An Improved Artificial Potential Field Approach to Real-Time Mobile Robot Path Planning in an Unknown Environment

Joe Sfeir, Maarouf Saad
Department of Electrical Engineering
Université du Québec, École de Technologie Supérieure
Montreal, Canada
Maarouf.saad@etsmtl.ca

Hamadou Saliah-Hassane
LICEF Research Center
Télé-université, TELUQ
Montreal, Canada
saliah@teluq.uqam.ca

Abstract— **This paper deals with the navigation of a mobile robot in an unknown environment. The approach developed is based on the Artificial Potential Field (APF) method in which the target creates a virtual potential that attracts the robot while obstacles create a virtual potential that repels the robot. A new form of repelling potential is proposed in order to reduce oscillations and to avoid conflicts when the target is close to obstacles. A rotational force is integrated as well, allowing for a smoother trajectory around the obstacles. Experiment results show the effectiveness of the proposed approach.**

Keywords-Artificial Potential Fields, mobile robot, navigation, unknown environment, rotational force, obstacle avoidance, FIRAS.

I. Introduction

The idea of a path planning approach is to make the robot move from its starting point towards a destination point, or goal, while avoiding obstacles on its way. The classical Artificial Potential Field (APF) method consists of assigning an attractive artificial potential field to the destination point that attracts the robot, and a repelling artificial potential field to the obstacles that repel the robot. Being under the influence of these two combined potentials, the robot moves to its destination while avoiding the obstacles on its way. The way this algorithm works is that although the potentials are only artificial, they can generate an artificial force field which, in turn, combined with the robot's state and artificial dynamics can produce a virtual velocity and acceleration that are used as an instantaneous reference to control the robot's pose.

Introduced in 1986 by Khatib [1], the Artificial Potential Field (APF) method assigns to the destination point a virtual potential that increasingly attracts the robot with distance and, to the obstacles, a virtual potential that increasingly repels the robot as it gets closer to them. Globally, the robot moves toward the destination point while avoiding the obstacles along its way.

Many different variations that take into account the whole environment have been developed. Khosla and Volpe [2] model the obstacles' potential as superquadrics. Siemiatkowska [3] uses fluid diffusion equations to generate a trajectory. These methods are global and require the prior knowledge of the robot's environment.

Local methods using APF are developed among others by Krogh [2] and Khatib [1]. The classic APF method is adapted to mobile robots. This method is straightforward and does not require excessive computational power. It suffers however from certain drawbacks inherent to the classical APF method, especially oscillations and local minima.

Oscillations are greatly reduced by Ren, McIsaac and Patel in [5] who use a more advanced method of calculating the gradient of the potential, using Newton's method. This makes the trajectory smoother, but drains the computational resources, which makes it less interesting for resource-restricted applications.

Park and Lee in reference [6] use a virtual obstacle potential that gets calculated when the robot is trapped in a local minimum. This reduces – though does not eliminate – the risk of not reaching the goal.

The GNRON (Goal Not Reachable when Obstacles are Nearby) problem is identified and eliminated by Ge and Cui in [7]. When an obstacle is close to the destination point, the robot's final resting position becomes shifted away from the destination point. The authors introduce a modified version of the obstacles' repulsion potential which eliminates the final position offset.

Lee in reference [8] offers a comparison of methods used in the literature (Table I).

TABLE I. COMPARISON OF APF METHOD CHARACTERISTICS

	FIRAS Function	Global Potentials	Superquadrics	Harmonic Functions	Navigation Function
Local Minima Problem Solved	No	No	Yes	Yes	Yes
GNRON Problem solved	No	Yes	Yes	Yes	Yes
Global Environment Knowledge Required	Yes	No	Yes	Yes	Yes

The method we use in our work is based on the classical APF approach detailed by Krogh in [4] and Khatib in [1]. The expression of the obstacle repelling field is modified from its original FIRAS form to take into account the fact that no prior knowledge of the environment is required, to avoid the Goal non Reachable when Obstacles are Nearby (GNRON) problem

978-1-4577-0820-6/11/$26.00 ©2011 IEEE

discussed by Krogh in [4] and to eliminate unnecessary potential peaks. We also introduce a new type of force that improves the robot trajectory by reducing oscillations near obstacles.

The paper is organized as follows. In section II the improved Artificial Potential Fields is presented with the details of the theoretical development. In section III we discuss experimental results and in section V we conclude with some final remarks.

II. Improved APF Method

In this section we introduce the idea of the APF method and develop the theory behind our approach.

The APF has two components: the goal's potential and the obstacles' potential.

$$\phi(x,y)=\phi_g(x,y)+\phi_o(x,y) \tag{1}$$

Where ϕ_g represents the goal's potential, ϕ_o the obstacles' potential and ϕ the total potential. This potential is only computed in the vicinity of the robot (x,y).

The expression of ϕ_g is [9]:

$$\phi_g(x,y)=\frac{1}{2}\xi\left((x-x_g)^2+(y-y_g)^2\right) \tag{2}$$

This expression is similar to that of a spring and the behavior of the robot (assuming that no obstacles are nearby) would be oscillatory just like in the classic second order spring-mass system. It is therefore necessary to introduce a damping factor in the robot's virtual dynamics.

In order for us to develop the expression of the obstacles' potential, we go back to the standard FIRAS function introduced by Khatib in [1]:

$$\phi_o=\begin{cases}\frac{1}{2}\eta\left(\frac{1}{d}-\frac{1}{\rho_0}\right)^2 & \text{if } d\le\rho_0\\ 0 & \text{if } d>\rho_0\end{cases} \tag{3}$$

Where

- η is a positive scaling factor
- d is the minimum distance between the robot and the obstacle
- ρ_0 is a positive constant denoting the obstacle's radius of influence

In our application, we assume that there is no prior knowledge of the environment, so it is not possible to compute d as given by Khatib in [1]. Therefore, we modify the definition of d to:

$$d=\min_{i=1\ldots n}(d_i) \tag{4}$$

Where:

- i is the i^{th} sensor on the robot's periphery (sonar)
- d is the distance sensed by sensor i

Another adjustment we take into consideration is the one given by Krogh in [4] regarding the GNRON problem. If we go back to the basic idea of APF, we can see that it is a form of gradient descent, where the coordinates of the goal point should represent a global minimum of the potential function ϕ. By examining the expression of the total potential (1), we notice that although the goal's potential is minimal at the goal point, there is no guarantee that the obstacles' potential will be minimal at the goal point, especially when there are obstacles near the goal which changes the global minimum's position preventing the robot from reaching the goal.

The solution proposed, which we adopt and modify here is straightforward: we multiply the obstacles' potential by a positive quantity that is null when the robot's coordinates are that of the goal point. Since both potentials are positive, the minimum will be reached when they are both null at the goal point, eliminating the GNRON problem. The quantity proposed by Krogh in [4] is simply the squared distance between the robot and the goal point, which would give the obstacles' potential:

$$\phi_o=\begin{cases}\frac{1}{2}\eta\left(\frac{1}{d}-\frac{1}{\rho_0}\right)^2\cdot\left((x-x_o)^2+(y-y_o)^2\right) \text{ if } d\le\rho_0\\ 0 \text{ if } d>\rho_0\end{cases} \tag{5}$$

This form of the obstacles' potential indeed does ensure that the global minimum will be located at the goal point. However, the fact that the multiplying quantity is highly dependant on the robot's position greatly distorts the shape of the obstacles' potential especially when the robot is very far from the goal point, which is exactly where the problem does not exist.

We introduce a modified quantity to minimize the distortion of the obstacles' potential when the robot is not near the goal, but to maintain the global minimum at that point:

$$\phi_o=\begin{cases}\frac{1}{2}\eta\left(\frac{1}{d}-\frac{1}{\rho_0}\right)^2\cdot\left(1-e^{-\frac{(x-x_o)^2+(y-y_o)^2}{R^2}}\right) \text{ if } d\le\rho_0\\ 0 \text{ if } d>\rho_0\end{cases} \tag{6}$$

Where R is the robot's radius.

With this form of repelling potential, the obstacles' potential is completely unaffected except when the robot is very close to the goal. This form of potential must also naturally assume that the robot is able to position itself at the goal, which means that there are no obstacles too close to the goal for the robot not to collide with them at the goal.

We also modify the repelling potential's essential structure:

$$\phi_o = \phi_m \cdot \left(\frac{\rho_0 - d}{\rho_0}\right)^\eta \cdot \left(1 - e^{-\frac{(x-x_o)^2+(y-y_o)^2}{R^2}}\right) \; ; \; 0 \le d \le \rho_0 \quad (7)$$

Where :

- ϕ_m is the maximum repelling potential reached only when the robot is in contact with the obstacle
- η is a constant greater than 1

In order to demonstrate that the robot's total potential is indeed bounded and to find the expression of the maximum potential, we introduce the virtual energy function:

$$E = 2\phi + \frac{1}{2}m\dot{x}^2 + \frac{1}{2}m\dot{y}^2 \quad (8)$$

Where ϕ is the total potential, $\dot{x}$ and $\dot{y}$ are the robot's velocity coordinates and m is the robot's virtual mass.

In order to make the calculations easier, we can translate the frame and place the origin at the goal point.

Similarly to a mechanical system, E represents the virtual mechanical energy of the system formed of the robot, the goal point and the obstacles.

We start by proving that E is a Lyapunov function.

Given the expression introduced in equation (7), we saw that the total potential is null at the goal point. Also, when the robot reaches the goal, it is going to stop moving, so its velocity coordinates become 0 as well. Therefore, the total energy is null at the goal point.

$$E(0,0) = 0 \quad (9)$$

Since the potential is always positive, we can also say that, given its expression in (8), the total energy is always positive.

$$E > 0; \forall(x, y) \neq (0,0) \quad (10)$$

Differentiating the energy function would give:

$$\dot{E} = 2\dot{\phi} + m\dot{x}\ddot{x} + m\dot{y}\ddot{y} \quad (11)$$

The time variation of the total potential can be given by:

$$\dot{\phi} = \frac{\partial\phi}{\partial t} = \dot{x}\frac{\partial\phi}{\partial x} = \dot{y}\frac{\partial\phi}{\partial y} \quad (12)$$

And the expression of the acceleration coordinates $\ddot{x}$ and $\ddot{y}$ are given by the virtual dynamics of the robot

$$\begin{cases} F_x - \lambda \cdot \dot{x} = m \cdot \ddot{x} \\ F_y - \lambda \cdot \dot{y} = m \cdot \ddot{y} \end{cases} \quad (13)$$

By combining equations (11), (12) and (13), the time derivative of the energy function becomes:

$$\dot{E} = -\lambda\dot{x}^2 - \lambda\dot{y}^2 \quad (14)$$

And since the virtual friction coefficient is positive, we get an always decreasing energy function:

$$\dot{E} < 0, \forall(x, y) \neq (0,0) \quad (15)$$

Which finally gives:

$$\begin{cases} E(0,0) = 0 \\ E > 0; \forall(x, y) \neq (0,0) \\ \dot{E} < 0; \forall(x, y) \neq (0,0) \end{cases} \quad (16)$$

The virtual energy function is therefore a Lyapunov function (as detailed by Nise in [10]) which implies that at any given time, the total energy is always less than the initial energy. Since the initial energy is limited to the goal's attraction potential (the robot is immobile), it is possible to deduce that:

$$\phi \le \frac{1}{2}\xi\left((x_0 - x_g)^2 + (y_0 - y_g)^2\right) \quad (17)$$

We therefore choose the maximum potential ϕ_m to be:

$$\phi_m = \frac{1}{2}\xi\left((x_0 - x_g)^2 + (y_0 - y_g)^2\right) \quad (18)$$

So at any given time,

$$\begin{cases} \phi < \phi_m \\ \phi_m = \phi_{init} \end{cases} \tag{19}$$

It is important to note that this property is true no matter what the expression of the repelling potential is, as long as the total potential and the friction coefficient are positive. This also means that it is impossible for the robot's trajectory not to converge, although not necessarily to the goal point.

With the potential calculated using equation (7), it is no longer necessary to impose an indefinitely growing potential around the obstacles to avoid collisions.

The value of parameter η determines the slope of the repelling potential around the obstacles, which means that the maximum value of the repelling force and the robot's acceleration can be fixed as desired (Fig. 1).

Globally, the modified potential is as follows:

$$\begin{cases} \phi = \phi_o + \phi_g \\ \phi_o = \phi_m \cdot \left(1 - e^{-\frac{D_g^2}{R^2}}\right) \cdot \left(\frac{\rho_0 - d}{\rho_0}\right)^{\eta} \\ \phi_g = \frac{1}{2} \cdot \zeta \cdot \left[(x - x_g)^2 + (y - y_g)^2\right] \end{cases} \tag{20}$$

Where D_g is the distance to the goal.

When the robot is close to an obstacle, the repelling force that is applied to it is perpendicular to the obstacle's surface. When the robot is heading towards the obstacle, this force has a tendency to make the robot turn around and head away from the obstacle. This situation can create undesirable oscillations in the robot's trajectory.

In order to reduce these oscillations, we use a rotational force that help the robot circumvent the obstacles rather than moving away from them. The idea of a circumnavigation force has been discussed in [11] and [12]. We base the expression of the rotational force on the gradient of the new form of repelling potential; we then rotate the vectors by 90 degrees, making the force rotational, as per (21) (Fig. 2).

$$\begin{cases} F_{c,x} = \mathrm{sgn}(\alpha_i) \cdot \frac{\eta' \cdot \phi_m}{\rho_0} \cdot \left(\frac{\rho_0 - d_i}{\rho_0}\right)^{\eta' - 1} \cdot \cos\left(\theta + \alpha_i - \frac{\pi}{2}\right) \\ F_{c,y} = \mathrm{sgn}(\alpha_i) \cdot \frac{\eta' \cdot \phi_m}{\rho_0} \cdot \left(\frac{\rho_0 - d_i}{\rho_0}\right)^{\eta' - 1} \cdot \sin\left(\theta + \alpha_i - \frac{\pi}{2}\right) \end{cases} \tag{21}$$

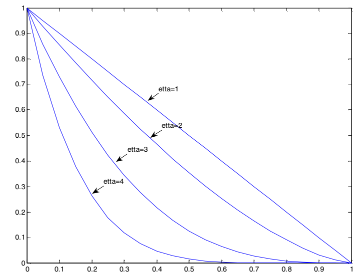

Figure 1. Normalized 2D representation of the new potential form for various values of η

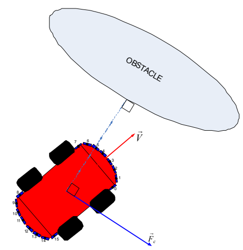

Figure 2. Rotational force parallel to the obstacle's surface

Where $\vec{F}_c$ is the rotational force, α_i is the angular position of the i^{th} sonar on the robot and η' is a positive constant. This expression is similar to that of the radial force, except its direction is perpendicular to the latter's. This force tends to make the robot move in parallel with the obstacle's surface, which makes for a smoother and more logical trajectory, since avoiding the obstacle does not necessarily mean heading away from it, except when the robot is too close to it; this is when the normal radial repelling force is dominant. Note that the rotational force is only local (around obstacles) and does not affect the global stability of the robot.

It is worthy of note that since this virtual force field is rotational, there is no scalar potential function from which it is derived. It is therefore not possible to find a third potential to add to the total potential in equation (20).

III. CASE STUDY

A. *Criteria of Comparison*

Since the output of the algorithm is the robot's trajectory, and since there is no theoretical benchmark to compare to, a few criteria of comparison between the results have to be established.

First, we consider the total trajectory length

$$L_t = \sum_{k=0}^{N} v_k \cdot \tau \tag{22}$$

Where v_k is the robot's instantaneous linear velocity at instant k and τ is the algorithm's sampling time.

Second, we introduce an oscillation coefficient which accounts for the trajectory's overall simplicity as well as for undesirable oscillations

$$C_o = \frac{1}{N}\sqrt{\sum_{k=0}^{N} \omega_k^{\ 2}} \qquad (23)$$

Where ω_k is the robot's rotational velocity at instant k.

It must be noted that a good trajectory is one that has a low value for both of these measures.

B. *Practical Results*

The experiment is done with a real skid-steer robot. The idea was to use the least amount of resources as possible, to validate the fact that the complete algorithm, from the artificial potentials down to the motor control can be implemented on very simple robots, using very limited sensing equipment.

In our experiment, we used a Pioneer P3AT robot. Although this robot is equipped with an array of advanced equipment (Laser range finder, GPS, onboard computer, WiFi connection…), our use of resources was limited to eight of the sixteen sonars that the robot has (8 front sonars at roughly 20 degree direction intervals). The onboard computer was not used; instead, we used the built-in microcontroller that directly controls the motors. For the purpose of interfacing, monitoring and recording of data, communication with a client computer was done via a wireless RS232 connection. The high-level calculation of potentials was done on the client computer. Localization was done at a low level by integrating the speeds of the wheels (Dead reckoning, not using any local or global positioning equipment).

The robot's control scheme consisted of an inside, low-level loop to control the motors' speed and compute the robot's position and of an outside, high-level loop to calculate the successive robot poses (the artificial potential field algorithm). The inner loop's step time was 100ms and the outer loop's was 1s. Due to this constraint, the robot's speed had to be topped at 0.3 m/s, to avoid instabilities due to large discontinuities of the potential fields.

Given the noisy nature of the sonars, we applied a filter on the sonars' signal by averaging 10 samples every second.

The output of the algorithm at each step was the vector $\vec{V} = (V_x, V_y)^T$. Using the robot's kinematic model, the high-level control loop took the form described in Figure 3.

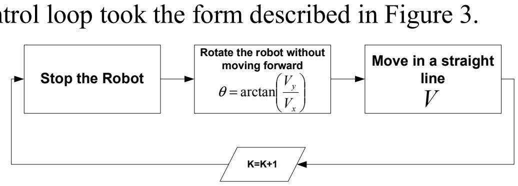

Figure 3. Robot "Stop-and-Go" control algorithm

A comparison was done between the standard FIRAS approach and the new potential field approach using the rotational force. Figure 4 shows the oscillatory nature of the FIRAS trajectory, especially when the robot is heading directly towards an obstacle (Perpendicular to its surface). It also shows how the intervention of the rotational force tends to smoothen the trajectory.

An objective comparison is made between the two approaches in Table II. Using the new potential form combined with the rotational force greatly reduces the coefficient of oscillation. It also reduces the total trajectory length.

TABLE II. PERFORMANCE COMPARISON BETWEEN FIRAS AND NEW POTENTIAL WITH ROTATIONAL FORCE USING REAL ROBOT

Measure	FIRAS	New Potential Form + Rotational Force
Total Trajectory Length (m)	8.4	7.3
Coefficient of Oscillation	15.6	7.7

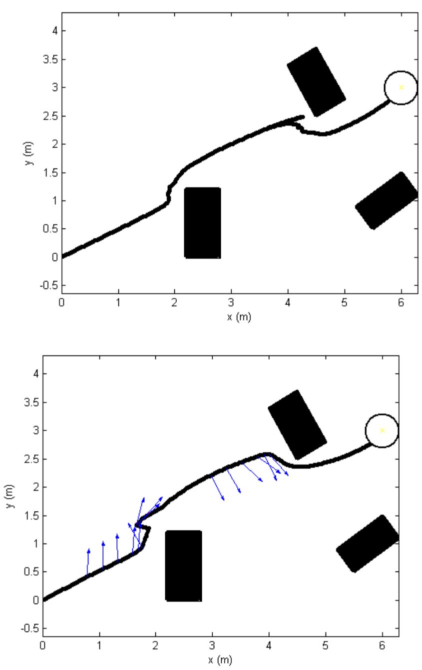

Figure 4. Real robot's trajectory. FIRAS (up) New potential function with rotational force (down). The arrows show the direction of the rotational force.

IV. CONCLUSION

In conclusion, when using the new form of the potential with the rotational avoidance force, even when using a minimum number of sensors with no prior knowledge of the environment, trajectory performance is dramatically improved. It was proven that adding the rotational avoidance force reduces oscillations of the trajectory when the robot is close to obstacles.

With the improvement of trajectory smoothness and length, there is still no mathematical proof that the robot will always escape local minima; it was shown however that chances of getting stuck can be improved.

Another issue that would be of interest is the choice of parameters. Indeed, it was shown that when the η parameter is chosen properly, the robot's trajectory is very smooth. However, there is still no objective approach to choosing the parameter's value other than trial and error.

[1] Khatib, O. 1986. « Real-time obstacle avoidance for manipulators and mobile robots » International Journal of Robotic Research, vol. 5, no.1, pp. 90-98

[2] Khosla, P., Volpe, R. 1998. « Superquadric artificial potentials for obstacle avoidance and approach », Proceedings of the IEEE International Conference on Robotics and Automation, (Philadelphia, PA, April 26-28)

[3] Siemiatkovska, B. 1994.« Cellular Neural Networks for Mobile Robot Navigation », Third IEEE International Workshop on Cellular Neural Networks and Their Applications, Dec. 18-21

[4] Krogh, B. 1984. « A Generalized Potential Field Approach to Obstacle Avoidance Control ». In Proceedings of the ASME Conference on Robotic Research : The Next Five Years and Beyond. Bethlehem, Pennsylvania, 1984.

[5] Ren, J., McIsaac, K. A., Patel R. V. 2006. « Modified Newton's method Applied to Potential Field-Based Navigation for Mobile Robots ». IEEE Transactions on Robotics, vol. 22, no.2, Apr. 2006, pp. 384-391

[6] Park, M. G., Lee, M. C. 2003. « Artificial Potential Field Based Path Planning for Mobile Robots using a Virtual Obstacle Concept ». In Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics. (Kobe, Japan, 20-23 Jul. 2003), pp. 735-740

[7] Ge, S. S., Cui, Y. J. 2000. « New Potential Functions for Mobile Robot Path Planning » IEEE Transactions on Robotics and Automation, vol. 16, no.5, Oct. 2000, pp. 615-620

[8] Lee, L.-F. 2004. « Decentralized Motion Planning Within an Artificial Potential Framework (APF) for Cooperative Payload Transport by Multi-Robot Collectives ». Mémoire de Maîtrise, Buffalo, New York, State University of New York at Buffalo, 193p.

[9] Latombe, J.C. 1990. Robot Motion Planning, 7th Edition. Springer, 651p.

[10] Nise, N. 2003. Control Systems Engineering, 4th Edition. Wiley, 983p.

[11] Gat, E. 1993. "Navigation templates: enhancements, extensions, and experiments," in Proceedings of the IEEE International Conference on Robotics and Automation, vol.1, 2-6 May 1993, pp.541-547

[12] Arkin, R. C. 1998. Behavior-Based Robotics. MIT Press, 491p.