

Article

A Novel Collision-Free Homotopy Path Planning for Planar Robotic Arms

Gerardo C. Velez-Lopez ¹, Hector Vazquez-Leal ^{2,3,*} , Luis Hernandez-Martinez ¹, Arturo Sarmiento-Reyes ¹, Gerardo Diaz-Arango ⁴ , Jesus Huerta-Chua ⁴, Hector D. Rico-Aniles ⁵ and Victor M. Jimenez-Fernandez ²

¹ Electronics Department, National Institute for Astrophysics, Optics and Electronics, Luis Enrique Erro 1, Santa María Tonantzintla, Cholula 72840, Puebla, Mexico; gcvelez@inaoep.mx (G.C.V.-L.); luish@inaoep.mx (L.H.-M.); jarocho@inaoep.mx (A.S.-R.)

² Facultad de Instrumentacion Electronica, Universidad Veracruzana, Cto. Gonzalo Aguirre Beltran S/N, Xalapa 91000, Veracruz, Mexico; vicjimenez@uv.mx

³ Consejo Veracruzano de Investigacion Cientifica y Desarrollo Tecnologico (COVEICYDET), Av. Rafael Murillo Vidal No. 1735, Xalapa 91069, Veracruz, Mexico

⁴ Instituto Tecnologico Superior de Poza Rica, Tecnologico Nacional de Mexico, Luis Donald Colosio Murrieta S/N, Poza Rica 93230, Veracruz, Mexico; gerardo.diaz@itspozarica.edu.mx (G.D.-A.); chua@itspozarica.edu.mx (J.H.-C.)

⁵ Electrical Engineering Department, North Central College, 30 N. Brainard St., Naperville, IL 60540, USA; hdricoaniles@noctrl.edu

* Correspondence: hvazquez@uv.mx



Citation: Velez-Lopez, G.C., Vazquez-Leal, H.;

Hernandez-Martinez, L.; Sarmiento-Reyes, A.; Diaz-Arango, G.; Huerta-Chua, J.; Rico-Aniles, H.D.; Jimenez-Fernandez, V.M. A Novel Collision-Free Homotopy Path Planning for Planar Robotic Arms. *Sensors* **2022**, *22*, 4022. <https://doi.org/10.3390/s22114022>

Academic Editors: Yuansong Qiao and Seamus Gordon

Received: 6 April 2022

Accepted: 18 May 2022

Published: 26 May 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Abstract: Achieving the smart motion of any autonomous or semi-autonomous robot requires an efficient algorithm to determine a feasible collision-free path. In this paper, a novel collision-free path homotopy-based path-planning algorithm applied to planar robotic arms is presented. The algorithm utilizes homotopy continuation methods (HCMs) to solve the non-linear algebraic equations system (NAES) that models the robot's workspace. The method was validated with three case studies with robotic arms in different configurations. For the first case, a robot arm with three links must enter a narrow corridor with two obstacles. For the second case, a six-link robot arm with a gripper is required to take an object inside a narrow corridor with two obstacles. For the third case, a twenty-link arm must take an object inside a maze-like environment. These case studies validated, by simulation, the versatility and capacity of the proposed path-planning algorithm. The results show that the CPU time is dozens of milliseconds with a memory consumption less than 4.5 kB for the first two cases. For the third case, the CPU time is around 2.7 s and the memory consumption around 18 kB. Finally, the method's performance was further validated using the industrial robot arm CRS Catalyst-5 by Thermo Electron.

Keywords: collision-free path planning; autonomous robot; robot arm; homotopy continuation methods

1. Introduction

Significant advances have been made in robotics with more powerful and versatile robots being developed. Currently, robots exhibit enhanced capabilities for performing autonomous and semi-autonomous tasks with a high degree of human interaction. Their applications have been expanded from the traditional ones used in industry and research environments to areas such as clinical surgery and rehabilitation therapy [1–3]. Moreover, the robotics field offer support functions such as automated navigation, warehouse management, and household management [2,4–9].

The design of robotic arms has been part of the robotics evolution by incorporating not only new materials and mechanical structures, but also novel application-specific models. An example of this evolution is hyper-redundant robots; these have a mechanical structure capable of deforming continuously according to their degrees of freedom (DoF) to adapt to

disorderly (unstructured) environments [10,11]; these robots resemble living organisms or their parts, such as snakes or elephant trunks; they are denoted as continuous manipulators and are widely applied in the medical area (for minimal invasive surgeries), for in-orbit servicing, grasping, and locomotion in unstructured environments [5,12,13]. However, working with this type of robot implies solving complex and computationally costly inverse kinematics and real-time collision-free path planning problems [10,11,14].

To perform its main task, every robot executes a sequence of movements. This action must be achieved safely, i.e., without colliding with any obstacle in the workspace. Commonly, this task is performed by a planning algorithm responsible for determining a collision-free path that allows the motion of the robot from an initial to a final configuration without colliding with any obstacle. Since obstacles may be static or moving, the algorithm may be recast as an off-line or on-line planner. On-line planning strategies generate the path to the goal during the movement, while off-line planners obtain the path to the goal before the movement begins [15].

Figure 1 shows the stages required for autonomous or semi-autonomous movement in robots [16]. The proposed work contributes to the advancement of collision-free path computations, represented in the blue block.

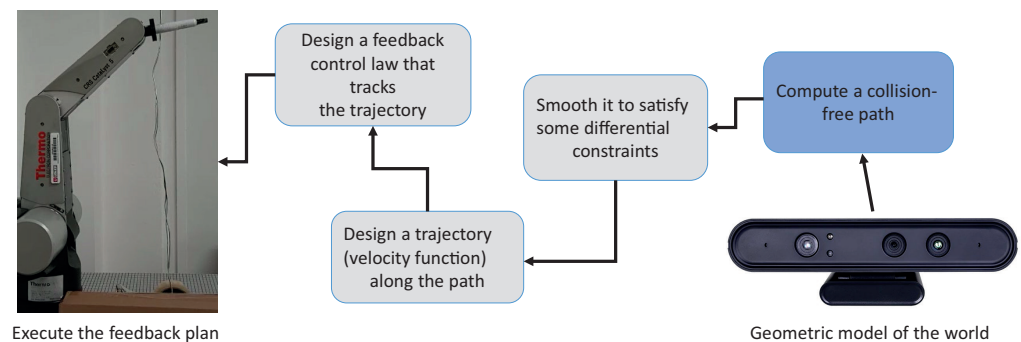


Figure 1. Refinement approach for autonomous robotic manipulators.

In the last few decades, several algorithms have been used to determine collision-free paths, and some of the most common are as follows: (a) sampling-based planning algorithms such as rapidly random trees (RRT) [16], probabilistic roadmap (PRM) [1,17], and the variants of each of them [16]; (b) graph-based algorithms such as visibility graph [18] and A* [19]; (c) heuristic-based algorithms such as ant colony [20] and genetic-based [3]; (d) deterministic-based methods, which include artificial potential fields (APFs) [21] and the homotopy-based path-planning method (HPPM) [22–24]. These algorithms and methods have been applied in mobile terrestrial robots, UAVs, car-like vehicles, and robotic manipulators [1,16,17,20,23,25–28]. However, these algorithms and methods still have several drawbacks such as falling into local minima, high computational cost, or long times to obtain a solution path, and some of these do not guarantee a solution path. Some algorithms present difficulties working in complex environments with narrow corridors, a high number of obstacles, or a high number of DoFs. Other path-planning algorithms need a post-processing stage to smooth out the obtained path for implementation in a real robot [16,23,29,30].

Path planning for planar robots is an interesting topic in robotics due to its principle of operation. It is at the root of multi-joint serial manipulator's path planning because many industrial robots operate with this configuration, such as the SCARA robots. Recently, some works have focused on strategies to handle collision-free path planning. In [31], a path-planning method based on an obstacle-free workspace was proposed for collision-free movements of multi-joint serial manipulators. The method presented in this work uses Monte Carlo and rapidly exploring random tree (RRT), which has advantages in solving the planning problem of a three-joint planar robot arm with only one circular obstacle. Amit Jena et al. proposed in [32] an optimal planning technique using geodesics to achieve

an accurate, as well as smooth trajectory for industrial robot manipulators. In this work, the authors validated its technique with a real SCARA robot; however, the workspace in the case study was obstacle-free. In [33], an optimum trajectory planning for a planar redundant manipulator was presented by minimizing the power consumption when its end-effector was commanded to move in its prescribed path. In this work, a model of the three-joint planar arm was used to validate the effectiveness of the methodology. The case studies in this work show the advantages of the planner to minimize the power consumption; however, only one polygonal obstacle is embedded in the workspace. Finally, a method for trajectory planning and control of planar robots with a passive rotational last joint was presented in [34]. This work validated the method for trajectory planning and control problems for the class of n-link planar robots with a passive rotational last joint. The case studies presented in this work were focused on showing the advantages of this method for trajectory planning and control; however, the presence of obstacles in the environment was not considered.

In this work, the operation principle of the path planning homotopic method (HPPM) is modified to deal with the planar robotic arm's constraints. Originally, the homotopy of continuation was formulated with the aim of solving non-linear systems of equations with multiple solutions. These systems are common in all areas that involve non-linear modeling of systems, such as physics, chemistry, electronics, robotics, fluid mechanics, etc. [23,24,35–39]. Although homotopy has proven to be a very useful tool to find a solution in non-linear systems, recently, its application has focused on solving inverse kinematics calculations in parallel manipulators [35,40], in the calculation of path planning in mobile robots [23,24], and path planning based on the homotopy class with restrictions [41].

In this work, the novel use of the homotopic continuation method for collision-free path planning for planar robotic arms is presented. This involves the modeling of the planar robotic arm and the obstacles in the homotopic formulation to generate the collision-free path. The obtained path is expressed as a sequence of movements for the robotic to move from position A to position B while avoiding obstacles. The paper is organized as follows. Section 2 presents the homotopy-based path planning formulation. The proposed method applied to planar robotic arms with obstacles is presented in Sections 3 and 4. Three case studies show the performance of the proposed method in Section 5. Section 6 presents a scenario where the proposed method is validated in an industrial robotic arm. Finally, Section 7 presents conclusions and future work.

2. Path Planning Using Homotopy-Based Formulations

The homotopy-based path-planning method (HPPM) assumes that the environment constrains the robot, and the obstacles present are mathematically modeled and contained in a non-linear algebraic equations system (NAES) represented by (1).

$$F(X) = 0; \quad \mathbb{R}^n \longrightarrow \mathbb{R}^n, \quad (1)$$

The use of a homotopy continuation method (HCM) modifies the NAES by introducing an additional parameter known as the homotopy parameter λ , yielding a set of homotopy formulas, expressed by (2).

$$H(F(X), \lambda) = \lambda F(X) + (1 - \lambda)G(X) = 0, \quad (2)$$

where $H(F(X), \lambda) : \mathbb{R}^{n+1} \longrightarrow \mathbb{R}^n$, $X \in \mathbb{R}^n$, $\lambda \in [0, 1]$ and $G(X) = 0$ is a function with a trivial or known solution. When $\lambda = 0$, the system in (2) reduces itself to:

$$H(F(X), \lambda)|_{\lambda=0} = G(X) = 0, \quad (3)$$

whereas at $\lambda = 1$, the system in (2) becomes:

$$H(F(X), \lambda)|_{\lambda=1} = F(X) = 0, \quad (4)$$

Equation (4) becomes the original equation system represented in (1).

A particular case of HCM arises if $G(X) = F(X) - F(X_0)$ in (2), which yields (5).

$$H(F(X), \lambda) = F(X) - (1 - \lambda)F(X_0) = 0, \quad (5)$$

where X_0 is the initial point. The formulation in (5) is known as Newton's homotopy, and it has been used to determine a collision-free path, namely the homotopy path-planning method (HPPM) [22,23].

The HPPM models the robot and the obstacles in the workspace as a set of non-linear algebraic equations. Singularities are created for obstacles so that they can be avoided [22,24]. This yields a system of non-linear equations that is solved by the HCM. The solution curve of $H(F(X), \lambda)$, is used as the path that the robot will travel from the initial position to the final. An enhanced version of the HPPM (EHPPM) allows better control over the repulsion effect that is caused by the singularities placed in the workspace [22–24].

The HPPM has been successfully applied to mobile robots [23,24]. Figure 2 shows the application of the EHPPM to generate a collision-free path for a mobile robot, showing a workspace with a mobile robot and 11 circular obstacles. The initial position of the robot is at (x_0, y_0) (red point), and the goal is at (a, b) (blue point), while the path to be followed by the robot is shown in blue.

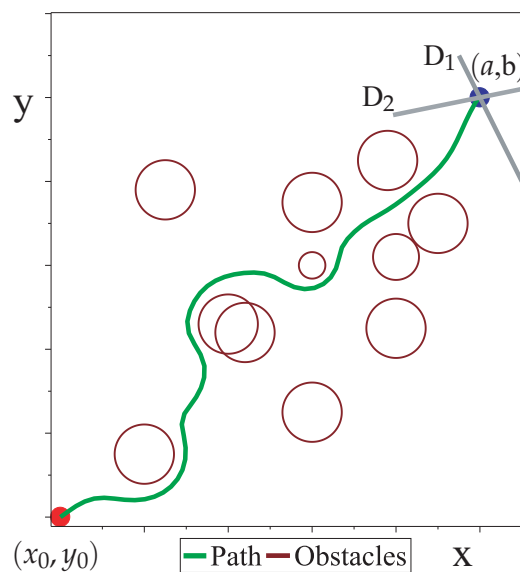


Figure 2. Path obtained with the EHPPM for a mobile robot.

The EHPPM uses the auxiliary Equations (6) and (7) to find the solution.

$$D_1(x, y) = -y - m_1x + (b + m_1a) = 0, \quad (6)$$

$$D_2(x, y) = -y - m_2x + (b + m_2a) = 0, \quad (7)$$

where D_1 and D_2 are straight lines that intersect at the goal (gray lines in Figure 2), m_1 and m_2 are the values of their slopes, and (a, b) represents the ordered pair of the point at the intersection of the straight lines [22–24].

The EHPPM models the workspace with Equations (8) and (9):

$$f_1(x, y) = D_1(x, y) = 0, \quad (8)$$

$$f_2(x, y) = D_2(x, y) + W(x, y) - Q = 0, \quad (9)$$

where $W(x, y)$ represents the singularities that model the obstacles in the workspace and $Q = W(a, b)$ removes the effect of singularities at the goal point (a, b) of the workspace [22–24]. When applying Newton's homotopy (5), the system of homotopic equations is (10).

$$H = \begin{cases} H_1(f_1(x, y), \lambda) = f_1(x, y) - (1 - \lambda)f_1(x_0, y_0) = 0, \\ H_2(f_2(x, y), \lambda) = f_2(x, y) - (1 - \lambda)f_2(x_0, y_0) = 0, \end{cases} \quad (10)$$

where (x_0, y_0) is the start point.

The EHPPM can model the obstacles by two separate sets, namely a set of circles (circular obstacles) and another set of ellipsoidal approximations (ellipsoidal obstacles) [22–24]. As a result, $W(x, y)$ can be represented by Equation (11).

$$W(x, y) = W_C(x, y) + W_R(x, y), \quad (11)$$

where $W_C(x, y)$ represents the set of circular obstacles and $W_R(x, y)$ represents the set of ellipsoidal obstacles.

On the one hand, the set of circular obstacles is defined by Equation (12).

$$W_C(x, y) = \sum_{i=1}^{i=k} \frac{PC_i}{C_i(x_i, y_i)}, \quad (12)$$

where PC_i is the repulsion parameter of each obstacle, k is the number of circular obstacles present in the workspace, and $C_i(x, y)$ represents a circular obstacle, modeled by Equation (13).

$$C_i(x, y) = (x - x_i)^2 + (y - y_i)^2 - rc_i^2 = 0, \quad (13)$$

where (x_i, y_i) is the center and rc_i is the radius of the i -th circle.

On the other hand, the set of ellipsoidal obstacles is modeled by Equation (14).

$$W_R(x, y) = \sum_{j=1}^{j=d} \frac{PR_j}{R_j(x_j, y_j)}, \quad (14)$$

where PR_j is the repulsion parameter of each ellipsoidal obstacle, d is the number of ellipsoidal obstacles present in the workspace, and $R_j(x, y)$ represents the ellipsoidal obstacle modeled by the equation of an ellipse expressed in (15).

$$R_j(x, y) = \left(\frac{x - x_j}{\alpha_j} \right)^{2\eta} + \left(\frac{y - y_j}{\beta_j} \right)^{2\eta} - 1 = 0, \quad (15)$$

where (x_j, y_j) is the center of the j -th ellipsoidal obstacle, α_j and β_j define the width and length, and η is an integer and defines the sharp of the vertex (in this work, $\eta = 2$).

2.1. Spherical-Path-Tracking Algorithm

The EHPPM uses the spherical tracking algorithm to follow the homotopy path at all times and avoids falling into discontinuities or closed curves [22–24,42]. This algorithm creates a hypersphere S of n dimensions, where the number of dimensions corresponds to the number of generated homotopic equations. The hypersphere S_i has a radius r_s with center O_i located on the homotopic curve γ . The circumference of the sphere touches at least two points (O_{i-1}, O_{i+1}) on the curve γ , as depicted in Figure 3.

Equation (16) describes the hypersphere for two homotopy functions.

$$S_i(x, y, \lambda) = (x - c_x)^2 + (y - c_y)^2 + (\lambda - c_\lambda)^2 - r_s^2 = 0, \quad (16)$$

where (c_x, c_y, c_λ) is the center and r_s is the radius of the hypersphere at each step of the spherical tracking.

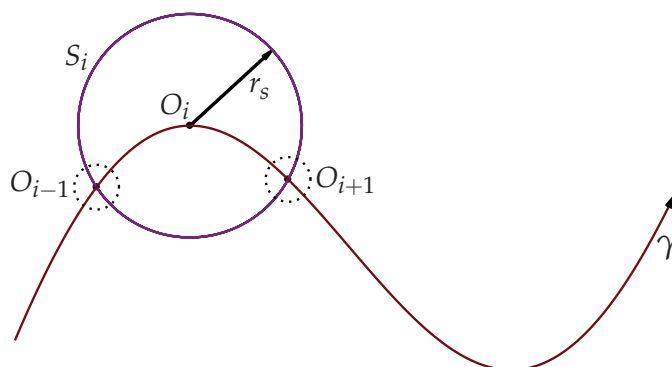


Figure 3. Representation of a hypersphere on the homotopic curve γ .

Using Newton’s homotopy (5), with Equations (10) and (16), the system of homotopy equations is expressed as:

$$H_S = \begin{cases} H_1(x, y, \lambda) = 0, \\ H_2(x, y, \lambda) = 0, \\ S_i(x, y, \lambda) = 0 \end{cases} \tag{17}$$

2.2. Predictor–Corrector Algorithm

The spherical path tracking of trajectories is complemented by the predictor–corrector algorithm, which helps follow the homotopic path without falling into discontinuities [22–24,42]. The predictor algorithm is used to generate the next point close to the homotopic trajectory γ such that the intersection between the hypersphere and the γ curve is achieved; in this work, the corrector was implemented with Broyden’s method [24,43], and the intersection between the hypersphere and the γ curve is achieved. Figure 4 shows the predictor–corrector algorithm where (x_i, y_i, λ_i) is the center of the hypersphere S_i and r_s represents the radius of the hypersphere. The predictor is the point $(x_j, y_j, \lambda_j) = (x_p, y_p, \lambda_p)$, and it is used as the starting point for the corrector, finding the intersection of the hypersphere with the curve γ at $j = 4$. Now, the next center of the hypersphere is at $(x_{j+1}, y_{j+1}, \lambda_{j+1})$.

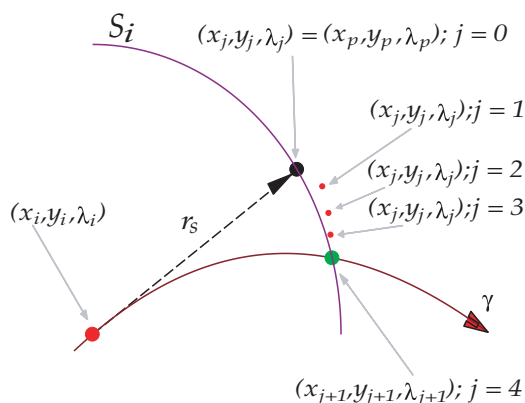


Figure 4. Representation of the predictor–corrector algorithm.

This algorithm repeats and updates the center of each hypersphere until it reaches $\gamma = 1$. In this work, two predictive methods were implemented: Euler’s predictor [23,24] and the vector predictor [42]. To start the homotopic path, Euler’s predictor is used (only two hyperspheres), then the vector predictor is the one that continues with the path until

reaching $\lambda = 1$; this is due to two reasons: by its formulation, the vector predictor always advances on the curve γ , and it requires a lower computational cost than Euler's predictor.

3. Proposed Scheme for Path Planning of Planar Arms

In this paper, a modified HPPM [22] is presented to obtain a collision-free path for planar robot arms. The proposal leaves aside the holonomic punctual robot and presents a novel scheme that has the ability to model redundant and hyper-redundant rigid planar robotic arms, with or without grippers, providing them with rigidity and orientation, and through the formulation of singular projections implemented, it guarantees the avoidance of obstacles. The proposed method is named the homotopy path-planning method for planar robotic arms (HPPM-PRA).

The inputs of the HPPM-PRA are the initial and final configuration of the planar robotic arm and the obstacle's position and size. The planar robotic arm configuration is described in terms of the angles of the links, as shown in Figure 5.

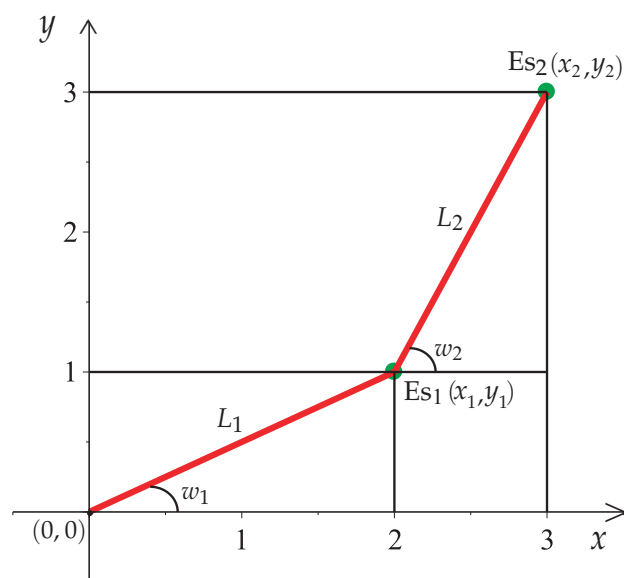


Figure 5. Representation of a two-link planar robot arm.

The HPPM-PRA works on the configuration space (C-space), and in the proposed method, it is shown with the angles w of the bonds; therefore, the algebraic equations are described using w -angles, as well as the obstacles. To describe the obstacles in the C-space, it is proposed to project the obstacles from the Euclidean space (x, y) to the C-space, by creating singularities just in the points where the links touch the perimeter of the obstacles. Therefore, the homotopy path avoids obstacles in the C-space.

Based on Figure 5, the formulas that determine the position of a two-link planar robot arm anchored to the point of origin $(0, 0)$ are described by (18).

$$\begin{aligned} Es_1 : (x_1, y_1) &= (L_1 \cos(w_1), L_1 \sin(w_1)), \\ Es_2 : (x_2, y_2) &= (x_1 + L_2 \cos(w_2), y_1 + L_2 \sin(w_2)), \end{aligned} \quad (18)$$

where L_1 and L_2 represent the length of the links and the angles w_1 and w_2 represent the position of links Es_1 and Es_2 . This process can be repeated for each link in the case of more than two links.

In order to avoid obstacles, each link has to be divided into n singular points. These singular points exhibit a projection over the perimeter of each of the obstacles. Figure 6 shows an example of a workspace with a two-link robot arm (Es_1, Es_2) and two circular

obstacles (C_1, C_2). The link Es_2 is divided into n singular points. Then, each singular point of Es_2 is represented by (19).

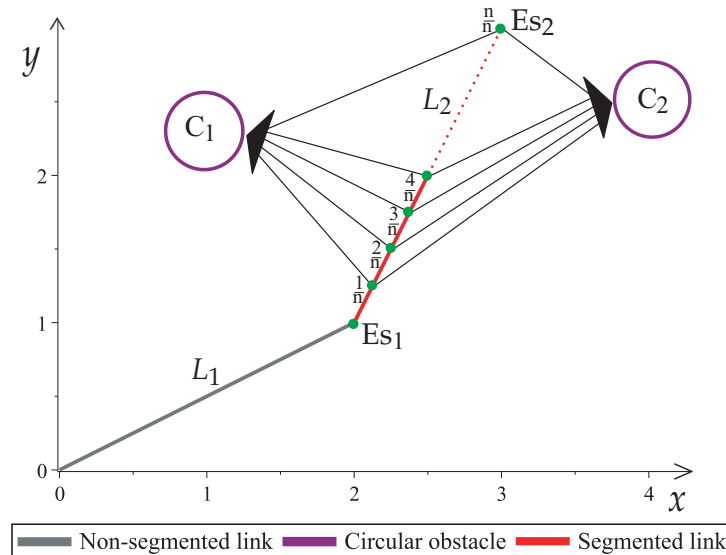


Figure 6. Singular projections of a two-link planar robot arm with two obstacles.

$$Es_2 : (x_k, y_k) = (L_1 \cos(w_1) + (q_k)L_2 \cos(w_2), L_1 \sin(w_1) + (q_k)L_2 \sin(w_2)), \quad (19)$$

$$k = 1, \dots, n,$$

where q_k helps calculate the position of each singular k -th point in the Euclidean space and represents the link's segments, as represented by Equation (20).

$$q_k = \frac{k}{n}, \quad k = 1, 2, \dots, n, \quad (20)$$

where k is the k -th segment and n is the number of segments.

The formulation for circular obstacles $C_i(w)$ and ellipsoidal obstacles $R_j(w)$ is expressed in (21) and (22).

$$C_i(w_1, w_2) = (x_k - x_i)^2 + (y_k - y_i)^2 - r_i^2, \quad i = 1, 2, \dots, c, \quad (21)$$

$$R_j(w_1, w_2) = \left(\frac{x_k - x_j}{\alpha_j} \right)^{2\eta} + \left(\frac{y_k - y_j}{\beta_j} \right)^{2\eta} - 1, \quad j = 1, 2, \dots, d, \quad (22)$$

where C_i is the i -th circular obstacle, R_j represents the j -th ellipsoidal obstacle, c is the number of circular obstacles, and d is the number of ellipsoidal obstacles in the workspace. The coordinates of each singular point in each link are represented by (x_k, y_k) , while (x_i, y_i) is the center of circular obstacle ($C_i(w_1, w_2)$) and (x_j, y_j) is the center of ellipsoidal obstacle ($R_j(w_1, w_2)$). r_i is the radius of each circular obstacle, and α_j and β_j are the base and height of the j -th ellipsoidal obstacle. To create rigid obstacles, n -singular points were used for each link by considering the size of the smallest obstacle in the workspace as a reference to set n . The formulation that represents the singularities created by the singular projections involving all the links is given by (23).

$$W(w_1, \dots, w_v) = \sum_{i=1}^c \sum_{t=1}^v \sum_{k=1}^n \frac{P_{C_i}}{C_i(w_1, \dots, w_v)} + \sum_{j=1}^d \sum_{t=1}^v \sum_{k=1}^n \frac{P_{R_j}}{R_j(w_1, \dots, w_v)}, \quad (23)$$

where P_{Ci} represents the repulsion parameter of each circular obstacle, P_{Rj} represents the repulsion parameter of each ellipsoidal obstacle, c represents the number of circular obstacles, d represents the number of ellipsoidal obstacles, v the number of links, and n the number of singular points of each link.

The HPPM-PRA uses a system of auxiliary equations to set the final position of the robotic arm given by (24).

$$l_k(w_1, \dots, w_v) = a_{0,k} + \sum_{j=1}^v a_{j,k} w_j = 0, \quad k = [1, 2, \dots, v], \quad (24)$$

Here, v is the number of links in the robot arm and a is a set of arbitrary constants. By evaluating (24) at the goal position $(w_{goal_1}, \dots, w_{goal_v})$ and selecting $a_{0,k}$ to set each linear equation to zero, the system of equations for a planar robot arm becomes:

$$\begin{aligned} f_1(w_1, \dots, w_v) &= l_1(w_1, \dots, w_v) = 0, \\ f_2(w_1, \dots, w_v) &= l_2(w_1, \dots, w_v) = 0, \\ &\vdots \\ f_v(w_1, \dots, w_v) &= l_v(w_1, \dots, w_v) + W(w_1, \dots, w_v) - Q = 0, \end{aligned} \quad (25)$$

where Q is a constant to guarantee that the solution of (25) is $(w_{goal_1}, \dots, w_{goal_v})$. It is important to note that the equation l_v was chosen to add the term $W - Q$; nonetheless, it is feasible to select another equation of (25).

The homotopy system to solve (25) using (5) is then (26).

$$H = \begin{cases} H_1(f_1(w_1, \dots, w_v), \lambda) = f_1(w_1, \dots, w_v) - (1 - \lambda)f_1(w_{start_1}, \dots, w_{start_v}) = 0, \\ H_2(f_2(w_1, \dots, w_v), \lambda) = f_2(w_1, \dots, w_v) - (1 - \lambda)f_2(w_{start_1}, \dots, w_{start_v}) = 0, \\ \vdots \\ H_v(f_v(w_1, \dots, w_v), \lambda) = f_v(w_1, \dots, w_v) - (1 - \lambda)f_v(w_{start_1}, \dots, w_{start_v}) = 0, \end{cases} \quad (26)$$

where $(w_{start_1}, \dots, w_{start_v})$ represents mathematically the starting point for the homotopy path and, physically, the initial position of the robotic arm.

Equation (27) is proposed to implement the hyperspherical path-tracking algorithm that was stated in (16).

$$S_i(w_1, \dots, w_v, \lambda) = (w_1 - c_1)^2 + (w_2 - c_2)^2 + \dots + (w_v - c_v)^2 + (\lambda - c_{v+1})^2 - r_s^2 = 0, \quad (27)$$

where $(c_1, c_2, \dots, c_{v+1})$ is the center of the hypersphere and r_s is the radius of the hypersphere. It is important to highlight that the first hypersphere's center (c_1, c_2, \dots, c_v) is equal to the starting point $(w_{start_1}, \dots, w_{start_v})$ of the homotopy at $\lambda = c_{v+1} = 0$. However, it is also feasible to use a variable-radius scheme.

The system of equations to be solved for each step of the path-tracking algorithm is given in (28).

$$H_S = \begin{cases} H_1(w_1, \dots, w_v, \lambda) = 0, \\ H_1(w_1, \dots, w_v, \lambda) = 0, \\ \vdots \\ H_v(w_1, \dots, w_v, \lambda) = 0, \\ S_i(w_1, \dots, w_v, \lambda) = 0, \end{cases} \quad (28)$$

where the center of the hypersphere S_i is updated at every i -th step of the path tracking, as explained in Section 2.2.

Workspace and C-Space

This section explains how to transform from the workspace to C-space and the relation to obstacles and the homotopic path computation process. On the one hand, the workspace is the graphical representation of the number and length of links, the initial and final position of the robotic arm (start, goal), and the number, position, and shape of obstacles in the real world. On the other hand, the C-space maps the allowed and forbidden configurations that the robot can perform without collisions in the workspace. The configurations are modeled in a space with as many dimensions as degrees of freedom the robot has. Figure 7a shows the workspace with a two-link planar robotic arm. The initial (start) and final (goal) positions of the robotic arm are represented by the solid red line and the blue asterisks, respectively. Figure 7b shows the C-space in terms of the angles w_1 and w_2 of the links.

The solid red dot marks the starting point (start), and the intersection of the auxiliary lines denotes the endpoint (goal) of the homotopic path (blue diamond).

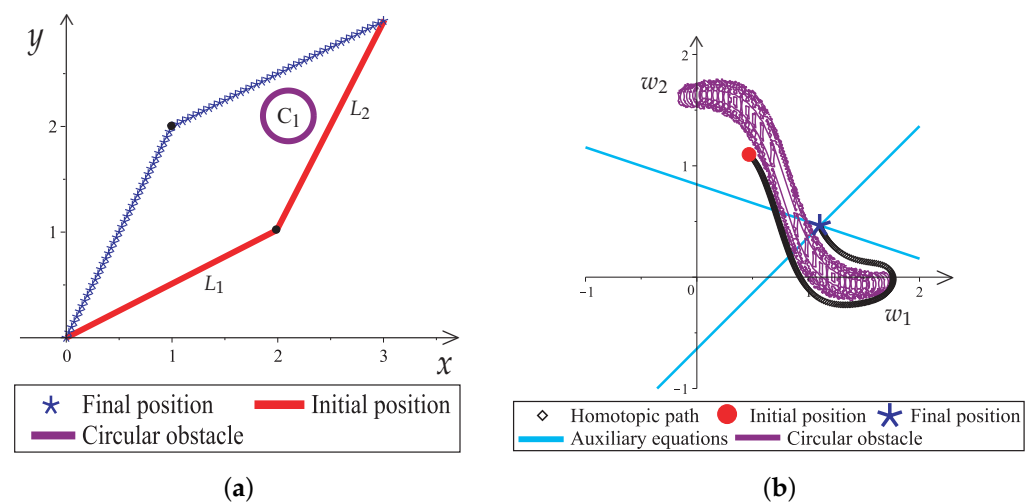


Figure 7. Workspace to C-space of a two-link planar robot arm with a circular obstacle. (a) Two-link planar robot workspace. (b) C-space representation of the two-link planar robot workspace.

It can be noted that the circular obstacle in the Euclidean space is transformed into an elongated amorphous shape in the C-space, and this is due to the singular projections that exist between the link of the robotic arm and the obstacle; this shape was obtained by plotting (23) using the Maple 18 “implicitplot” command. It should be mentioned that for more than three dimensions, it is not easy to visualize the C-space.

4. HPPM-PRA Procedure Steps

The HPPM-PRA is a straightforward procedure that is easy to implement. This requires some basic steps:

1. Capturing of the workspace. In this step, a camera or system to capture the environment is used to generate the geometrical representation of the robot’s workspace. This procedure requires an image-processing algorithm; however, this is not a topic of this work; thus, the workspace is considered known a priori. Then, for this work, only circular obstacles (21) and ellipsoidal obstacles (22) are considered to represent the workspace in the case studies of Section 5.
2. Setting the robot parameters. The number and length of the links and the start and end configurations are given to the HPPM-PRA.
3. Repulsion parameter assignation. Assign the repulsion parameter to each obstacle of the workspace.
4. Model workspace obstacles in the C-space. The singular projections (W) that represent the forbidden configurations (obstacle collision space) are employed to establish the Q value using (23).

5. Generate the auxiliary equations. These are used to set the final configuration of the robotic arm.
6. Generate non-linear equation system. This represents the entire problem and contains the characteristics of the robot and the workspace.
7. Homotopy continuation formulation (Equation (26)). In this step, the original system of non-linear equations of the previous step is converted to a homotopic system.
8. Hyperspherical tracking. The hyperspherical tracking algorithm is employed to calculate each point of the solution path.
9. Robotic arm executing. Finally, the obtained homotopic path is followed by the robotic arm.

Figure 8 shows the flowchart of the proposed methodology.

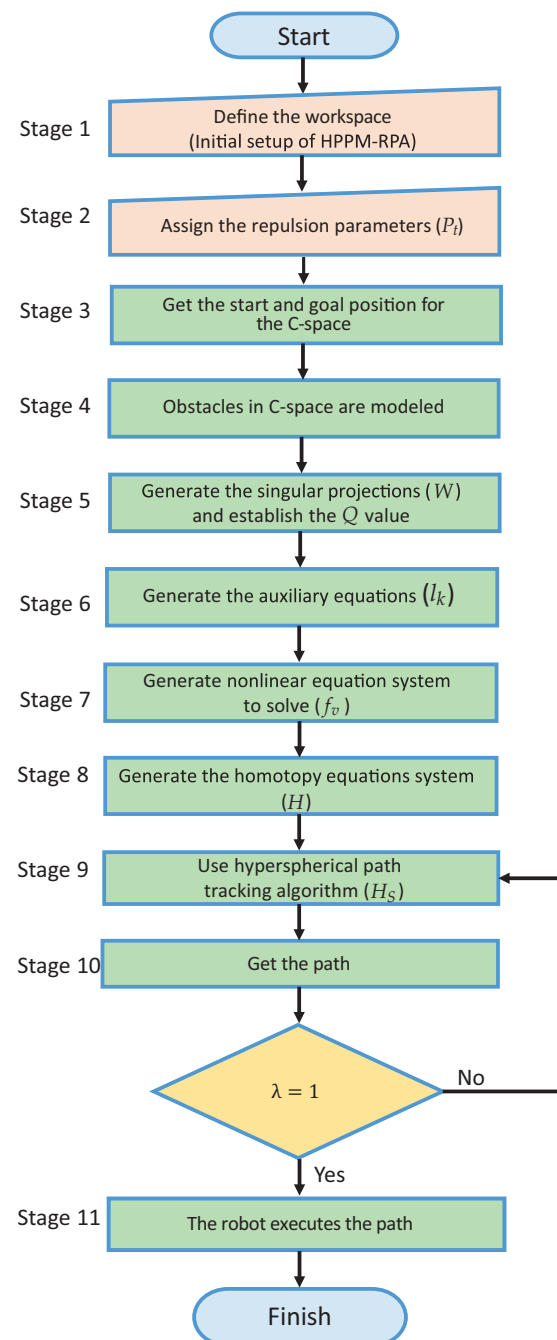


Figure 8. HPPM-PRA method flowchart.

Algorithms 1–5 describe the stages that model the proposed method. Algorithm 1 describes the general procedure of the HPPM-PRA. This procedure requires the workspace configuration as the input and uses four algorithms that build the homotopy system (H). The homotopy system is solved by using the spherical path-tracking algorithm to generate the path for the planar robotic arm.

The function of Algorithm 2 is to transform the workspace to the C-space, and the information obtained (w_{start}, w_{goal}, a_0) is used by the following algorithms.

Algorithm 3 is responsible for obtaining the value of Q from the evaluation of the circular and ellipsoidal obstacles in the final position (w_{goal}). Furthermore, this algorithm is used in each step of the hyperspherical procedure to obtain the value of W .

Algorithm 4 builds, based on the auxiliary equations ($l_k(w_1, \dots, w_v)$), the system of non-linear equations (f_1, \dots, f_v), which Algorithm 5 uses to generate the homotopic system (H).

Algorithm 1 HPPM-PRA general procedure.

Require: $R(w), C(w), Es_{start}, Es_{goal}, L, v$ ▷ The task to be solved is proposed
Require: P_C, P_R ▷ Assign the repulsion parameter
1: Get (w_{start}, w_{goal}, a_0) ▷ See Algorithm 2
2: Get the value of Q ▷ See Algorithm 3
3: Set the non-linear equation system to solve (f_1, \dots, f_v) ▷ See Algorithm 4
4: Generate the homotopy equation (H) ▷ See Algorithm 5
5: Create hypersphere S_i ▷ w_{start} is used as the center of the first hypersphere
6: Formulation of the homotopy system (H_S)
7: iteration=0 ▷ A temporary variable is used as the counter
8: **while** ($\lambda \neq 1$) **do** ▷ Use the hyperspherical path-tracking algorithm, until $\lambda = 1$
9: **if** ($iteration < 2$) **then**
10: Euler's predictor (H_S) ▷ Euler's predictor is used
11: **else**
12: Vector predictor (H_S) ▷ The vector predictor is used
13: **end if**
14: Broyden's method (H_S) ▷ The corrector method is used
15: The numeric homotopy path is stored
16: Update the center of the hypersphere (S_i)
17: iteration++
18: **end while**
Ensure: The numerical homotopy path ▷ The robotic arm can execute the path

Algorithm 2 Transformation from workspace to C-space.

1: **function** C-SPACE($start, goal, a_{j,k}, v, L_v$)
2: j, k ▷ Temporary variables
3: **for** ($j = 0; j < v; j++$) **do**
4: $w_{start}[j] = \arccos\left(\frac{start[j+1]-start[j]}{L[j]}\right)$ ▷ Start position in C-space.
5: $w_{goal}[j] = \arccos\left(\frac{goal[j+1]-goal[j]}{L[j]}\right)$ ▷ Goal position in C-space.
6: **end for**
7: **for** ($j = 0; j < v; j++$) **do** ▷ Calculation of the value of a_0 .
8: **for** ($k = 0; k < v; k++$) **do**
9: $a_0[j] = (w_{goal}[k+1])(a[j][k])$
10: **end for**
11: $a_0[j] = w_{goal}[0]$
12: **end for**
13: **return** (w_{start}, w_{goal}, a_0) ▷ Returns the values from the C-space
14: **end function**

Algorithm 3 Get the value of (Q, W) .

```

1: function GET  $Q(w_{goal}, C_w, R_w, P_C, L_v, c, v, n)$ 
2:    $add_{obs}[v] = 0, add[n] = 0, j, k, i$  ▷ Temporal variables.
3:   for ( $j = 0; j < c; j++$ ) do
4:     for ( $k = 0; k < v; k++$ ) do
5:       for ( $i = 0; i < n; i++$ ) do
6:          $x_k[i] = \cos(w_{goal}[k])(L[k])\left(\frac{i+1}{n}\right)$ 
7:          $y_k[i] = \sin(w_{goal}[k])(L[k])\left(\frac{i+1}{n}\right)$  ▷  $(x_k, y_k)$  are the coordinates of each
           singular point for a given link
8:          $add_{obs}[k] += \frac{P_C[j]}{(x_k[i]-x[j])^2+(y_k[i]-y[j])^2-(r[j])^2}$  ▷ The equation of the circular
           obstacle  $C(w)$  can be replaced by the equation of ellipsoidal obstacle  $R(w)$ 
9:       end for
10:       $add[j] += add_{obs}[k]$ 
11:    end for
12:     $add_{obs}[v] = 0$  ▷ Temporal variable is cleared.
13:     $Q += add[j]$ 
14:  end for
15:  return  $(Q)$  ▷ The value of  $Q$  is obtained.
16: end function

```

Algorithm 4 Set non-linear equation system to solve.

```

1: function SET  $f_1, \dots, f_v(a_0, a_{j,k}, W, Q, v)$ 
2:    $j, k$  ▷ Temporary variables
3:   for ( $j = 1; j < v; j++$ ) do
4:     for ( $k = 1; k < v; k++$ ) do
5:        $l[k] += (a[j][k])(w[j])$ 
6:     end for
7:      $l[k] += a_0[j]$  ▷ The system of auxiliary equations is obtained  $l_k(w_1, \dots, w_v)$ 
8:   end for
9:   for ( $k = 1; k < v - 1; k++$ ) do
10:     $f[k] = l[k]$ 
11:  end for
12:   $f[k+1] = l[k] + W - Q$ 
13:  return  $f$  ▷ Returns  $f_1, \dots, f_v$ 
14: end function

```

Algorithm 5 Generate the homotopy system.

```

1: function GENERATE  $H(w_{start}, v, f_1, \dots, f_v, \lambda)$ 
2:   for ( $k = 1; k < v; k++$ ) do
3:      $H[k] = f[k] - (1 - \lambda)f[k](w_{start})$ 
4:   end for
5:   return  $f$  ▷ Returns  $H_1, \dots, H_v$ 
6: end function

```

5. Case Studies

In this section, three case studies are presented, which show the capacity of the proposed method to obtain the path in planar robot arms. These studies evaluate the performance of the proposed method with the environment, which includes narrow corridors and circular and ellipsoidal obstacles. The implementation can be modified to any number of links, their length, and added grippers. For all the case studies shown in this section, the following color convention is used: circular obstacles (C_u) are purple; ellipsoidal obstacles (E_u) are gray; the gripper is brown; the initial position of the robot arm is red;

the final position is blue; the homotopic path is shown in black. The proposed method was implemented using the C++ programming language, and the animations were performed in Maple software. All the case studies were executed on a personal computer with a Core i5-4210u@1.7GHz processor and 8 GB of RAM. Here, it is important to note that no special specialized package, library, or hardware was used to help reduce the computing time.

5.1. Case Study 1

In this case study, a three-link planar robot arm with two circular obstacles is presented as shown in Figure 9. To reach the final position, the robot arm must pass through a narrow corridor generated by the circular obstacles C_1 and C_2 .

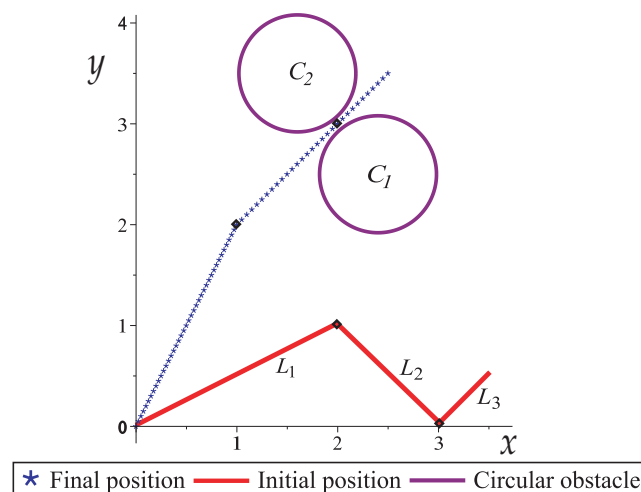


Figure 9. Workspace of a planar robot arm with three links and two circular obstacles.

Table 1 shows the center, radius, and proposed repulsion parameter (P) of circular obstacles, the length of each link, and the proposed constants of auxiliary equations.

Table 1. Parameters of Case Study 1.

Obstacle	Type of Obstacle	x_c	y_c	r_c	P
C_1	Circular	2.4	2.5	0.58	-0.1
C_2	Circular	1.6	3.5	0.58	0.1
Link length	$L_1 = \sqrt{5}, L_2 = \sqrt{2}, L_3 = \sqrt{0.5}$				
Constants of auxiliary equations	$l_1 : (a_0 = -5.03, a_1 = 1, a_2 = 3, a_3 = 2),$ $l_2 : (a_0 = -3.46, a_1 = 1, a_2 = -1, a_3 = 4),$ $l_3 : (a_0 = -0.32, a_1 = 1, a_2 = 1, a_3 = -2)$				
Initial state of the robot arm links	$L_1 : \{(0,0), (2,1)\}, L_2 : \{(2,1), (3,0)\}, L_3 : \{(3,0), (3.5,0.5)\}$				
Final state of the robot arm links	$L_1 : \{(0,0), (1,2)\}, L_2 : \{(1,2), (2,3)\}, L_3 : \{(2,3), (2.5,3.5)\}$				

Figure 10 depicts the collision-free path for the robotic arm of Figure 9. From Figure 10a to Figure 10e, the collision-free movements of the planar robotic arm are observed until it reaches the final position (goal) in the narrow corridor generated by the two obstacles (see Figure 10e).

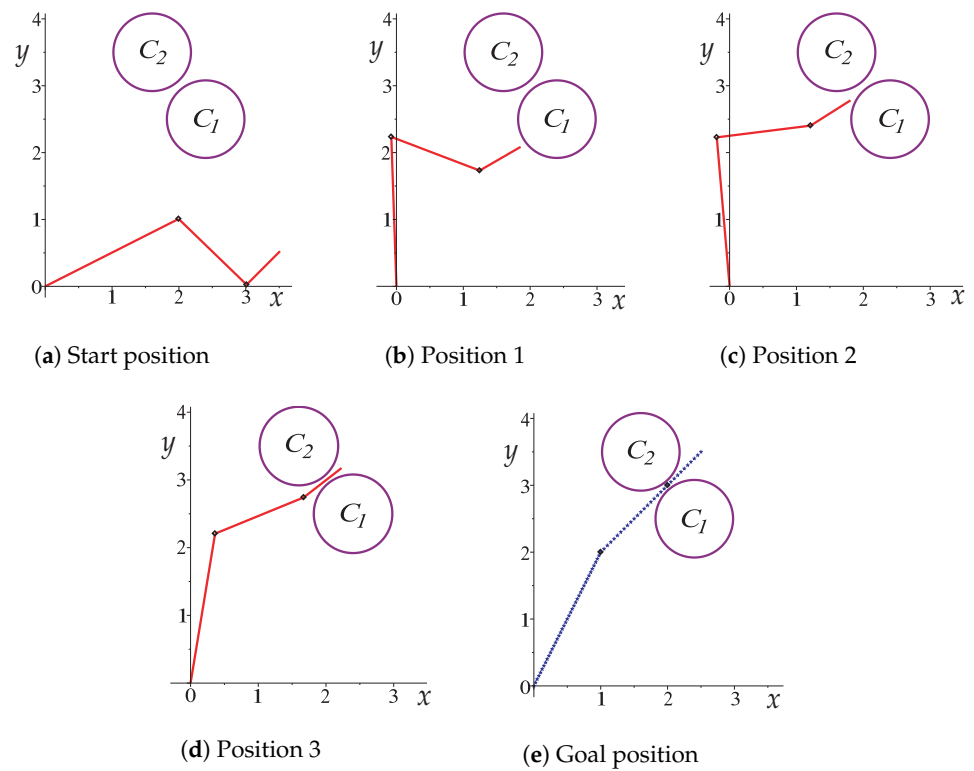


Figure 10. Sequence of images that describe the collision-free path of Case Study 1.

Figure 11 shows the changes that each angle of movement w of the robotic arm has; the marked points are the nodes of the solution path shown in Figure 10. Figure 12 shows the change in joint angle movement, where it is observed that the greatest change obtained between each movement does not exceed 0.03 radians, confirming that the path obtained is smooth.

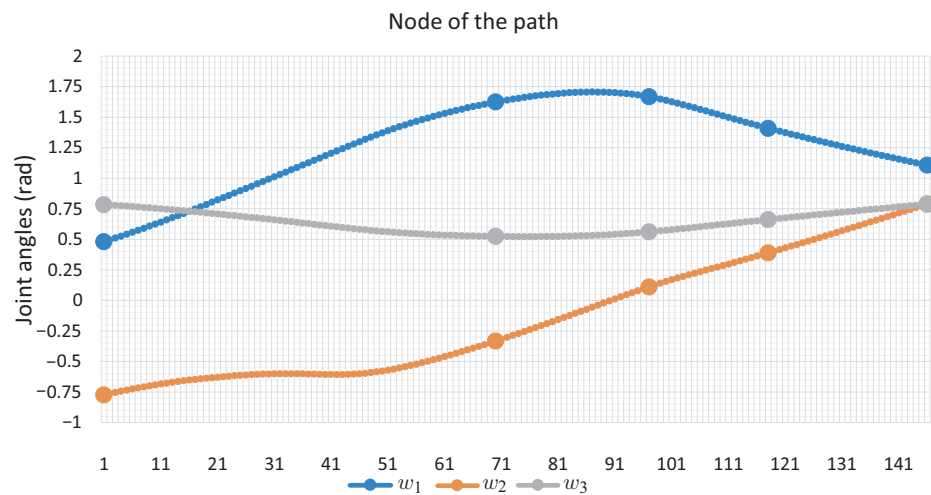


Figure 11. Joint angles motion of Case Study 1.

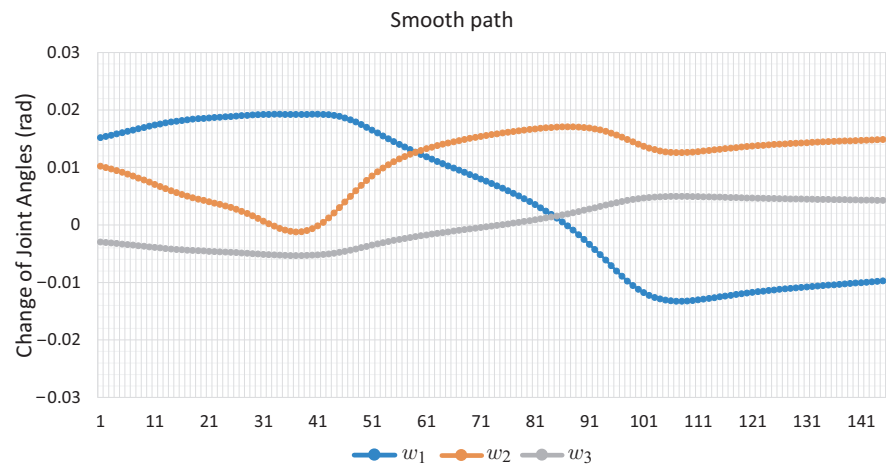


Figure 12. Change of joint angle motion of Case Study 1.

It is important to note that the robotic arm touches the perimeter of the obstacles; nevertheless, it does not collide with them due to the safeguard radius (rt) [22,23], as depicted in Figure 13. This case study shows that the HPPM-PRA can obtain paths even in narrow corridors where probabilistic methods may fail.

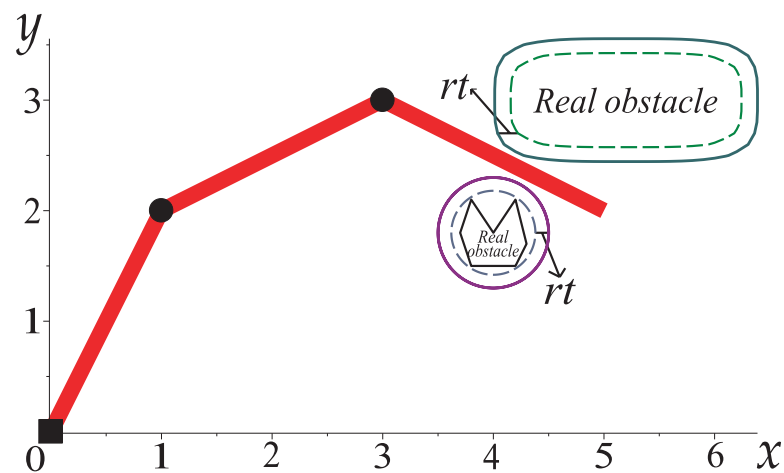


Figure 13. Representation of a safeguard radius (rt) of an obstacle in a workspace with a robotic arm.

5.2. Case Study 2

For this case study, a robot arm with six links (of the same length) and a gripper is depicted. The robotic arm starts from a rest position where the gripper is closed and moves to grip a circular object (C_1) that is between the ellipsoidal obstacles (E_1, E_2), as depicted in Figure 14. Table 2 shows the configuration parameters for this case study.

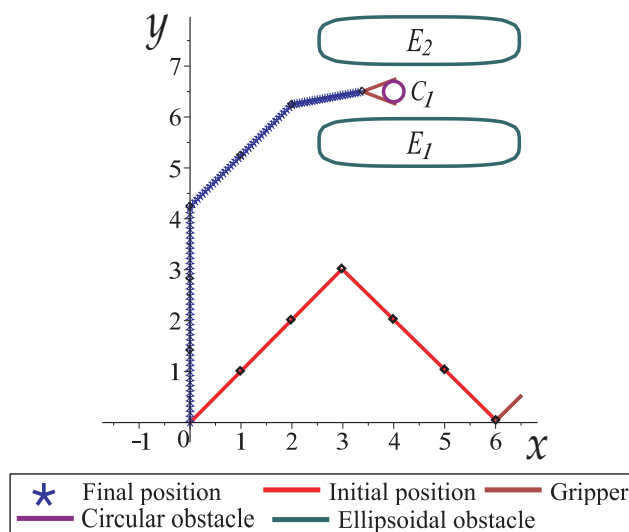


Figure 14. Case Study 2 shows a planar robotic arm with six links and a gripper, two ellipsoidal obstacles, and a circular obstacle.

Table 2. Parameters of Case Study 2.

Obstacle	Type of Obstacle	x_c	y_c	r_c	α_c	β_c	P
C_1	Circular	4.0	6.5	0.2	-	-	-0.0002
E_1	Ellipsoid	4.5	5.5	-	15.0	0.05	-0.8
E_2	Ellipsoid	4.5	7.5	-	15.0	0.05	0.8
Link length	$L_{1-6} = \sqrt{2}, L_{gripper} = \sqrt{0.5}$						
Constants of auxiliary equations	$l_1 : (a_0 = -13.05, a_1 = 1, a_2 = 1, a_3 = 2, a_4 = 3, a_5 = 4, a_6 = 5, a_7 = 5, a_8 = 4),$ $l_2 : (a_0 = -6.26, a_1 = 1, a_2 = -1, a_3 = 2, a_4 = -3, a_5 = 4, a_6 = -5, a_7 = 5, a_8 = -4),$ $l_3 : (a_0 = 9.19, a_1 = 1, a_2 = -1, a_3 = -2, a_4 = -3, a_5 = -4, a_6 = -5, a_7 = 5, a_8 = 4),$ $l_4 : (a_0 = -5.73, a_1 = 1, a_2 = 3, a_3 = -2, a_4 = -3, a_5 = 1, a_6 = 3, a_7 = 7, a_8 = -3),$ $l_5 : (a_0 = -21.87, a_1 = 1, a_2 = 7, a_3 = 2, a_4 = -3, a_5 = 6, a_6 = 7, a_7 = 8, a_8 = 1),$ $l_6 : (a_0 = 0.00, a_1 = 1, a_2 = -1, a_3 = -7, a_4 = 7, a_5 = 4, a_6 = 3, a_7 = 1, a_8 = -4),$ $l_7 : (a_0 = -13.88, a_1 = 1, a_2 = 8, a_3 = 4, a_4 = -3, a_5 = -3, a_6 = -6, a_7 = 6, a_8 = 8),$ $l_8 : (a_0 = -20.24, a_1 = 1, a_2 = 10, a_3 = 1, a_4 = -3, a_5 = -4, a_6 = 10, a_7 = 6, a_8 = -8)$						
Initial state of the robot arm links	$L_1 : \{(0, 0), (1, 1)\}, L_2 : \{(1, 1), (2, 2)\}, L_3 : \{(2, 2), (3, 3)\},$ $L_4 : \{(3, 3), (4, 2)\}, L_5 : \{(4, 2), (5, 1)\}, L_6 : \{(5, 1), (6, 0)\},$ $L_7 : \{(6, 0), (6.5, 0.5)\}, L_8 : \{(6, 0), (6.5, 0.5)\}$						
Final state of the robot arm links	$L_1 : \{(0, 0), (0, \sqrt{2})\}, L_2 : \{(0, \sqrt{2}), (0, 2\sqrt{2})\},$ $L_3 : \{(0, 2\sqrt{2}), (0, 3\sqrt{2})\}, L_4 : \{(0, 3\sqrt{2}), (1, 5.24)\},$ $L_5 : \{(1, 5.24), (2, 6.24)\}, L_6 : \{(2, 6.24), (3.39, 6.5)\},$ $L_7 : \{(3.39, 6.5), (4.05, 6.75)\}, L_8 : \{(3.39, 6.5), (4.05, 6.25)\}$						

Figure 15 shows how the robotic arm moves from the initial position to the final position to grip the circular object. The robotic arm goes through the narrow corridor whilst

the gripper opens gradually to hold the object; the HPPM-PRA considers the robotic arm and gripper as a unified structure during simulation. In this case study, a simple gripper was proposed, but it can be modeled in different ways, depending on the specific needs.

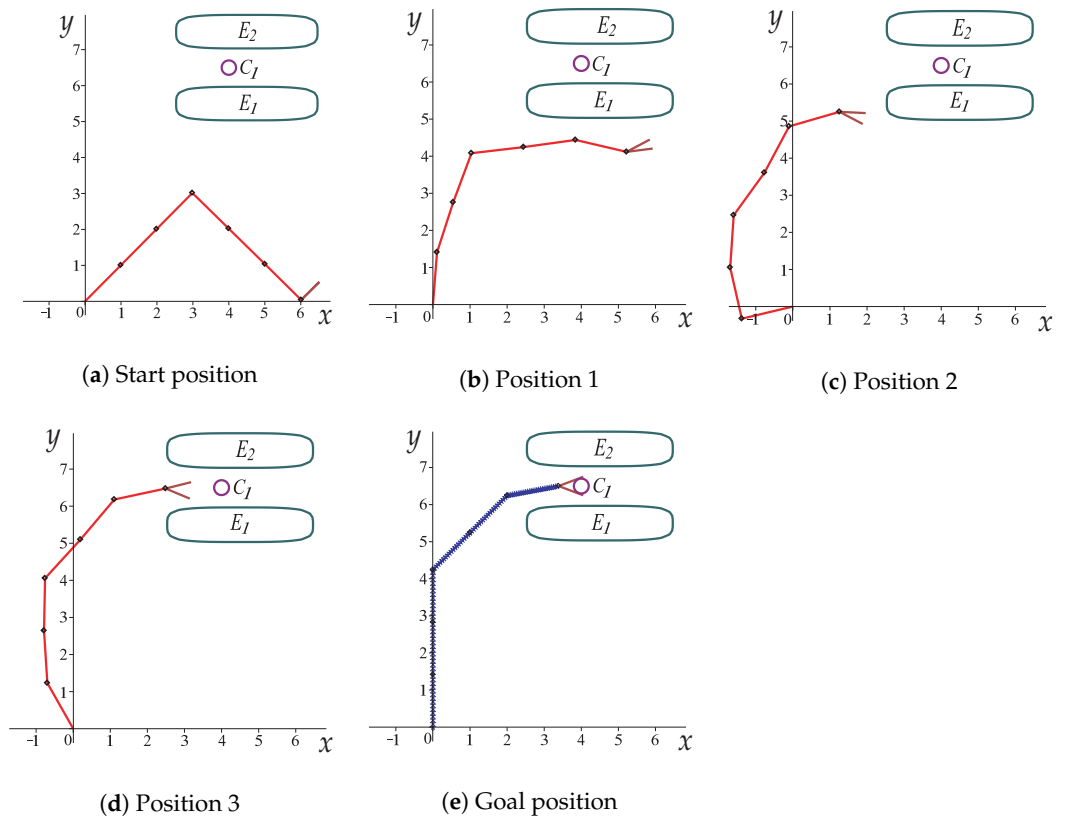


Figure 15. Movement of a six-link planar robot arm with a gripper, obstacles, and a circular object to grip.

Figure 16 shows the angle w of each joint for the full path; the points marked in Figure 16 correspond to the images in Figure 15. Figure 17 shows the change in joint angle movement, where it is observed that the greatest change obtained between each movement does not exceed 0.03 radians for the full path, so the path is smooth.

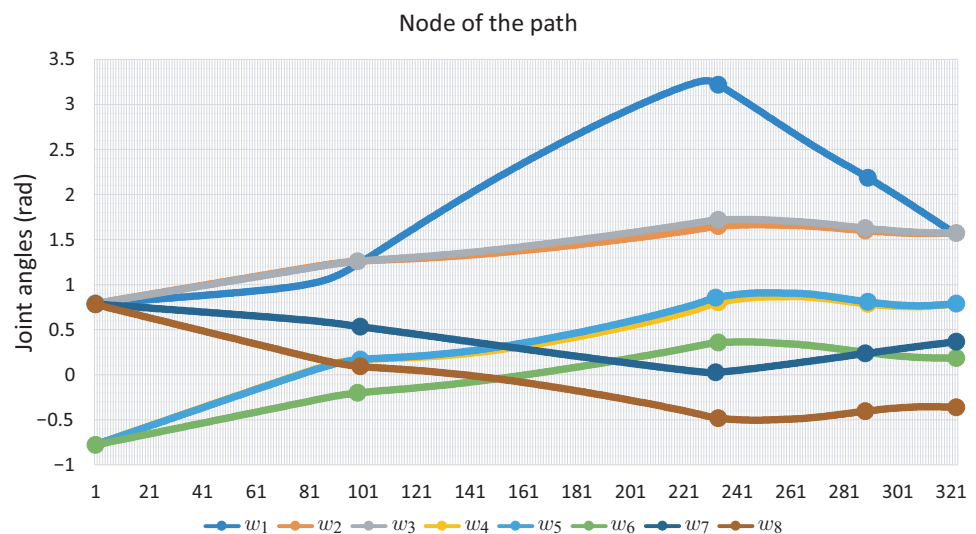


Figure 16. Joint angle motion for Case Study 2.

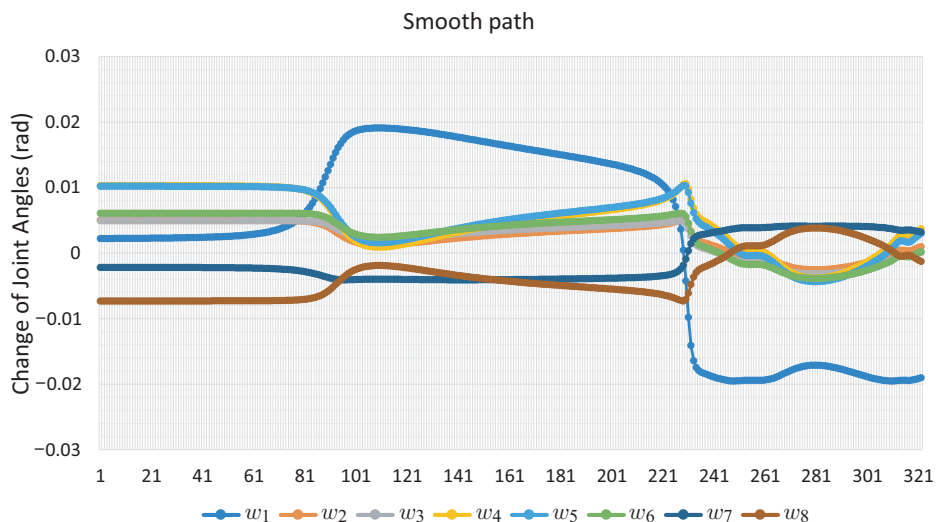


Figure 17. Change of joint angle motion of Case Study 2.

5.3. Case Study 3

This case study aims to show the suitability of the proposed method applied to a hyper-redundant robot. For this, a twenty-link hyper-redundant robot arm is used. Figure 18 depicts the scenario proposed for this case study. The robot arm must evade the circular obstacle C_2 , continue to move through the path formed by objects E_1 – E_5 , and finally, grab the circular object (C_1). In this scenario, a gripper is no longer necessary, since the robotic arm can grab the object with its links; this is an advantage of the hyper-redundant arms [10,11].

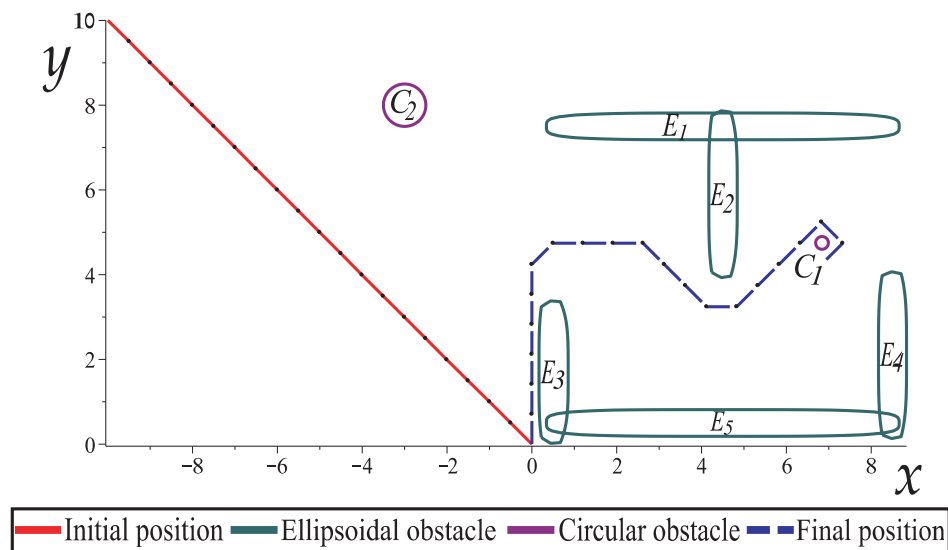


Figure 18. Workspace of a planar robotic arm with twenty link, five ellipsoidal obstacles, and one circular obstacle.

Figure 19 presents the sequence of robot arm simulation of the study case 3. In this picture, each position corresponds to a marked point in Figure 20. Figure 20 shows the evolution of the angles w along the path obtained for Case Study 3.

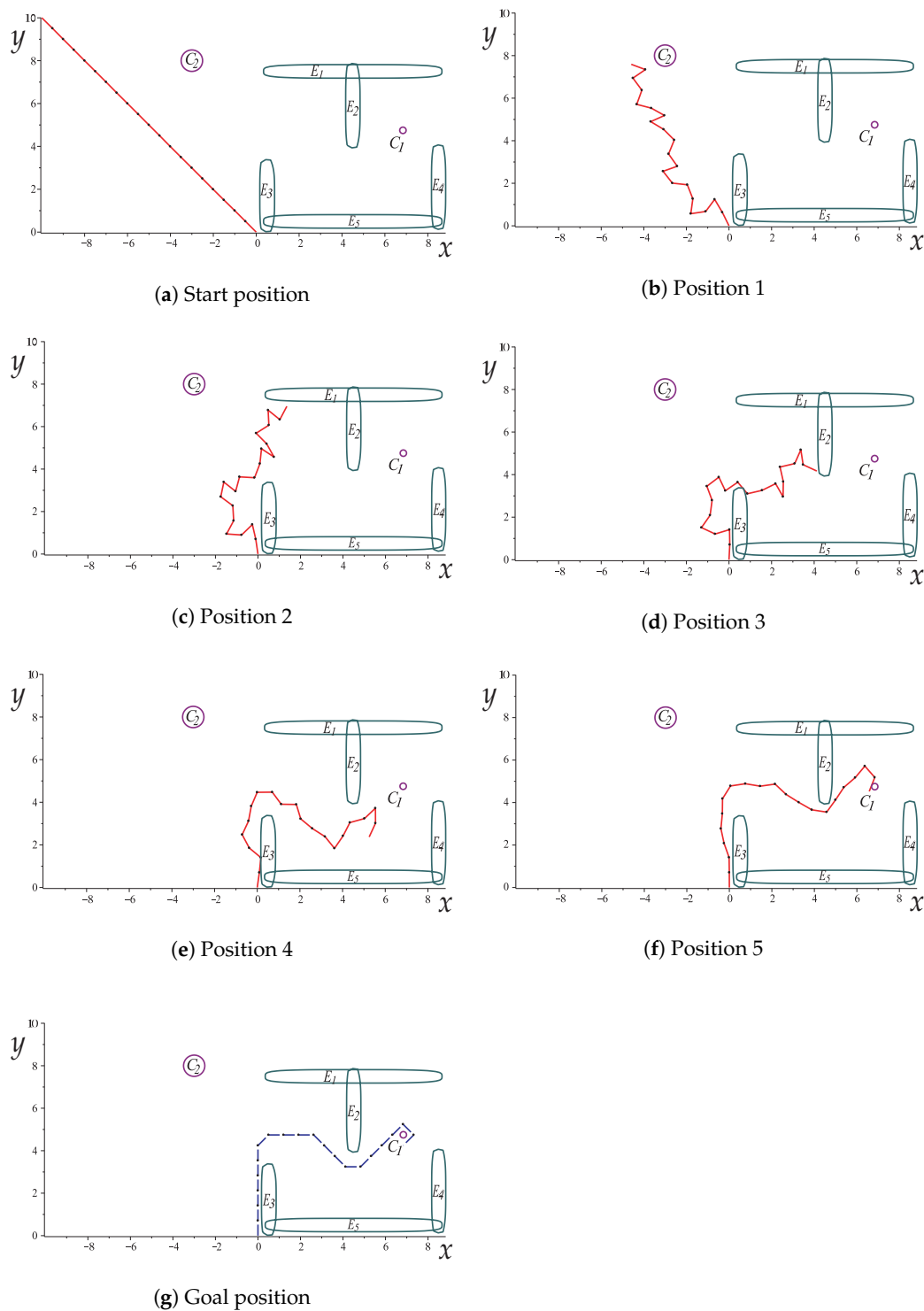


Figure 19. The sequence of images that describe the collision-free path of Case Study 3.

Figure 20 shows the evolution of the angles w along the path obtained for this. It can be seen that it exhibits a smooth displacement throughout the points that correspond to the images in Figure 21. The smoothness of the path can be validated by the nature of the homotopy continuation methods, which generate a continuous solution curve. Figure 21 shows the change in joint angle movement, where it is observed that the greatest change obtained between each movement does not exceed 0.02 radians for the full path, so the path is smooth.

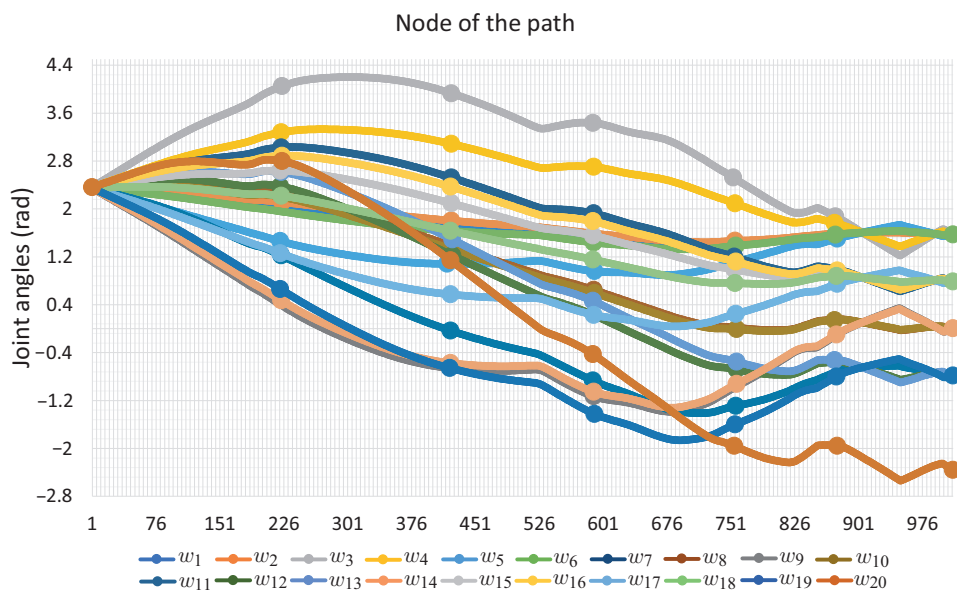


Figure 20. Joint w -angle motion for Case Study 3.

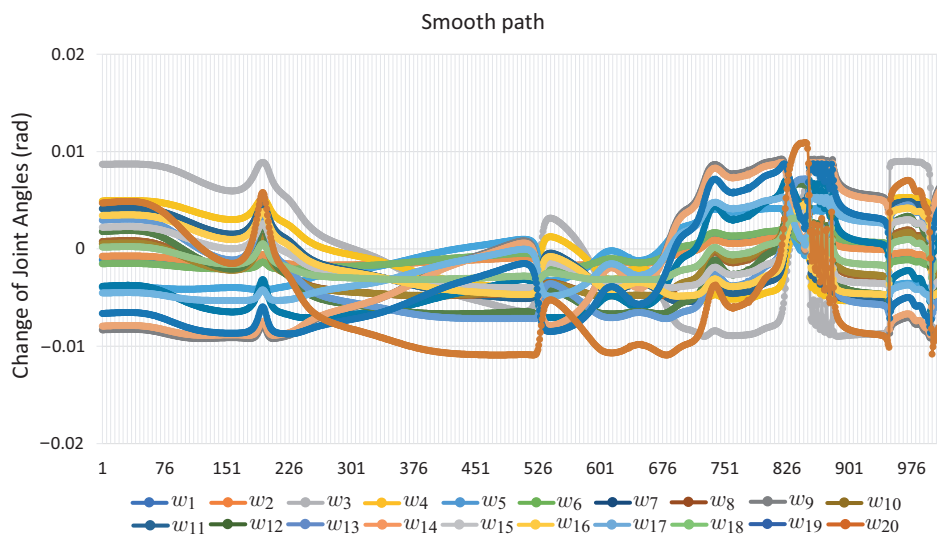


Figure 21. Change of joint angle motion of Case Study 3.

Table 3 shows the simulation parameters. The successful implementation of the HPPM-PRA on hyper-redundant robot arms to compute a collision-free path opens the possibility for the method to be applied to various areas of science such as medicine and exploration, among others [5,10,12–14,44,45].

Table 4 shows the computation time, memory consumption, total number of hyperspheres used to trace the homotopy path, radius of the hyperspheres, and characteristics of the workspace (number of links, the amount and type of obstacles).

Table 3. Parameters of Case Study 3.

Obstacle	Type of Obstacle	x_c	y_c	r_c	α_c	β_c	P
C_1	Circular	6.84	4.75	0.15	-	-	0.0000001
C_2	Circular	-3.0	8.0	0.5	-	-	-1.2
E_1	Ellipsoid	4.5	7.5	-	300.0	0.01	450.0
E_2	Ellipsoid	4.5	5.9	-	0.02	15.0	60.0
E_3	Ellipsoid	8.5	2.1	-	0.02	15.0	-0.0001
E_4	Ellipsoid	4.5	0.5	-	300.0	0.01	10.0
E_5	Ellipsoid	0.5	1.7	-	0.02	8.0	-63.0
Link length	$L_{1-20} = \sqrt{0.5}$						
Constants of auxiliary equations	$l_1 : (a_0 = -42.41, a_1 = 1, a_2 = 1, a_3 = 2, a_4 = 3, a_5 = 4, a_6 = 5, a_7 = 5, a_8 = 4, a_9 = 3, a_{10} = 2, a_{11} = 1, a_{12} = -5, a_{13} = -4, a_{14} = -3, a_{15} = -2, a_{16} = -1, a_{17} = -1, a_{18} = -2, a_{19} = -3, a_{20} = -4),$ $l_2 : (a_0 = 18.84, a_1 = 1, a_2 = -1, a_3 = 9, a_4 = 7, a_5 = -2, a_6 = 1, a_7 = -5, a_8 = -4, a_9 = -3, a_{10} = 6, a_{11} = -1, a_{12} = 5, a_{13} = 3, a_{14} = -6, a_{15} = -8, a_{16} = 1, a_{17} = -10, a_{18} = 2, a_{19} = 3, a_{20} = 8)$						
Initial state of the robot arm links	$L_1 : (-0.5, 0.5), L_2 : (-1, 1), L_3 : (-1.5, 1.5), L_4 : (-2, 2),$ $L_5 : (-2.5, 2.5), L_6 : (-3, 3), L_7 : (-3.5, 3.5), L_8 : (-4, 4),$ $L_9 : (-4.5, 4.5), L_{10} : (-5, 5), L_{11} : (-5.5, 5.5), L_{12} : (-6, 6),$ $L_{13} : (-6.5, 6.5), L_{14} : (-7, 7), L_{15} : (-7.5, 7.5), L_{16} : (-8, 8),$ $L_{17} : (-8.5, 8.5), L_{18} : (-9, 9), L_{19} : (-9.5, 9.5), L_{20} : (-10, 10)$						
Final state of the robot arm links	$L_1 : (0, \sqrt{2}), L_2 : (0, 2\sqrt{2}), L_3 : (0, 3\sqrt{2}), L_4 : (0, 4\sqrt{2}), L_5 : (0, 5\sqrt{2}),$ $L_6 : (0, 6\sqrt{2}), L_7 : (0.5, 4.74), L_8 : (1.2, 4.74), L_9 : (1.91, 4.74),$ $L_{10} : (2.62, 4.74), L_{11} : (3.12, 4.24), L_{12} : (3.62, 3.74), L_{13} : (4.12, 3.24),$ $L_{14} : (4.82, 3.24), L_{15} : (5.32, 3.74), L_{16} : (5.82, 4.24), L_{17} : (6.32, 4.74),$ $L_{18} : (6.82, 5.24), L_{19} : (7.32, 4.74), L_{20} : (6.82, 4.24)$						

Table 4. Results obtained in the three case studies carried out.

Study Case	Time	Memory	Hyperspheres	Hypersphere Radius	Number of Links	Circular Obstacle	Ellipsoid Obstacle
1	3.3 ms	1.404 KB	146	0.02	3	2	-
2	61.1 ms	4.308 KB	323	0.02	8	1	2
3	2.71 s	18.272 KB	1012	0.02	20	2	5

It can be seen that the computation time and memory consumption increase as the number of links and obstacles increases. The ellipsoidal obstacles have a higher computational cost than the circular obstacles. This is because the numerical problems that path tracking faces are due to the exponent of the ellipsoid formulation. The number of hyperspheres shown in Table 4 indicates the total movements made by the robotic arm to complete the task, so the proposed method obtains smooth paths for all case studies; this can be validated with Figures 11, 12, 16, 17, 20 and 21. The nature of homotopy is to create smooth paths to solve the NAES; therefore, the paths obtained with the proposed method will be smooth. The radius of the hypersphere was assigned as 0.02 with the purpose of standardizing the value, but this can be changed for any case study. It is suggested to use a value between 0.001 and 0.04; the smaller the size, the greater the total number of hyperspheres to be obtained is and vice versa.

Case Study 1 was the one that obtained the best results in CPU time, hyperspheres, and memory consumption. This is because the complexity of this workspace is minor since it only has two obstacles and an arm with three links. For Case Study 2 and Case Study 3, the computation time and memory consumption increased because of the greater number of obstacles and links and the presence of ellipsoidal obstacles.

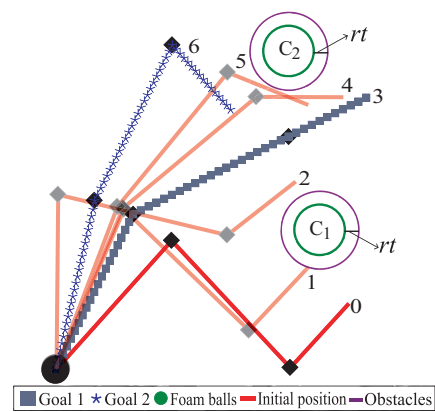
In the first two case studies, the computation time was less than 65 milliseconds and the memory consumption was less than 5 KB; this means that the proposed method can be implemented in embedded systems with limited memory. The memory consumption in all cases was less than 20 KB, demonstrating the low memory consumption of the proposed method. In contrast, the state-of-the-art hyper-redundant robotic arms, such as the one proposed in Case Study 3, require high-performance equipment with outstanding processor and memory consumption. They even use mathematical strategies to obtain favorable results [10,11,14], not to mention that, sometimes, they cannot find the path or only work with specific scenarios.

6. Implementation of the Proposed Method in the CRS Catalyst-5 Robot

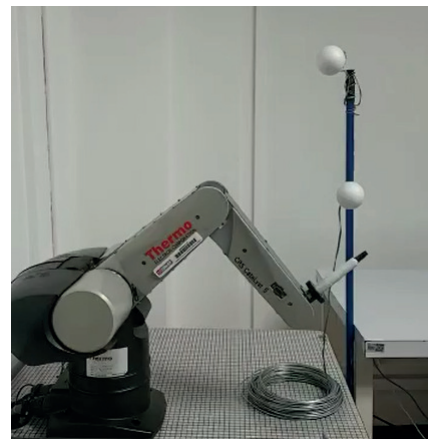
The proposed method was validated through its implementation on a real robotic arm model, CRS CataLyst-5. The method so far only works with planar robot arms; thus, the movement of the CRS-CataLyst-5 robot was limited to two axes.

To carry out the implementation, the test was divided into two stages:

1. First, the workspace to be solved was established. A three-link robot arm with normalized dimensions regarding the CRS CataLyst-5 robot was used. The circular obstacles had a tolerance radius rt , which guaranteed no collision of the robot arm with the obstacle (foam balls). For this case, two goals (goal1 and goal2) were set and are depicted in Figure 22 by the line formed by gray boxes and the blue line formed by asterisks, respectively. The movements of the robot arm are semi-transparent. The robot arm first reaches Goal 1 (the first homotopic path has been followed). Then, the endpoint of this path is used as the starting position to obtain the second homotopic path and reach Goal 2. In this way, a single path is obtained capable of avoiding obstacles and meeting both goals. The computation time and memory consumption were 2 milliseconds and 0.924 KB, respectively. The sequence of movements is executed by the robotic arm, as shown in Figure 22a.
2. The second stage of this process is to adjust the numeric homotopy path data to the correct instructions for the CRS CataLyst-5 arm to follow the path. The CRS-CataLyst-5 robot has five degrees of freedom, a teach pendant, and a controller for interpreting and processing the instructions sent by the computer through its Robcomm3 software to generate the movements of the robot [46,47]. Figure 22b depicts the robotic arm workspace. From Figure 23 (implementation), the sequence of movements of Figure 22a (simulation) is corroborated.

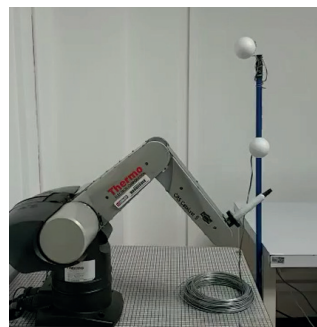


(a)

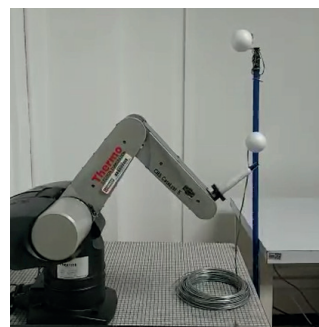


(b)

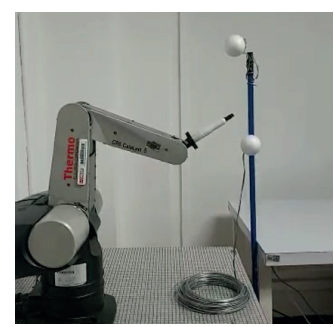
Figure 22. Path planning for the CRS CataLyst-5 robot. (a) Robot arm simulation. (b) Configuration of the CRS CataLyst-5 robot arm.



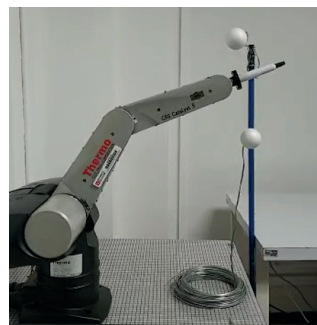
(a) Start position (0)



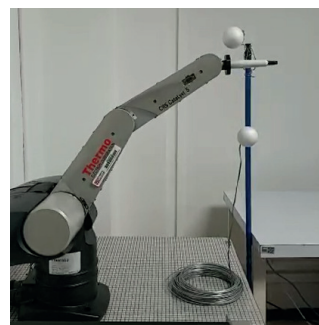
(b) 1



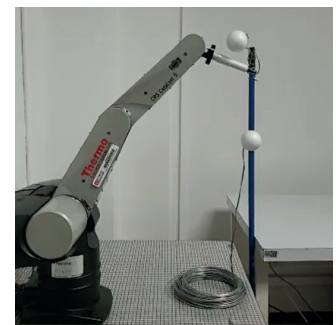
(c) 2



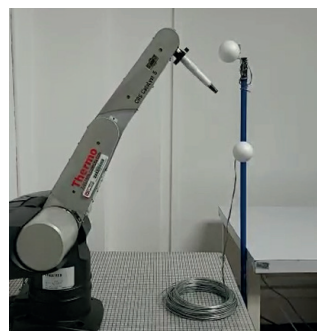
(d) Goal 1 (3)



(e) 4



(f) 5



(g) Goal 2 (6)

Figure 23. Sequence of images that describe the movement of the CRS CataLyst-5 robot avoiding two obstacles.

7. Conclusions and Future Work

In this work, a novel method for collision-free path planning for robotic arms using homotopy continuation methods was presented. This proposal is flexible since it is capable of working in the simulation with different characteristics of the robotic arm, such as different link lengths, an arbitrary number of links, or adding grippers. Moreover, the homotopy path-planning method for planar robotic arms (HPPM-PRA) can work with circular and ellipsoidal obstacles. Obstacle avoidance is achieved by strategically adding mathematical singular points to the links. The behavior of the evasion is controlled by the repulsion parameter assigned to each obstacle, which allows obtaining a path that approaches or moves away from each obstacle when searching for the goal position.

The HPPM-PRA was tested in three different scenarios to validate the capability of the method to work with different robot arms. The results of the case studies showed 20 KB of maximum memory expended in the implementations. These results validate the low memory consumption of the HPPM-PRA. A remarkable result of the HPPM-PRA is that it can reach a goal while avoiding obstacles and crossing narrow corridors using low computational resources, which is a complex task for probabilistic methods. The HPPM-PRA method was validated with a CRS CataLyst-5 robot for practical implementations. For this, it was necessary to consider a normalized workspace and the limitations of the software and hardware of the CRS CataLyst-5 robot.

The HPPM-PRA represents a novel proposal for the homotopy continuation methods [22–24,35–43,48,49] because it faces the problem of path planning by representing the configuration space (C-space) of the robot arm by using a system of algebraic equations and strategically allocated singularities, which are fundamental during the process of circumventing the obstacles. The proof of concept proved to be effective for implementation on real robots and prepares the way for implementation on scrolling robots in 3D configuration spaces. However, the extension of this project to 3D environments is left as future work, since the representation of the configuration space must be modified in the HPPM formulation by using the Denavit–Hartenberg representation.

Author Contributions: Conceptualization, H.V.-L. and G.D.-A.; Data curation, J.H.-C. and V.M.J.-F.; Formal analysis, G.C.V.-L., H.V.-L. and G.D.-A.; Investigation, G.C.V.-L. and L.H.-M.; Methodology, H.V.-L., G.D.-A. and H.D.R.-A.; Project administration, L.H.-M. and A.S.-R.; Resources, J.H.-C., H.D.R.-A. and V.M.J.-F.; Software, G.C.V.-L. and G.D.-A.; Supervision, L.H.-M., A.S.-R., G.D.-A., J.H.-C., H.D.R.-A. and V.M.J.-F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: Gerardo Cesar Velez Lopez gratefully acknowledges the financial support provided by the National Council for Science and Technology of Mexico (CONACyT) through an academic scholarship under contract 711139.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Baek, D.; Hwang, M.; Kim, H.; Kwon, D.S. Path planning for automation of surgery robot based on probabilistic roadmap and reinforcement learning. In Proceedings of the 2018 15th International Conference on Ubiquitous Robots (UR), Honolulu, HI, USA, 26–30 June 2018; pp. 342–347.
2. Bauzano, E.; Estebanez, B.; Garcia-Morales, I.; Munoz, V.F. Collaborative human–robot system for hals suture procedures. *IEEE Syst. J.* **2014**, *10*, 957–966. [[CrossRef](#)]
3. Roy, R.; Mahadevappa, M.; Kumar, C. Trajectory path planning of EEG controlled robotic arm using GA. *Procedia Comput. Sci.* **2016**, *84*, 147–151. [[CrossRef](#)]

4. Kopperger, E.; List, J.; Madhira, S.; Rothfischer, F.; Lamb, D.C.; Simmel, F.C. A self-assembled nanoscale robotic arm controlled by electric fields. *Science* **2018**, *359*, 296–301. [[CrossRef](#)] [[PubMed](#)]
5. Flores-Abad, A.; Ma, O.; Pham, K.; Ulrich, S. A review of space robotics technologies for on-orbit servicing. *Prog. Aerosp. Sci.* **2014**, *68*, 1–26. [[CrossRef](#)]
6. Kularatne, D.; Bhattacharya, S.; Hsieh, M.A. Optimal path planning in time-varying flows using adaptive discretization. *IEEE Robot. Autom. Lett.* **2017**, *3*, 458–465. [[CrossRef](#)]
7. Grushko, S.; Vysocký, A.; Oščádal, P.; Vocetka, M.; Novák, P.; Bobovský, Z. Improved Mutual Understanding for Human-Robot Collaboration: Combining Human-Aware Motion Planning with Haptic Feedback Devices for Communicating Planned Trajectory. *Sensors* **2021**, *21*, 3673. [[CrossRef](#)]
8. Zhao, S.; Zhao, J.; Sui, D.; Wang, T.; Zheng, T.; Zhao, C.; Zhu, Y. Modular Robotic Limbs for Astronaut Activities Assistance. *Sensors* **2021**, *21*, 6305. [[CrossRef](#)]
9. Rybus, T.; Wojtunik, M.; Basmadji, F.L. Optimal collision-free path planning of a free-floating space robot using spline-based trajectories. *Acta Astronaut.* **2022**, *190*, 395–408. [[CrossRef](#)]
10. Martín Barrio, A.; Terrile, S.; Barrientos, A.; del Cerro, J. Robots Hiper-Redundantes: Clasificación, Estado del Arte y Problemática. *Rev. Iberoam. Autom. Inform. Ind.* **2018**, *15*, 351–362. [[CrossRef](#)]
11. Cowan, L.S.; Walker, I.D. The importance of continuous and discrete elements in continuum robots. *Int. J. Adv. Robot. Syst.* **2013**, *10*, 165. [[CrossRef](#)]
12. Lau, K.; Leung, E.; Poon, C.; Lau, J.; Yam, Y.; Chiu, P. Development of an Endoscopic Surgical Robotic System—from Bench to Animal Studies. In Proceedings of the Hamlyn Symposium on Medical Robotics, London, UK, 25–28 June 2017; p. 1.
13. Berthet-Rayne, P.; Gras, G.; Leibrandt, K.; Wisanuvej, P.; Schmitz, A.; Seneci, C.A.; Yang, G.Z. The i 2 snake robotic platform for endoscopic surgery. *Ann. Biomed. Eng.* **2018**, *46*, 1663–1675. [[CrossRef](#)] [[PubMed](#)]
14. Deng, J.; Meng, B.H.; Kanj, I.; Godage, I.S. Near-optimal Smooth Path Planning for Multisection Continuum Arms. In Proceedings of the 2019 2nd IEEE International Conference on Soft Robotics (RoboSoft), Seoul, Korea, 14–18 April 2019; pp. 416–421.
15. Shiller, Z. Off-line and on-line trajectory planning. In *Motion and Operation Planning of Robotic Systems*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 29–62.
16. LaValle, S.M. *Planning Algorithms*; Cambridge University Press: Cambridge, UK, 2006.
17. Murray, S.; Floyd-Jones, W.; Qi, Y.; Sorin, D.J.; Konidaris, G.D. Robot Motion Planning on a Chip. In Proceedings of the Robotics: Science and Systems, Ann Arbor, MI, USA, 18–22 June 2016.
18. Brass, P.; Vigan, I.; Xu, N. Shortest path planning for a tethered robot. *Comput. Geom.* **2015**, *48*, 732–742. [[CrossRef](#)]
19. Fu, B.; Chen, L.; Zhou, Y.; Zheng, D.; Wei, Z.; Dai, J.; Pan, H. An improved A* algorithm for the industrial robot path planning with high success rate and short length. *Robot. Auton. Syst.* **2018**, *106*, 26–37. [[CrossRef](#)]
20. Baghli, F.Z.; Elbakkali, L.; Lakhal, Y. Optimization of arm manipulator trajectory planning in the presence of obstacles by ant colony algorithm. *Procedia Eng.* **2017**, *181*, 560–567. [[CrossRef](#)]
21. Gai, S.N.; Sun, R.; Chen, S.J.; Ji, S. 6-DOF robotic obstacle avoidance path planning based on artificial potential field method. In Proceedings of the 2019 16th International Conference on Ubiquitous Robots (UR), Jeju, Korea, 24–27 June 2019; pp. 165–168.
22. Vazquez-Leal, H.; Marin-Hernandez, A.; Khan, Y.; Yıldırım, A.; Filobello-Nino, U.; Castañeda-Sheissa, R.; Jimenez-Fernandez, V.M. Exploring collision-free path planning by using homotopy continuation methods. *Appl. Math. Comput.* **2013**, *219*, 7514–7532.
23. Diaz-Arango, G.; Vázquez-Leal, H.; Hernandez-Martinez, L.; Pascual, M.T.S.; Sandoval-Hernandez, M. Homotopy path planning for terrestrial robots using spherical algorithm. *IEEE Trans. Autom. Sci. Eng.* **2017**, *15*, 567–585. [[CrossRef](#)]
24. Diaz-Arango, G.; Vázquez-Leal, H.; Hernandez-Martinez, L.; Jimenez-Fernandez, V.M.; Heredia-Jimenez, A.; Ambrosio, R.C.; Huerta-Chua, J.; Cos-Cholula, D.; Hernandez-Mendez, S. Multiple-Target Homotopic Quasi-Complete path-planning method for Mobile Robot Using a Piecewise Linear Approach. *Sensors* **2020**, *20*, 3265.
25. Liu, Y.R.; Huang, M.B.; Huang, H.P. Automated grasp planning and path planning for a robot hand-arm system. In Proceedings of the 2019 IEEE/SICE International Symposium on System Integration (SII), Paris, France, 14–16 January 2019; pp. 92–97.
26. Kingston, Z.; Moll, M.; Kavraki, L.E. Sampling-based methods for motion planning with constraints. *Annu. Rev. Control Robot. Auton. Syst.* **2018**, *1*, 159–185. [[CrossRef](#)]
27. Vien, N.A.; Toussaint, M. POMDP manipulation via trajectory optimization. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 242–249.
28. Samaniego, R.; Rodríguez, R.; Vázquez, F.; López, J. Efficient path planning for articulated vehicles in cluttered environments. *Sensors* **2020**, *20*, 6821.
29. Kang, T.W.; Kang, J.G.; Jung, J.W. A Bidirectional Interpolation Method for Post-Processing in Sampling-Based Robot Path Planning. *Sensors* **2021**, *21*, 7425.
30. Jeon, G.Y.; Jung, J.W. Water sink model for robot motion planning. *Sensors* **2019**, *19*, 1269. [[CrossRef](#)] [[PubMed](#)]
31. Cao, Y.; Guo, M.; Li, Y. Obstacle-free workspace based path planning for serial manipulator. In Proceedings of the 2017 36th Chinese Control Conference (CCC), Dalian, China, 26–28 July 2017; pp. 6897–6900.
32. Jena, A.; Sahu, P.K.; Bharat, S.C.; Biswal, B. Optimal trajectory planning of a 3R SCARA manipulator using geodesic. In Proceedings of the 2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), Delhi, India, 4–6 July 2016; pp. 1–6.

33. Chembuly, V.S.; Voruganti, H.K. Trajectory planning of redundant manipulators moving along constrained path and avoiding obstacles. *Procedia Comput. Sci.* **2018**, *133*, 627–634. [[CrossRef](#)]
34. De Luca, A.; Oriolo, G. Trajectory planning and control for planar robots with passive last joint. *Int. J. Robot. Res.* **2002**, *21*, 575–590. [[CrossRef](#)]
35. Shafiee-Ashtiani, M.; Yousefi-Koma, A.; Iravanimesh, S.; Bashardoust, A.S. Kinematic analysis of a 3-UPU parallel Robot using the Ostrowski-Homotopy Continuation. In Proceedings of the 2016 24th Iranian Conference on Electrical Engineering (ICEE), Shiraz, Iran, 10–12 May 2016; pp. 1306–1311.
36. Nor, H.M.; Rahman, A.; Ismail, A.I.M.; Majid, A.A. Super Ostrowski homotopy continuation method for solving polynomial system of equations. *MATEMATIKA Malays. J. Ind. Appl. Math.* **2016**, *32*, 53–67.
37. Jiménez-Islas, H.; Martínez-González, G.M.; Navarrete-Bolaños, J.L.; Botello-Álvarez, J.E.; Oliveros-Munoz, J.M. Nonlinear homotopic continuation methods: A chemical engineering perspective review. *Ind. Eng. Chem. Res.* **2013**, *52*, 14729–14742. [[CrossRef](#)]
38. Dong, B.; Yu, B.; Yu, Y. A homotopy method for finding all solutions of a multiparameter eigenvalue problem. *SIAM J. Matrix Anal. Appl.* **2016**, *37*, 550–571. [[CrossRef](#)]
39. Jiménez-Islas, H.; Calderón-Ramírez, M.; Martínez-González, G.M.; Calderón-Álvarado, M.P.; Oliveros-Muñoz, J.M. Multiple solutions for steady differential equations via hyperspherical path-tracking of homotopy curves. *Comput. Math. Appl.* **2020**, *79*, 2216–2239. [[CrossRef](#)]
40. Mirmohammad, S.H.; Yousefi-Koma, A.; Mohtasebi, S.S. Direct kinematics of a three revolute-prismatic-spherical parallel robot using a fast homotopy continuation method. In Proceedings of the 2016 4th International Conference on Robotics and Mechatronics (ICROM), Tehran, Iran, 26–28 October 2016; pp. 410–415.
41. Gregoire, J.; Čáp, M.; Frazzoli, E. Locally optimal multi-robot navigation under delaying disturbances using homotopy constraints. *Auton. Robot.* **2018**, *42*, 895–907. [[CrossRef](#)]
42. Vázquez-Leal, H.; Castañeda-Sheissa, R.; Rabago-Bernal, F.; Hernández-Martínez, L.; Sarmiento-Reyes, A.; Filobello-Niño, U. Powering multiparameter homotopy-based simulation with a fast path-following technique. *ISRN Appl. Math.* **2011**, *2011*, 610637. [[CrossRef](#)]
43. Velez-Lopez, G.C.; Hernández-Martínez, L.; Diaz-Arango, G.; Vazquez-Leal, H. A Tool to Solve Nonlinear Algebraic Equations Systems. In Proceedings of the 2019 16th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), Mexico City, Mexico, 11–13 September 2019; pp. 1–4.
44. Giorgio, I.; Del Vecovo, D. Non-linear lumped-parameter modeling of planar multi-link manipulators with highly flexible arms. *Robotics* **2018**, *7*, 60. [[CrossRef](#)]
45. Cianchetti, M.; Laschi, C.; Menciassi, A.; Dario, P. Biomedical applications of soft robotics. *Nat. Rev. Mater.* **2018**, *3*, 143–153. [[CrossRef](#)]
46. CRS Robotics Corporation (Thermo CRS Limited). *A255 Robot Arm User Guide For Use with C500C Controller*; CRS Robotics Corporation: Burlington, ON, Canada, 2000.
47. Rios, E.M.L. *Prácticas Didácticas del Laboratorio de Procesos Automatizados e Integrados por Computadora (LPAIC)*; Instituto Politecnico Nacional: Mexico City, Mexico, 2010.
48. Torres-Muñoz, D.; Hernandez-Martinez, L.; Vazquez-Leal, H. Spherical continuation algorithm with spheres of variable radius to trace homotopy curves. *Int. J. Appl. Comput. Math.* **2016**, *2*, 421–433. [[CrossRef](#)]
49. Vázquez-Leal, H.; Jimenez-Fernandez, V.; Benhammouda, B.; Filobello-Niño, U.; Sarmiento-Reyes, A.; Ramirez-Pinero, A.; Marin-Hernandez, A.; Huerta-Chua, J. Modified hyperspheres algorithm to trace homotopy curves of non-linear circuits composed by piecewise linear modelled devices. *Sci. World J.* **2014**, *2014*, 938598. [[CrossRef](#)] [[PubMed](#)]