A Modified Potential Fields Method for Robot Navigation Applied to Dribbling in Robotic Soccer

Bruno D. Damas, Pedro U. Lima, and Luis M. Custódio

Instituto de Sistemas e Robótica
Instituto Superior Técnico, Av. Rovisco Pais, 1, 1049-001, Lisboa, Portugal
bdamas@math.ist.utl.pt, {pal,lmmc}@isr.ist.utl.pt
http://socrob.isr.ist.utl.pt

Abstract. This paper describes a modified potential fields method for robot navigation, especially suited for unicycle-type non-holonomic mobile robots. The potential field is modified so as to enhance the relevance of obstacles in the direction of the robot motion. The relative weight assigned to front and side obstacles can be modified by the adjustment of one physically interpretable parameter. The resulting angular speed and linear acceleration of the robot can be expressed as functions of the linear speed, distance and relative orientation to the obstacles. For soccer robots, moving to a desired posture with and without the ball are relevant issues. To enable a soccer robot to dribble a ball, i.e., to move while avoiding obstacles and pushing the ball without losing it, under severe restrictions to ball holding capabilities, a further constraint among the angular speed, linear speed and linear acceleration is introduced. This dribbling behavior has been used successfully in the robots of the RoboCup Middle-Size League ISocRob team.

1 Introduction

The navigation problem for a mobile robot in an environment cluttered with obstacles is a traditional problem in Robotics. Some variations, such as dynamics *vs.* static obstacles or non-holonomic *vs.* holonomic robots make it harder to solve [5]. Other issues such as a car with a trailer moving backwards or pushing an object can also be of practical interest. In the latter case, constraints must be imposed on the robot linear and angular velocities so as to ensure that the pushed object is not lost.

An algorithm (Freezone) to solve the navigation problem for a mobile robot endowed with omni-directional vision, sonars and odometry, with a particularization for soccer robots, has been introduced in previous work [6]. The algorithm was designed to move the robot towards a desired posture while avoiding obstacles, using omni-directional vision-based self-localization to reset the odometry after some relevant events, and a sonar ring to detect the obstacles. The application of this algorithm to robotic soccer was mainly focused on moving the robot, without the ball, towards a desired posture. However, nothing is said on how

G.A. Kaminka, P.U. Lima, and R. Rojas (Eds.): RoboCup 2002, LNAI 2752, pp. 65–77, 2003.

to move the robot *with* the ball and simultaneously avoiding other robots (e.g., dribbling). Only a few teams of the RoboCup Middle-Size league are capable of dribbling the ball. Dribbling is accomplished either by a suitable mechanical design of the robot [2] or by path planning [7]. In the latter work, problems may be experienced in the presence of fast moving robots. Furthermore, it is not clear how the constraints on angular and linear speeds are specified.

Some of the design features of the Freezone algorithm were conceived to avoid problems displayed by other navigation methods available in the literature (see [6] and the references therein). Among those is the well-known potential fields algorithm [3]. The original potential fields algorithm was designed to drive holonomic vehicles. Nevertheless, it can be modified in different ways to handle non-holonomic constraints such as by projecting the resulting field on the possible acceleration vectors, as in the generalized potential fields method [4].

This paper introduces an alternative approach where the generalized potential field is modified so as to enhance the relevance of obstacles in the direction of the robot motion. The relative weight assigned to front and side obstacles can be modified by the adjustment of one physically interpretable parameter. Furthermore, the resulting angular speed and linear acceleration of the robot, obtained under the modified potential field method, can be expressed as functions of the linear speed, distance and relative orientation to the obstacles. This formulation enables the assignment of angular and linear velocities for the robot in a natural fashion, physically interpretable. Moreover, it leads to an elegant formulation of the constraints on angular speed, linear speed and acceleration that enable a soccer robot to dribble a ball, i.e., to move while avoiding obstacles and pushing the ball without losing it, under severe restrictions to ball holding capabilities. It is shown that, under reasonable physical considerations, the angular speed must be less than a non-linear function of the linear speed and acceleration, which reduces to an affine function of the acceleration/speed ratio when a simplified model of the friction forces on the ball is used and the curvature of the robot trajectory is small. This dribbling behavior has been used successfully in the robots of the RoboCup Middle-Size League ISocRob team.

This paper is organized as follows: in Section 2, the generalized potential fields method, its virtues and shortcomings, are revisited. Section 3 describes the modified potential fields method introduced in this paper. The application of the method to dribbling a ball in robotic soccer is introduced in Section 4, by determining physical constraints on the expressions for angular and linear acceleration obtained in the previous section. In Section 5 some experimental results are presented and Section 6 concludes the paper.

2 Generalized Potential Fields Method

The traditional potential fields method of avoiding obstacles consists of evaluating a repulsive force for each obstacle. That evaluation is made taking into account the distance to the obstacle and the relative velocity between the robot and the obstacle(s). An attractive force that tends to drive the robot to its target is also calculated. Each of these forces has the direction of the object that gave rise to it. The attractive force accelerates the robot towards its target while the repulsive forces accelerate in the opposite direction of the obstacles.

In the generalized potential fields method [4] the absolute value of each repulsive vector is obtained using

$$|a| = \frac{\alpha v}{2d\alpha - v^2} \ , \tag{1}$$

where α is the maximum acceleration available to the robot and v and d are respectively the velocity component in the obstacle direction and the distance to that obstacle. Expression (1) arises when the repulsive potential is defined as the inverse of the critical time interval until a collision happens. This potential is infinite when the estimated time until a collision takes place equals the time needed to stop the robot using full backward acceleration.

This method has some serious drawbacks: it is not always possible for non-holonomic vehicles to accelerate in the direction given by the resulting force vector, and so the potential fields concept is not fully applicable; also, when an obstacle is close enough, the singularity of (1) is reached due to errors caused by the sensors sampling time and the unavoidable noise contained in the sensors measures, leading the robot to an undesirably unstable behavior.

Despite not being well suited for non-holonomic vehicles, the potential fields method is very appealing, since it allows the use of other several navigation methods within the framework of a behavior-based architecture [1], using an independent potential fields module for obstacles avoidance and other modules such as path planning or pose stabilization to drive the robot to its target. In fact, the potential fields method implicitly defines such a behavior-based architecture, where the evaluation of the sum of repulsive forces acts as one module and the evaluation of the attractive vector acts as another module, the robot actuations being simply a result of the vectorial sum of the output of each module.

Therefore, a solution more suitable than just replacing, in the navigation system, the potential fields method by a different method, is to modify it for non-holonomic vehicles in such a way that the method modularity is preserved.

3 Modified Potential Fields Method – The Unicycle Case

The kinematic model of the unicycle vehicle represented in Fig. 1 is given by

$$\begin{bmatrix} v \\ w \end{bmatrix} = \begin{bmatrix} r/2 & r/2 \\ r/2L & -r/2L \end{bmatrix} \begin{bmatrix} w_R \\ w_L \end{bmatrix} \tag{2}$$

where v is the speed of the robot, $w = \dot{\theta}$ is the angular velocity of the robot, w_R and w_L are the rotating speeds of the right and left wheels, r is the wheels radius and L is half of the distance between the contact points of each wheel.

The non-holonomic nature of a unicycle vehicle does not allow movements in arbitrary directions. The instantaneous velocity of the robot has always the same direction as the robot heading (the vehicle body frame is depicted in Fig. 1). So

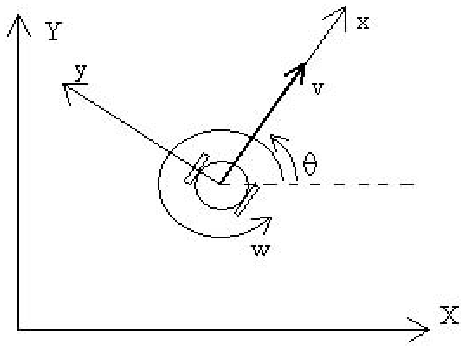


Fig. 1. Kinematics model

it is much more natural to state the repulsion acceleration in two independent components: the first component is the normal acceleration (along the y-axis) and is given by $a_y = vw$; the second component, the tangential acceleration of the robot (along its direction of motion, the x-axis), is equal to the time derivative of the instant velocity. The key point here is that the vectorial sum of these two acceleration components does not necessarily need to have the direction of the fictitious line that connects the obstacle and the robot, as was the case when using the generalized method. In fact it can be a better approach to design the robot behavior separately in terms of its angular and linear speed in the presence of obstacles.

3.1 Potential Fields

The idea behind the potential fields method is the analogy with the movement of electrically charged particles in free space: each one is repelled by the particles with equal signs and attracted to the particles with opposite signs. The force exerted by one particle on another has always the direction of that particle, with an orientation opposite to the particle if the particles have the same sign and the opposite orientation when the particles have different signs. The intensity of the electrostatic force does not depend on the velocity of the particles: since the field is radial it is sufficient to know the distances between them to completely define the potential function. This is a natural consequence of the absence of restrictions on the movement. Nevertheless it is not much useful to act regarding a repulsive force generated by an obstacle whose position can hardly be reached due to the robot kinematics restrictions. Instead of using a Euclidean distance, one can "shape" the potential field to the non-holonomic nature of the robot. In the unicycle case, in the absence of slippage, there is a restriction of movement along the y-axis: v_y is necessarily equal to zero for all times, and so it is convenient to increase the repulsive force along the x-axis since the velocity has only a component along that axis. There are many different possible potential field shapes: the triangular potential field and the elliptic potential field are only two examples. The former is described by the equation

$$|y| = -\frac{|x|}{m} + d \tag{3}$$

while the latter is given by

$$\frac{y^2}{d^2} + \frac{x^2}{(md)^2} = 1 \ . \tag{4}$$

In both cases x and y are the obstacle coordinates in the vehicle referential, d is the potential value for that particular obstacle and m is a constant that defines the potential field "stretch" along the feasible direction of movement (the x-axis in the unicycle case). The constant m usually has a value greater than 1, meaning that the potential value of an obstacle placed along the y-axis equals the potential value of the same obstacle placed at a distance m times larger along the x-axis. If the potential value is expressed in terms of x and y, then

$$d = \frac{|x|}{m} + |y| \tag{5}$$

and

$$d = \sqrt{y^2 + \frac{x^2}{m^2}} \tag{6}$$

for the triangular and elliptic potential fields, respectively. It can also be useful to express these potential fields in polar coordinates, respectively,

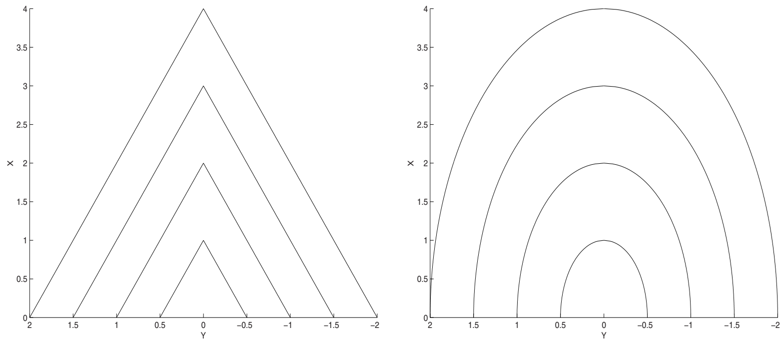
$$d = r\left(\frac{1}{m}|\cos\varphi| + |\sin\varphi|\right) \tag{7}$$

and

$$d = r\sqrt{\frac{1}{m^2}\cos^2\varphi + \sin^2\varphi} \ , \tag{8}$$

where φ, the orientation of the obstacle relative to the robot, and r, the obstacle distance, are obtained by the usual transformations, $r = \sqrt{x^2 + y^2}$ and $\varphi = \arctan(y/x)$. The contour lines for both potential fields can be seen in Fig. 2. Note that, in the generalized potential fields method [4], the potential fields are described by $d = r/\cos\varphi$, since only the velocity component in the obstacle direction is taken into account. Note that generally the potential value corresponds to a distance to the robot using a different, non-Euclidian metric. The generalized potential fields method also leads to a "stretch" of the potential field in the direction of movement (see Fig. 3), as is the case of the triangular and elliptic potential fields when $m > 1$. Also note that if we set $m = 1$ in the elliptic case a circular potential field is obtained and the distance in terms of potential becomes an Euclidean distance.

Up to now nothing has been said about the navigation through the obstacles and the repulsive forces themselves; in fact, the only purpose of this section was to conceive the idea of a non-Euclidean distance that can prove itself more useful when taking into consideration the non-holonomic restrictions of the robot. The navigation algorithms will be presented in the next sub-sections.



(a) Triangular field (b) Elliptic field

Fig. 2. Field countour lines, with $m = 2$

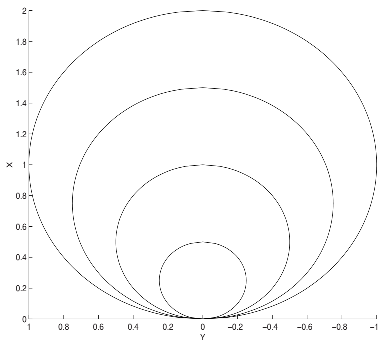


Fig. 3. Potential fields used in (1)

3.2 Normal Repulsive Acceleration

The total applied acceleration in the direction perpendicular to the movement is simply equal to the sum of individual normal accelerations, $a_y = a_{y1} + a_{y2} + a_{y3} + \cdots$. Since $a_y = vw$, where v is the robot linear speed, the last expression can be written as

$$w = w_1 + w_2 + w_3 + \cdots \quad . \tag{9}$$

This means that the total normal acceleration of the robot is given by the sum of individual angular velocities applied to that robot, apart a scale factor.

Should the robot generate the same path independently of its linear speed, given an obstacle configuration and the initial conditions, one must ensure that its curvature function C for those obstacles is independent of the linear speed. Noting that $C = w/v$, if, for each obstacle i,

$$w_i = c(d_i) \cdot v \quad , \tag{10}$$

where $c(d_i)$ is a function of the distance d_i to the obstacle (measured using the chosen potential field), then $C = c(d_i)$ and the curvature function becomes dependent only on the position of that obstacle. Generally that function is assumed to decrease with distance: once again there are several candidates for the curvature function. $c(d)$ could be any of the following:

$$|c(d)| = \frac{G}{d - D} \ , \tag{11}$$

$$|c(d)| = \frac{G}{(d - D)^2} \ , \tag{12}$$

$$|c(d)| = \begin{cases} G(1 - d/D) & \text{if } d < D \\ 0 & \text{otherwise} \end{cases} \ . \tag{13}$$

G is an overall gain and D is a parameter that controls the derivative of the curvature function with respect to d, the distance of the target to the center of the robot. D has distance units and in the case of (11) and (12) must be dimensioned in order to guarantee that $D < R$, where R is the robot radius (if it were not so the curvature function could reach a singularity). A careful choice of values of D and G is critical in what concerns to the robot performance.

The signal of the curvature function is given by

$$\frac{c(d)}{|c(d)|} = \begin{cases} 1 & -\frac{\pi}{2} \leq \varphi < 0 \\ -1 & 0 < \varphi \leq \frac{\pi}{2} \end{cases} \ . \tag{14}$$

For $\varphi = 0$ the signal is undefined: it can be randomly assigned, but when multiple obstacles exist there are other possible approaches (see Section 3.5).

3.3 Tangential Repulsive Acceleration

The total tangential acceleration is also given by the sum of the individual tangential components, $a_x = a_{x1} + a_{x2} + a_{x3} + \cdots$, which can be transformed to

$$\dot{v} = \dot{v}_1 + \dot{v}_2 + \dot{v}_3 + \cdots \ . \tag{15}$$

For each obstacle, the tangential repulsive acceleration can be projected in several ways: usually it should increase when the obstacle gets closer and should decrease when the robot goes slower. This acceleration depends on the speed of the robot and the distance to the target as well,

$$\dot{v} = F(d, v, \ldots) \ , \tag{16}$$

although it can also depend on the time derivatives of v and d when a dynamic relation is used instead of a static one (e.g., a PID controller).

There is no need to use the same parameters, not even the same potential field shapes, when modeling the normal and the tangential repulsive accelerations: those two components are actually independent.

3.4 Attractive Acceleration

To drive the robot to its desired final posture an attractive module is needed. This module can consist of a path-follower or a posture stabilizer by state feedback. For example, a simple controller one can design is

$$\begin{cases} w_c = K_w(\theta_{ref} - \theta) \\ \dot{v}_c = K_v(v_{ref} - v) \end{cases} , \tag{17}$$

where θ_{ref} and v_{ref} are respectively the desired angle and velocity and K_v and K_w are controller parameters to be tuned. θ_{ref} is defined as

$$\theta_{ref} = \arctan\left(\frac{y_{ref} - y}{x_{ref} - x}\right) , \tag{18}$$

where (x_{ref}, y_{ref}) is the robot target position and (x, y) its current position. The control algorithm is simple and the study of its stabilization properties is out of the scope of this work: the goal is simply to achieve the target position with a simple controller. Nevertheless, despite its simplicity, these controllers have proven to be quite satisfactory when conjugated with the obstacle avoidance modules.

Equations (9) and (15) simply state that after the obstacle avoidance modules are designed the modules responsible for getting the robot to its target posture can be added by simply summing the respective acceleration components.

3.5 Multiple Obstacles

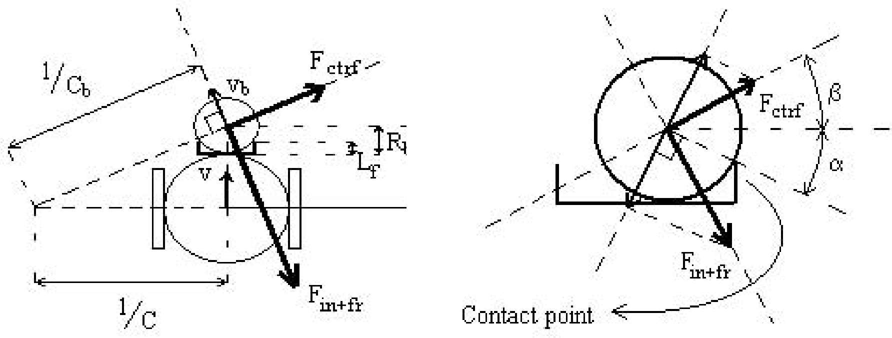
Although (9) and (15) are extremely attractive, suggesting a natural sum of the tangential and normal components relative to the respective obstacles, such an approach has serious drawbacks: two small obstacles placed side by side would give rise to a repulsive force much more stronger than the repulsive force caused by an obstacle with an equivalent size and placed at the same position. Moreover in many cases an autonomous robot has access only to measurements provided by, e.g., a sonar or infrared ring placed around it, and has no clue on whether the reading of two contiguous sensors belongs to distinct obstacles or to the same object. A possible solution is to consider only the most critical obstacle at each side of the robot, determining the nearest left and right obstacles and discarding all the others. In the tangential repulsion case it suffices to get the nearest front obstacle, and so the repulsive accelerations become defined as

$$w_{obs} = c(d_{LMax}) \cdot v + c(d_{RMax}) \cdot v \tag{19}$$

and

$$\dot{v}_{obs} = F\left(d_{FMax}, v, \cdots\right) , \tag{20}$$

where d_{LMax}, d_{RMax} and d_{FMax} are respectively the minimum obstacle distance at the left side, right side and front side of the robot, and F is a suitable function. When the nearest obstacle is located precisely in front of the robot, (19)



(a) Overview (b) Detail

Fig. 4. Forces acting on the ball

becomes undefined; it is not recommended then to choose randomly the side to which assign that obstacle, since such an approach can be a cause of undesirable unstability. One can calculate the second nearest obstacle and then assign the nearest obstacle to the second nearest obstacle side, creating a kind of hysteresis that prevents the robot "hesitation". The robot actuations are finally given by

$$\begin{aligned} w &= w_c + w_{obs} \\ \dot{v} &= \dot{v}_c + \dot{v}_{obs} \end{aligned} \quad , \tag{21}$$

where w_c and $\dot{v}_c$ are the attractive accelerations that try to drive the robot to its final target and w_{obs} and $\dot{v}_{obs}$ are the repulsive accelerations due to the obstacles, defined in (19) and (20).

4 Dribbling

To keep the ball controlled near the robot while the robot moves is a crucial and a challenging problem under the RoboCup Middle-Size League rules. ISocRob and other teams developed a flipper mechanism in order to dribble a ball better.

It is only possible to keep the ball between the flippers while navigating through obstacles if the inertial and the friction forces exerted on the ball are able to balance or overcome the torque originated by the centrifugal force at the contact point (see Fig. 4). This means that

$$\sin(\alpha+\beta)F_{ctrf} \leq \cos(\alpha+\beta)(F_{fr}+F_{in}) \ , \tag{22}$$

where F_{ctrf}, F_{fr} and F_{in} are respectively the centrifugal, the friction and the inertial forces, and where the angles are given by

$$\alpha = \arcsin\frac{R_b - L_f}{R_b} \tag{23}$$

and

$$\beta = \arctan \frac{L}{1/C} \quad . \tag{24}$$

L is the distance between the midpoint of the robot and the midpoint of the ball, R_b is the ball radius, C is the instant curvature of the robot and L_f is the flippers width. It is assumed that the robot is turning left, i.e., $w > 0$ and that $v > 0$. Note that, although the ball is "attached" to the robot, its velocity is only equal to the robot velocity if the instant curvature is null; in the most general case, since $C_b v_b = Cv$, the robot and the ball speeds are related by

$$v_b = \frac{1}{\cos\beta} v \quad , \tag{25}$$

where v and v_b are the robot speed and the ball speed. C_b is the instant curvature of the ball, which can be obtained by

$$\frac{1}{C_b} \approx \sqrt{\left(\frac{1}{C}\right)^2 + L^2} \quad . \tag{26}$$

The inertial, centrifugal and friction forces can be replaced, according to their definitions, by

$$F_{ctrf} = m_b C_b v_b^2 = m_b \frac{C}{\cos\beta} v^2 \quad , \tag{27}$$

$$F_{in} = m_b \dot{v}_b = m_b \frac{\dot{v}}{\cos\beta} \tag{28}$$

and

$$F_{fr} = m_b a_{fr} \quad , \tag{29}$$

where m_b is the mass of the ball and a_{fr} is acceleration caused by the friction force. Expression (22) consequently becomes $Cv^2 \leq \cot(\alpha + \beta)[\cos(\beta) a_{fr} + \dot{v}]$, leading to

$$w \leq \cot(\alpha + \beta) \left[\frac{\cos(\beta)}{v} a_{fr} + \frac{\dot{v}}{v} \right] \quad . \tag{30}$$

The friction between the ball, the robot and the floor is usually very hard to model accurately, as it usually does not depend exclusively on the ball speed and its derivatives. If, for the sake of simplicity, only the term proportional to the ball speed is taken into account when evaluating the friction force, e.g.,

$$a_{fr} = \mu_{fr} v_b = \mu_{fr} \frac{v}{\cos\beta} \quad , \tag{31}$$

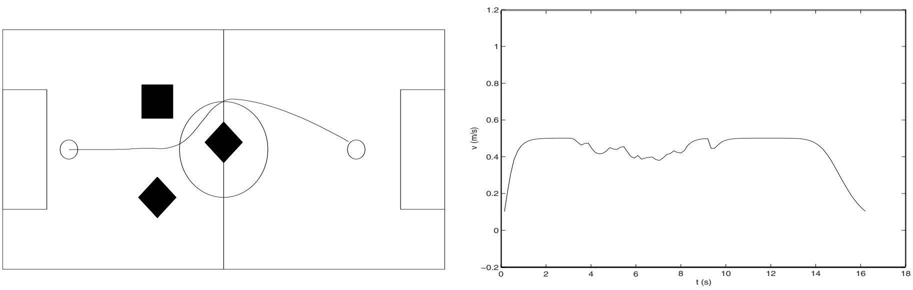
where μ_{fr} is the friction coefficient, then (30) becomes

$$w \leq \cot(\alpha + \beta) \left[\mu_{fr} + \frac{\dot{v}}{v} \right] \quad . \tag{32}$$

Finally, when the curvature C of the robot is small enough, corresponding to a large curved path, (32) simplifies to

$$w \leq \cot(\alpha) \left[\mu_{fr} + \frac{\dot{v}}{v} \right] \quad . \tag{33}$$

Since α is constant, (33) can be written as



(a) Path

(b) Speed profile

Fig. 5. Obstacle avoidance ($v_{ref} = 0.5m/s$)

$$w \leq A + B\frac{\dot{v}}{v} \ , \tag{34}$$

where $A = \cot(\alpha)\mu_{fr}$ and $B = \cot(\alpha)$.

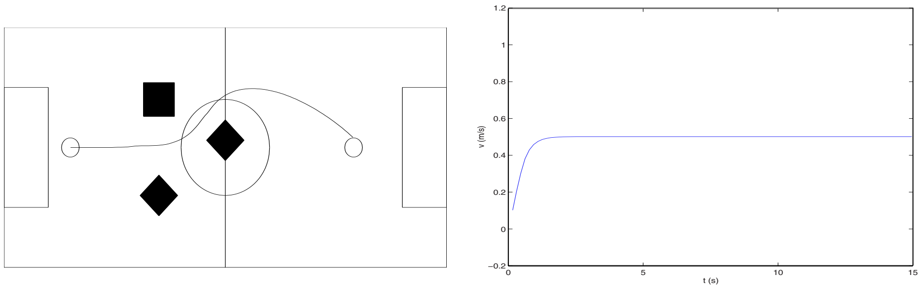
The constant B is easily obtained since it depends only on the geometry of the robot and the size of the ball. Constant A must be determined empirically. Note that (34) is only a valid expression when the curvature is small enough; in the most general case one should use (32). This model assumes that the robot is always turning to the same side. When this is not the case and the robot curvature function changes the ball goes from one flipper to another. Usually that leads to some bouncing, which can actually be a serious problem. Note also that a more sophisticated friction model may be needed to get better results.

Expression (34) states the dribbling fundamental restriction on the robot movement. Usually the angular velocity is bounded in order to meet condition (34), although other more complex schemes may be found, restricting both w and $\dot{v}$, that meet that condition.

5 Experimental Results

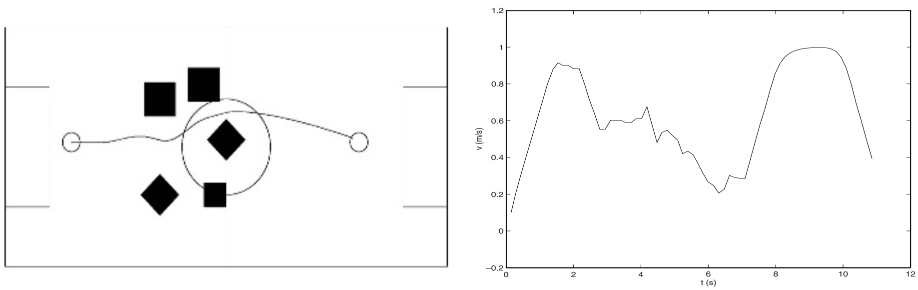
The attractive acceleration components were obtained using very simple controllers, namely those referred on (17), with $K_w = 3$ and $K_v = 0.4$. The repulsive normal acceleration was based on (13), while the tangential acceleration was based on a PD controller whose error is a function of distance also given by (13). Both normal and tangential repulsions use an elliptic field with $m \approx 2$. All the experiments were performed using the robots of the ISocRob team. The start point was $(-3.5, 0)$ — left side of the camp — and the target position was $(3.0, 0)$ — in the right side of the camp.

Note that, as pointed out in Section 3.2, theoretically (10) makes the curvature function independent of the robot speed. However the robot dynamics effectively contribute to a degradation of the robot performance, especially at high speeds. To take that effect into account, the parameter D of equation (13) is a linear function of v, providing the robot with a faster response to obstacles



(a) Path (b) Speed profile

Fig. 6. Obstacle avoidance with dribbling restrictions ($v_{ref} = 0.5m/s$)



(a) Path (b) Speed profile

Fig. 7. Obstacle avoidance in a highly cluttered environment ($v_{ref} = 1.0m/s$)

at high speeds. Fig. 5 presents the robot path in the presence of obstacles and the correspondent speed profile. Fig. 6 shows how the behaviour of the robot changes when the dribbling restriction is active, with $A = 0.3$ and $B' = 0.19$. The value $B' = B/T$, where T is the sampling time, is referred because a discrete time version of equation (34) was used.

In Fig. 6(a), after the mid-field obstacle, the robot follows a wider path to keep ball. It is also visible in Fig. 6(b) that the speed never decreases, since this would leed to a ball loss.

Finally, Fig. 7 shows the robot response in a cluttered environment. The dribbling limitations presented in Section 4 create considerable difficulties to the task of traversing such a cluttered environment, unless the reference speed of equation (17) is a very low one, decreasing the normal acceleration obtained from the obstacle avoidance module (see equation (10)).

6 Conclusions

This paper introduced a modified version of the generalized potential fields method. The modification allows different potential field shapes “stretchable” by changes in one parameter. It also allows a decoupled specification of the tangential and normal components of the acceleration caused by an obstacle. These accelerations can be seen as disturbances acting on the robot efforts to go to its target posture, based on a suitable closed loop guidance controller. This algorithm, such as all the other potential field based methods, can lead to situations where the robot becomes trapped in a local minima situation, in particular in highly cluttered environments. This does not pose too much of a problem since that is precisely the underlying philosophy of that kind of method: to provide a simple and fast, although non-optimal, way of moving to a desired posture while avoiding collisions with other objects (this method is computationally unexpensive, since it only needs to perform some simple calculations for each obstacle distance measured). The independence of the normal and tangential components of repulsive acceleration formulated in Section 3 can nevertheless provide a better way of avoiding those local minima if the normal acceleration is preferred over the tangential acceleration, leading to a behavior where the robot only brakes when it has no place to turn. The escape from local minima should be left to a path-planner, embebbed in the attractive module.

The method enables an elegant formulation of the required constraints for a soccer robot to keep the ball while moving towards a given posture, also known as dribbling. The general case and a simplified version, affine on the ratio between the linear acceleration and velocities, are presented. These results have been successfully applied to the RoboCup Middle-Size League robots of the IsocRob team, leading to goals after a dribble, or 180 degrees turns with the ball.

References

1. R. Arkin, *Behavior-Based Robotics*, The MIT Press, 1998
2. M. Jamzad, A. Foroughnassiraei, E. Chiniforooshan, R. Ghorbani, M. Kazemi, H. Chitsaz, F. Mobasser, S. B. Sadjad, "Design and Construction of a Soccer Player Robot ARVAND", in *RoboCup 1999: Robot Soccer World Cup III*, pp 745–749, M. Veloso, E. Pagello, H. Kitano, Editors, 2000
3. O. Khatib, "Real-Time Obstacle Avoidance for Manipulators and Mobile Robot", 1985 IEEE Int. Conf. On Robotics and Automation, St. Louis, Missouri, pp. 500–505, March 25–28, 1985
4. B. H. Krogh, "A Generalized Potential Field Approach to Obstacle Avoidance Control", Proc. of Intern. Robotics Research Conference, Bethlehem, Pennsylvania, August, 1984
5. J. C. Latombe, *Robot Motion Planning*, Kluwer Academic Publ., 1991
6. C. Marques, P. Lima, "Multi-Sensor Navigation for Soccer Robots", in *RoboCup 2001 Book*, A. Birk, S. Coradeschi, S. Tadokoro, Editors, Springer-Verlag, Berlin, to be published soon.
7. T. Weigel, W. Auerbach, M. Dietl, B. Dumler, J. S. Gutmann, K. Marko, K. Muller, B. Nebel, B. Szerbakowski, M. Thiel, "CS Freiburg: Doing the Right Thing in a Group", in *RoboCup 2000: Robot Soccer World Cup IV*, pp 52–63, P. Stone, T. Balch, G. Kraezschmar, Editors, 2001