We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



1

Local Autonomous Robot Navigation using Potential Fields

Miguel A. Padilla Castañeda, Jesús Savage, Adalberto Hernández and Fernando Arámbula Cosío
University
Country

Chapter Abstract

The potential fields method for autonomous robot navigation consists essentially in the assignment of an attractive potential to the goal point and a repulsive potential to each of the obstacles in the environment. Several implementations of potential fields for autonomous robot navigation have been reported. The most simple implementation considers a known environment where fixed potentials can be assigned to the goal and the obstacles. When the obstacles are unknown the potential fields have to be adapted as the robot advances, and detects new obstacles. The implementation of the potential fields method with one attraction potential assigned to the goal point and fixed repulsion points assigned to the obstacles, has the important limitation that for some obstacle configurations it may not be possible to produce appropriate resultant forces to avoid the obstacles. Recently the use of several adjustable attraction points, and the progressive insertion of repulsion points as obstacles are detected online, have proved to be a viable method to avoid large obstacles using potential fields in environments with unknown obstacles. In this chapter we present the main characteristics of the different approaches to implement local robot navigation algorithms using potential fields for known and partially known environments. Different strategies to escape from local minima, that occur when the attraction and repulsion forces cancel each other, are also considered.

1. Introduction: The Potential Fields Method for Obstacle Avoidance

The local autonomous robot navigation problem consists of the calculation of a viable path between two points, an starting and a target point. The local navigation approach should produce an optimum (usually shortest) path, avoiding the obstacles present in the working environment. In general, the obstacles and the target could be static or dynamic. The obstacles could also be known *a priory* (e.g. the different walls in a building) or could be unknown (e.g. persons walking nearby the robot). In this chapter are presented the following aspects of a potential fields scheme for autonomous robot navigation: The potential and force field functions; The use of single or multiple attraction points; The construction of an objective function for field optimization; The field optimization approach in known and unknown environments. In the last section of the chapter we present hybrid

approaches to recover from local minima of the potential field. During the chapter we have only considered potential fields defined in cartesian space, where attractive or repulsive potentials are a function of the position of the target or the obstacle. Recently, potential fields defined in a 2D trajectory space, using the path curvature and longitudinal robot velocity, have been reported (Shimoda et al., 2005).

1.1 Previous works on artificial potential fields for autonomous robot navigation

Artificial potential fields for autonomous robot navigation were first proposed by Khatib (1990). The main idea is to generate attraction and repulsion forces within the working environment of the robot to guide it to the target. The target point has an attractive influence on the robot and each obstacle tends to push away the robot, in order to avoid collisions. Potential field methods provide an elegant solution to the path finding problem. Since the path is the result of the interaction of appropriate force fields, the path finding problem becomes a search for optimum field configurations instead of the direct construction (e.g. using rules) of an optimum path. Different approaches have been taken to calculate appropriate field configurations.

Vadakkepat et al. (2000) report the development of a genetic algorithm (GA) for autonomous robot navigation based on artificial potential fields. Repulsion forces are assigned to obstacles in the environment and attraction forces are assigned to the target point. The GA adjusts the constants in the force functions. Multiobjective optimisation is performed on 3 functions which measure each: error to the target point, number of collisions along a candidate path, and total path length. This scheme requires *a priory* knowledge of the obstacle positions in order the evaluate the number of collisions through each candidate path. Kun Hsiang et al. (1999), report the development of an autonomous robot navigation scheme based on potential fields and the chamfer distance transform for global path planning in a known environment, and a local fuzzy logic controller to avoid trap situations. Simulation and experimental results on a real AGV are reported for a simple (4 obstacles) and known environment. McFetridge and Ibrahim (1998) report the development of a robot navigation scheme based on artificial potential fields and fuzzy rules. The main contribution of the work consists in the use of a variable for the evaluation of the importance of each obstacle in the path of the robot. Simulation results on a very simple environment (one obstacle) show that use of the importance variable produces smoother and shorter trajectories. Ge and Cui (2002) describe a motion planning scheme for mobile robots in dynamic environments, with moving obstacles and target point. They use potential field functions which have terms that measure the relative velocity between the robot and the target or obstacle.

The main disadvantage of artificial potential field methods is its susceptibility to local minima (Borenstein and Koren, 1991), (Grefenstette and Schultz, 1994). Since the objective function for path evaluation is usually a multimodal function of a large number of variables. Additionally, in the majority of works on artificial potential fields for robot navigation, a single attraction point has been used. This approach can be unable to produce the resultant forces required to avoid a large or several, closely spaced, obstacles (Koren and Borenstein, 1991). An scheme based on a fixed target attraction point and several, moving, auxiliary attractions points was reported in Arámbula and Padilla (2004). Multiple auxiliary attractions points with adjustable position and force intensity enable navigation around large obstacles, as well as through closely spaced obstacles, at the cost of increased

complexity of the field optimisation. A GA has been successfully used to optimise potential fields with a large number of unknown obstacles and four auxiliary attraction points. The approach is fast enough for on-line control of a mobile robot.
In the following section we present different potential and force field functions which have been used for robot navigation. In section 3 we present the main characteristics of potential fields with one, as well as several attraction points. In section 4 we present the basics of field optimization: objective function construction; function optimization in known and unknown environments. In section 5 we introduce a hybrid method to avoid local minima during field optimization.

2. Potential field and force field functions

The first formulation of artificial potential fields for autonomous robot navigation was proposed by Khatib (1990). Since then other potential fields formulation have been proposed (Canny 1990, Barraquand 1992, Guldner 1997, Ge 2000, Arámbula 2004).
In general, the robot is represented as a particle under the influence of an scalar potential field U, defined as:

$$U = U_{att} + U_{rep} \tag{1}$$

where U_{att} and U_{rep} are the attractive and repulsive potentials respectively.
The attraction influence tends to pull the robot towards the target position, while repulsion tends to push the robot away from the obstacles. The vector field of artificial forces $\mathbf{F}(\mathbf{q})$ is given by the gradient of U:

$$\mathbf{F}(\mathbf{q}) = -\nabla U_{att} + \nabla U_{rep} \tag{2}$$

where ∇U is the gradient vector of U at robot position $\mathbf{q}$(x, y) in a two dimensional map.
In this manner, $\mathbf{F}$ is defined as the sum of two vectors $\mathbf{F}_{att}(\mathbf{q}) = -\nabla U_{att}$ and $\mathbf{F}_{rep}(\mathbf{q}) = \nabla U_{rep}$, as shown in eq. 3.

$$\mathbf{F}(\mathbf{q}) = \mathbf{F}_{att}(\mathbf{q}) + \mathbf{F}_{rep}(\mathbf{q}) \tag{3}$$

2.1 Artificial Potential Fields Formulation

The most commonly used form of potential field functions proposed by Kathib (1990) is defined as:
Attraction potential field

$$U_{att} = \frac{1}{2}\xi\, d^2 \tag{4}$$

where $d = \left|\mathbf{q} - \mathbf{q}_a\right|$; $\mathbf{q}$ is the current position of the robot; $\mathbf{q}_a$ is the position of an attraction point; and ξ is an adjustable constant.

Repulsion potential field

$$U_{rep} = \begin{cases} \frac{1}{2}\eta\left(\frac{1}{d} - \frac{1}{d_0}\right)^2 & d \le d_0 \\ 0 & d > d_0 \end{cases} \tag{5}$$

where $d = \left|\mathbf{q} - \mathbf{q}_o\right|$ for the robot position $\mathbf{q}$ and the obstacle position $\mathbf{q}_o$, d_0 is the influence distance of the force and η is an adjustable constant.
The corresponding force functions are:
Attraction force

$$\mathbf{F}_{att}(\mathbf{q}) = -\nabla U_{att} = -\xi(\mathbf{q} - \mathbf{q}_a) \tag{6}$$

where $\mathbf{q}$ is again the robot position, $\mathbf{q}_a$ the position of the attraction point and ξ is an adjustable constant.
Repulsion force

$$\mathbf{F_{rep}(q)} = \nabla U_{rep} = \begin{cases} \eta\left(\frac{1}{d} - \frac{1}{d_0}\right)\frac{(\mathbf{q} - \mathbf{q_0})}{d^3} & d \le d_0 \\ 0 & d > d_0 \end{cases} \tag{7}$$

where $d = \left|\mathbf{q} - \mathbf{q}_o\right|$ for the robot position $\mathbf{q}$ and the obstacle position $\mathbf{q}_o$, d_0 is the influence distance of the force and η is an adjustable constant.
The above formulation is popular due to its mathematical elegance and its simplicity; unfortunately, it suffers of oscillations and local minima under some obstacle configurations could cause problems, such: trap situations due to local minima, oscillations in narrow passages or impossibility of passing between closely spaced obstacles.
Some different potential fields have been reported in the past in order to solve these problems. Ge and Cui (2000) proposed a modified formulation of Eq. 5 and Eq. 7 for repulsion forces for solving the problem of having a non-reachable target when it is placed nearby obstacles due to the fact that as the robot approaches the goal near an obstacle, the attraction force decrease and becomes drastically smaller than the increasing repulsion force. The modified repulsion potential takes the form of:

$$U_{rep} = \begin{cases} \frac{1}{2}\eta\left(\frac{1}{d} - \frac{1}{d_0}\right)^2 \left|\mathbf{q} - \mathbf{q_{goal}}\right|^n & d \le d_0 \\ 0 & d > d_0 \end{cases} \tag{8}$$

The term $\left|\mathbf{q}-\mathbf{q}_{\mathbf{goal}}\right|$ is the distance between the robot and the goal position. The introduction of this term ensures that the total potential $U = U_{att} + U_{rep}$ arrives at its global minimum 0, if and only if $\mathbf{q} = \mathbf{q}_{\mathbf{goal}}$. The corresponding repulsion force is given by:

$$\mathbf{F_{rep}}(\mathbf{q}) = \nabla U_{rep} = \begin{cases} \mathrm{F}_{\mathrm{rep1}}\,\mathbf{n}_{OR} + \mathrm{F}_{\mathrm{rep2}}\,\mathbf{n}_{RG} & d \le d_0 \\ 0 & d > d_0 \end{cases} \tag{9}$$

where

$$\mathbf{F_{rep1}} = \eta\left(\frac{1}{d} - \frac{1}{d_0}\right)\frac{\left|\mathbf{q}-\mathbf{q}_{goal}\right|^n}{d^2} \tag{10}$$

$$\mathbf{F_{rep2}} = \frac{\eta^2}{2}\left(\frac{1}{d} - \frac{1}{d_0}\right)^2\left|\mathbf{q}-\mathbf{q}_{goal}\right|^{n-1} \tag{11}$$

$\mathbf{n}_{OR} = \nabla d$ and $\mathbf{n}_{RG} = -\nabla\left|\mathbf{q}-\mathbf{q}_{goal}\right|$ are two unit vectors pointing from obstacle to the robot and from the robot to the goal, respectively. In this way, $F_{rep1}\mathbf{n}_{OR}$ repulses the robot away from the obstacle, while $F_{rep2}\mathbf{n}_{RG}$ attracts the robot toward the goal. Although this approach solves the problem of nonreachable goals which are nearby, still suffers of local minima at some obstacle configurations and combinations of η and ξ.

Arámbula and Padilla (2004) modified equations 6 and 7 experimentally in order to amplify the effect of repulsion in obstacles and designed a potential field scheme with movable and adjustable, in real time, auxiliary attraction points in order to reduce the risk of the robot to being trapped in local minima. The modified artificial attraction force $\mathrm{F}_{\mathbf{att}}$ used for the target point and for each of the auxiliary attraction points is:

$$\mathbf{F}_{att}(\mathbf{q}) = -\nabla U_{att} = -\xi(\mathbf{q}-\mathbf{q}_a)\frac{1}{\left|\mathbf{q}-\mathbf{q}_a\right|} \tag{12}$$

The aim of normalization of Eq. 6 is to produce an attraction force independent of the distance between the robot and the target point (Eq.12). The artificial repulsion force F_{rep} is defined as:

$$\mathbf{F}_{rep}(\mathbf{q}) = \nabla U_{rep} = \begin{cases} \eta \cdot sqrt\left(\frac{1}{d} - \frac{1}{d_0}\right)\frac{(\mathbf{q}-\mathbf{q}_o)}{d^3} & d \le d_0 \\ 0 & d > d_0 \end{cases} \tag{13}$$

As the robot gets closer to an obstacle, the repulsion force of the closest obstacle points grows in the opposite direction of the robot trajectory. If the robot distance to an obstacle

point is higher than d_0, that obstacle position has no effect on the robot. An steep repulsion force function is needed in order to enable navigation through narrow passages, however it was observed that taking the square root of $(1/d - 1/d_0)$ in Eq.13 provides a light increase of the repulsion forces at mid distances (as shown in Fig.1) enabling, in turn, a safer obstacle avoidance. The constant η is also adjustable in real time as the robot moves by a genetic algorithm as is explained in section 4.

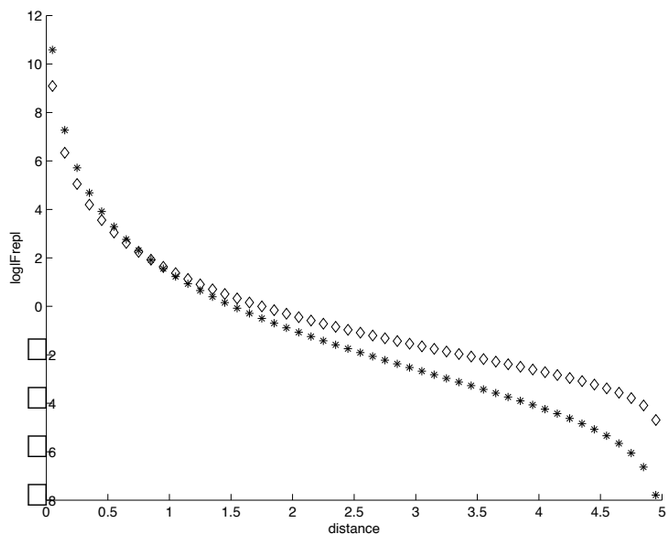


Figure 1. Plot of the magnitude of equation 9 (diamonds) and the same equation without taking the square root of $(1/d\text{-}1/d_0)$ (stars)

2.2 Distance Fields as Potential Fields

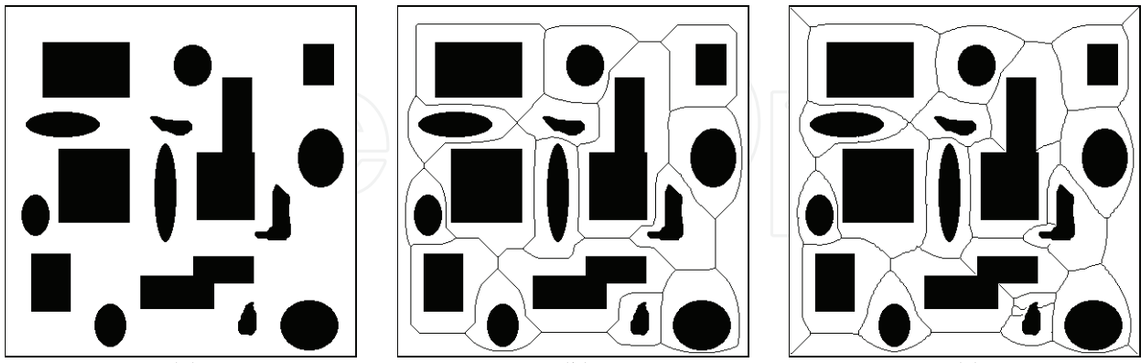
Canny and Lin (1990) and Barraquand et al (1992) used a similar approach based on distance functions for building the potential field. Canny and Lin (1990) used the Euclidean distance field as a non-negative continuous and differentiable function defined as:

$$U = U_{att} = \min_i \eta(D_i(O_i, x)) \tag{14}$$

where $D_i(O_i, x)$ is the shortest Euclidean distance between an obstacle O_i and the position x of the robot and η is and adjustment constant. In this manner, the potential tends to zero as the robot approaches the obstacles, so the robot moves along the skeleton of the distance field that represents the path of maximum attraction potential. Unfortunately, under certain obstacle configurations the resulting potential field may contain local maxima, specially if the robot is near obstacle concavities.

Barraquand et al (1992) used a simple algorithm that computes the potential U as a grid where at the goal position x_{goal} is setting up the value of *0* and then progressively by region growing incrementing in 1 the value of the free obstacle neighbors and infinity in obstacle positions. Then the navigating path is found by tracking the flow of the negative

gradient vector field $-\nabla U$ starting from the initial robot position x_{init}. The idea behind this approach is to produce a free of local minima potential field.
Fig. 2 shows an example of an obstacle workspace and the potential fields produced by the calculation of the distance fields of both approaches.



(a) (b) (c)

Figure 2. Example of a workspace with distance field as potential field. a) Obstacle configuration; b) Skeleton of the potential field produced by Canny and Lin (1990); c) Skeleton of the potential field produced by Barraquand et al (1992)

2.3 Harmonic Artificial Potential Fields

Some authors have proposed the use of harmonic functions for building artificial potential fields which satisfies the $\nabla^T \nabla U = 0$ in order to avoid the problem of local minima (Connolly 1990, Utkin 1991, Guldner 1997). The generalized harmonic potential of a point charge q is:

$$U(r) = \begin{cases} \dfrac{q}{r^{n-2}} & n = 1,3,4,... \\ q \ln \dfrac{1}{r} & n = 2 \end{cases} \tag{15}$$

for $r > 0$ in $\Re^n$ where $r = \sqrt{\sum_n x_i^2}, x = (x_1, x_2, ..., x_n)$ and the gradient $-\nabla U$ described by:

$$-\nabla U(r) = \begin{cases} (n-2)\dfrac{q}{r^{n-1}} \vec{e}_r & n = 1,3,4,... \\ \dfrac{q}{r} \vec{e}_r & n = 2 \end{cases} \tag{16}$$

where $\vec{e}_r$ denotes a unit vector in radial direction.

In particular, Guldner et al (1997) introduced *the harmonic dipole potential* based on electrostatics, where points on the workspace represent *point charges* within a *security zone* inside ellipsoidal gradients. For a single obstacle, they defined the gradient of the harmonic

potential field for a dipole charge as a *security circle* with radius R with a unit charge at the target point in the origin of the circle and a positive obstacle charge $q < 1$ defined as:

$$q = \frac{R}{R + D} \tag{17}$$

where D is the distance between the two charges. For multiple obstacles, independent security zones are determined for each obstacle in a transformed space and mapped in the original space without overlapping. When computing the navigation path, the method only considered the closest obstacle to the robot at each time and requires to switch obstacle potentials when the robot cross between security zones; in order to avoid discontinuities when switching potentials between obstacles, the resultant potential near the border of two zones is calculated by the weighted contribution of the obstacles, where the weight depends on the distance to the s*ecurity borders* of the obstacles.

2.4 Physical Fields as Artificial Potential Fields

Physical analogies for potential fields for robot navigation have been reported in the past for electrostatics (Guldner et al 1997), incompressible fluids dynamics (Keymeulen et al 1994), gaseous substance diffusion (Schmidt and Azarm 1992), mechanical stress (Masoud et al 1994) and steady-state heat transfer (Wang and Chirikjian 2000). For example, Wang and Chirikjian (2000) used temperature as the artificial potential field because in heat transfer the heat flux points in the direction of a negative temperature gradient; temperature monotonically decreases on the path from any point to the sink. In the analogy, the goal is treated as the sink that pulls the heat in and the obstacles as zero or very low thermal conductivity. With this approach the temperature is characterized as the harmonic field without local minima of the form:

$$\nabla \cdot (K \nabla T) = q \tag{18}$$

$$\int_{\Omega} q \mathrm{dV} = 0 \tag{19}$$

$$\left(\frac{\partial T}{\partial n} \right)_{\Gamma} = 0 \tag{20}$$

$$\underline{f} = -K \nabla T \tag{21}$$

where T is the temperature over the workspace, q indicates the heat sources and sink, K is the thermal conductivity which is a function of space coordinates, Ω is the configuration space where the robot moves and Γ is the boundary of this configuration space, n expresses the unit normal vector and f is the heat flux. Numerical solution is obtained from finite difference or finite element methods.

3. Attraction point configurations

In order to avoid trap situations or oscillations in the presence of large or closely spaced obstacles (Koren and Borenstein, 1991), in a map modelled as a two dimensional grid, several auxiliary attraction points can be placed around the goal cell (Fig. 3). Each attraction

force $\mathbf{F}_{att}^{i}$ located at cell c_i, depends on the corresponding value of ξ_i (Eq. 6), which needs to be adjusted by an optimization algorithm as described in the next section. The effect of auxiliary attraction points has been evaluated in two modalities (Arámbula and Padilla, 2004): (1) auxiliary points placed at a fixed distance (of 15 cells) from the goal cell; and (2) auxiliary points placed at a variable distance (between 0 and 15 cells), which is adjusted automatically with a GA. Results from both approaches are shown in section 4. The use of auxiliary attraction points with a force strength and position automatically adjusted with a GA, allows for the generation of resultant force vectors which enable the robot to avoid large obstacles, as shown in Fig. 4.

3.1. Multiple attraction points

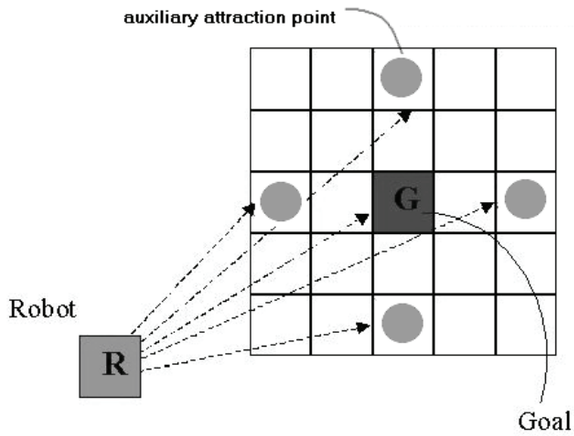


Figure 3. Attraction field composed of 5 attraction cells with adjustable position and force intensity

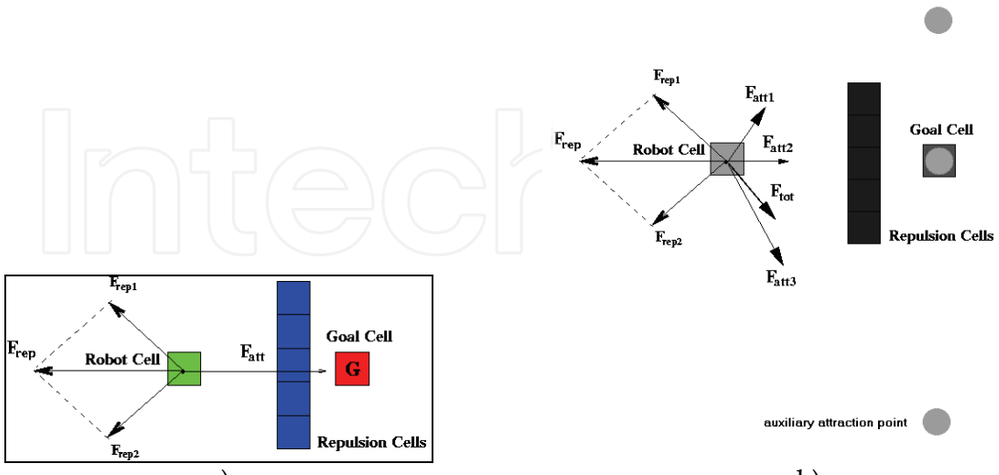


Figure 4. a) A large obstacle which can not be avoided with one attraction point only; b) use of auxiliary attraction points of varying force intensity and position allow for the generation of resultant forces which guide the robot around the obstacle

4. Potential field optimization for obstacle avoidance

4.1 Pre-calculated potential fields

When the environment where a robot navigates is of the type of an office or a house, and it is known in advance, then the objects and walls can be represented using polygons.

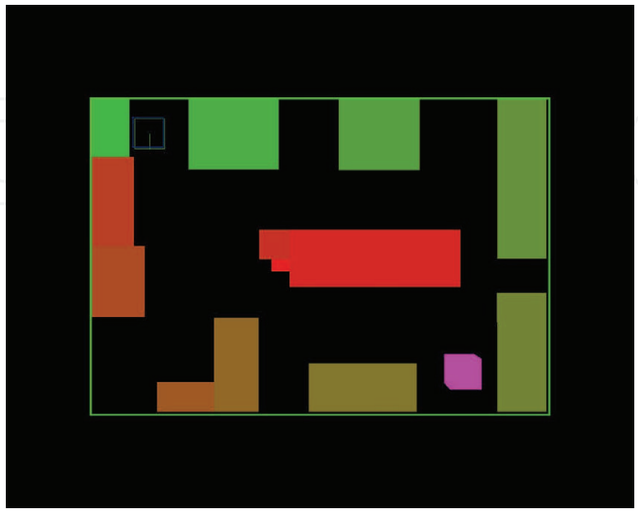


Figure 6. Representation of the testing environment using polygons

Each polygon consists of a clockwise ordered list of its vertices. Representing the obstacles as polygons makes easier the definition of forbidden areas, which are areas which are not allowed for the robot to enter. They are built by growing the polygons that represent the objects by a distance greater than the radius of the robot, to consider it as a point and not as a dimensioned object (Lozano Pérez 1979). It is possible to create the configuration space in this way when the robot has a round shape. Fig. 6. shows a representation of a polygonal testing environment example testing environment. From the polygonal representation it is found the free space where the robot can navigate with this approach, which is formed by a set of equally spaced cells in which there are not obstacles, as it is shown in Fig. 7.

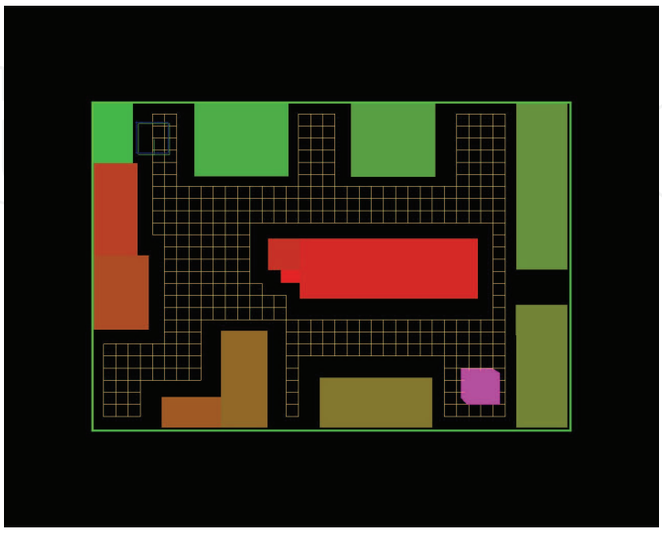


Figure 7. Representation of the free space using cells

For each cell it is calculated the repulsion forces that each of the obstacles generates, they are added and the resulting force is obtained, Fig. 8 shows the repulsion force map for the environment.

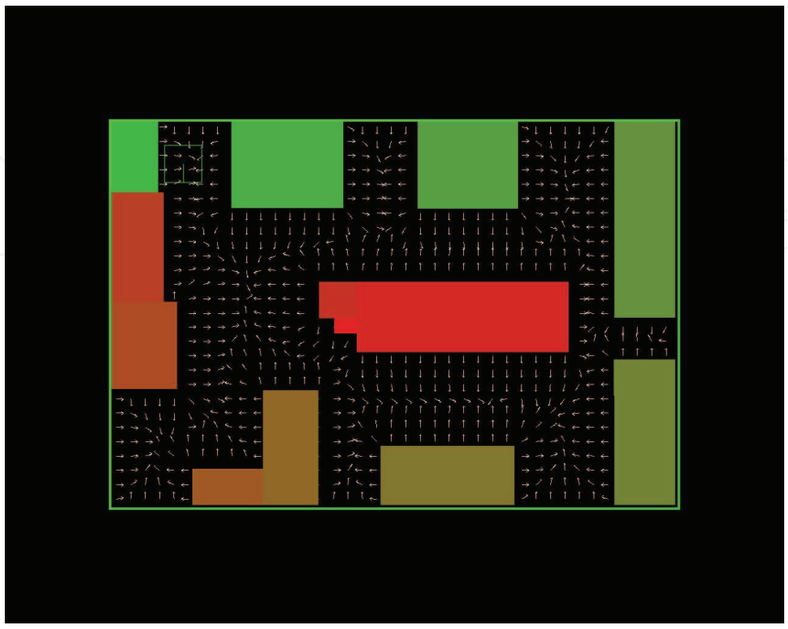


Figure 8. Repulsion force map for the environment shown in Fig. 6

By calculating in advance the repulsion force map liberates the robot's processors to perform other tasks, then knowing the destination the attraction force is calculated in each of the cells and added to the repulsion force calculated before. Figure 9 shows the attraction and repulsions force map, in which a robot navigates from the upper left corner to the lower right one.

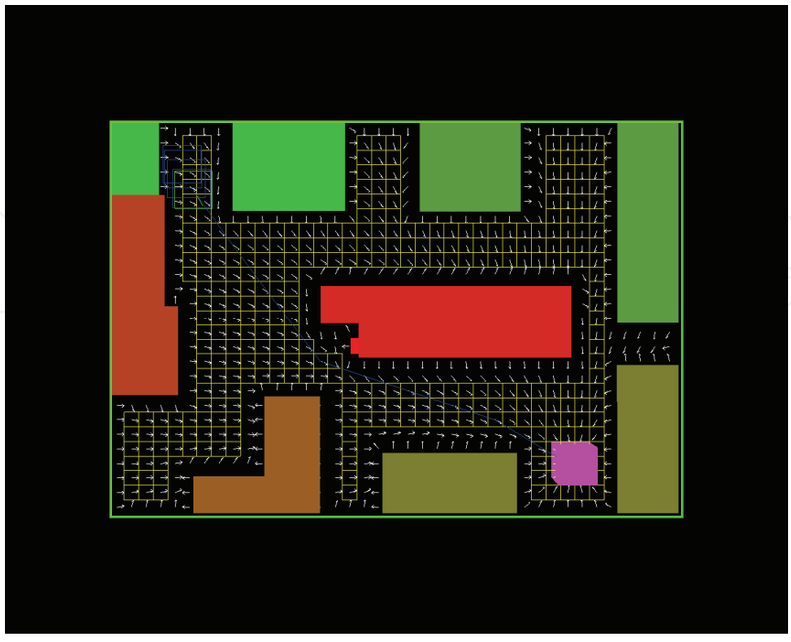


Figure 9. Attraction and repulsions force map

The use of this kind of repulsion and attraction force maps improves the performance of the robot, because it is not necessary to calculate for each of the robot positions the repulsion forces on-line.

4.2 Optimization approaches

4.2.1 Objective functions

An objective function for robot navigation should measure the optimality of a path between two points. The main criteria to determine the optimality are: minimum travel distance, and safe obstacle avoidance throughout the path. Then the objective function should provide optimum values (minima or maxima) for the shortest travel paths, with maximum distances to each obstacle in the path. Objective functions are usually constructed by the system developer, according to the navigation conditions: known or unknown obstacles; one or several attraction points; navigation map. To illustrate we present two objective functions which have been succesfully used for local obstacle avoidance.

As mentioned in the introduction Kun Hsiang et al. (1999), reported the development of an autonomous robot navigation scheme based on potential fields and the chamfer distance transform for global path planning in a known environment, and a local fuzzy logic controller to avoid trap situations. The chamfer distance transform produces a matrix where each entry is the distance to the closest obstacle, these distances are used to calculate the repulsion forces exerted by the obstacles on the robot. The attraction force of the goal point is a constant with a user defined magnitude. A fuzzy logic controller based on two objective functions was developed to avoid trap situations where the robot is not able to avoid an obstacle using only the potential field functions. The objective functions measure: the angle between the repulsive force of the closest obstacle and the resultant force (Ec. 22); the distance to the closest obstacle (Eq. 23). The controller tries to maximize the distance to the obstacles. An stop condition is used when the robot reaches the goal.

$$\varphi=\theta_{obs} - \theta \tag{22}$$

where:
θ_{obs} is the angle of the repulsive force;
θ is the angle of the resultant force.

$$diff=M(xi+1, yi+1)- M(xi,yi) \tag{23}$$

where:
M(xi+1, yi+1) is the distance to the closest obstacle at the next position;
M(xi,yi) is the distance to the closest obstacle at the current position.

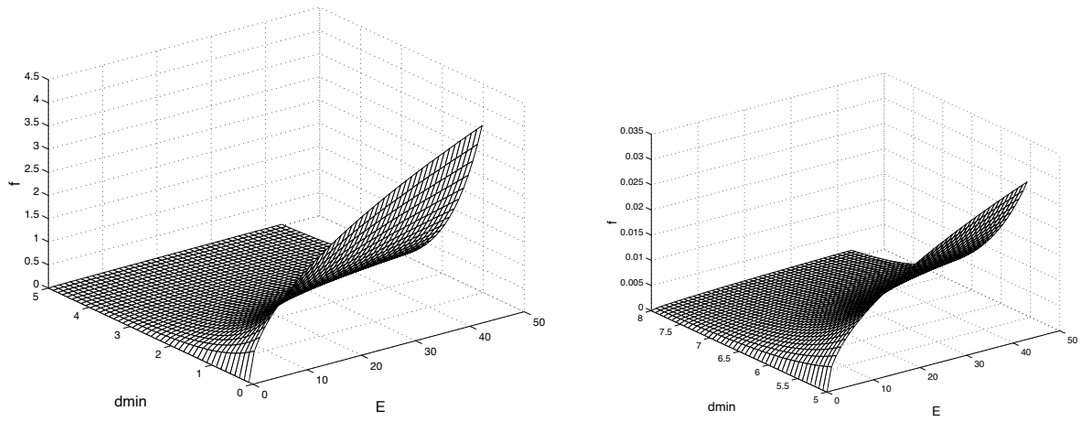
In Arambula and Padilla (2004) is reported an objective function to evaluate force field configurations which correspond to an optimum robot position (i.e. positions closer to the goal cell which also avoid obstacles). The objective function value of each candidate force field configuration is evaluated with two criteria: minimisation of the error distance E between the robot and the goal cell; and maximisation of the distance d_{min} to the closest obstacle cell. Equation 24 shows the objective function, which produces optimum (minimum) values for minimum E, and maximum d_{min}

$$\mathrm{f}(\mathbf{q}) = \begin{cases} \sqrt{E/2} \cdot e^{-d_{\min}} & d_{\min} > 0 \\ 2000 & d_{\min} = 0 \end{cases} \tag{24}$$

where:
d_{min} is the distance to the closest obstacle cell

$$E = \left| q_{rx} - q_{gx} \right| + \left| q_{ry} - q_{gy} \right|$$

$\mathbf{q}_r$ is a candidate cell for the new robot position;
$\mathbf{q}_g$ is the goal cell;
The construction of the objective function (f) favors robot paths that run away from the obstacles and result in decreasing distance to the goal cell. The case where d_{min}= 0 (which corresponds to a collision) is severely penalised. In Fig. 5a is shown the plot of Eq.24 for: 0<=E<=44 and 0.1<= d_{min}<=5.



(a) (b)

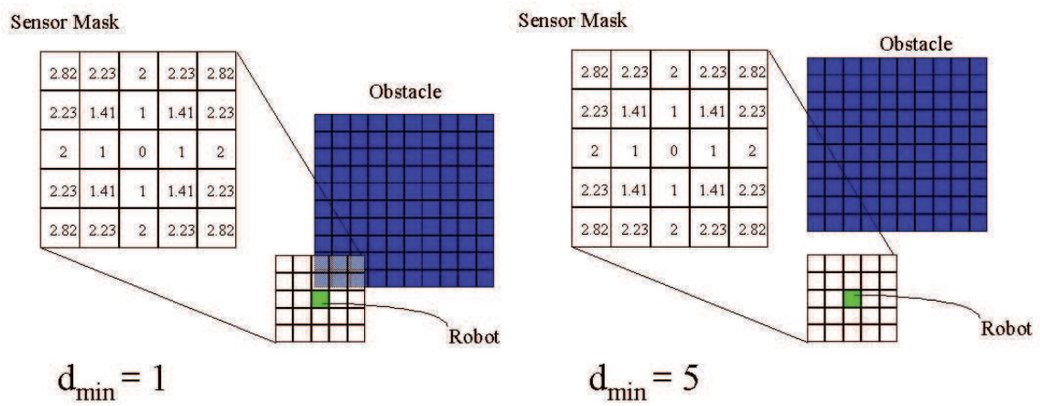
Figure 5.(a) Plot of Eq.24 for: 0<=E<=44, 0.1<= d_{min}<=5; (b) Plot of Eq.24 for: 0<=E<=44, 5<= d_{min}<=8

As shown in Fig. 5a, f gives non-optimum high values for small d_{min} and large E, although smaller values of f can be achieved through increased d_{min} or smaller E, the absolute optimum value of f=0 will only be achieved for E=0. In Fig. 5b is shown the plot of f in the range: 0<=E<=44, 5<=d_{min}<=8; as can be observed, at a predefined maximum value of d_{min}=5, f still shows an slope which guarantees that optimum values correspond to decreasing E.

4.2.2 Adaptive potential fields

If the robot navigates in an environment with unknown obstacles it is necessary to detect and avoid obstacles as the robot moves towards the goal. In Arámbula and Padilla (2004) was reported an scheme for online obstacle detection. The robot is represented as a particle

R that moves in the configuration space C, modelled as a two dimensional grid, where each cell c_i inside C can be occupied by the robot, the goal or the obstacles. There is also an associated obstacle map M of the same size of C. The obstacle map is initially empty, and it is filled at the positions of the obstacles detected by the robot, as it moves inside C. The goal cell, and 4 auxiliary attraction points exert an attraction force on *R* given by Eq. 12, while each of the detected obstacle cells exerts repulsion forces given by Eq. 13. For obstacle detection, a $5x5$ grid simulates the robot sensors. When R moves, the positions of the sensors in the mask are updated and used to calculate the distance $d_{\min}$ to the closest detected obstacle (Fig. 10a). A predefined distance is assigned to obstacles outside of the detection mask, as shown in Fig. 10b.



(a) (b)

Figure 10. Examples of obstacle sensing. a) The robot detects an obstacle at $d_{\min} = 1$; b) The robot does not detect any obstacle and sets $d_{\min}$ to a predefined value of 5

In order to avoid trap situations or oscillations in the presence of large or closely spaced obstacles (Koren and Borenstein, 1991), 4 auxiliary attraction points have been placed around the goal cell (Fig. 3). Each attraction force $\mathbf{F}_{att}^{i}$ located at cell c_i, depends on the corresponding value of ξ_i, which is automatically adjusted by a genetic algorithm described in the next section. The effect of auxiliary attraction points was evaluated in two modalities: (1) auxiliary points placed at a fixed distance (of 15 cells) from the goal cell; and (2) auxiliary points placed at a variable distance (between 0 and 15 cells), which is adjusted automatically by the GA. Results from both approaches are reported in section 4.3. Use of auxiliary attraction points with a force strength and position automatically adjusted by the GA, allows for the generation of resultant force vectors which enable the robot to avoid large obstacles, as shown in Fig. 12.

4.2.2.1 Adaptive field optimization using genetic algorithms

Genetic algorithms are an efficient technique to optimise difficult functions in large search spaces. By testing populations of solutions represented as strings (called chromosomes) in an iterative process, a GA is able to find a near optimal solution in a robust manner, with the ability to produce a "best guess" from incomplete or noisy data (Goldberg, 1989) . A GA was used to optimise the values of the variables (ξ_i) of 5 attraction points and the values of the variables (η_j) of up to 155 obstacle cells. Each variable has a range of {0, 1000} and was binary coded with 20 bits of resolution in order to maintain a large number of values for the repulsion and attraction forces. A chromosome is formed by concatenation of the 160 binary coded variables. As mentioned above two modalities of the approach were evaluated: (1) with auxiliary attraction points placed at fixed positions, and (2) with auxiliary attraction points placed at variable distance from the goal cell. To implement modality (2), four additional binary variables in the range {0, 15} and coded with 4 bits each, are included in the chromosomes.

The GA searches for optimum values of ξ_i and η_j in a given binary string (chromosome) which move the robot to a position such that *f* (Eq. 24) has a minimum value. Only those η_j which correspond to obstacle cells detected by the robot are used to calculate the force fields given by Eq. 13, the rest of the repulsion weights in the string is ignored. At each generation of the GA, every chromosome in the current population is decoded and the value of Eqs. 12, and 13 is calculated, with this values is calculated the resultant force and the corresponding robot position. This robot position is evaluated with Eq. 24 and assigned a selection probability based on its objective function value (smaller values of the objective function correspond to higher selection probabilities). Each chromosome in the current population is assigned a number of copies with probability P_s using stochastic universal sampling (SUS) for selection and the ranking method to assign probabilities (Chipperfield et. al, 1995). Single point crossover is applied to the copies (offspring) with a probability of 0.6, mutation is applied to each string with a probability of 0.01 per bit. Finally, the next generation of the GA is formed using fitness based reinsertion with a generation gap of 0.8. This process continues until the robot reaches the goal cell or 200 generations (robot steps) are completed. Below is shown the pseudocode of the GA for robot navigation.

```
Pop= Random initialisation of 50 binary chromosomes
Step_count=0
While step_count<200
        Calculate Fatt (Eq.8), Frep (Eq.9), and the next robot position for each chromosome
        in Pop;
        Calculate f (Eq.10) for each robot position ;
        Assign a probability of selection (Ps) to each chromosome using the ranking
        method;
        Assign copies to each chromosome using SUS with probability of selection Ps;
        Mutate and cross the copies (offspring);
        Reinsert offspring in Pop with a generation gap of 0.8;
        Calculate f for Pop;
        Select best chromosome and move the robot to the corresponding position;
```

```
Increment step_count;
If(d=0)
        finish
end
```

4.3 Potential Field Optimization in a Partially known environment: Experiments and results

The genetic algorithm described above was implemented in Matlab using the GA toolbox developed at the University of Sheffield (Chipperfield et. al, 1995). A cell map of 40x40 cells simulating a five-room floor was used for evaluation. Random obstacle distributions were used, as shown in Fig. 11.

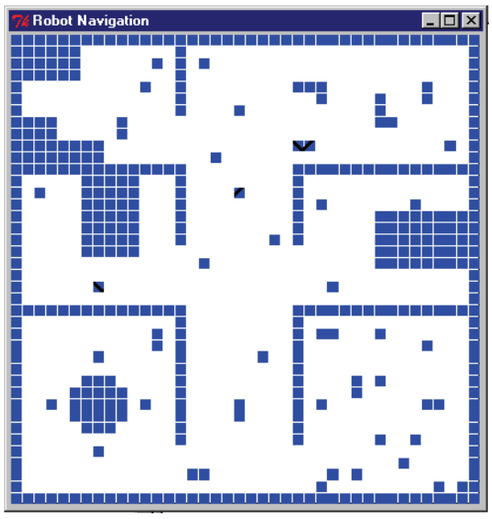


Figure 11. Cell map simulating a five-room floor with random obstacle

Ten experiments were performed, the start and goal positions for each experiment are shown in table 1, the origin is placed at the top-left corner of the cell map. Two intermediate goal points have been used to guide the robot through the corridor corner as well as through the door of the appropriate room. The positions of the intermediate goal points are also shown in table 1. The robot travels from the start position to each successive intermediate goal point and to the final goal point.

Two modalities of the navigation algorithm were evaluated: (1) with auxiliary attraction point placed at fixed positions, and (2) with auxiliary attraction points placed at variable distance from each goal cell. In table 2 are shown the results of the 20 experiments performed, the first column shows the experiment number corresponding to table 1. Columns two and three show respectively, the total distance traveled by the robot (measured in cells), and the deviation (as a percentage) from the optimum shortest path. Auxiliary attraction points were placed at a fixed distance of five cells from each goal position. Columns four and five show respectively, the total distance traveled by the robot and the deviation percentage, for auxiliary attraction points placed at a variable distance, which is automatically adjusted by the GA.

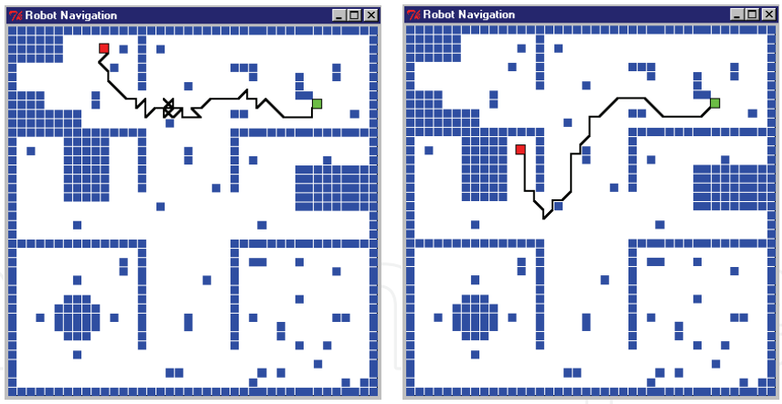
Exp. No	(start)-(goal)	intermediate goal 1	intermediate goal 2
1	(34, 9)-(11, 3)	(20,10)	(15,10)
2	(34, 9)-(13, 14)	(20,10)	(15,21)
3	(34, 9)-(3, 26)	(20,10)	(15,38)
4	(34, 9)-(36, 27)	(20,10)	(25,38)
5	(34, 9)-(37, 14)	(20,10)	(25,22)
6	(34, 4)-(3, 6)	(20,10)	(15,10)
7	(34, 4)-(3, 14)	(20,10)	(15,21)
8	(34, 4)-(12, 26)	(20,10)	(15,38)
9	(34, 4)-(30, 30)	(20,10)	(25,38)
10	(34, 4)-(38, 22)	(20,10)	(25,22)

Table 1. Start-goal and intermediate goal positions of each experiment

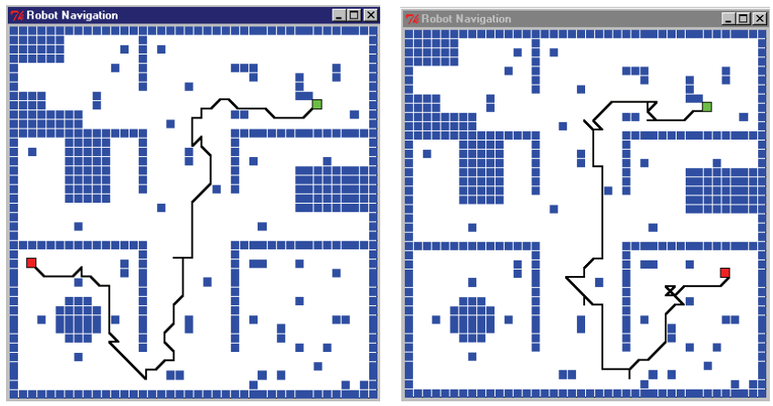
Exp.No.	Total distance 1 (cells)	Deviation from optimum 1 (%)	Total distance 2 (cells)	Deviation from optimum 2 (%)
1	34	17.2	48	65.5
2	44	29.4	34	0
3	collision	Collision	69	6.1
4	68	21.4	81	44.6
5	46	15.0	65	62.5
6	40	21.2	43	30.3
7	49	11.4	48	9.0
8	70	27.2	81	47.3
9	75	41.5	70	32.0
10	48	17.0	72	75.6
		Average: 22.3%		Average: 37.3%

Table 2. Experiment results: **Total distance 1**, and **Deviation from** optimum **1** obtained with auxiliary attraction points placed at fixed distance (five cells) from the goal; **Total distance 2**, and **Deviation from optimum 2** obtained with auxiliary attraction points placed at variable distance from the goal

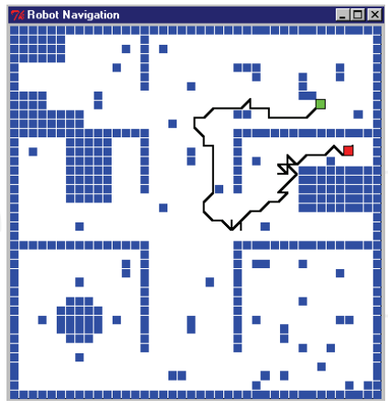
From the results shown in table 2, the average deviation from the optimum path length is larger (37% vs. 22%) for the second approach, this is most likely because we have a larger and more complex search space which results in a higher probability of suboptimal points being chosen by the GA. However the second approach was able to produce a feasible path without collisions for all the experiments. In contrast the first approach (using fixed auxiliary attraction points) was not able to reach the goal for experiment 3. In Fig. 12 are shown five paths produced by the second approach. The average time for path completion on a Pentium III PC at 750MHz is 115s with an average path length of 56 cells (i.e.2.05 s/step).



(1) (2)



(3) (4)



(5)

Figure 12. Paths produced by the navigation algorithm, using auxiliary attraction points placed at variable distance from the goal cell. Start-goal positions are as given in table 1

5. Hybrid Approaches to Recover from Local Minima

Hybrid approaches can be used to modify a potential field configuration in which a local minimum has been detected, for example Fig. 13, shows a robot that found an obstacle in the middle of the path between the origin and the goal and it is oscillating back and forth, due to the repulsion and attraction forces. First the repulsion forces repealed the robot from the obstacle, and when the robot is a little far away from it, the attraction force pushed it back to the obstacle, and then the repulsion force acts again repeating the whole process.
The potential field configuration can be modified by the addition of attraction forces that allow the robot to exit the local minima. By using the position of the known obstacle, additional attraction forces are added in places that will take the robot out of the local minimum. Usually additional attraction points are added in some of the vertices of the obstacles, as is shown in Fig. 14.

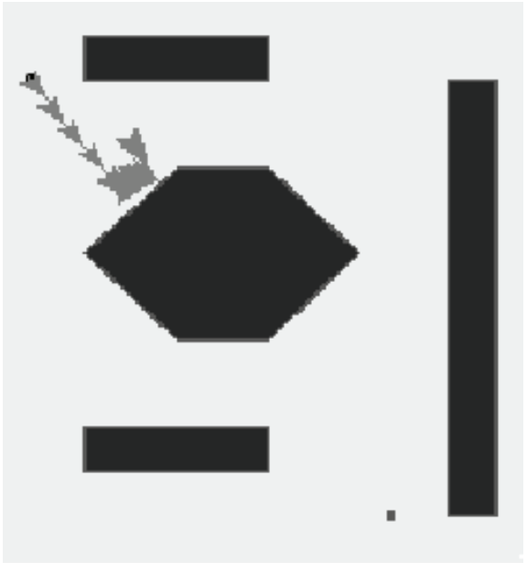


Figure 13. The robot is stuck in a local minimum

Basically the hybrid approach finds the obstacle in which the robot got stuck, then using its vertices $V=(v_1,v_2,...,v_N)$ it selects the vertices v_i, $v_{i+1},...,v_{k-1},v_k$, where v_i is the closest vertex from the stuck point, v_{i+1} is the clockwise vertex from v_i and v_k is the closest vertex to the goal. Using these selected vertices the approach places a new goal to reach at v_{i+1} disabling the original goal, after the goal in v_{i+1} is reached a new goal is issued at the next selected vertex and so on until v_k is reached. Finally the original goal is set again. In the Fig. 14 we can see that four additional attraction forces where added to the space to take the robot away from the local minimum.
There are cases in which this approach does not work because there are obstacles so large that can generate several local minima in which the robot can get stuck again. In this case another approach is to have a robotics behavioral architecture that consists of several behaviors in parallel (Arkin 1998), each of the behaviors generates an output according to the readings of the sensors connected to them and its internal state. Then a referee selects the output of one of the behaviors according to a selection mechanism and sends it to the robot's actuators. Figure 15 shows this type of architecture with two behaviors, one with potential fields and the other with an state machine.

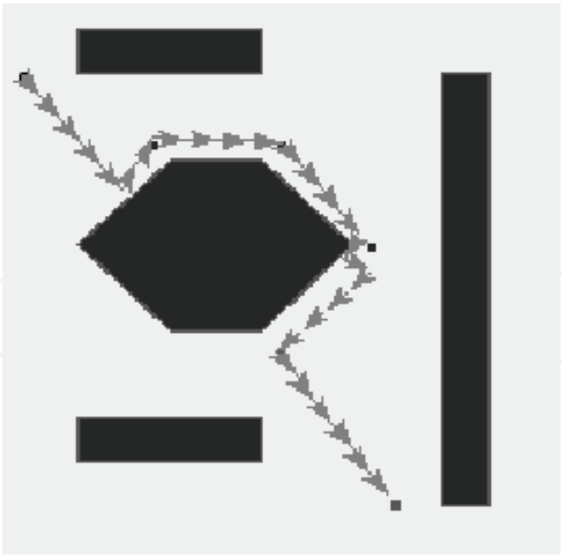


Figure 14. Four additional attraction forces are added to the environment to take the robot out of the local minima

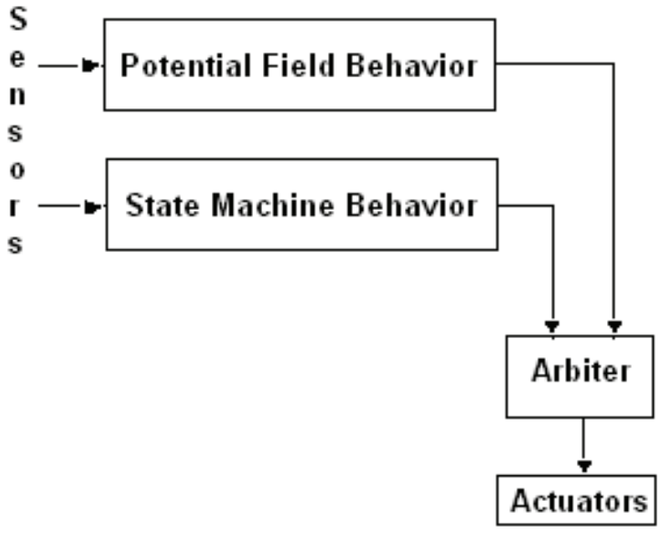


Figure 15. Behavior architecture used to control the movements of a robot

The function of the state machine behavior is to detect when the robot gets stuck in a local minima and take it out of it. After it takes the robot out of the local minima the referee selects again the potential field behavior. Figure 16 shows the behavior that the robot follows to avoid an obstacle. When the robots senses an obstacle in the left or in the right it will go backward first and then turn to the right or to the left accordingly, if it finds the obstacle in front of it, it goes backward then turns to the left 90 degrees, goes forward and then turns to the right and forward again. This simple behavior allows the robot to avoid local minima.

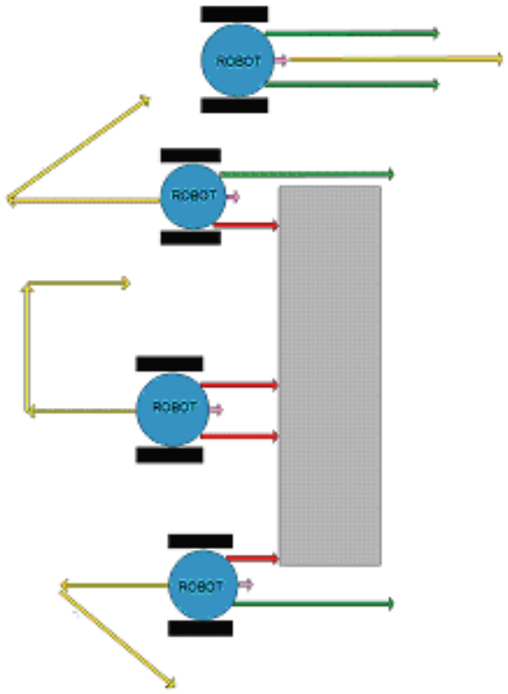


Figure 16. Robot behavior to take a robot out of a local minimum

6. References

Arámbula Cosío F. and Padilla Castañeda M.A. (2004), Autonomous robot navigation using adaptive potential fields, *Mathematical and Computer Modelling*, Volume 40, Issue 9-10, 1141-1156 .

Arkin R.C. (1998), Behavior-Based Robotics, Cambridge, MA: The MIT Press.

Borenstein J. and Koren Y. (1991), The vector field histogram-fast obstacle avoidance for mobile robots, *IEEE Transactions on Robotics and Automation 7,* 278-288.

Canny J.F. and Lin M.C. (1990), An opportunistic global path planner, *IEEE Int. Conf. on Robotics and Automation*, 1554-1559.

Chipperfield A., Fleming P., Pohlheim H. and Fonseca C. (1995), *Genetic Algorithm Toolbox for Use with* MATLAB *User's Guide,* Automatic Control and Systems Engineering, University of Sheffield, U.K.

Connolly C.I., Burns J.B., and Weiss R (1990), Path planning using Laplace's equation, *Proc. lEEEConf. on Robotics and Automation,* pp 2102-2106, Cincinnati, OH.

Ge S.S. And Cui Y.J. (2002), "Dynamic motion planning for mobile robots using potential field method", Autonomous Robots, 13, 207-222.

Ge S.S. and Cui Y.J. (2000), New Potential Functions for Mobile Robot Path Planning, *IEEE Transactions on Robotics and Automation*, vol. 16, No. 5, pp.615-620.

Goldberg D.E. (1989), *Genetic Algorithms in Search, Optimisation, and Machine Learning,* Addison-Wesley, MA.

Grefenstette J. and Schultz A.C. (1994), An evolutionary approach to learning in robots, In *Proceedings of the Machine Learning Workshop on Robot Learning, Int. Conf. on Robot Learning,* pp. 65-72, New Brunswick, N J.

Guldner J., Utkin V., Hashimoto H. (1997), Robot Obstacle Avoidance in n-Dimensional Space Using Planar Harmonic Artificial Potential Fields, *Journal of Dynamic Systems, Measurement, and Control*, vol. 119, pp. 160-166

Keymeulen D. and Decuyper J. (1994), The fluid dynamics applied to mobile robot motion: the stream field method, *IEEE Int. Conf. on Robotics and Automation*, 378-385.

Khatib O. (1990), Real-time obstacle avoidance for manipulators and mobile robots, In *Autonomous Robot Vehicles,* (Edited by I.J. Cox and G.T. Wilfong), pp. 396-404, Springer-Verlag.

Koren Y., Borenstein J., Potential field methods and their inherent limitations for mobile robot navigation. *In: Proceedings of the IEEE Int. Conf. on Robotics and Automation,* 1398-1404 (1991).

Kun-Hsiang Wu, Chin-Hsing Chen, and Juing-Ming Ko, Path planning and prototype design of an AGV. *Mathematical and Computer Modelling*, 30**,** 147-167 (1999).

Lozano-Pérez T. (1979), An algorithm for planning collision-free paths among polyhedral obstacles, *Communications of the ACM*, vol.~22 pp.~560--570.

Masoud A.A., Masoud S.A., and Bayoumi M.M. (1994), Robot navigation using a pressure generated mechanical stress field: the biharmonic potential approach, *IEEE Int. Conf. on Robotics and Automation*, 124-129.

McFetridge L. and Ibrahim M.Y. (1998), New technique of mobile robot navigation using a hybrid adaptive fuzzypotential field approach, *Computers Ind. Engng*. 35 (3-4), 471-474.

Schmidt G.K. and Azarm K. (1992), Mobile robot navigation in a dynamic world using an unsteady diffusion equation strategy, *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 642-647.

Shimoda S., Kuroda Y., Iagnemma K., (2005), Potential field navigation of high speed unmanned ground vehicles on uneven terrain, *IEEE Int. Conf. on Robotics and Automation*, Barcelona, Spain, 2828-2833.

Utkin V.I., Drakunov S., Hashimoto H., and Harashima F. (1991), Robot path obstacle avoidance control via sliding mode approach, *Proc. lEEE/RSJ Int. Workshop on Intelligent Robots and Systems,* pp. 1287-1290, Osaka, Japan.

Vadakkepat P., Kay Chen Tan, Wang Ming-Liang, Evolutionary artificial potential fields and their application in real time robot path planning. *Proc. of the 2000 Congress on Evolutionary Computation,* pp. 256-263, July (2000).

Barraquand, J., Langlois, B., Latombe, J.C., Numerical Potential Field Techniques for Robot Path Planning, *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 22, No. 2, pp. 224-241, March/April, 1992.

Wang, Y., Chirikjian, G., A New Potential Field Method for Robot Path Planning, *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 977-982, 2000.



Motion Planning

Edited by Xing-Jian Jing

ISBN 978-953-7619-01-5
Hard cover, 598 pages
Publisher InTech
Published online 01, June, 2008
Published in print edition June, 2008

In this book, new results or developments from different research backgrounds and application fields are put together to provide a wide and useful viewpoint on these headed research problems mentioned above, focused on the motion planning problem of mobile ro-bots. These results cover a large range of the problems that are frequently encountered in the motion planning of mobile robots both in theoretical methods and practical applications including obstacle avoidance methods, navigation and localization techniques, environmental modelling or map building methods, and vision signal processing etc. Different methods such as potential fields, reactive behaviours, neural-fuzzy based methods, motion control methods and so on are studied. Through this book and its references, the reader will definitely be able to get a thorough overview on the current research results for this specific topic in robotics. The book is intended for the readers who are interested and active in the field of robotics and especially for those who want to study and develop their own methods in motion/path planning or control for an intelligent robotic system.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Miguel A. Padilla Castaneda, Jesus Savage, Adalberto Hernandez and Fernando Arambula Cosío (2008). Local Autonomous Robot Navigation Using Potential Fields, Motion Planning, Xing-Jian Jing (Ed.), ISBN: 978-953-7619-01-5, InTech, Available from: http://www.intechopen.com/books/motion_planning/local_autonomous_robot_navigation_using_potential_fields

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2008 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License, which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen