Journal of Intelligent and Robotic Systems (2006) 45: 101–114
DOI: 10.1007/s10846-005-9008-2

Exploratory Navigation Based on Dynamical Boundary Value Problems

MARCELO TREVISAN[1], MARCO A. P. IDIART[1], EDSON PRESTES[2] and PAULO M. ENGEL[2]
[1]*Instituto de Física, UFRGS, P.O. Box 15051, 91501-970 Porto Alegre, RS-Brazil; e-mail: tmarcelo@if.ufrgs.br, idiart@if.ufrgs.br*
[2]*Instituto de Informática-UFRGS, P.O. Box 15064, 91501-970 Porto Alegre, RS-Brazil; e-mail: prestes@inf.ufrgs.br, engel@inf.ufrgs.br*

(Received: 9 September 2004; in final form: 1 June 2005)

Abstract. The paper presents a general framework for concurrent navigation and exploration of unknown environments based on discrete potential fields that guide the robot motion. These potentials are obtained from a class of partial differential equation (PDE) problems called boundary value problems (BVP). The boundaries are generated from sensor readings and therefore they change as the robot moves. This framework corresponds to an extension of our previous work (Prestes, E., Idiart, M. A. P., Engel, P. and Trevisan, M.: Exploration technique using potential fields calculated from relaxation methods, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2001, p. 2012; Prestes, E., Engel, P. M., Trevisan, M. and Idiart, M. A.: Exploration method using harmonic functions, *Robot. Auton. Syst.* **40**(1) (2002), 25–42). Here, we propose that a careful choice of the PDE and the boundary conditions can produce efficient exploratory behaviors in sparse and dense environments. Furthermore, we show how to extend the exploratory behavior to produce new ones by changing dynamically the boundary function (the value of the potential at the boundaries) as the exploration takes course. Our framework is validated through a series of experiments with a real robot in office environments.

Key words: autonomous navigation, boundary value problems, environment exploration.

1. Introduction

Potential fields are traditionally used in robotics for path planning (Rimon and Koditschek, 1988; Barraquand et al., 1992; Latombe, 1993; Khatib, 1996). The idea is to use the interplay between repulsion from obstacles and the attraction from the goal positions to produce the desirable behavior. There are two major forms of potential calculation: First, the potential can be calculated as the sum of the potentials produced by each individual obstacle and goal. Such superposition is of relatively low computational cost but suffers from local minima that could trap the robot. Second, the potential can be calculated from the solution of a

PDE, for instance the Laplace's equation (Kim and Khosla, 1992; Connolly and Grupen, 1993)

$$\nabla^2 p(\mathbf{r}) = 0 \tag{1}$$

where the obstacles and goals are treated as boundary conditions. The minima are no longer present but the computational cost can be large since every time a new obstacle is found the potential has to be recalculated by the relaxation of the whole map.

A successful exploratory behavior consists in a good combination of a search procedure with a path planning approach. The potential field method is capable of providing such combination. This problem has been widely studied within the scope of simultaneous localization and mapping (Thrun et al., 2000, 2003). In this paper we present a general framework for concurrent navigation and exploration based on our previous works (Prestes et al., 2001, 2002) that use partial relaxations of harmonic functions for the potential field. Our framework belongs to the second type of potential calculations described above, so as our original method.

In our original method (Prestes et al., 2001, 2002) the potential field indicates both the nearest attainable frontier and the path leading to it in a single calculation, in contrast with exploration algorithms where these tasks are performed in different modules (Yamauchi, 1997; Yamauchi et al., 1998; Rao et al., 1993; Feder et al., 1999; Yang et al., 2003; Solanas and Garcia, 2004) and time can be wasted unnecessarily with unreachable frontiers.

Another important property of the method is that the potential calculation is weakly dependent on the complexity of the environment since the associated relaxation algorithm depends on the number of cells in the grid representing the environment and not on the specific placement of obstacles on it. Bug like approaches (Lumelsky and Skewis, 1990; Lumelsky and Stepanov, 1986) focus on completeness rather than optimality and tend to deteriorate as the intricacy of the environment mounts. They guarantee to find a path if it exists but pay little attention to minimizing its length. Our method produces short and smooth[★] exploratory paths reducing significantly the odometric errors and the hitting probability in presence of noise. This is an advantage upon cell decomposition methods or wavefront methods (Batavia and Nourbakhsh, 2000; Connolly and Grupen, 1993; Connolly, 1994).

The specific contributions of this paper are:

– We demonstrate that the BVP field method can be generalized beyond the harmonic function method, keeping its original properties while introduc-

[★] Almost all paths produced by the gradient on a potential calculated from a BVP are infinitely differentiable along their length. An exception is a saddle point which still is piecewise differentiable (Sabersky et al., 1971).

ing new flexibility to deal with different environment statistics such as sparse and dense environments.
– We show that a variety of new exploratory behaviors can be generated by dynamically changing the boundary function (the value of the potential at the boundaries) as the exploration takes course.

The paper is organized as follows. In the second section, we revisit the harmonic method for exploration. In the third section we show the generalization of the Laplace's equation and the qualitative effects of the changes in the navigation field and its applications. In the fourth section we introduce the idea of exploration driven by boundary features. In the fifth section we discuss the implementation of the method in its many forms. In the sixth section we present experimental and the simulation results. Finally, in the last section, we discuss the conclusions.

2. Harmonic Potential Fields for Exploration

Given an environment with obstacles and a desired goal position, the harmonic method for path planning is implemented as follows: In the space of independent degrees of freedom (the c-space), the boundary of the BVP is defined such that on the obstacle surfaces the potential field value is set to 1 while in the surface, or point, defining the goal position it is set to 0. Then the equation (1) is calculated. Given an initial point $\mathbf{r}(0) = \mathbf{R}_0$ the planner returns as the path the gradient descent curve on $p(\mathbf{r})$ starting at $\mathbf{R}_0$ and ending at the goal.

The extension of the harmonic function method to exploration is straightforward. The BVP is calculated only in the explored region. This region is limited by two types of boundaries, either they are surfaces of obstacles or they are frontiers of exploration. The frontiers are free-space limits that indicate that no sensor readings were acquired or considered beyond that point, see Figure 1.

The solutions of the Laplace's equation do not possess local minima (maxima).[★] It allows the existence of saddle points, but saddle points are unstable points and do not represent any problem in real world robotics since trapping by saddle points can only occur in infinite precision dynamics.

[★] This can be seen if we use Gauss' divergence theorem for the field $\mathbf{E} = -\nabla p$. Gauss' theorem reads in two dimensions

$$\int_A \nabla^2 p \, \mathrm{d}A = \int_c \nabla p \cdot \widehat{\mathbf{n}} \, \mathrm{d}\ell$$

where c is any closed line on the plane, internal to the boundaries, the vector $\widehat{\mathbf{n}}$ its normal vector, and A its area. Laplace's $\nabla^2 p = 0$ equation implies (see the right hand side) that the flux of trajectory lines through any closed line vanishes. It indicates that trajectories that enter a given region, internal to the boundaries, always leave that region, and therefore local minima are absent.

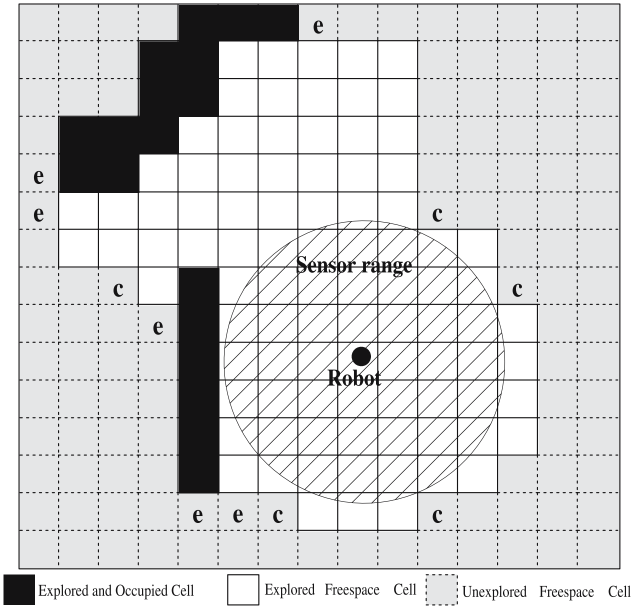

Figure 1. Region of the environment where the BVP is calculated. Searching for edges of walls, regions labeled as **e**, represents a wall-following exploring behavior and for unexplored concave regions, labeled as **c**, it corresponds to a space-filling exploring behavior (see Section 6.3).

A gradient descent dynamics in such potential generates paths that are analogous to lines of force in *Classical Mechanics* such that the force is $\mathbf{F} = -\nabla p$, with the following properties (Feynman et al., 1972):

(1) They are open curves, i.e., they do not present closed loops. This is clearly seen since the vector field is derived from a potential *(conservative field*) and therefore the curl vanishes ($\nabla \times \mathbf{F} = 0$).
(2) They connect points in the obstacle surface (*high potential*) to points in the frontier (*low potential*). This is consequence of the fact that local minima or maxima are not allowed by the equation that defines the PDE.
(3) Two lines of force never cross each other. This is also a consequence of the fact that the force field is derived from a potential. The potential is an analytical function and has the gradient defined in all its points.

The properties of the lines of force guarantee that the robot when following the gradient always moves towards a frontier with an unexplored region, if it exists. Once the robot attains its goal, transposing the frontier, the explored region grows larger and the frontier recedes.

3. Beyond Harmonic Functions

An approximation for the solution of the Laplace's equation in a generic environment is obtained by iteration of

$$p(\mathbf{r}, t + \Delta t) = \sum_{\mathbf{r}' \in \aleph(r)} \frac{1}{4} p(\mathbf{r}', t) \tag{2}$$

where $\mathbf{r}$ are points in a square lattice, and the sum is over the four next neighbors of each point, represented by the set $\aleph(\mathbf{r})$. Depending on how the time is updated (synchronously or asynchronously) this procedure is called Jacobi's Method or Gauss–Seidel Method, with different convergence performances. This expression is a particular instance of the more general master equation

$$p(\mathbf{r}, t + \Delta t) - p(\mathbf{r}, t) = \sum_{\mathbf{r}' \in \aleph(\mathbf{r})} p(r', t)\omega_{\mathbf{r}'\mathbf{r}} - p(\mathbf{r}, t) \sum_{\mathbf{r}' \in \aleph(\mathbf{r})} \omega_{\mathbf{r}\mathbf{r}'} \tag{3}$$

that describes a Markov process of a random walker in a square grid. $p(\mathbf{r}, t)$ is interpreted as the probability to find a particle in the site $\mathbf{r} = (i, j)$ at time t, and $\omega_{\mathbf{r}\mathbf{r}'} = \omega(\mathbf{r}|\mathbf{r}')$ is the transition probability, or the conditional probability that if the particle is in state $\mathbf{r}$ at time t, it will jump to site $\mathbf{r}'$ at time $t + \Delta t$.

Considering that the transition probabilities are independent on the particular site the walker is on, we can write a more general relation for the probability at the steady state

$$p(\mathbf{r}) = \sum_{\mathbf{r}' \in \aleph(\mathbf{r})} p(\mathbf{r}')\omega_{\mathbf{r}'\mathbf{r}} \tag{4}$$

where the only constraints over ω are that $\omega_{\mathbf{r}\mathbf{r}'} > 0$ and $\Sigma_{\mathbf{r}'} \omega_{\mathbf{r}\mathbf{r}'} = 1$. As $p(\mathbf{r})$ at $\mathbf{r}$ is the weighted average of the probability (potential) values at its neighbor sites it is straightforward to prove that $p_{\min} \leq p(\mathbf{r}) \leq p_{\max}$ where $p_{\min}$ and $p_{\max}$ are the minimum and maximum values of p on the neighborhood of $\mathbf{r}$. Therefore there is always a nearby value of p lower or greater than the value at $\mathbf{r}$. In other words, it shows that for the generalized equation (4) local minima are also absent.

In order to illustrate the effect of the generalization equation (4) consider the asymmetric case $\omega_{\text{left}} \neq \omega_{\text{right}}$ and $\omega_{\text{up}} \neq \omega_{\text{down}}$. It reflect a bias in the displacement of the walker in the direction of the current

$$\mathbf{v} = \left(\omega_{\text{right}} - \omega_{\text{left}}, \omega_{\text{up}} - \omega_{\text{down}}\right).$$

Figure 2 shows the gradient descent in this biased potential (*dashed arrows*), for $\mathbf{v} = (1, 1)$, and the gradient descent of the solution of the Laplace's equation (1)

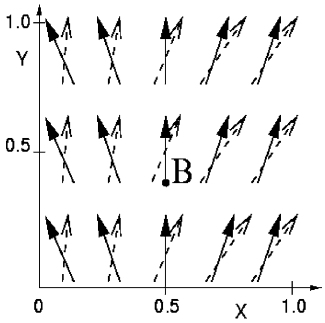

Figure 2. Gradient descent of the solutions of the equation (4) (*dashed arrows*) and the Laplace's equation (*solid arrows*) in a *square region* $L \times M$ where $L = M = 1$.

(*solid arrows*), for the conditions that $p(x, 0) = 1$, $p(x, M) = 0$, $p(0, y) = 0$ and $p(L, y) = 0$, which represents a square region with a wall (*obstacle*) at the bottom ($y = 0$) and three frontiers of exploration (*free-space limits*) at the *top* ($y = M$), at *left* ($x = 0$) and *right* ($x = L$).

The contribution of the bias can be understood supposing that a robot is placed at point **B** (see Figure 2). At this point the gradient descent of the Laplace's solution points straight to the top, so it guides the robot away from the obstacle. In the left and the right neighborhood of **B** the gradient points slightly to the left and the right side of the environment respectively. This symmetry comes from the boundaries. Now, considering the gradient descent of the solution of the equation (4), we see that at the point **B** the gradient points to the right side of the environment. The axial symmetry around $x = 0.5$ axis is broken. In this case the symmetry of solution is not determined exclusively by the boundaries but also by the direction of vector **v**. In the experiments section we discuss the impact on the robots behavior by this symmetry breaking (see Section 6.1).

4. Exploration Driven by Boundary Features

Environmental features were found to be important sources of spatial information for rats foraging for food (O'Keefe and Burgess, 1996). These features that take in fixed and mobile obstacles, walls and goals are geometric determinants of the environment boundaries. Although it does not seem so obvious at first sight, the boundaries are good cues to orient the exploration of an unknown environment and to construct routes between locations of interest (Smith and Husbands, 2002). By selecting different features in the boundaries we can generate different strategies for the complete exploration of the environment. These strategies are associated with the potential values attributed to the different boundary configurations used to calculate the potential in (3). For example, consider the environment represented in Figure 1. Regions labeled as '**e**' are edges of walls. When the robot explores these regions first it should exhibit a wall-following behavior. But if the robot avoids the walls seeking to fill unexplored concave regions, like the ones labeled as '**c**,' it should exhibit a space-filling behavior.

As we stated above, the behavior of the robot is given by the boundary conditions, where the boundary are held at a potential defined by some function along the boundary, i.e.,

$$p(\mathbf{r}) = f(\mathbf{r}) \quad \text{for} \quad \mathbf{r} \in \partial\Gamma \tag{5}$$

where $\partial\Gamma$ is the borderline between the explored and unexplored space of the environment Γ. Wall-following exploring behavior is achieved just setting a low potential ($f(\mathbf{r}) = 0$) in wall edges and their neighborhood. On the other hand, space-filling exploring behavior is achieved setting low potential in unexplored concave regions. In Section 6.3, we explore this flexibility to generate two distinct behaviors: wall-following and space-filling.

5. Implementation

We have implemented our approach in the Nomad 200 platform (Nomadic). The Nomad 200 is an integrated system with six sensory modules: tactile, infrared, ultrasonic, basic vision, structured light vision and compass systems. We used only its sonar capabilities, which is comprised of 16 channels that give range information from 0.5 m to 6.5 m.

The robot stores an internal $N_x \times N_y$ grid. Each grid cell represents a $\Delta L \times \Delta L$ real-world square area and stores the following attributes:

- potential ($p(\mathbf{r})$): represents the potential value in the cell centered in position $\mathbf{r} = (i, j)$;
- state: indicates the cell situation, which can be: *not explored*, *free-space* or *occupied*. The attribute value *not explored* indicates that the cell was never visited and its potential value is set, according to its neighborhood, to a value between 0 and 1. The value *free-space* indicates that the cell was explored, and it is free, therefore its potential value can be updated. The attribute value *occupied* indicates that the cell is occupied by an object in the real-world and its potential value is set to 1.
- certainty ($c(\mathbf{r})$): represents the certainty value for the cell centered in the position $\mathbf{r} = (i, j)$. This value is updated using the HIMM method proposed by Borenstein (Borenstein and Koren, 1991). The larger this value, the greater the certainty in the presence of the object.

After discretization the equations (1) or (4) are iterated numerically using the Gauss–Seidel method (Prestes et al., 2001).

6. Experimental Results

In this section we present a series of results obtained using the Nomad 200 robot. The experiments were performed on a Celeron 475 MHz running Linux over a radio Ethernet.

A traditional problem with the use of harmonic functions is its high computational cost. Even though the iteration equations are fairly simple the convergence of the potential values are considerable slow. The number of iterations (n) required to converge the potential below an error $\epsilon < 10^{-p}$ for a potential region of M cells is approximately $n \simeq 1/4\ p\ M$ for the Gauss–Seidel method (Press et al., 1992). It scales with the number of grid points.

To improve the computation speed we interrupt the iterations before total convergence because, in the process of exploration, most of the time the robot is near to a frontier and explored cells close to this region suffer more boundary influence than farther cells, and consequently converge faster. In (Prestes et al., 2001, 2002), we showed that it gives excellent results if the relaxation method is Gauss-Seidel and it is enough to provide the directions for the robot movement.

In our experiments, we consider that the iteration rate is 20 iteration/s★ and the robot vision radius is equal to 2 m.

6.1. DENSE ENVIRONMENTS VERSUS SPARSE ENVIRONMENTS

We have demonstrated that exploration with harmonic functions works very well in dense environments (Prestes et al., 2002). We consider dense an environment in which there are always obstacles under the sensors range of the robot. For sensors ranges of order of 2 m office environments are typically dense environments. In (Prestes et al., 2002) we show exploratory behaviors covering office environments of order of 200 m^2 using grid sizes of 100 $\times$ 100 cells, where each cell corresponds to a squared region of 15 cm $\times$ 15 cm in real space. Typical exploration times are 5 min with an average speed of 12.7 cm/s and path lengths very close to the optimal.

For sparse environments,★★ however, harmonic functions are no longer a good choice. Here we show that the flexibility on the choice of the PDE that generates the exploratory behavior can help to improve performance.

The first experiment was performed in a sparse 14 m $\times$ 16 m rectangular environment with a central column and two small obstacles close to opposite corners. Each grid cell was chosen to correspond to a region of 50 cm $\times$ 50 cm.

★ The rate indicates that the potential is updated 20 times by second; after each interval the robot adjusts its head to the direction pointed by the gradient of its current position.

★★ In a sparse environment the robot can travel many steps without sensing any obstacles. The sparseness of an environment adds a degree of complexity to the exploration process. Without guidance from the walls the robot have to decide the sequence of steps that leads to efficient exploration of empty spaces.

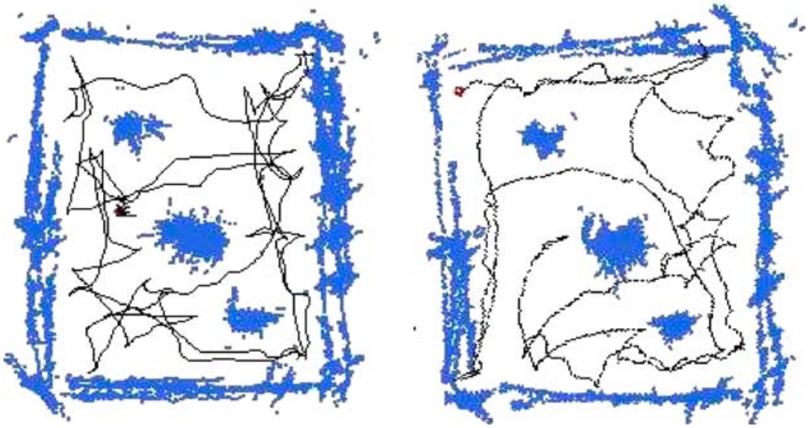

Figure 3. Examples of exploration sequences using the harmonic potential and the biased potential with $\mathbf{v} = (0.7, 0.7)$. For the harmonic trial, $L = 194.9$ m and for the biased, $L = 156.1$ m.

The original form of the room is slightly distorted due to odometric errors that arise from excessive changes in the robot's direction of movement.

Figure 3 shows the results of two experimental trials for exploration using harmonic potential equation (1) and the modified equation (4) (called *biased potential*) introduced in Section 3, with $\mathbf{v} = \epsilon(\cos\theta, \sin\theta)$, where $\epsilon = 1$ and $\theta = 45°$. In the experiments, we observed that the exploratory performance is not affected by the choice of θ. The robot trajectories and the history of sonar readings throughout the entire experimental time are simultaneously shown.

Repeating the experiment four times with the robot starting at different positions in the environment we obtained an average path length of $L = 185.5(1 \pm 0.08)$m using the harmonic method and $L = 140.3(1 \pm 0.18)$m using the asymmetric equation, what indicates an improvement of 30% in performance.

6.2. EXPLORING DYNAMIC ENVIRONMENTS

The main goal of an exploratory process is to represent the real environment consistently in a map. In general this consistence is difficult to be maintained in the presence of dynamic obstacles. These obstacles demand the robot controller a short reactive time, a fast mapping process and, to still produce smooth trajectories, to reduce the odometric errors.

In this section, we present some results obtained exploring a room at our Lab using the Nomad Robot. The robot was in charge of mapping the environment while avoiding dynamic obstacles, like people. The system treats moving obstacles in a similar way as static obstacles. The main difference being that they are represented over a short period of time on the HIMM grid. When the robot revisits the environment, it observes the sensor readings and updates internally

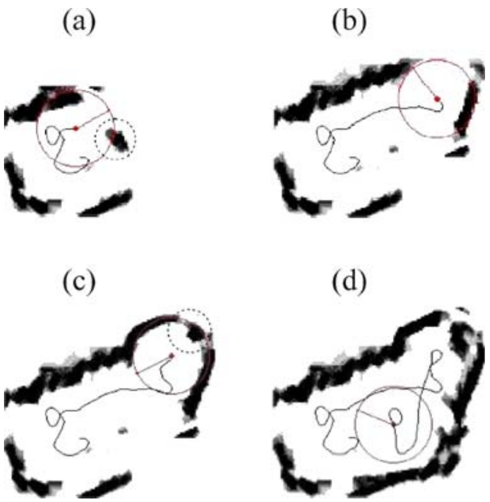

Figure 4. Experiment in an environment with dynamic obstacles. Snapshots of the mapping process of a room with people moving freely and their possible effect in exploration. People when detected are illustrated as *dashed circles*.

the new environment configuration releasing the free positions once marked as obstacles and marking the positions occupied currently.

The environment corresponds to nearly 7 m $\times$ 18 m. The average speed and the vision radius of the robot are 25.4 cm/s and 2 m, respectively. The sensors are read every 500 ms,[★] which permits to identify any changes in the environment; and we used the partial relaxation already proposed in (Prestes et al., 2002), which produces fast reactions to unexpected events.

Figure 4 shows a sequence of snapshots of a typical exploration process. People are illustrated as *dashed circles*.

In Figure 4(a), a person was detected at the center the room leaving a trace in the map. But that was gradually erased during the subsequent exploration. The person detected in Figure 4(c) on the other hand left a more permanent trace since it caused the robot to change its direction of exploration, closing what could be a passage to an unexplored region.

6.3. EXPLORATION USING BOUNDARY FEATURES

We repeated the experiments shown in Section 6.1, but this time using the smart boundary conditions introduced in Section 4. The environment and the robot configuration are the same of the last experiment (see Section 6.1).

We consider the edges of walls and the unexplored concave regions as boundary features (see Figure 1). These features are implicitly associated to grid

[★] This value depends on the robot speed. The larger the robot speed the smaller should be the time between readings.

cells which form the **e**-regions and **c**-regions, respectively. The identification of **c** and **e** regions is made by a simple heuristic. The **e**-regions are identified by verifying if exists an unexplored cell near a free-space cell and near an occupied cell. The **c**-regions are identified by checking if exists an unexplored cell with at least half of its neighbors that are free-space cells.

Here, initially, the robot looks for **e**-regions. However, in sparse environments, it finds only **c**-regions. These regions are attractive and the robot pursues them. When an **e**-region is eventually found, the system sets to low potential the cells associated to it in order to attract the robot towards them. At the same time, the cells related to **c**-regions are no longer attractive and are considered as obstacles, i.e., they are updated to have high potential. Therefore, as seen in Section 4, the robot shows a wall-following behavior.

When all **e**-regions have been explored, the robot switches its behavior to space-filling behavior. The space-filling behavior is obtained setting the cells associated to **c**-region to low potential. Then, the robot pursues them until all the environment is completely explored or when it finds another **e**-region. In the latter case, the robot repeats the process described before.

In Figure 5, we observe initially a wall-following exploring behavior until the walls no longer attract the robot. Then, the robot behavior changes to space filling. The use of the boundary features improves dramatically the exploration of the sparse environment. The robot's stereotypical behaviors reduce odometric errors and consequently the map is cleaner than the map in Figure 3.

Using the boundary feature control, we obtained much better performances in contrast to the experiments shown in Figure 3 (see Section 6.1). In Figure 5, the robot travelled $L \approx 87.7$ m for exploring the environment completely, which

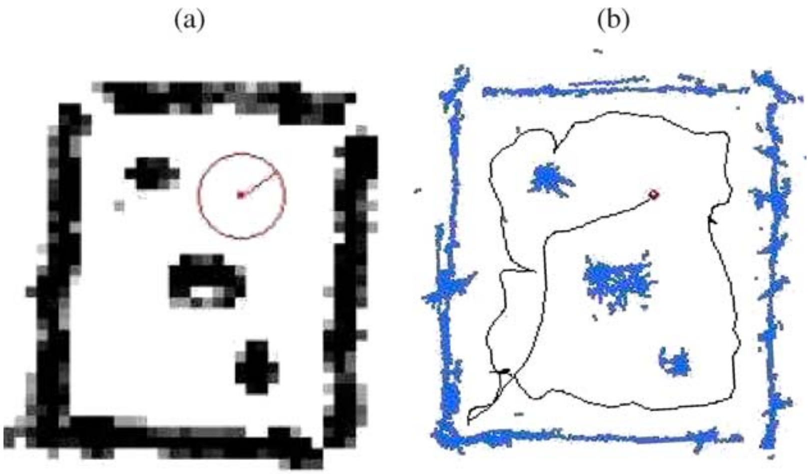

Figure 5. Exploration of a sparse environment using the boundary features concept. (a) Grid map obtained after exploration. (b) history of sonar readings throughout the entire experimental time.

corresponds to 56% of the path followed by the robot using biased potential. These results are preliminary and therefore, permit only a qualitative analysis. However, we can assert the benefits of using the boundary feature control. Currently, we are studying new spatial structures to be used as boundary feature, as well, new behaviors to be associated to them.

During the execution of the wall-following behavior the parameter ϵ is set to zero in equation (4), returning to the Laplace's equation, because the local environment configuration is similar to a dense environment. During the execution of the space-filling behavior, we consider the same value for v used in Section 6.1, since the environment is sparse.

7. Conclusion

In this work, we extend and generalize the harmonic method used to exploratory navigation proposed by authors in (Prestes et al., 2001, 2002). The generalized BVP is sound and preserves the mathematical properties of the harmonic method while keeping robustness and implementation easiness of the harmonic method. One of the great advantages of this potential calculation is the smooth trajectories generated from following the gradient of the potential. This smoothness implies in small odometric errors when the robot moves in a non-sparse environment with low speed (Prestes et al., 2004). Furthermore, it permits to treat adequately dynamic obstacles, like people, as seen in Section 6.2.

Furthermore, we proposed the use of dynamic boundaries to cause different behaviors of the robot. This is important when the environment configuration requires a multi-behavior robot. The great advantage of dynamical boundaries is that they can be integrated into different potential field methods where the field indicates paths toward unexplored regions of the space. In the future, we intend to apply this idea with stereo vision because it allows to extract several useful features that can be interpreted as dynamic boundaries.

A difficulty with potential strategies based on BVPs is the high computational cost of keeping the potential updated and relaxed during the progress of exploration. It has an impact in the controller performance when the potential regions grow. A possible solution is to use a more efficient grid sampling like quadtrees to account for inhomogeneous distributions of objects (Zelek, 1998) or partial relaxation.

Although we use particular equations to sparse and dense environments, the vector $\mathbf{v}$ allows a smooth transition between the Laplace equation and the equation (4) by adjusting just one parameter whenever it is required. For instance, when $\mathbf{v} = (0, 0)$, the equation (4) corresponds to the Laplace equation. In addition, our theory is thought to be directly applied to path planning task as well, i.e., we have a unique principle that provides a unified understanding and solution to the problems related to different tasks: path planning and exploration of unknown environments.

Acknowledgements

This work has been partially supported by Brazilian agencies CNPq, FINEP and FAPERGS.

References

Barraquand, J., Langlois, B., and Latombe, J.: Numerical potential field techniques for robot path planning, *IEEE Trans. Syst. Man Cybern.* **22** (1992), 224–240.

Batavia, P. and Nourbakhsh, I.: Path planning for the Cye Robot, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2000, pp. 15–20.

Borenstein, J. and Koren, Y.: Histogramic in-motion mapping for mobile robot obstacle avoidance, *IEEE J. Robot. Autom.* **7**(4) (1991), 535–539.

Connolly, C. I.: Harmonic functions and collision probabilities, *Int. J. Rob. Res.* **16**(4) (1994), 497–507.

Connolly, C. I. and Grupen, R. A.: On the application of harmonic functions to robotics. *J. Robot. Syst.* **10** (1993), 931–946.

Feder, H. J. S., Leonard, J. J., and Smith, C. M.: Adaptive mobile robot navigation and mapping. *Int. J. Rob. Res.* **18**(7) (1999), 650–668.

Feynman, R. P., Leighton, R. B., and Sands, M.: *The Feynman Lectures on Physics*, Vol. 2. Addison-Wesley, 1972.

Khatib, M.: Sensor based motion control for mobile robots. Ph.D. thesis, Laboratoire d'Automatique et d'Analise des Systmes, Toulouse, France, 1996.

Kim, J. and Khosla, P.: Real-time obstacle avoidance using harmonic potential functions, *IEEE Trans. Robot. Autom. Syst.* **8** (1992), 338–349.

Latombe, J.-C.: *Robot Motion Planning*, Kluwer Academic Publishers, Assinippi, Norwell, Massachusetts, 1993.

Lumelsky, V. and Skewis, T.: Incorporatin range sensing in the robot navigation function, *IEEE Trans. Syst. Man Cybern.* **20**(5) (1990), 1058–1069.

Lumelsky, V. and Stepanov, A.: Dynamic path planning for a mobile automaton with limited information on the environment, *IEEE Trans. Automat. Contr.* (1986), 1058–1063.

O'Keefe, J. and Burgess, N.: Geometric determinants of the place fields of hippocampal neurons, *Nature* **381** (1996), 425–428.

Press, W., Teukolsky, S., Vetterling, W., and Flannery, B.: *Numerical Recipes in C: The Art of Scientific Computing*, 2nd edn, Cambridge University Press, 1992.

Prestes, E., Idiart, M. A. P., Engel, P., and Trevisan, M.: Exploration technique using potential fields calculated from relaxation methods, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2001, p. 2012.

Prestes, E., Engel, P. M., Trevisan, M., and Idiart, M. A.: Exploration method using harmonic functions, *Robot. Auton. Syst.* **40**(1) (2002), 25–42.

Prestes, E., Idiart, M. A., Trevisan, M., and Engel, P. M.: Autonomous learning architecture for environmental mapping, *J. Intell. Robot. Syst.* **39** (2004), 243–263.

Rao, N. S. V., Kareti, S., Shi, W., and Iyengar, S. S.: Robot navigation in unknown terrains: introductory survey of non-heuristic algorithms. Technical Report ORNL/TM-12410, Oak Ridge National Laboratory, Oak Ridge, Tennessee, 1993.

Rimon, E. and Koditschek, D.: Exact robot navigation using cost functions, in: *IEEE/RSJ International Conference of Robotics and Automation*, 1988, pp. 1791–1796.

Sabersky, R., Acosta, A., and Hauptmann, E.: *Fluid Flow*, MacMillan Publishing Co., 1971.

Smith, L. and Husbands, P.: Visual landmark navigation through large-scale environments, in: *EPSRC/BBSRC International Workshop on Biologically-Inspired Robotics: The Legacy of W. Grey Walter*, 2002, pp. 272–279.

Solanas, A. and Garcia, M. A.: Coordinated multi-robot exploration through unsupervised clustering of unknown space, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2004, pp. 717–721.

Thrun, S., Burgard, W., and Fox, D.: A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping, in: *IEEE International Conference on Robotics and Automation*, 2000, pp. 321–328.

Thrun, S., Hähnel, D., Ferguson, D., Montermerlo, M., Triebel, R., Burgard, W., Baker, C., Omohundro, Z., Thayer, S., and Whittaker, W.: A system for volumetric robotic mapping of abandoned mines, in: *IEEE International Conference on Robotics and Automation*, 2003, pp. 4270–4275.

Yamauchi, B.: A frontier based exploration for autonomous exploration, in: *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Monterey, CA, 1997, pp. 146–151.

Yamauchi, B., Schultz, A., Adams, W., and Graves, K.: Mobile robot exploration and map-building with continuous localization, in: *IEEE International Conference on Robotics and Automation*, Leuven, Belgium, 1998, pp. 3715–3720.

Yang, Y., Brock, O., and Grupen, R. A.: Exploiting redundancy to implement multi-objective behavior, in: *IEEE International Conference on Robotics and Automation*, 2003, pp. 3385–3390.

Zelek, J. S.: A framework for mobile robot concurrent path planning and execution in incomplete and uncertain environments, in: *AIPS-98 Workshop on Integrating Planning, Scheduling & Execution in Dynamic & Uncertain Environments*, Pittsburgh, PA, 1998.