy works satisfactorily in the domain as well as in the boundary.



**Fig. 16.** Robot soccer field in pixel values.



**Fig. 17.** Nearest robots  $R_1$  and  $R_3$  from the teams approaching the ball.

The proposed Bezier curve based path planning is implemented in the MiroSot simulation league developed in a C++ environment using thread functions. A two-dimensional simulation of the multi-agent robot soccer system is developed as per MiroSot small league field specifications [37] shown in Fig. 16. In this simulation, in addition to a Bezier curve based path planning module, an ANN based decision making module [10] and a fuzzy neural network based action selection module [31] are also incorporated. It simulates the top view of the soccer robot system, which is equivalent to the image of the soccer field captured by a video camera. In the simulation, the home team is coloured RED while the opposing team is coloured BLUE. Fig. 17 shows a simulation window in which the nearest robots are approaching the ball along the Bezier paths. The Bezier paths and corresponding polygons are highlighted in this figure. The defensive behaviour of the RED robot is projected in Fig. 18. In this figure the team BLUE ( $R_3$ ) kicks the ball into the goal area of the team RED. At the same time, the nearest robot of the team RED ( $R_2$ ) exhibits the defensive strategy so that the ball is intercepted and kicked away. The accuracy of the ball passing behaviour and goal shooting behaviour is demonstrated in Fig. 19. In this figure the BLUE robot ( $R_3$ ) standing near the goal area of the home team receives a direct pass from its team-mate  $R_4$  so that it shoots the ball into the goal post avoiding the goalkeeper and scores a goal. These simulations show that coordination, corporation, decision making and intelligence can be possible with a Bezier curve based path planning. The highlighted robot paths in Figs. 18 and 19 show the traces of obstacle avoidance and coordination in the movements of the robots.



Fig. 18. RED ( $R_2$ ) robot defending the attack of BLUE ( $R_3$ ) robot.



Fig. 19. Robot BLUE ( $R_3$ ) scoring from the direct pass of the team-mate ( $R_4$ ).

## 8. Conclusion

An effective and efficient path planning technique for the soccer robots based on Bezier curves has been proposed in this paper. The salient features of the Bezier curves have been found compatible for the robot path planning. The proposed algorithm defines a Bezier path using four control points, and these control points have been easily locatable from the known configuration of the field at each sampling time. The limitations on the tangential and radial accelerations of the robot have been given due consideration during the proposed path planning. Effective obstacle avoidance techniques have been proposed utilizing the properties of the Bezier curve for avoiding collisions with the static and dynamic obstacles in the field. The robot soccer simulation windows show that motion coordination among the agents is possible with the Bezier curve based path planning.

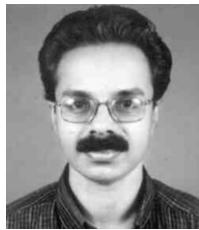
## References

- [1] M.E. Mortenson, Geometrical Modelling, John Wiley & Sons, 1985.
- [2] D.F. Rogers, J.A. Adams, Mathematical Elements for Computer Graphics, 2nd ed., McGraw-Hill international editions, 1989.
- [3] M. Al-Khatib, J.J. Saade, An efficient data-driven fuzzy approach to the motion planning problem of a mobile robot, International Journal of Fuzzy Sets and Systems 134 (2003) 65–82.
- [4] Y. Arai, T. Fujii, H. Asama, H. Kaetsu, I. Endo, Collision avoidance in multi-robot system based on multi-layered reinforcement learning, International Journal of Robotics and Autonomous Systems 29 (1999) 21–32.
- [5] C. Belha, V. Isler, G.J. Pappas, Discrete abstractions for robot motion planning and control in polygonal environments, IEEE Transactions on Robotics 21 (5) (2005) 864–874.
- [6] C. Ferrari, E. Pagello, J. Ota, T. Arai, Multi-robot motion coordination in space and time, International Journal of Robotics and Autonomous Systems 25 (1998) 219–229.
- [7] T. Fraichard, A. Scheuer, From reeds and shepp's to continuous-curvature paths, IEEE Transactions on Robotics 20 (6) (2004) 1025–1035.
- [8] H.P. Huang, C.C. Liang, Strategy-based decision making of a soccer robot system using a real-time self-organizing fuzzy decision tree, International Journal of Fuzzy Sets and Systems 127 (2002) 49–64.
- [9] X.J. Jing, Behaviour dynamics based motion planning of mobile robots in uncertain dynamic environments, International Journal of Robotics and Autonomous Systems 53 (2005) 99–123.
- [10] K.G. Jolly, K.P. Ravindran, R. Vijayakumar, R. Sreerama Kumar, Intelligent decision making in multi-agent robot soccer system through compounded artificial neural networks, International Journal of Robotics and Autonomous Systems 55 (7) (2007) 589–596.
- [11] D.H. Kim, J.H. Kim, A real-time limit-cycle navigation method for fast mobile robots and its application to robot soccer, International Journal of Robotics and Autonomous Systems 42 (2003) 17–30.
- [12] M. Lepetic, G. Klancar, I. Skrjanc, D. Matko, B. Potocnik, Time optimal path planning considering acceleration limits, International Journal of Robotics and Autonomous Systems 45 (2003) 99–210.
- [13] T.C. Liang, J.S. Liu, G.T. Hung, Y.Z. Chang, Practical and flexible path planning for car-like mobile robot using maximal-curvature cubic spiral, International Journal of Robotics and Autonomous Systems 52 (2005) 312–335.
- [14] T.K. Nagy, R. D'Andrea, P. Ganguly, Near-optimal dynamic trajectory generation and control of an omni-directional vehicle, International Journal of Robotics and Autonomous Systems 46 (2004) 47–64.
- [15] E. Papadopoulos, I. Papadimitriou, I. Poulopakis, Polynomial-based obstacle avoidance techniques for nonholonomic mobile manipulator systems, International Journal of Robotics and Autonomous Systems 51 (2005) 229–247.
- [16] K.H. Park, Y.J. Kim, J.H. Kim, Modular Q-learning based multi-agent cooperation for robot soccer, International Journal of Robotics and Autonomous Systems 35 (2) (2001) 109–122.
- [17] N.A. Philip, A.A. Nikos, Obstacle representation by bump-surfaces for optimal motion-planning, International Journal of Robotics and Autonomous Systems 51 (2005) 129–150.
- [18] O. Purwin, R. D'Andrea, Trajectory generation and control for four wheeled omni directional vehicles, International Journal of Robotics and Autonomous Systems 54 (2006) 13–22.
- [19] Z. Qu, J. Wang, C.E. Plaisted, A new analytical solution to mobile robot trajectory generation in the presence of moving obstacles, IEEE Transactions on Robotics 20 (6) (2004) 978–993.
- [20] G. Springer, P. Hannah, R.J. Stonier, S. Smith, Simple strategies for collision-avoidance in robot soccer, International Journal of Robotics and Autonomous Systems 21 (1997) 191–196.
- [21] P. Svestka, M.H. Overmars, Coordinated path planning for multiple robots, International Journal of Robotics and Autonomous Systems 23 (1998) 125–152.
- [22] P. Vadakkepat, X. Penga, B.K. Queka, T.H. Leea, Evolution of fuzzy behaviours for multi-robotic system, International Journal of Robotics and Autonomous System 55 (2) (2007) 146–161.
- [23] H. Weber, A motion planning and execution system for mobile robots driven by stepping motors, International Journal of Robotics and Autonomous Systems 33 (2000) 207–221.
- [24] J.M. Yang, J.H. Kim, Sliding mode control for trajectory tracking of nonholonomic wheeled mobile robots, IEEE Transactions on Robotics and Automation 15 (3) (1999) 578–587.
- [25] S.X. Yang, M. Meng, An efficient neural network method for real-time motion planning with safety consideration, International Journal of Robotics and Autonomous Systems 32 (2000) 115–128.
- [26] Yong K. Hwang, Narendra Ahuja, A potential field approach to path planning, IEEE Transactions on Robotics and Automation 8 (1) (1992) 23–32.
- [27] L. Benamati, C. Cosma, P. Fiorini, Path planning using flat potential field approach, in: Proc. of the IEEE Int. Conf. on Robotics and Automation, 2005, pp. 103–108.
- [28] X.J. Jing, Y.C. Wang, Artificial coordinating field based coordinating collision avoidance, in: Proc. of the IEEE Int. Conf. on Robotics, Intelligent Systems and Signal Processing, China, 2003, pp. 126–130.
- [29] M. Kazemi, M. Mehrandezh, K. Gupta, Sensor-based robot path planning using harmonic function-based probabilistic roadmaps, in: Proc. of the IEEE Int. Conf. on Robotics and Automation, 2005, pp. 84–89.
- [30] K. Kiguchi, K. Watanabe, T. Fukuda, Trajectory planning of mobile robots using DNA computing, in: Proc. IEEE Int. Conf. on Computational Intelligence in Robotics and Automation, Canada, 2001, pp. 380–385.
- [31] K.G. Jolly, R. Sreerama Kumar, R. Vijayakumar, A neuro-fuzzy based intelligent goal shooting behaviour in a multi-agent robot soccer system, in: Proc. of the Int. Conf. on Frontiers in Design and Manufacturing Engg, ICDM-'08, Karunya University, Coimbatore, India, Macmillan Advanced Research Series, 2008, ISBN 10:0230-63460-5, pp. 273–278.
- [32] A.V. Pantola, F.R. Salvador, Pose estimation for a soccer game using neural network, in: Int. Conf. on Computational Intelligence in Robotics and Autonomous Systems, Singapore, 2001, pp. 47–52.

- [33] L. Podsedkowski, J. Nowakowski, M. Idzikowski, I. Visvary, Nonholonomic mobile robots—a new solution for path planning in changing environment, in: Proc. of the IEEE Int. Conf. on Robotics and Automation, 1999, pp. 1–8.
- [34] H. Qianwei, M. Hongxu, Z. Hui, Collision-avoidance mechanism of multi-agent system, in: Proc. of the 2003 IEEE Int. Conf. on Robotics, Intelligent Systems and Signal Processing, China, 2003, pp. 1036–1040.
- [35] K. Sugihara, J. Smith, Genetic algorithm for adaptive motion planning of an autonomous mobile robot, in: Proc. of the IEEE Int. Conf. on Robotics and Automation, 1997, pp. 138–143.
- [36] A. Yahja, S. Singh, A. Stentz, Recent results in path planning for mobile robots operating in vast outdoor environments, in: Proc. IEEE Sym. on Image, Speech, Signal Processing and Robotics, Hong Kong, 1998, pp. 1–6.
- [37] <http://www.fira.net>.



**K.G. Jolly** is working as a lecturer in the department of Mechanical Engineering, NSS College of Engineering, Palakkad, Kerala, India. He did his B.Tech at NSS College of Engineering, Palakkad and M. Tech at Indian Institute of Technology, Kharagpur in Machine Dynamics. Currently he is working towards his Ph.D. at National Institute of Technology Calicut, Calicut, Kerala. His areas of interest include machine dynamics, robotics, multi-agent system and artificial intelligence.



**R. Sreerama Kumar** is working as a professor in the department of Electrical Engineering, National Institute of Technology Calicut, Kerala, India. He did his B. Tech. at NSS College of Engineering, Palakkad, M. Tech at Indian Institute of Technology, Madras and Ph.D. at Indian Institute of Science, Bangalore. Dr.Sreerama Kumar is the recipient of the prestigious national award, instituted by the Indian Society for Technical Education, for Promising Engineering Teacher for the year 2003 for creative work done in technical education. He is a Fellow of the Institution of Engineers (India) and Senior Member of The Institution of Electrical and Electronics Engineers (USA). He has authored five books, and has more than 60 technical publications in reputed journals and conferences to his credit. His field of interest includes soft computation, robotics and system dynamics.



**R. Vijayakumar** is working as a professor in the department of Mechanical Engineering, National Institute of Technology Calicut, Kerala, India. He did his B. Sc (Engg) degree in Mechanical Engineering at the University of Kerala, M. Tech degree in Mechanical Engineering at Indian Institute of Technology, Kanpur and Ph.D. degree in Mechanical engineering at the University of Calicut in 2004.