Fast and Robust Homotopy Path Planning Method for Mobile Robotics

G. Diaz-Arango, L. Hernandez-Martinez,
A. Sarmiento-Reyes
INAOE Electronics Department,
Puebla, Mexico
Email: {gerardo_diaz, luish, jarocho}@inaoep.mx

H. Vazquez-Leal
Maestría en ingeniería electrónica y computación,
Facultad de Instrumentación Electrónica,
Universidad Veracruzana,
Cto. Gonzalo Aguirre Beltrán S/N, 91000 Xalapa, VER, México
Email: hvazquez@uv.mx

***Abstract*—In this paper, we propose a new scheme to find a free-collision path for complex environment maps. This scheme considers the obstacles in the configuration space as prohibited circular areas. Once the obstacles are modelled, the Non-Linear Algebraic Equation System (NAES) is formulated. Next, we apply a Homotopy Path Planning Method (HPPM) to find a set of points that conforms the solution path on the free-space. HPPM uses the Homotopy Continuation Method (HCM) and the Spherical Algorithm (SA) to obtain the solutions of the NAES emanating from the map. In the present work, we propose a scheme for partitioning complex maps into a set of local maps. This is useful for reducing the computation time. Several simulations of complex environment maps with thousands of obstacles are presented. Furthermore, a strategy to obtain the shortest path is applied. Finally, a simulation of the path tracking on Robot Operating System (ROS) for a differential mobile robot is shown.**

Keywords: Homotopy continuation methods, Path Planning, Mobile robot, ROS, Spherical Algorithm, Complex map.

I. Introduction

In recent decades, spend attention has been dedicated to solve the path planning issue for mobile robots. The main function of a path planning method is to find collision-free path from an initial position until a goal position is reached. The most common algorithms are based on cell decomposition, artificial potential fields and probabilistic approaches. However, the conditions for proper functionality are different for each application. On one hand, a path is cut off due local minima present in artificial potential fields planning method [1], [2]. The decomposition cell methods are limited by the resolution of the grid [3] and the probabilistic approach algorithms exhibit a bad performance in narrow maps [4], [5]. On another approach, path panning methods based on HCM have proved great performance to find a successful path [6], [7]. These methods model the work space, the obstacles and the robot as a NAES. Subsequently, the equation system is solved applying HCM for finding the set of solutions that constitute the path. In this work we present an efficient path planning method for terrestrial mobile robots. This paper is organized as follows. In Section **II**, the concepts of the Homotopy Continuation Methods is presented. The Path Planning Method based on HCM and proposed obstacles representation are presented in Section **III**. In Section **IV**, the Spherical tracking and the predictor-corrector scheme are briefly explained. The proposed technique to find successful paths for complex maps is presented in Section **V**. Some simulations for environment maps with thousand obstacles and the tracking by a 3P-DX robot performed in MobileSim software are shown in Section **VI**. Finally, the conclusions are presented in Section **VII**.

II. Homotopic Continuation Method

Homotopy continuation methods are used to find multiple solutions of NAES of the form:

$$f(x) = 0: \quad \mathbb{R}^n \longrightarrow \mathbb{R}^n, \tag{1}$$

HCM creates a continuous perturbation on the original equations system $f(x)$ by transforming it into a Homotopic system. In this work we use Newton's Homotopy and represent the system as:

$$H(x, \lambda) = \lambda f(x) + (1 - \lambda)(f(x) - f(x_0)) = 0, \tag{2}$$

where λ is the homotopy parameter, x_0 is the starting point, $H(x, \lambda) : \mathbb{R}^{n+1} \longrightarrow \mathbb{R}^n, \quad x \in \mathbb{R}^n$. The behaviour of λ is described as:

- If $\lambda = 0$, the homotopy system have a trivial solution $H(f(x), 0) = f(x) - f(x_0) = 0$.
- If $\lambda = 1$, the homotopy system have the solutions of the original system $H(f(x), 1) = f(x) = 0$.

During the continuous perturbation from $\lambda = 0$ to $\lambda = 1$ a sequence of solution points is obtained. This sequence, named Homotopic curve, constitutes the solution path (γ) of the NAES [6], [8], [9].

III. Obstacles representation and Homotopy Path Planning Method

HPPM uses the system of equations that model the configuration space and applies Newton's homotopy to convert it into a homotopic system of equations. Next, it applies the SA for proper tracking the homotopic curve. For this system, the curve represents a sequence of points that describes a continuous path from a start point to a goal point while avoiding collisions with obstacles [6], [7]. The configuration environment is presented as a system of two Non-Linear

978-1-4799-5341-7/16/$31.00 ©2016 IEEE

Algebraic Equations $f_1(x,y)=0$ and $f_2(x,y)=0$. For both equations the unique solution lies on the goal point $B(a,b)$ [6], [7]. The configuration space is represented by the equations:

$$f_1(x,y) = L_1(x,y) = 0, \tag{3}$$

$$f_2(x,y) = L_2(x,y) + W(x,y) - Q = 0, \tag{4}$$

where $W(x,y)$ represents the obstacles in the map and $Q = W(a,b)$ eliminates the effect of the obstacles in the solution point. L_1 and L_2 are two different straight line equations of the form:

$$L_1(x,y) = -y + m(x-a) + b = 0, \tag{5}$$

where $m_1 = -4$ and $m_2 = -1$. These straight lines intersect only at the target point $B(a,b)$ as shown in Figure 1a.

Applying Newton's homotopy to equations (3) and (4), the system is transformed into:

$$H = \begin{cases} H_1(f_1(x,y),\lambda) = f_1(x,y) - (1-\lambda)f_1(x_0,y_0) = 0, \\ H_2(f_2(x,y),\lambda) = f_2(x,y) - (1-\lambda)f_2(x_0,y_0) = 0, \end{cases} \tag{6}$$

where (x_0, y_0) is the initial point A.

The works [6], [7] explain that the presence of obstacles in the map produces a set of solution homotopic curves depicted in Figure 1a, which shows the contour of the obstacles in solid line. Besides, γ_0 represents the solution path, DT is the direct path from A to B, IHC and EHC represents the internal and external homotopic curves of each obstacle respectively. In this work, we propose the obstacles representation as a circles of the form:

$$|C_i(x,y)| + C_i(x,y) = 0, \tag{7}$$

where $\forall\ (x,y) \in \mathbb{R}^2$ such that $\sqrt{(x-x_i)^2 + (y-y_i)^2} \leq rc_i$ i.e. the equation 7 has solution for the all points within the circle. Then, the $W(x,y)$ in the equation 4 can be defined by:

$$W(x,y) = \sum_{i=0}^{i=k-1} \frac{p_i}{|C_i(x,y)| + C_i(x,y)}, \tag{8}$$

$$C_i(x,y) = (x-x_i)^2 + (y-y_i)^2 - rc_i^2 = 0, \tag{9}$$

where the circular shape of each obstacle is defined by C_i equation 9, k is the number of obstacles in the map, p_i is the repulsion parameter for the i-th obstacle, rc_i is the ratio of the circumference and (x_i, y_i) represents the center of a circular obstacle [6]. Our proposed obstacle representation removes all IHC on the map, as shown in Figure 1b. Using 8, we can represent the obstacles as prohibited areas due that their function is undefined for the points inside of these.

IV. Spherical Tracking Algorithm and Non-convergence to Solution

Previous works have demonstrated the effectiveness of SA for tracking homotopic curves [9], [10]. The center of initial sphere is set on the point (x_0, y_0). Then, the intersection point between the homotopic curve and sphere is calculated. Subsequently in each step the center of a new sphere is placed on the previous intersection point. Figure 2 shows the schematic procedure of the spherical algorithm. As a result, the homotopic system is augmented with the SA:

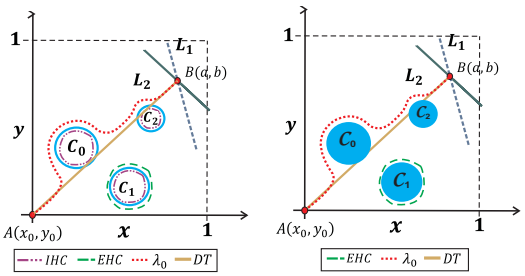


Figure 1: Homotopic paths presents in configuration space map.

$$H_S = \begin{cases} H_1(x,y,\lambda) = 0, \\ H_2(x,y,\lambda) = 0, \\ S_i(x,y,\lambda) = 0. \end{cases} \tag{10}$$

for three dimensions, the sphere is represent as:

$$S_i(x,y,\lambda) = (x-c_x)^2 + (y-c_y)^2 + (\lambda - c_\lambda)^2 - r^2 = 0, \tag{11}$$

where r is the radius and (c_x, c_y, c_λ) is the center of the sphere in each spherical tracking step.

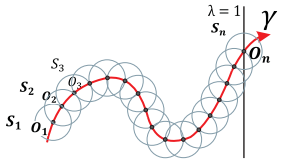


Figure 2: Spherical tracking for a homotopy curve.

Predictor-Corrector Scheme

A proper predictor-corrector scheme allows SA to reach a correct homotopy curve. In this work, we use the scheme reported in [7]. Figure 3 shows the point (x_p, y_p, λ_p) as a predictor, this represents the intersection between the sphere S_i and the tangent vector $(\vec{v}_p)$. Once the predictor point is known,then the Newton-Raphson method (N-R)is applied as corrector to find the intersection point $(x_{i+1}, y_{i+1}, \lambda_{i+1})$ between the homotopy curve and the sphere.

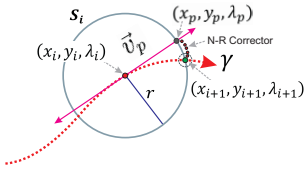


Figure 3: Predictor-corrector scheme.

The predictor point is calculated by using the next expression:

$$(x_p, y_p, \lambda_p) = (c_x, c_y, c_\lambda) + r\|\vec{v}_p\| \tag{12}$$

where, (c_x, c_y, c_λ) is the center of the sphere, r is the sphere radius and $\vec{v_p}$ is the tangent vector [9]. N-R can be recast as:

$$(x_{j+1}, y_{j+1}, \lambda_{j+1}) = (x_j, y_j, \lambda_j) - [J(H_S))]^{-1} H_S, \tag{13}$$

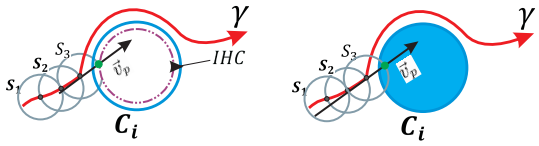
where j, is the current iteration, (x_j, y_j, λ_j) is the point obtained during each iteration, $[J(x_j, y_j, \lambda_j)]^{-1}$is the inverse

matrix of the Jacobian $H_S(x_j, y_j, \lambda_j)$. The tolerance criterion for N-R method in this work is set as:

$$\| H_S(x_{j+1}, y_{j+1}, \lambda_{j+1}) \| < 1 \times 10^{-6}, \quad (14)$$

and the maximum number of iterations is set to $j_{max} = 20$.

The existence of IHC due to the original obstacle representation produces traps inside the obstacles. When the predictor point is placed within the obstacles edge, the tracking could cycled and not reach the target. On the other hand, for the new representation if the predictor point is placed on the prohibited area, then the SA is truncated due a zero-division present in the equation 8. Figure 4 show the predictor point that produces spherical tracking failure.



(a) Representation of obstacle using a circumference.

(b) Representation of obstacle using a circular area.

Figure 4: Predictor point issue.

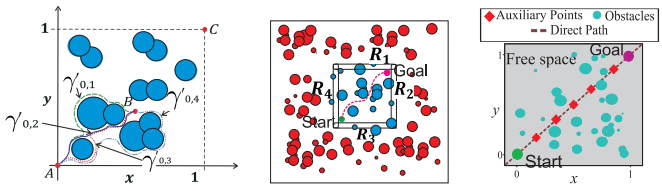
V. Solving Complex Configuration Maps and Finding a Shortest Path

In the previous work [7] we explain how a set of auxiliary paths (γ_0') can be found by using intermediate points and different obstacle repulsion parameters. Figure 5a shows the auxiliary path for the proposed obstacle representation.This Figure shows the initial point A, goal point C, auxiliary point B and auxiliary paths $\gamma_{0,1}'$, $\gamma_{0,2}'$, $\gamma_{0,3}'$ and $\gamma_{0,4}'$. In order to avoid the trivial path that passes along the edges of the map we propose to confine the interest region by using walls. The walls are represented by elliptical approximations 16. The four obstacles are added to 15 and then, the expression is redefined as:

$$W(x,y) = W(x,y) + \sum_{i=1}^{i=4} \frac{p_i}{|R_i(x,y)| + R_i(x,y)}, \quad (15)$$

$$R_i(x,y) = \left(\frac{x - x_i}{\alpha}\right)^{2\eta} + \left(\frac{y - y_i}{\beta}\right)^{2\eta} - 1 = 0, \quad (16)$$

where α is the width, β is the height, η defines the corners, p_i is the repulsion parameter and (x_i, y_i) is the center for each wall [6]. Only the obstacles within the interest area are considered. The repulsion parameter for walls and circular obstacles in the edges automatically chosen such that the path is forced to stay inside, as shown in Figure 5b.



(a) Auxiliary paths for a complex environment map.

(b) Creating local maps confined by walls.

(c) Selection of proper auxiliary points.

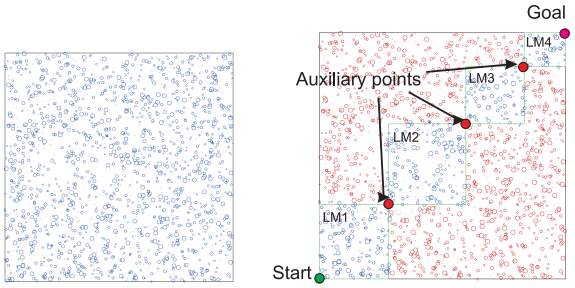
Figure 5: Strategies to solve a complex environment map.

In this work, we proposed an automatically procedure to chose the auxiliary points by using N-R to solve the NAES that represent the free-space 17 and the direct path 18. The initial conditions of the NR are a set of points of the direct path uniformly separated.

$$FS = \sum_{i=0}^{i=k-1} |C_i(x,y)| - C_i(x,y), \quad (17)$$

$$DT = -y + \left(\frac{b - y_0}{a - x_0}\right)(x - x_0) + y_0, \quad (18)$$

where (x_0, y_0) is the initial point and (a, b) is the goal point. A set of auxiliary, goal and initial points are depicted in Figure 5c. We apply a similar scheme as a the reported in [7] and a local confined maps scheme for generated a set of auxiliary paths for complex environments. Subsequently, the shortest path is built using a combination of the shortest auxiliary paths. Figure 6a shows a complex map with 2000 obstacles and Figure 6b presents the corresponding sectioned map, where the interest obstacles are denoted in blue and the irrelevant in red.



(a) Environment 2-D map with 2000 obstacles.

(b) Complex map sectioned on four local maps.

Figure 6: Scheme for solve 2-D complex environment maps.

VI. Experiments

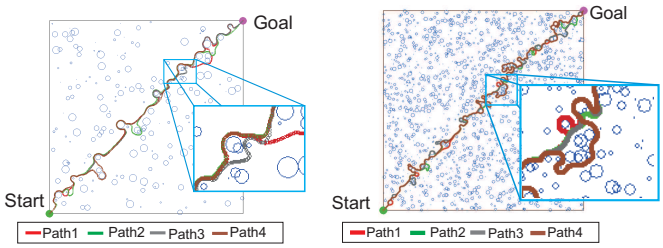
The efficiency of the proposed scheme is validated using a series of simulations for full complex maps and for a one sectioned map. Subsequently, one example of successful path is applied to a virtual Model of a P3-DX robot on ROS. All simulations presented in this work, were performed on C++ using a double precision data on a PC with Intel Core $i5$ at $2.5GHz$ and $6GB$ of RAM.

A. Successful path for maps with 200 and 2000 obstacles

We consider two study cases one with 200 obstacles and another with 2000 obstacles. For both cases, initial and the target points are set at $(0, 0)$ and $(1, 1)$ respectively. In order to avoid the trivial path, the maps are confined in the normalized space using four rectangular obstacles. Figures 7a and 7b show four successful paths for each configuration. The computation time and length of each path are presented in Table I.

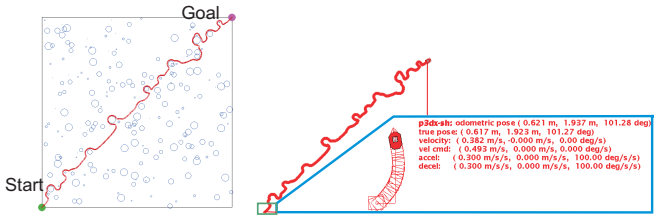
Environment maps								
N.Obstacles	200 Figure 7a				2000 Figure 7b			
Path	1	2	3	4	1	2	3	4
Steps	919	898	894	999	7165	6404	7406	6953
Time (ms)	504	483	504	564	41190	38840	48561	39305
Path length	2.10143	2.06822	2.01062	2.2497	2.59544	2.20463	2.57591	2.40284

Table I: Computation time and length in normalized units for four successful path of two environment maps.



(a) Map with 200 obstacles. (b) Map with 2000 obstacles.

Figure 7: Successful paths for two different environment maps.

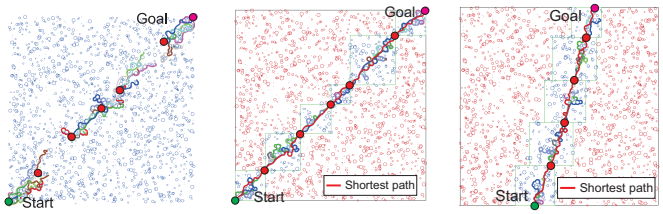


(a) Environment 2-D map with 200 obstacles. (b) Path tracking by Pioneer 3P-DX robot.

Figure 9: Scheme for solve 2-D maps with many obstacles.

B. *Shortest path*

Figure 8a shows unsuccessful auxiliary paths can be observed for the strategy reported in [7] due the complexity of the map. For the same configuration, the proposed scheme of this work reaches to goal point successful, as shown in Figure 8b. Figure 8c shows a shortest and auxiliary paths for another configuration of initial and goal points for map with 2000 obstacles.



(a) Auxiliary paths for method reported in [7]. (b) Auxiliary paths and shortest path using local maps. (c) Auxiliary paths and shortest path using local maps.

Figure 8: Complex environment map with 2000 obstacles solved by sectioned strategy.

Complex Environment map with 2000 obstacles			
Figure	8a	8b	8c
Auxiliary points	5	5	4
Auxiliary paths	10	10	10
Steps	unsuccessful	6458	7027
Time (ms)	395623	12455	15344
Shortest Path length	unsuccessful	1.6263	1.33849
Start-Goal	(0,0)-(1,1)	(0,0)-(1,1)	(0.38,0)-(0.68,1)

Table II: Comparative of a proposed approach and the scheme reported in [7].

It shows a decrease of 26385 ms for path planning of the Figures 8b and path 2 in 7b i.e. the computation time is reduced 67.93 per cent. In similar way, the length decreased 26.2 per cent.

C. *ROS simulations*

In this paper, we show the effective tracking of the obtained paths using MobileSim and ROS. In this case study, we use the package of the differential-drive mobile robot Pioneer 3P-DX. Figure 9b shows the simulation on ROS for a mobile robot 3P-DX developed on a square map of 40 meters per side.

VII. Conclusions

In this work, a modified strategy to solve complex environment maps is proposed. In same way, we proposed a scheme for reducing the complexity of maps with many obstacles by using local maps. Additionally, this approach is used to find the shortest path and to reduce the computing time. The experiments prove that the proposed scheme for HPPM is appropriate to find a successful path in complex configuration maps with thousand obstacles. The ROS simulation verified the advantages of HPPM for applications in real robots. Finally, the computing times in Tables I and II show that the proposed scheme for solving complex maps reduces the calculation time by approximately 60 percent.

References

[1] M. C. Lee and M. G. Park, "Artificial potential field based path planning for mobile robots using a virtual obstacle concept," *Advanced Intelligent Mechatronics IEEE/ASME International Conference*, vol. 2, pp. 735–740, 2003.

[2] M. A. P. Castañeda, J. Savage, A. Hernández, and F. A. Cosío, "Local autonomous robot navigation using potential fields," *Motion Planning. InTech*, 2008.

[3] H. dong Zhang, B. hua Dong, Y. wan Cen, R. Zheng, S. Hashimoto, and R. Saegusa, "Path planning algorithm for mobile robot based on path grids encoding novel mechanism," in *Natural Computation, 2007. ICNC 2007. Third International Conference on*, vol. 4, pp. 351–356, Aug 2007.

[4] C. bae Moon and W. Chung, "Kinodynamic planner dual-tree rrt (dt-rrt) for two-wheeled mobile robots using the rapidly exploring random tree," *Industrial Electronics, IEEE Transactions on*, vol. 62, pp. 1080–1090, Feb 2015.

[5] M. Elbanhawi and M. Simic, "Sampling-based robot motion planning: A review," *Access, IEEE*, vol. 2, pp. 56–77, 2014.

[6] H. Vazquez-Leal, A. Marin-Hernandez, Y. Khan, A. Yildirim, U. Filobello-Nino, R. Castaneda-Sheissa, and V. Jimenez-Fernandez, "Exploring collision-free path planning by using homotopy continuation methods," *Applied Mathematics and Computation*, vol. 219, pp. 7514–7532, 2013.

[7] G. Diaz-Arango, A. Sarmiento-Reyes, L. Hernandez-Martinez, H. Vazquez-Leal, D. Lopez-Hernandez, and A. Marin-Hernandez, "Path optimization for terrestrial robots using homotopy path planning method," in *Circuits and Systems (ISCAS), 2015 IEEE International Symposium on*, pp. 2824–2827, May 2015.

[8] T. L. Wayburn and J. D. Seader, "Homotopy continuation methots for computer-aided process design," *Computers and Chemical Engineering*, vol. 11, pp. 7–25, 1987.

[9] J. M. Oliveros-Munoz and H. Jiménez-Islas, "Hyperspherical path tracking methodology as correction step in homotopic continuation methods," *Chemical Engineering Science*, vol. 97, pp. 413–429, 2013.

[10] Y. K., "Simple algorithms for tracing solution curves," *IEEE International Symposium on Circuits and Systems*, vol. 6, pp. 2801–2804, 1992.