IN-HAPTICS: Interactive Navigation using Haptics

Richard D. Walker IV * University of California, San Diego
Sean B. Andersson † Boston University
Calin A. Belta ‡ Boston University
Pierre E. Dupont § Boston University

ABSTRACT

We present a computational framework and experimental platform for robot navigation that allows for a user-friendly, graphical and haptic interaction with the human operator during the deployment process. The operator can see, feel, and manipulate the artificial potential field that drives the robot through an environment cluttered with obstacles. We present a case study in which the operator rescues a robot trapped in a local minimum of a navigation potential field.

Index Terms: H.5.2 [Information Interfaces and Presentation]: User Interfaces—Haptic I/O; I.2.9 [Artificial Intelligence]: Robotics—Autonomous vehicles

1 INTRODUCTION

The aim in robot motion planning is to be able to specify a task in a high-level, expressive language and to have the robot automatically convert the specification into a set of low level primitives, such as feedback controllers, to accomplish the task [4, 12, 13]. Navigation is arguably the most common motion planning problem, in which a robot is deployed from a specification given simply as "Go from A to B and avoid obstacles." Potential-based motion planners for navigation are based on the idea that a robot configuration can be driven to a desired value in the same way in which a particle moves in a force field. They have been successfully applied to a variety of robotic applications. Earlier work focused on motion planning for robotic manipulators [8, 15, 16]. Since then, such methods have become commonplace in path planning [12], with numerous examples in just the past few years (e.g. [3, 7, 9, 14]).

A potential function is required to have a minimum value at the goal and high values at the obstacles. It is usually constructed through the superposition of attractive (to the goal) and repulsive (from the obstacles) functions. As a result, such functions usually exhibit local minima, which are undesired locations where the robot can get trapped. Other limitations include barred passage between closely spaced obstacles and oscillations in narrow passages [10]. A variety of approaches have been developed to overcome these problems. For example, a modified Newton's method has been developed to overcome the oscillations in the presence of obstacles and narrow passages [17]. Several works address the local minima problem. Some are based on the augmentation of the potential field with a search-based planner. For example, the Randomized Path Planner (RPP) [1] uses a variety of potential functions, and, when stuck at a local minimum, it performs a random walk, with the goal of escaping the local minimum. Assuming a successful path exists, the RPP will find it with high probability. If no such path exists, however, the scheme has no recourse for modifying the fields. Others propose the construction of a special type of potential function, called a navigation function [18, 19]. While guaranteed to have exactly one minimum, a navigation function can only be applied to a limited class of configuration spaces, which are diffeomorphic to sphere spaces. Finally, local minima can be dealt with by using potential field methods locally, and path planning and path relaxation methods globally [11]. Most of these approaches have the main disadvantage that they are "static" or "off-line": they only apply to static environments and the human operator cannot intervene to change the task specification or modify the robot trajectory during the deployment process.

As robotics become increasingly present in different applications, from those in everyday life with service and medical robots, to military settings with unmanned drones and human-robot teams, the need for an intuitive approach for high-level supervision and interaction with autonomous systems becomes ever more apparent. With its ability to transmit information and commands both visually and tactilely, a combined graphical and haptic system is one feasible scheme for such an intuitive interface.

With this motivation, in this paper we propose a computational framework and experimental platform for robot navigation that allows for a user-friendly interaction with the human operator during the deployment process. The interaction is achieved through a combined haptic and graphical interface (see Fig. 1). The interface allows the operator to see, feel, and manipulate a potential field. In particular, the potential field can be "pulled" to eliminate local minima, which allows for the use of simple potential functions, *e.g.,* sums of attractive and repulsive potentials. The interface can also be used to adjust the potential function to changes in the environment, such as the appearance of new obstacles. These changes take effect in real-time. The robot in this work is controlled by a simple gradient descent on the potential field and thus the user modifications to the field translate immediately to changes in the motion of the robot. The approach is similar to the use of haptics for motion control in animation [6, 20] but is targeted at motion planning rather than direct control.

While in this paper we focus on potential-based navigation, we believe that this graphical / haptic interface can be used to allow for human intervention during deployment using other navigation techniques, such as cell decompositions [4]. Indeed, the graphical interface could display the vector fields in the partitioned environment, and their intensities can be felt by the operator through the haptic device. Since recent works show that cell-decomposition-based motion planning allows for rich specification languages (see [2] for a review), such as temporal logics [5], we envision that such techniques will allow for high-level, user-friendly, human-robot interaction with applications in a variety of areas.

2 EXPERIMENTAL PLATFORM

The IN-HAPTICS platform, schematically depicted in Fig. 1, consists of a wheeled robot, a human supervisor, a graphical interface, and a haptic interface. The robot moves autonomously through its workspace along a path determined by an artificial potential field. This potential field is displayed to the human supervisor both on the screen of the graphical interface and on the haptics device. At any time, the supervisor can modify the potential field using the haptics interface. After modification, the changes to the potential field are transmitted over a wireless link to the robot and immediately implemented.

*e-mail: rwalker4@gmail.com
†e-mail:sanderss@bu.edu
‡e-mail:cbelta@bu.edu
§pierre@bu.edu

IEEE Haptics Symposium 2010
25 - 26 March, Waltham, Massachusetts, USA
978-1-4244-6820-1/10/$26.00 ©2010 IEEE

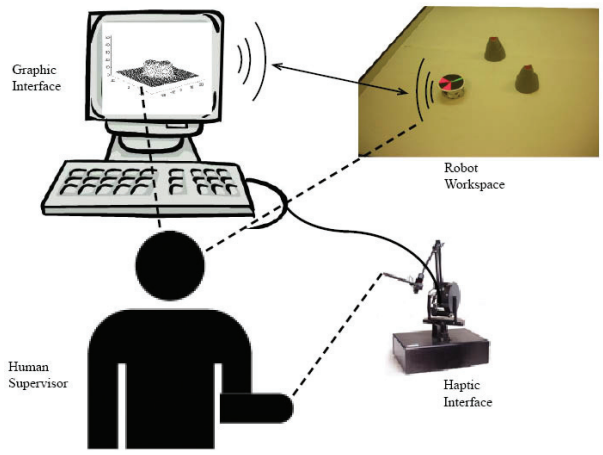


Figure 1: IN-HAPTICS: An autonomous robot moves under the influence of an artificial potential field. This field is displayed simultaneously on a graphical interface and haptics device. At any time, a human supervisor can use the haptics device to modify the potential field and adjust the behavior of the robot.

In our implementation, we use a K-Team Khepera III robot. The main algorithm and the graphical interface are implemented in Matlab on a PC. The graphical interface allows the user to switch between "modification mode", during which the haptics device is used to adjust the potential field, and "display mode" during which the haptics device is used to tactilely explore the field. The PC is connected to a SensAble Technologies PHANToM Premium. Modification of the potential field is achieved by first selecting a set of points of the potential field using the stylus of the haptics device and then using the stylus to define the desired values. The algorithm, described in Sec. 3, connects these new points with a smooth surface and then blends this surface into the surrounding field.

3 POTENTIAL FIELD REPRESENTATION AND MODIFICATION

Robot navigation in the workspace is achieved using a standard artificial potential field approach [12]. Under this scheme, a high value for the potential field is assigned to obstacles and other regions to avoid while the global minimum of the potential field is assigned to the goal destination. While there are many ways to build such a field, we take a simple approach and create the potential field as a sum of repulsive fields, one for each obstacle in the environment, and an attractive field with a global minimum at the goal location.

The attractive potential ϕ_{atr} is modeled as a combination of a decaying potential and a potential well to ensure a negative gradient of magnitude zero at the goal destination:

$$\phi_{atr}(r) = \begin{cases} r^2, & r > r_{rob}, \\ \sqrt{r^2 - r_{rob}} + r_{rob}^2, & r \leq r_{rob}, \end{cases} \quad (1)$$

where r is the distance between the robot's position and the goal position and r_{rob} is the radius of the robot.

For simplicity, we model each obstacle as a simple disk that is large enough to circumscribe the obstacle it represents. Each disk is enlarged by the radius of the robot. For each such disk, the repulsive potential at a distance r from the center of an obstacle of radius r_{obs} is

$$\phi_{rep}(r, r_{obs}) = \min\left(\left(\frac{1}{r - r_{obs} - r_{rob}}\right)^2, B\right) \quad (2)$$

where B is a finite constant large enough to prevent collision of the robot and the obstacle.

The full potential field at a position $\mathbf{r} = (x, y)$ with N obstacles at positions $\mathbf{r_i}$, each of radius r_{obs}^i, and with a goal at $\mathbf{r_g}$ is then given by

$$\phi(\mathbf{r}) = \phi_{atr}\left(\|\mathbf{r} - \mathbf{r_g}\|\right) + \sum_{i=1}^{N} \phi_{rep}\left(\|\mathbf{r} - \mathbf{r_i}\|, r_{obs}^i\right) \quad (3)$$

where $\|\cdot\|$ denotes Euclidean length.

This potential field is displayed both graphically and on the haptics device. To modify it, the user first enters "modification mode" using the graphical interface. Using the stylus, the user then selects a point and indicates its desired value, selects another point and indicates its desired value, and so on. When the user completes the point selection process, regular mode is again entered using the graphical interface. The potential field is then modified according to the following algorithm.

The set of points selected by the user defines a convex hull, indicating the area to be modified as well as the potential field values for the points defining the hull. Within this hull, a grid of points is defined where $p_i = (x_i, y_i, z_i)$ denotes the position of the i^{th} point in the grid. Note that the height z_i is interpreted as the potential field value. This points are then connected in "nearest-neighbor" fashion using virtual springs. The boundary conditions defined by the user-selected field values for the points defining the convex hull establish a set of initial conditions for the deflection of the springs. To determine a smooth potential field surface within the convex hull, the position of these points are numerically evolved according to

$$p_i(k+1) = p_i(k) + \frac{\alpha}{n_i} \sum_{j=1}^{n_i} \left(d_{ij}(k) - \bar{d}_{ij}\right) \hat{d}_{ij}(k) \quad (4)$$

where k is the time index in the numerical solution, α is the spring constant, n_i is the number of nearest neighbors to point p_i, d_{ij} is the distance to the j^{th} neighbor at time k, $\bar{d}_{ij}$ is the unstretched length of the spring to the i^{th} neighbor, and $\hat{d}_{ij}(k)$ is a unit vector pointing from point p_i to the j^{th} neighbor.

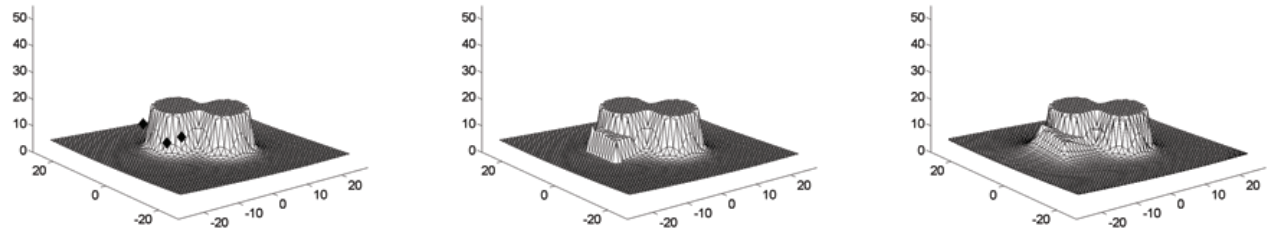
Under this scheme, points on the edges of the convex hull are connected only to points on the interior and thus feel a net force pulling then into the hull, leading to a distortion of the hull. To prevent this, additional points are defined along the edges of the hull. These points are constrained to move only in the vertical direction but otherwise follow the same dynamics. The final values of the modified region are then determined by evolving the system (4) to its equilibrium state.

To connect the modified region to the remainder of the potential field an exponential decay is applied to points near, but outside the convex hull as follows. A rectangular grid is defined near the modified region. For each point within a fixed distance of the hull, the modified potential field value is found by evolving

$$\phi(k+1) = \phi(k)\, e^{\left(-\frac{1}{r}\right)} + \phi_{hull}\left(1 - e^{\left(-\frac{1}{r}\right)}\right) \quad (5)$$

to its equilibrium value. Here r is the shortest distance between the point and the convex hull, ϕ_{hull} is the potential field value at the closest point on the hull, and k is the iteration index.

In Fig. 2 we illustrate the effect of this algorithm. The original potential field, together with a collection of points selected by the user, is shown in Fig. 2(a). The modified potential field after evolving (4) is shown in Fig. 2(b). Note that the edges of the modified surface discontinuously connect to the rest of the field. The smooth, final field after evolving (5) is shown in Fig. 2(c).



(a) Original Potential Field and Selected Points (b) Potential Field after Modification (c) Exponential Decay of Modified Region

Figure 2: Modifying a potential field. **(a)** Using the haptics device, the user selects a group of points. **(b)** The region inside the convex hull of the selected points is modified according to the algorithm described in the text. **(c)** To prevent discontinuities, the modified region is connected to the original potential field using an exponential decay.

4 Mobile Robot Dynamics and Control

The dynamics of the Khepera robot are modeled as a kinematic nonholonomic system, that is

$$\begin{pmatrix} \dot{x}(t) \\ \dot{y}(t) \end{pmatrix} = v \begin{pmatrix} \cos\theta(t) \\ \sin\theta(t) \end{pmatrix} \tag{6}$$

where v is the speed of the robot and θ is the heading direction. The robot is controlled using a standard steepest descent in the potential field by controlling the heading direction according to

$$\dot{\theta}(t) = \beta\left(\theta_{\nabla\phi} - \theta(t)\right) \tag{7}$$

where β is a gain and $\theta_{\nabla\phi}$ is the direction of the gradient of the potential field. The speed v is given by

$$v = \min\left(\|\nabla\phi\|, v_{max}\right), \tag{8}$$

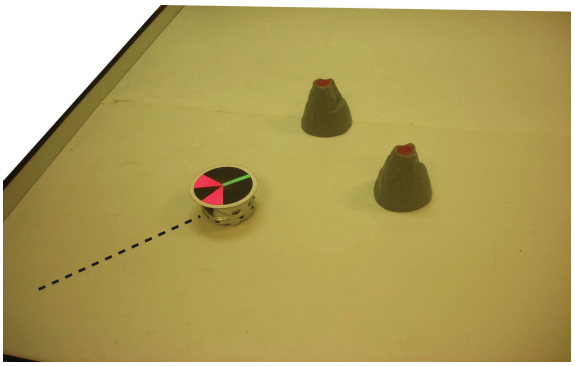
where v_{max} is a threshold used to limit the maximum speed.

When the robot becomes trapped in a local minima, the speed of the robot goes to zero. After user modification of the field, there may be a large difference between the current heading of the robot and the direction of the gradient of the new potential field. As a practical measure, the resulting large turning speeds are undesirable. Therefore, if a large change in the direction of the field is detected, the robot speed is set to 0 until the heading is approximately aligned to the new gradient direction. Once aligned, the motion is then governed by (6) and (7).

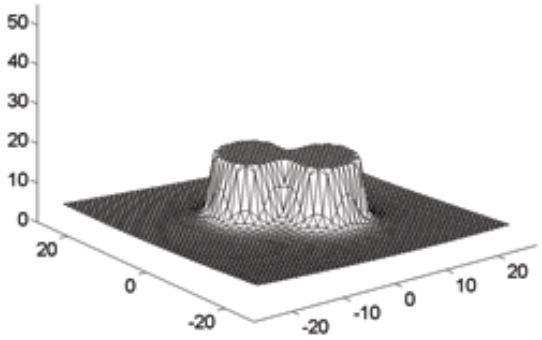
5 Experimental Results

In this section, we present an example in which In-Haptics is used to rescue the robot from a local minimum of a navigation potential function. The environment, shown in Fig. 3(a), consists of two obstacles that are placed close to each other and such that they separate the robot from its goal position. Note that this is a typical situation in which a trapping local minimum of the potential function is known to occur [4]. The potential function in the robot configuration space is shown in Fig. 3(b). The combination of the attractive and repulsive potentials results in a local minimum at the point where the downward sloping attractive potential meets the upward sloping repulsive potentials from the obstacles.

The robot moves autonomously under the control algorithms described in Sec. 4 and gets trapped into the local minimum where it stops (Fig. 3(a)). Upon noticing the trapping situation, the user modifies the potential field in such a way that the local minimum is eliminated (see Sec. 3). The modified potential is shown in Fig. 4(b) and the resulting motion of the robot, which escapes the undesired local minimum, is presented in Fig. 4(a).

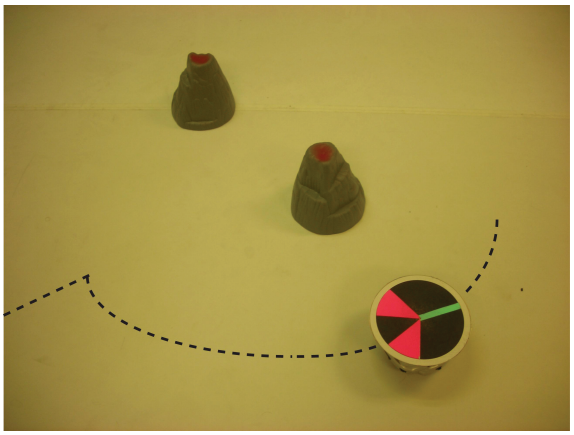


(a) The robot is trapped in a local minimum

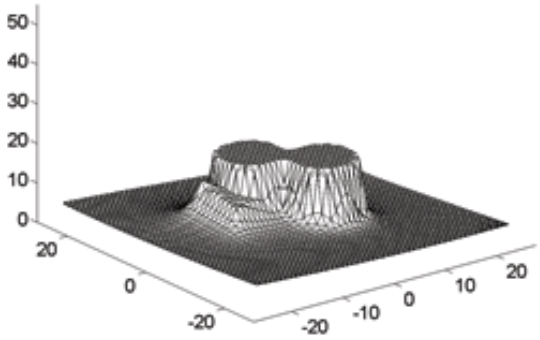


(b) The initial potential field with trapping local minimum

Figure 3: A typical situation in which a robot is trapped in a local minimum of an artificial potential function: two close obstacles separate the robot from its goal.



(a) Escaping the local minimum



(b) Modified potential field

Figure 4: After user-modification of the potential field, the local minimum is eliminated and the robot autonomously steers around the obstacles and proceeds to the goal.

REFERENCES

[1] J. Barraquand, L. Kavraki, J.-C. Latombe, R. Motwani, T.-Y. Li, and P. Raghavan. A random sampling scheme for path planning. *Int. J. Rob. Res.*, 16(6):759–774, 1997.

[2] C. Belta, A. Bicchi, M. Egerstedt, E. Frazzoli, E. Klavins, and G. J. Pappas. Symbolic planning and control of robot motion. *IEEE Robotics and Automation Magazine, special issue on grand challenges for robotics*, 14(1):61–71, 2007.

[3] G. Chesi and Y. S. Hung. Global path-planning for constrained and optimal visual servoing. *IEEE Transactions on Robotics*, 23(5):1050–1060, 2007.

[4] H. Choset, K. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. Kavraki, and S. Thrun. *Principles of Robot Motion: Theory, Algorithms, and Implementations*. MIT Press, Boston, MA, 2005.

[5] E. M. Clarke, O. Grumberg, and D. A. Peled. *Model Checking*. The MIT Press, 2000.

[6] B. R. Donald and F. Henle. Using haptic vector fields for animation motion control. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3435–3442, 2000.

[7] L. Huang. Velocity planning for a mobile robot to track a moving target - a potential field approach. *Robotics and Autonomous Systems*, 57(1):55–63, 2009.

[8] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 500–505, 1985.

[9] D. H. Kim and S. Shin. Local path planning using a new artificial potential function composition and its analytical design guidelines. *Advanced Robotics*, 20(1):115–135, 2006.

[10] Y. Koren and J. Borenstein. Potential field methods and their inherent limitations for mobile robot navigation. *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on*, pages 1398–1404 vol.2, Apr 1991.

[11] B. H. Krogh and C. E. Thorpe. Integrated path planning and dynamic steering control for autonomous vehicles. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1664–1669, 1986.

[12] J. C. Latombe. *Robot Motion Planning*. Kluger Academic Pub., 1991.

[13] S. M. LaValle. *Planning algorithms*. Cambridge University Press, Cambridge, UK, 2006.

[14] C. C. Lin, L. W. Kuo, and J. H. Chaung. Potential-based path planning for robot manipulators. *Journal of Robotic Systems*, 22(6):313–322, 2005.

[15] T. Lozano-Pérez, J. L. Jones, E. Mazer, and P. A. O. an dW. Erick L. Grimson. Handey: A robot system that recongnizes, plans, and manipulates. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 843–849, 1987.

[16] W. S. Newman. *High-speed robot control in complex environments*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 1987.

[17] J. Ren, K. a. Mcisaac, and R. v. Patel. Modified newton's method applied to potential field based navigation for nonholonomic robots in dynamic environments. *Robotica*, 26(3):285–294, 2008.

[18] E. Rimon. *Exact robot navigation using artificial potential functions*. PhD thesis, Yale University, New Haven, CT, USA, 1990. Adviser-Koditschek,, Daniel E.

[19] E. Rimon and D. E. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation*, 8(5):501–518, 1992.

[20] W. Son, K. Kim, N. M. Amato, and J. C. Trinkle. A generalized framework for interactive dynamic simulation for multirigid bodies. *IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics*, 34(2):912–924, 2004.