

Research Article

Robot Path Planning with Avoiding Obstacles in Known Environment Using Free Segments and Turning Points Algorithm

Imen Hassani , Imen Maalej , and Chokri Rekek

Control and Energy Management Laboratory (CEM Lab), National Engineering School of Sfax, University of Sfax, Sfax, Tunisia

Correspondence should be addressed to Imen Hassani; imenhassani@yahoo.fr

Received 23 February 2018; Revised 24 April 2018; Accepted 13 May 2018; Published 11 June 2018

Academic Editor: Luis Gracia

Copyright © 2018 Imen Hassani et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Currently, the path planning problem is one of the most researched topics in autonomous robotics. That is why finding a safe path in a cluttered environment for a mobile robot is an important requirement for the success of any such mobile robot project. In this work, a developed algorithm based on free segments and a turning point strategy for solving the problem of robot path planning in a static environment is presented. The aim of the turning point approach is to search a safe path for the mobile robot, to make the robot moving from a starting position to a destination position without hitting obstacles. This proposed algorithm handles two different objectives which are the path safety and the path length. In addition, a robust control law which is called sliding mode control is proposed to control the stabilization of an autonomous mobile robot to track a desired trajectory. Finally, simulation results show that the developed approach is a good alternative to obtain the adequate path and demonstrate the efficiency of the proposed control law for robust tracking of the mobile robot.

1. Introduction

Nowadays, robots are considered as an important element in society. This is due to the replacement of humans by robots in basic and dangerous activities. However, designing an efficient navigation strategy for mobile robots and ensuring their securities are the most important issues in autonomous robotics.

Therefore, the path planning problem is one of the most interesting and researched topics. The aim of the robot path planning is to search a safe path for the mobile robot. Also the path is required to be optimal. In this sense, several research works tackling the path planning problem have been proposed in the literature [1–4]. Until now, many methods have been used for path planning of mobile robots. Among these strategies, the geometry space method such as Artificial Potential Field [5, 6], Agoraphobic Algorithm [7], and Vector Field Histogram [8, 9]. These methods give the heading angle for avoiding obstacles. The strategy of dynamic windows has been used in [10, 11]. This approach is a velocity-based local planner that calculates the optimal collision-free velocity

for a mobile robot. Another method used in [12] is named turning point searching algorithm which consists of finding a point around which the mobile robot turns without hitting obstacles.

In the other side, several research works for tracking control of a wheeled mobile robot have gained attention in the literature [13–16]. The nonholonomic system suffers of nonlinearity and uncertainty problem. Because of this uncertainty, the trajectory error for a wheeled mobile robot has always been produced and can not be eliminated. In this sense, many tracking methods are proposed in the literature as Proportional Integral Derive (PID) controller [17] but this controller becomes instable when it is affected by the sensor sensitivity [18]. Furthermore, a fuzzy logic controller is used in [19] but this control law has a slow response time due to the heavy computation [20]. Other works used sliding mode controller in various applications [15, 16]. The aim advantage of this control system is its insurance for stability, robustness, fast response, and good transient [21].

The aim of the developed strategy is to solve the problem when the robot is located between two obstacles such as

the following: how the robot can detect that the distance between the two obstacles is safe enough to reach the target without collision and how to avoid obstacles and move between two obstacles in the shortest path. That is why this work is based on selecting safe free segments in an environment encumbered by obstacles firstly. After that, a developed turning point searching algorithm is applied to determinate the endpoint of the safe free segment which gives the shortest path. This strategy is inspired from the approach given by Jinpyo and Kyihwan [12]. In fact, the strategy presented in [12] handles two fundamental objectives: the path length and the path safety. This approach is focused firstly on searching the endpoint of a free segment which gives the shortest path. Hence, if the distance of the free segment selected is larger than the robot diameter, the endpoint is considered as a turning point. If this is not the case, it must replay the algorithm to search a new endpoint of the free segments. The disadvantages of this strategy are that it is focused firstly on finding the shortest path without taking into consideration the safety and, after that, it is focused on ensuring a safe path navigation which leads to an extensive and heavy computation and needs more time for planning the adequate path for a mobile robot. In order to overcome these disadvantages, our developed algorithm serves to ensure at first the path safety by selecting the safest free segments. Then, it searches the path length by determining the endpoint of the safest free segments which gives the shortest path. Using this strategy, we can rapidly determine the safest and the shortest path. Moreover, once the path is planned, a tracking law based on sliding mode controller is used for the robot to follow the designed trajectory.

Our contribution is to develop a new algorithm for solving the problem of robot path planning with static obstacles avoidances. This planning, also called static path plan, presents the advantage of ensuring safety and shortness of the path. Moreover, the proposed algorithm is characterized by a reactive behavior to find a collision-free trajectory and smooth path. On the other side, the mobile robot should track the trajectory without collision with obstacles. So, a sliding mode control is proposed for guaranteeing robustness, stability, and reactivity.

The rest of this paper is organized as follows. Section 2 presents the mobile robot model used in this work. The different steps of the proposed algorithm for path planning purpose are described in detail in Section 3. In Section 4, a sliding mode controller is used for trajectory tracking. Finally, simulation results and conclusion are presented and analyzed in Sections 5 and 6, respectively.

2. Mobile Robot Model

Several research works for autonomous navigation have been applied to different types of mobile robots [22, 23]. In this work, we consider the Khepera IV mobile robot which has two independent driving wheels that are responsible for orienting and commanding the platform by acting on the speed of each wheel. Thus, the schematic model of the wheeled mobile robot Khepera IV is shown in Figure 1.

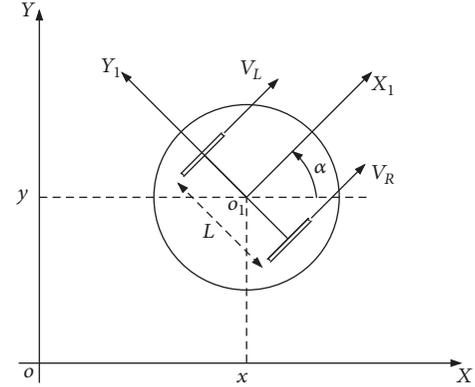


FIGURE 1: Schematic representation of Khepera IV.

The kinematic model of a nonholonomic mobile robot is given as follows:

$$\begin{aligned} \frac{dx}{dt} &= \frac{V_R + V_L}{2} \cos \alpha \\ \frac{dy}{dt} &= \frac{V_R + V_L}{2} \sin \alpha \\ \frac{d\alpha}{dt} &= \frac{V_R - V_L}{L} \end{aligned} \quad (1)$$

where (x, y) are the robot's Cartesian coordinates, α is the angle between the robot direction and X axis, V_R and V_L are, respectively, the robot right and left wheel velocities, and L is the distance between the two wheels.

3. Path Planning Algorithm

In order to solve the path planning problem, an algorithm based on finding the turning point of a free segment is proposed.

3.1. Principle of the Proposed Algorithm. A free segment is considered as the distance between two endpoints of two different obstacles (see Figure 2). It searches the endpoint of a safe segment where the mobile robot turns around this point without hitting obstacles.

When there are no obstacles, the path planning problem does not arise. In fact, the robot moves from an initial position p_i to a goal position p_g in a straight line which will be considered as the shortest path. However, when the mobile robot encounters with n obstacles as shown in Figure 2 ($n = 3$), the robot should be turning without collision with obstacles. So, the major problem is how to determinate a suitable path from a starting point to a target point in a static environment. To solve this problem our developed algorithm is proposed to search for a turning point of a safe free segment which gives the shortest path and allows the robot to avoid obstacles. Once the turning point is located, a dangerous circle with radius R_d is fixed in this point. In this case, our proposed strategy aims to search for the turning point P_{tp} of the safe free segment around which the robot turns safely.

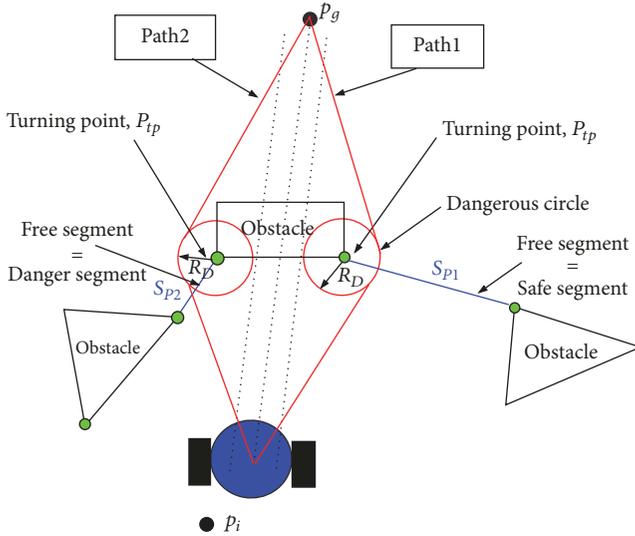


FIGURE 2: Framework of navigation.

For ensuring safety, we select the segment whose distance S_{p_i} ($i = 1, \dots, n - 1$) is larger than the robot diameter D_r with a margin for security δ ($S_{p_i} \geq D_r + \delta$). On the other hand, the segment whose distance S_{p_i} is smaller than the robot diameter is considered as a danger segment (see Figure 2). In this work, we take into account only safe segments and danger segments are ignored. Furthermore, and to determinate the shortest path, we have determined the point P_{tp} of the safest segment which gives the shortest path. Then a dangerous circle is fixed at this point and the robot turns and moves towards the tangential direction to this circle. Even when there is a danger problem, our proposed algorithm will be reactive to allow the robot to avoid obstacles and reach the goal. In this case, the robot reserves the determined turning point and searches for a new turning point to avoid collision with obstacles. To more clarify our strategy, the different notions of the algorithm are incorporated in Figure 2 and the basic principle is summarized in a flowchart presented in Figure 3.

3.2. Static Path Planning Steps. The aim of this section is to find a safe path as short as possible. In this approach, it is defined as the path having the tangential direction to the circle located on the searched turning point.

3.2.1. Selection of the Safe Path. The safe path aims to find a free path that helps the robot to reach the target without hitting obstacles of the environment. The selection of a safe segment needs to follow the next steps:

- (i) Step 1: Find out all free segments of the environment (see Figure 4). Equations (2) and (3) show how to determinate the value of the distance S_{p_1} that connects points p_2 and p_3 and the distance S_{p_2} that connects points p_4 and p_5 :

$$S_{p_1} = \sqrt{(x_{p_3} - x_{p_2})^2 + (y_{p_3} - y_{p_2})^2} \quad (2)$$

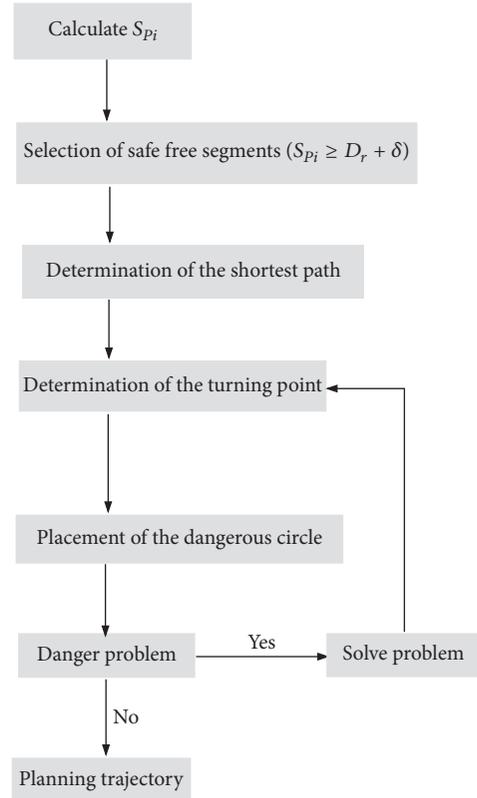


FIGURE 3: The proposed algorithm.

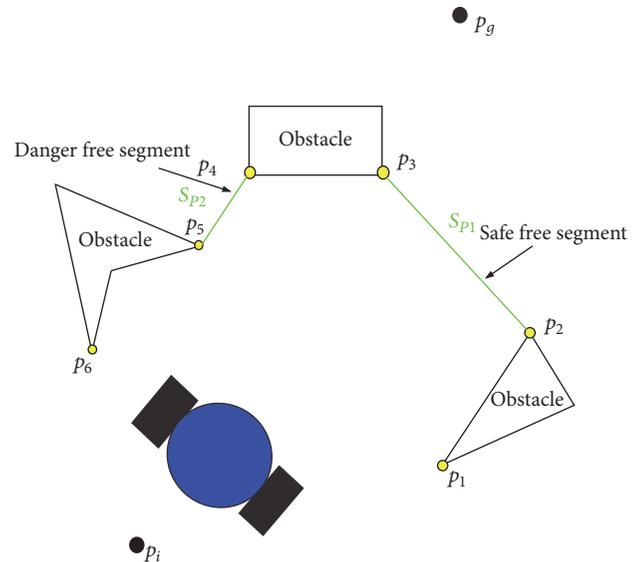


FIGURE 4: Determination of free segments (safe-danger).

$$S_{p_2} = \sqrt{(x_{p_5} - x_{p_4})^2 + (y_{p_5} - y_{p_4})^2} \quad (3)$$

where (x_{p_i}, y_{p_i}) ($i=2..5$) corresponds to the coordinate of endpoints of free segments.

- (ii) Step 2: The segment whose distance S_{p_i} ($i = 1, \dots, n - 1$) is larger than D_r is considered as a safe segment.

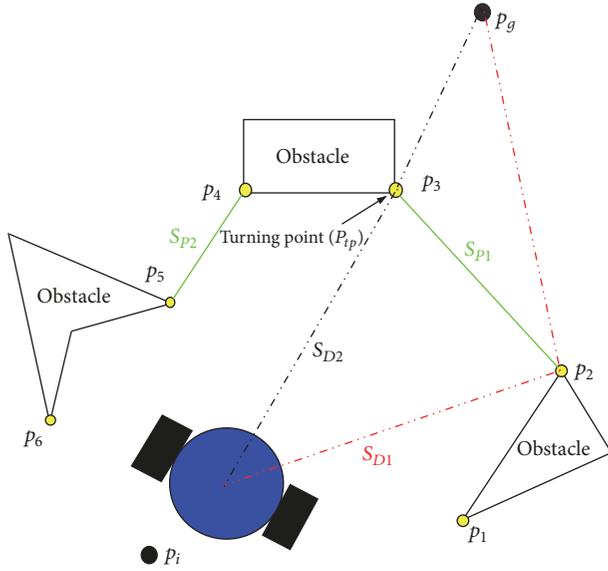


FIGURE 5: Determination of the shortest path.

However, the segment whose distance S_{p_i} is smaller than D_r is considered as a danger segment. Only safe segments are taken into consideration for the rest of this work. Danger segments whose number is n_d are ignored. In this step, we define the number of safe segments as

$$n_s = \begin{cases} n-1 & \text{if } S_{p_i} \geq (D_r + \delta) \\ n-1-n_d & \text{elsewhere.} \end{cases} \quad (4)$$

Once the safety criteria are handled, in the next section we are interested to determinate the shortest path.

3.2.2. Determination of the Shortest Path. When the robot goes to reach the target position, it is important to do it in the shortest path as possible. The objective of determining the shortest path can be divided into three steps:

- (i) Step 1: Calculate distances S_{D1} and S_{D2} between the robot and the target with consideration of the safe free segment (see Figure 5). These distances should be calculated as follows:

$$S_{D1} = \sqrt{(x_{p_2} - x)^2 + (y_{p_2} - y)^2} + \sqrt{(x_t - x_{p_2})^2 + (y_t - y_{p_2})^2} \quad (5)$$

$$S_{D2} = \sqrt{(x_{p_3} - x)^2 + (y_{p_3} - y)^2} + \sqrt{(x_t - x_{p_3})^2 + (y_t - y_{p_3})^2} \quad (6)$$

- (ii) Step 2: It concerns the determination of the turning point which is defined as the point around which the

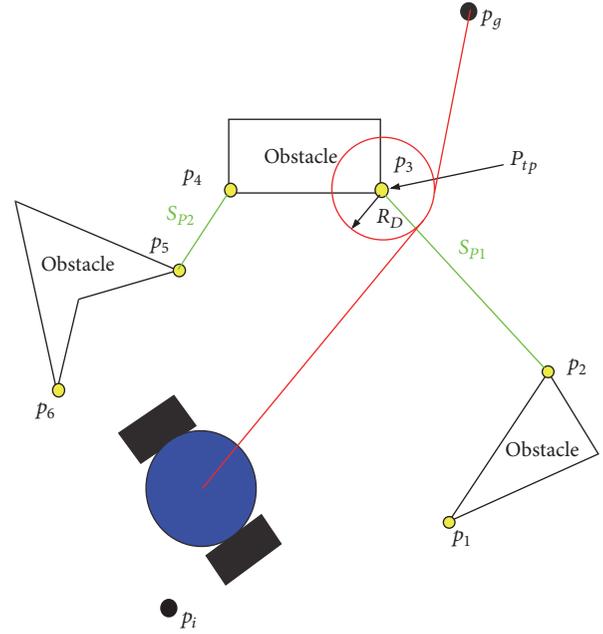


FIGURE 6: Dangerous circle placement.

mobile robot turns for avoiding obstacles; the process is achieved after comparing the distances S_{D1} and S_{D2} . The endpoint of the safe free segment which gives the shortest path corresponds to the searched turning point P_{tp} as shown in Figure 5.

- (iii) Step 3: It concerns the placement of the dangerous circle. Once the turning point is determined, a dangerous circle with radius R_D is fixed at this point as shown in Figure 6.

3.3. Problems Examination. Even the adequate path is determined, some problems can persist whose results make the robot damaged and can not avoid obstacles. Some problem cases are highlighted in this work.

3.3.1. Collision Danger Problem. Path planning problem means that the path should be safe enough to go through without collision. However, a collision danger problem can persist in some cases:

- (i) Case 1: If there is an intersection between the robot and the obstacle. To better concretize the problem, Figure 7 is given: path 1 presents an example of a mobile robot where it is entrapped by the obstacle and it can not avoid it. To remove the collision between the robot path and obstacle, path 2 is presented and turned around a second dangerous circle with radius R_D . So, we can conclude that path 2 is safe enough for the robot to go to the destination point without collision.
- (ii) Case 2: If the distance between the line tangent of the dangerous circle and the endpoint of an obstacle d (see Figure 8) is less than the robot radius ($d \leq R_r$),

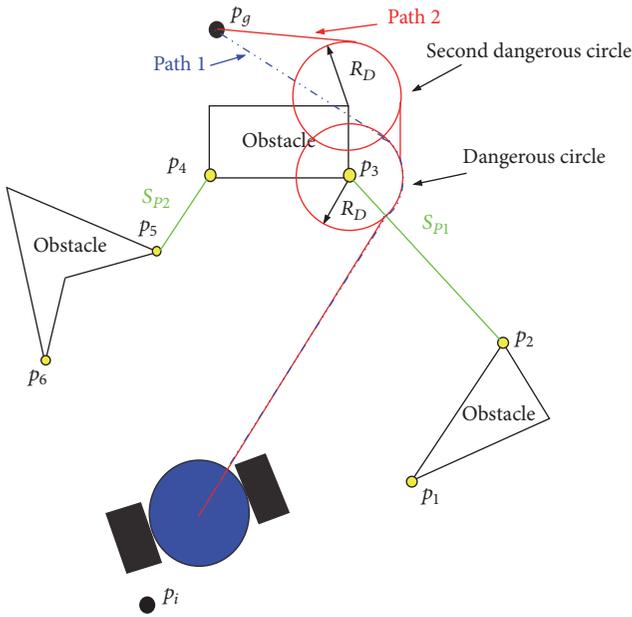


FIGURE 7: Collision danger problem-case 1.

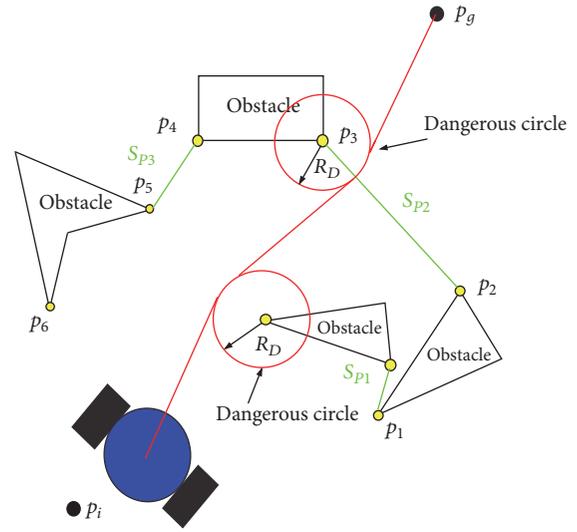


FIGURE 9: Placement of the turning point.

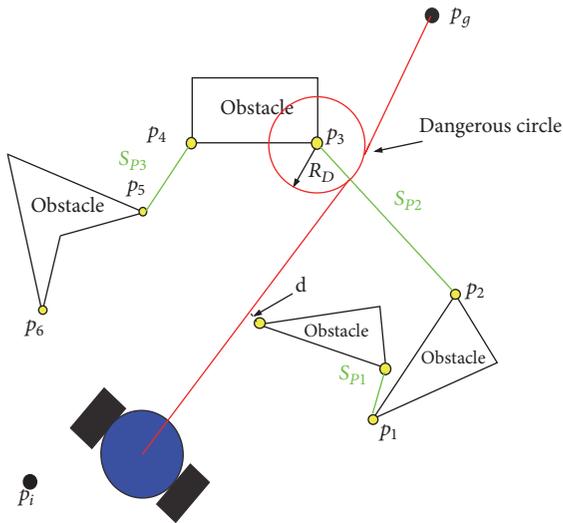


FIGURE 8: Collision danger problem-case 2.

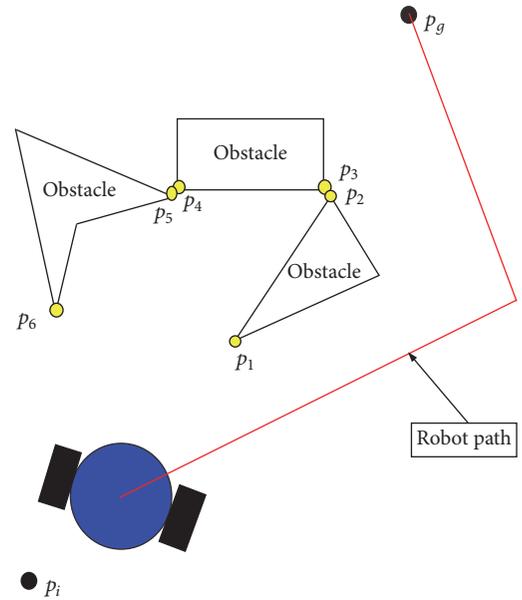


FIGURE 10: Local minima problem resolution.

a turning point algorithm is applied and a dangerous circle is centered at the adequate turning point (see Figure 9).

3.3.2. *Problem of Local Minima.* A local minima problem can exist when all segments are danger or the robot is entrapped with obstacles. To escape from such a situation, the robot goes far away from those obstacles until reaching the target (see Figure 10).

4. Sliding Mode Control

After planning the path of the robot Khepera IV, a sliding mode controller is proposed for robust tracking trajectory

([15, 16]). In this strategy, two positions are needed to be known as shown in Figure 11: the desired position $p_r=(x_r, y_r, \alpha_r)$ which is defined as the desired position to be reached and the current robot position $p=(x, y, \alpha)$ which is defined as its real position at this moment. Furthermore, the difference between the reference position p_r and the current position p is called the tracking error position $p_e=(x_e, y_e, \alpha_e)$. The expression of p_e is defined in equation (7) as follows:

$$p_e = \begin{bmatrix} x_e \\ y_e \\ \alpha_e \end{bmatrix} = \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_r - x \\ y_r - y \\ \alpha_r - \alpha \end{bmatrix}. \quad (7)$$

Tracking trajectory can be introduced as finding the adequate control vector $q = (v, w)^T$ (v is the linear velocity

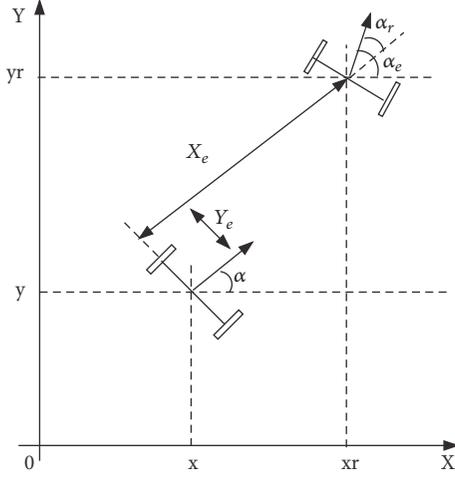


FIGURE 11: Tracking error.

of the wheeled mobile robot and w is its angular velocity). So that the error position p_e converges asymptotically to zero. The autonomous mobile robot is controlled according to

$$\begin{aligned} V_R &= v + \frac{L.w}{2} \\ V_L &= v - \frac{L.w}{2}. \end{aligned} \quad (8)$$

The process of designing a sliding mode controller is divided into two steps:

- (i) Step 1: The choice of the sliding surface: s is defined as the switching function because the control switches its sign on the sides of the switching $s = 0$. Therefore, $x_e = 0$ is chosen at the first switching function. When $x_e = 0$, the Lyapunov candidate function is defined as $V = (1/2)y_e^2$. Then, we determine the time derivative of V :

$$\begin{aligned} \dot{V} &= y_e \dot{y}_e = y_e (-x_e w + v_r \sin(\alpha_e)) \\ &= -x_e y_e w - v_r y_e \sin(\arctan(v_r y_e)). \end{aligned} \quad (9)$$

We notice that $\dot{V} \leq 0$ because $v_r y_e \sin(\arctan(v_r y_e)) \geq 0$. We define $\alpha_e = -\arctan(v_r y_e)$ as a switching candidate function. Then, the expression of the vector of sliding surfaces is given as follows:

$$s = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} x_e \\ \alpha_e + \arctan(v_r y_e) \end{bmatrix}. \quad (10)$$

- (ii) Step 2: The determination of the control law: the designing of a sliding mode controller needs firstly to establish an analytic expression of the adequate condition under which the state moves towards and reaches a sliding mode. However, a chattering phenomenon can be caused by the finite time delays for computations and limitations of control. That is

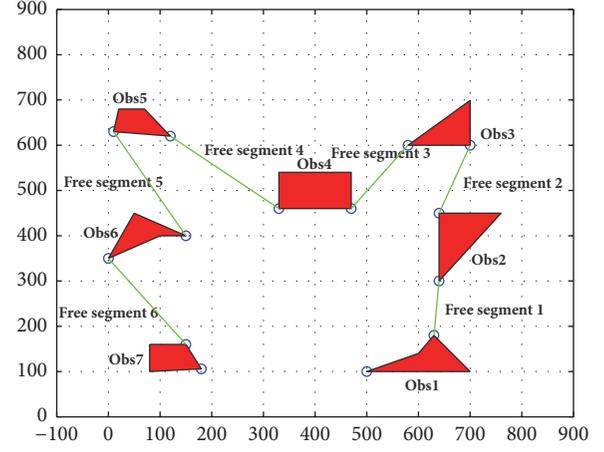


FIGURE 12: Environment mapping.

why the switching function is defined as a saturation function. The control law is defined then as

$$\dot{s} = -k \text{sat}(s). \quad (11)$$

It is noted that the reaching control system is not only able to establish the reaching condition but also able to specify the dynamic of the switching function. By differentiating the vector of the sliding surfaces defined in equation (10), we obtain

$$\begin{aligned} \dot{s} &= \begin{bmatrix} \dot{s}_1 \\ \dot{s}_2 \end{bmatrix} = \begin{bmatrix} -k_1 \text{sat}(s_1) \\ -k_2 \text{sat}(s_2) \end{bmatrix} = \begin{bmatrix} \dot{x}_e \\ \dot{\alpha}_e + \frac{\partial \gamma}{\partial v_r} \dot{v}_r + \frac{\partial \gamma}{\partial y_e} \dot{y}_e \end{bmatrix} \\ &= \begin{bmatrix} y_e w + v_r \cos \alpha_e - v \\ w_r + \frac{\partial \gamma}{\partial v_r} \dot{v}_r + \frac{\partial \gamma}{\partial y_e} (v_r \sin \alpha_e - x_e w) - w \end{bmatrix} \end{aligned} \quad (12)$$

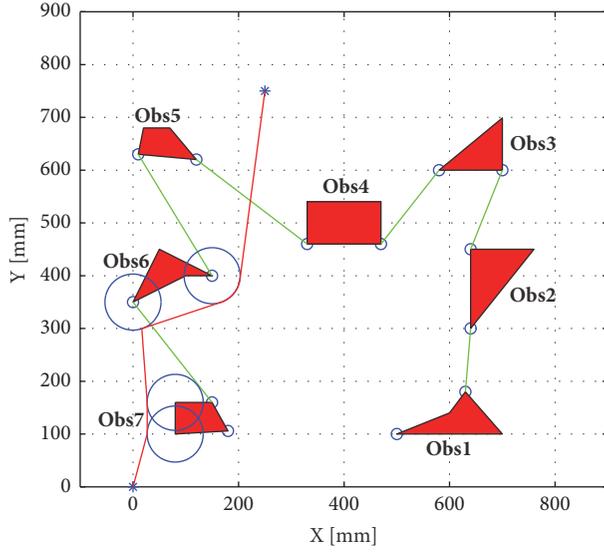
where

$$\begin{aligned} \frac{\partial \gamma}{\partial v_r} &= \frac{y_e}{1 + (v_r y_e)^2} \\ \text{and } \frac{\partial \gamma}{\partial y_e} &= \frac{v_r}{1 + (v_r y_e)^2}. \end{aligned} \quad (13)$$

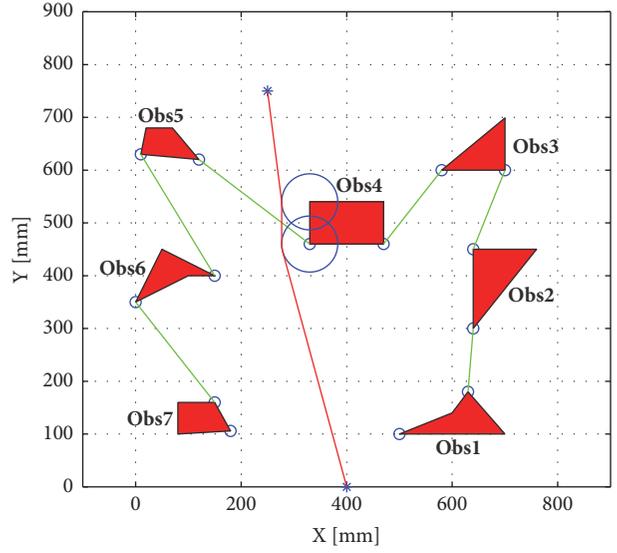
5. Simulation Results

In mobile robot navigation, the building of the environment is considered an essential issue to carry out motion planning operations. In this section, to demonstrate the basic ability of the proposed algorithm, we present some simulation results. In all simulations, we will present results of an environment including seven obstacles which are placed with an arbitrary way (see Figure 12). Table 1 presents the initial center coordinates of static obstacles.

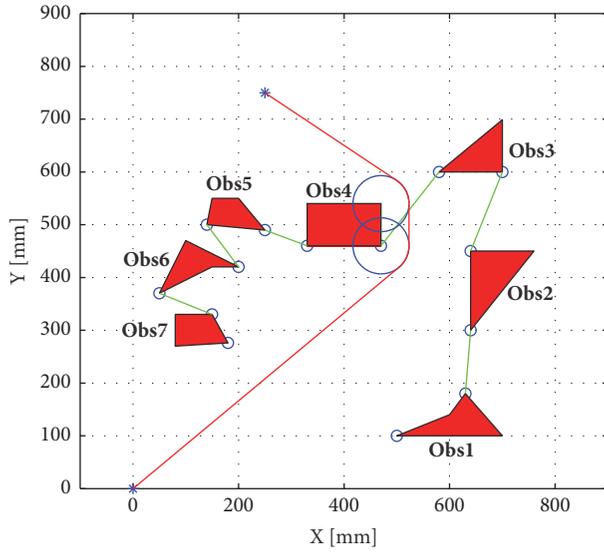
The simulations are performed for the cases where the target coordinate (x_t, y_t) is fixed while the robot position changed.



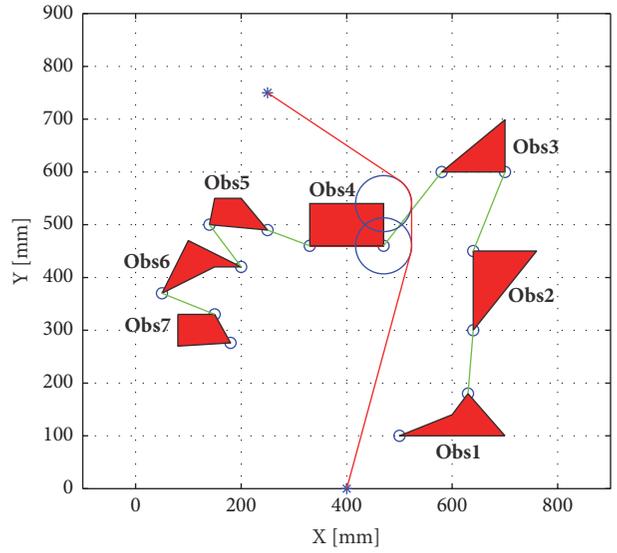
(a) Navigation with safe segments $((x_r, y_r)=(0, 0))$.



(b) Navigation with safe segments $((x_r, y_r)=(400, 0))$.



(c) Navigation with safe and danger segments $((x_r, y_r)=(0, 0))$.



(d) Navigation with safe and danger segments $((x_r, y_r)=(400, 0))$.

FIGURE 13: Path planning $((x_t, y_t)=(250, 750))$.

TABLE 1: Center coordinates of obstacles.

Obstacles	x_{Obs}	y_{Obs}
Obstacle 1	550	100
Obstacle 2	640	400
Obstacle 3	640	600
Obstacle 4	400	500
Obstacle 5	70	680
Obstacle 6	100	400
Obstacle 7	150	130

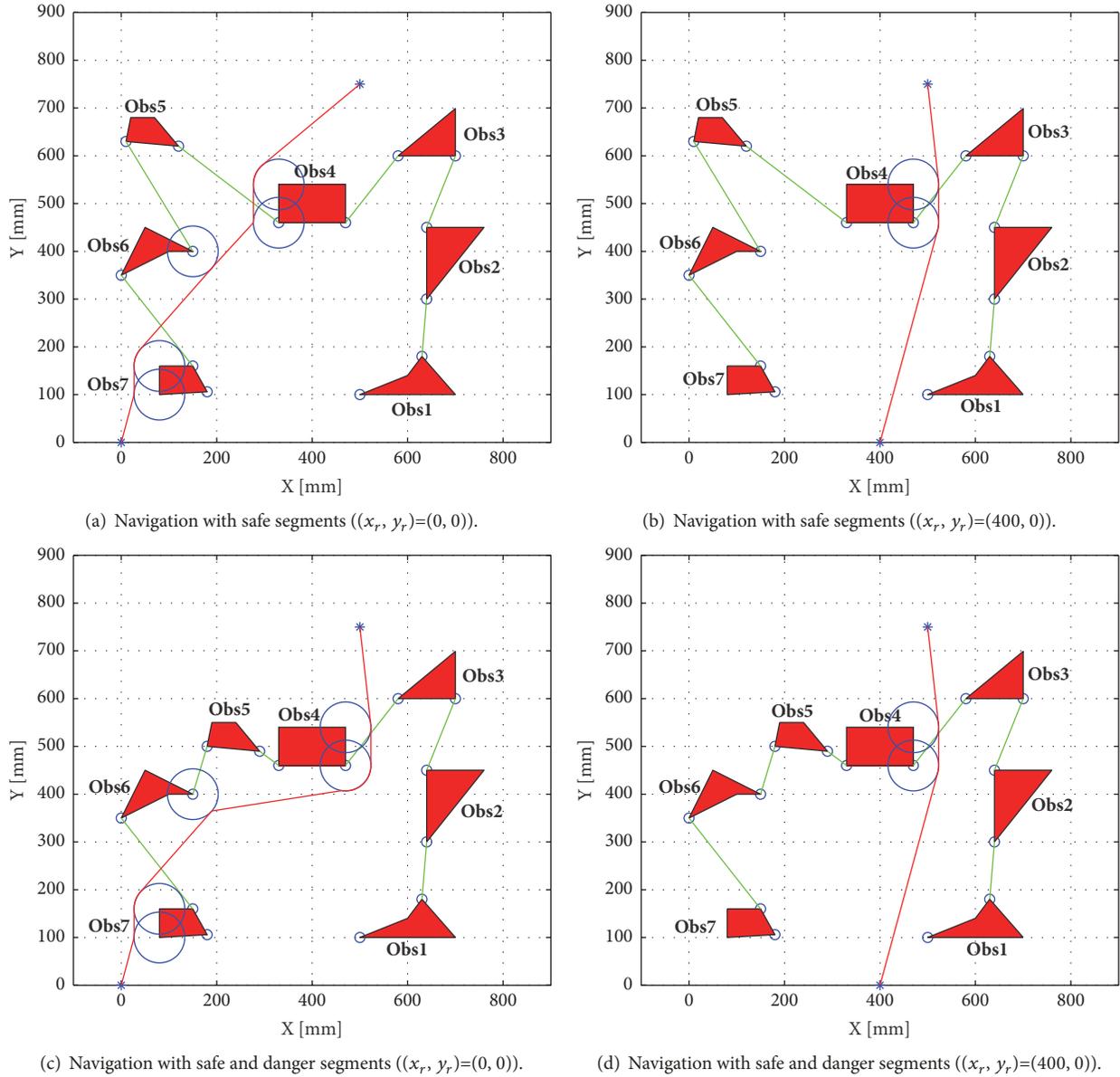
TABLE 2: Center coordinates of obstacles $((x_t, y_t)=(250, 750))$.

Obstacles	x_{Obs}	y_{Obs}
Obstacle 1	550	100
Obstacle 2	640	400
Obstacle 3	640	600
Obstacle 4	400	500
Obstacle 5	200	550
Obstacle 6	150	420
Obstacle 7	150	300

In this section, we present the case when the robot starts from the initial positions $(x_r, y_r)=(0, 0)$ and $(x_r, y_r)=(400, 0)$ as shown in Figures 13(a) and 13(b), where all free segments are safe. We notice that the robot turns around circles which

are located in the adequate turning points and reaches the target for each modification of the robot position.

Even the obstacle centers changed their positions as shown in Table 2, and the path navigation changes are shown

FIGURE 14: Path planning $((x_t, y_t)=(500, 750))$.

in Figures 13(c) and 13(d) because of the appearance of danger segments.

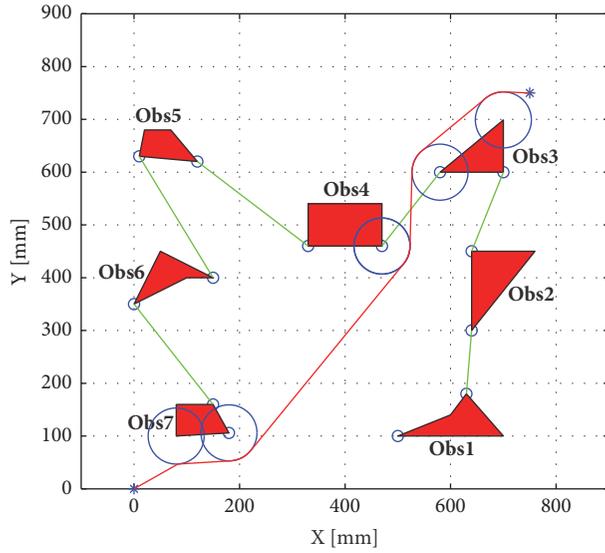
Figure 16 illustrates the navigation of the mobile robot with safe segments and danger segments. That robot starts from different initial positions $((x_r, y_r)=(0, 0))$ (see Figures 14(a) and 14(c)) and $((x_r, y_r)=(400, 0))$ (see Figures 14(b) and 14(d)). The obstacle center coordinates are addressed in Table 3.

Another simulation results present the case where all free segments are safe (see Figures 15(a) and 15(b)). The robot turns around the dangerous circles until reaching the desired target. By changing obstacle centers as shown in Table 4, we remark the appearance of dangerous segments. The robot takes into account just the free segments and moves in the safe path (see Figures 15(c) and 15(d)).

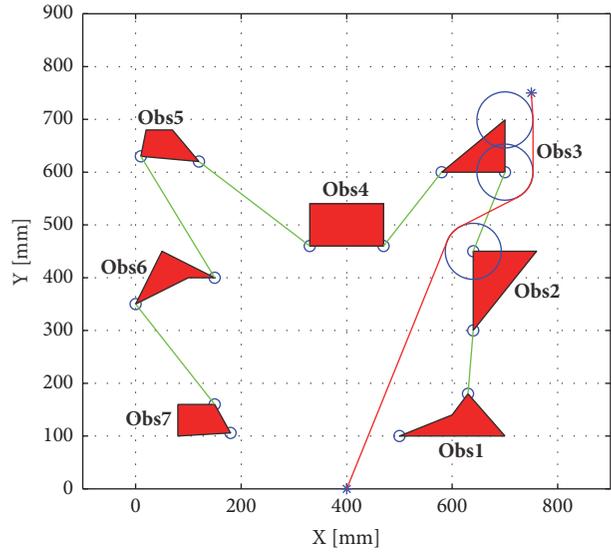
TABLE 3: Center coordinates of obstacles $((x_t, y_t)=(500, 750))$.

Obstacles	x_{Obs}	y_{Obs}
Obstacle 1	550	100
Obstacle 2	640	400
Obstacle 3	640	600
Obstacle 4	400	500
Obstacle 5	240	550
Obstacle 6	100	420
Obstacle 7	150	250

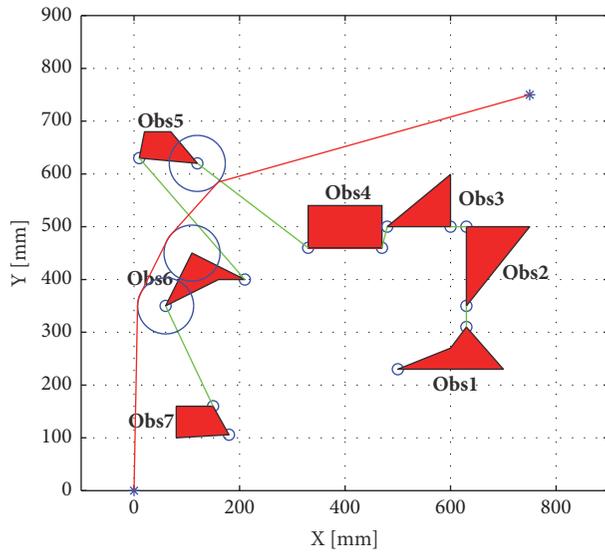
Figures 16(a) and 16(b) show that the mobile robot ensures reaching the destination with avoiding different obstacles. Table 5 shows the center obstacle positions. In



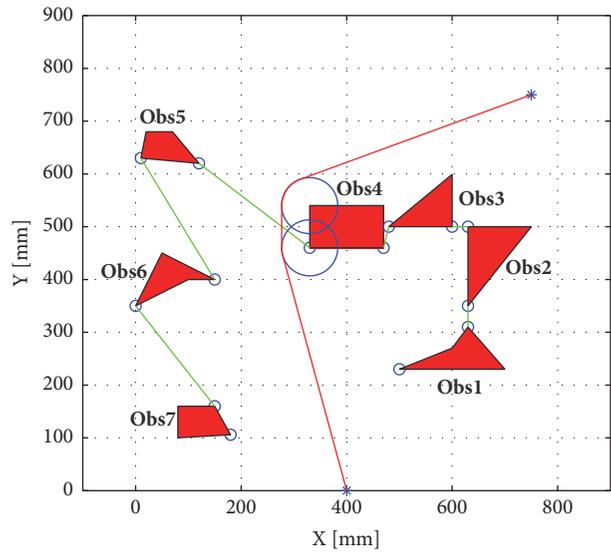
(a) Navigation in case safe segments $((x_r, y_r)=(0, 0))$.



(b) Navigation in case safe segments $((x_r, y_r)=(400, 0))$.



(c) Navigation in case safe and danger segments $((x_r, y_r)=(0, 0))$.



(d) Navigation in case safe and danger segments $((x_r, y_r)=(400, 0))$.

FIGURE 15: Path planning $((x_t, y_t)=(750, 750))$.

TABLE 4: Center coordinates of obstacles $((x_t, y_t)=(750, 750))$.

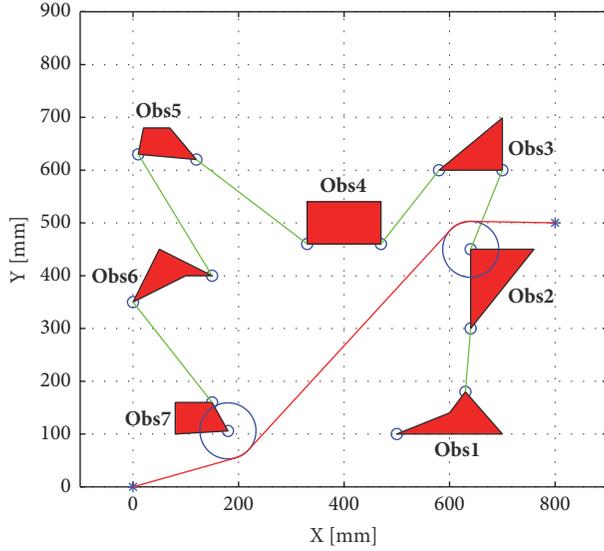
Obstacles	x_{Obs}	y_{Obs}
Obstacle 1	550	230
Obstacle 2	630	450
Obstacle 3	540	500
Obstacle 4	400	500
Obstacle 5	70	680
Obstacle 6	160	400
Obstacle 7	150	130

TABLE 5: Center coordinates of obstacles $((x_t, y_t)=(800, 500))$.

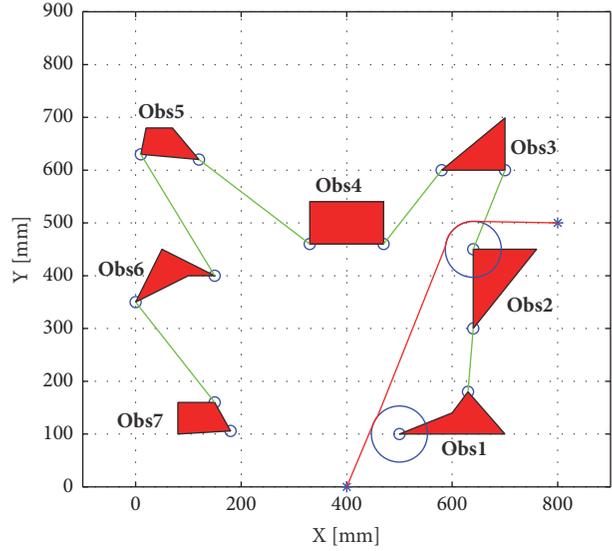
Obstacles	x_{Obs}	y_{Obs}
Obstacle 1	550	240
Obstacle 2	550	400
Obstacle 3	540	510
Obstacle 4	400	510
Obstacle 5	200	560
Obstacle 6	150	430
Obstacle 7	150	260

this case, we constat that there is a local minima problem. Therefore, the robot goes far away from obstacles and moves directly to the target (see Figures 16(c) and 16(d)).

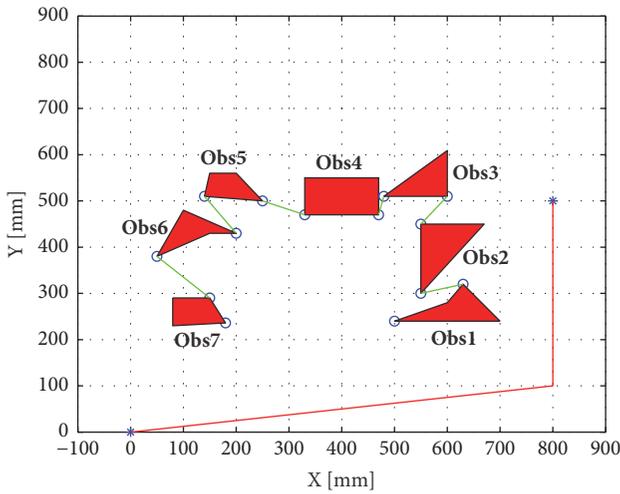
From all simulation results, it is obvious to see that the developed strategy is very reactive because the robot achieves the obstacle avoidance in each modification of the robot



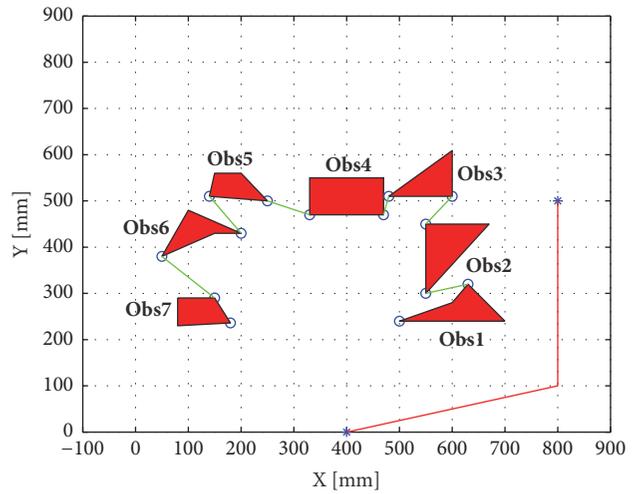
(a) Navigation with safe segments $((x_r, y_r)=(0, 0))$.



(b) Navigation with safe segments $((x_r, y_r)=(400, 0))$.



(c) Navigation with danger segments $((x_r, y_r)=(0, 0))$.



(d) Navigation with danger segments $((x_r, y_r)=(400, 0))$.

FIGURE 16: Path planning $((x_r, y_r)=(800, 500))$.

and the target positions and in presence of safe and danger segments.

After planning the safest and the shortest path, it is required for the mobile robot to track reference trajectories based on sliding mode controller. Figure 17 shows that the mobile robot always follows the reference trajectory.

To more illustrate the performance of the sliding mode controller, the error positions, and the two speeds (right and left) of the wheels for the cases. Figures 15(a) and 16(b) were presented in Figures 18 and 19. Figure 18 shows that the tracking errors tend to zero which allows concluding that the proposed control law system provides a good tracking trajectory.

In addition to this, Figure 19 presents the evolution of two speeds (right and left) of the wheels. For example, for Figure 19(b), initially the mobile robot advances with

the same speeds for both wheels. As soon as obstacle 1 is detected, the control system provides a larger right wheel speed compared to the left wheel speed. After passing obstacle 1, the two speeds are equal until the robot reaches the target. As soon as obstacle 2 is detected, the controller system provides a larger right wheel speed than the left wheel speed. After passing obstacle 2, we notice that the speed of the left wheel is larger than the right wheel. This is to turn the mobile robot to the target position. Once the robot is oriented towards the target, the two speeds are equal until the robot reaches the target.

6. Conclusion

In this paper, an algorithm which searches for a turning point based on free segments is presented. It handles two

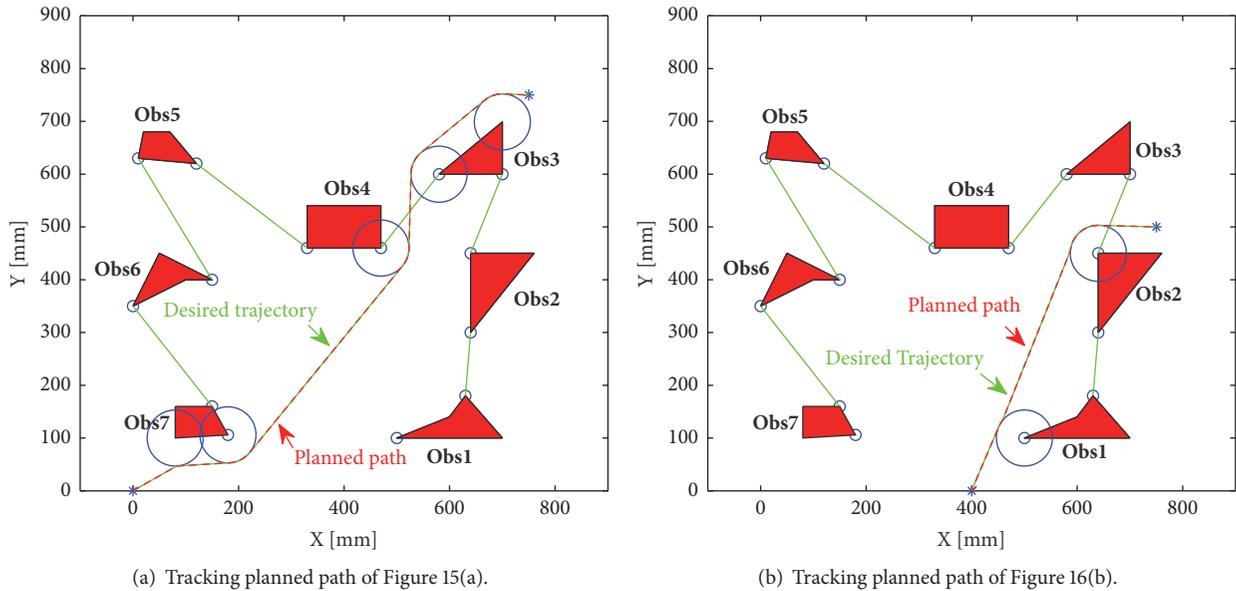


FIGURE 17: Tracking planned path.

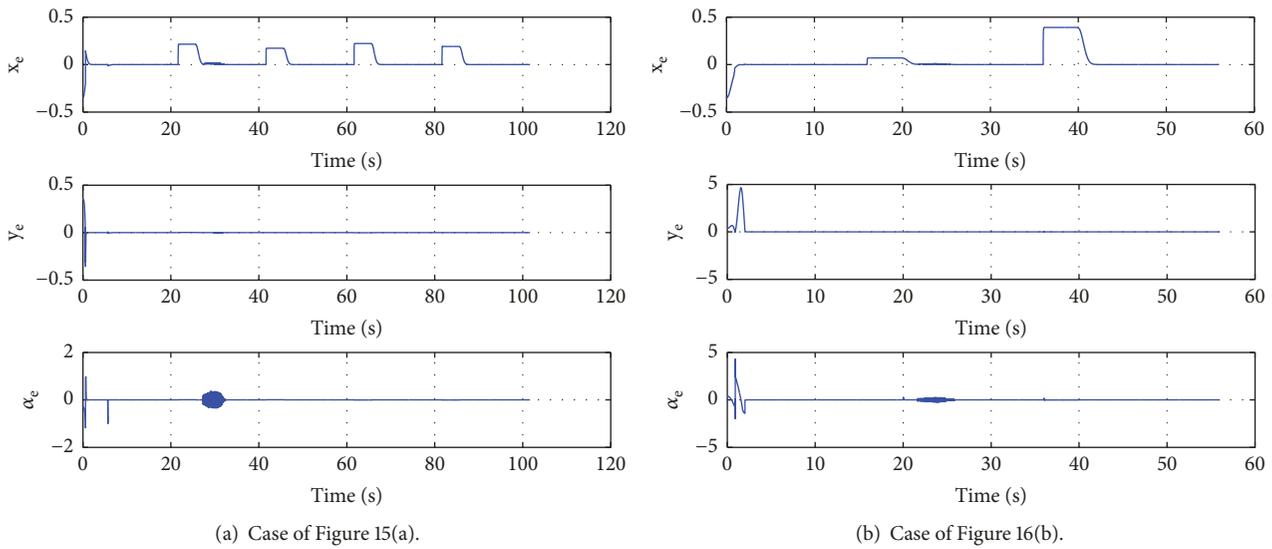


FIGURE 18: Tracking Errors (x_e, y_e, α_e).

different objectives: the safe path and the path length. The advantage of the developed algorithm is that the robot always can move from the initial position to the target position, not only safely, but also on the shortest path regardless the shape of the obstacles and the change of goal position in the known environment. In the other side, the proposed sliding mode control is an important method to deal with the system. This controller demonstrates a good tracking performances such as robustness, stability and fast response. Simulation results are performed on a platform Khepera IV to demonstrate that the proposed method is a good alternative to solve the path planning and trajectory tracking problems.

As a future work, it could be interesting to determinate paths in dynamic environment.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

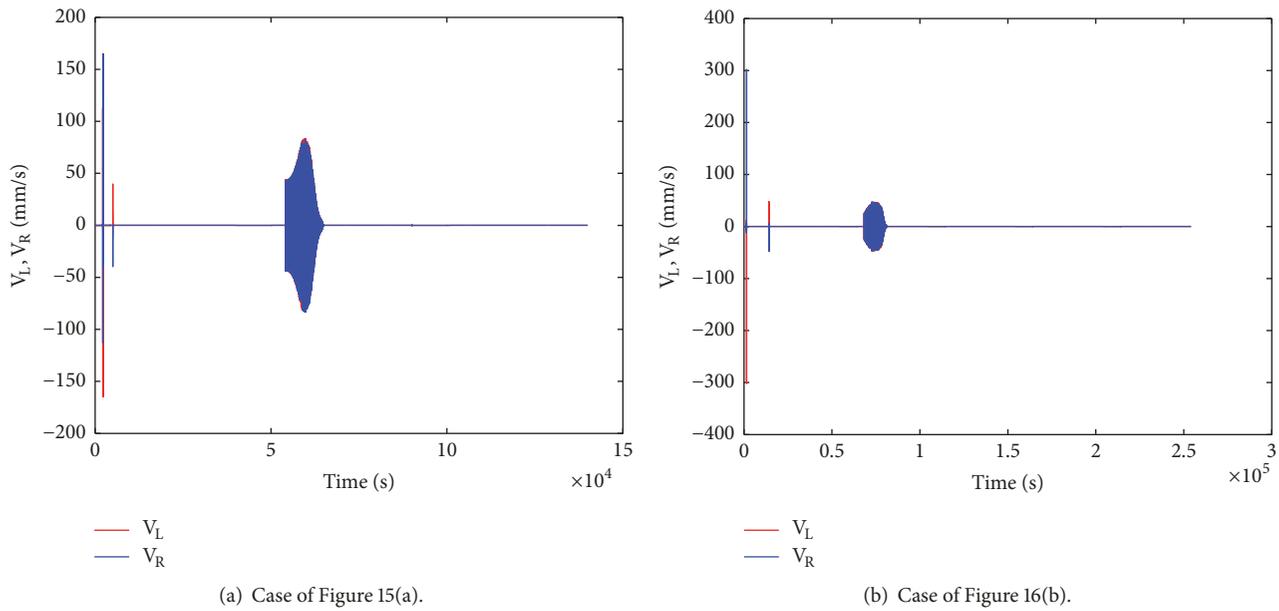


FIGURE 19: Evolution of the two speeds (right and left).

References

- [1] D. Xin, C. Hua-hua, and G. Wei-kang, "Neural network and genetic algorithm based global path planning in a static environment," *Journal of Zhejiang University Science*, vol. 6A, no. 6, pp. 549–554, 2005.
- [2] C. Yang, H. Jianda, and W. Huaiyu, "Quadratic programming-based approach for autonomous vehicle path planning in space," *Chinese Journal of Mechanical Engineering*, vol. 25, no. 4, pp. 665–673, 2012.
- [3] J.-H. Liang and C.-H. Lee, "Efficient collision-free path-planning of multiple mobile robots system using efficient artificial bee colony algorithm," *Advances in Engineering Software*, vol. 79, pp. 47–56, 2015.
- [4] A. Hidalgo-Paniagua, M. A. Vega-Rodríguez, J. Ferruz, and N. Pavón, "Solving the multi-objective path planning problem in mobile robotics with a firefly-based approach," *Soft Computing-A Fusion of Foundations, Methodologies and Applications*, vol. 21, no. 4, pp. 949–964, 2017.
- [5] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1398–1404, Sacramento, CA, USA, April 1991.
- [6] H. Seki, S. Shibayama, Y. Kamiya, and M. Hikizu, "Practical Obstacle Avoidance Using Potential Field for A Nonholonomic Mobile Robot with Rectangular Body," in *Proceedings of the 13th IEEE International Conference on Emerging Technologies And Factory Automation*, pp. 326–332, Hamburg, Germany, 2008.
- [7] M. Y. Ibrahim and L. McFetridge, "The Agoraphilic algorithm: A new optimistic approach for mobile robot navigation," in *Proceedings of the 2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics Proceedings*, pp. 1334–1339, Como, Italy, July 2001.
- [8] J. Borenstein and Y. Koren, "The vector field histogram—fast obstacle avoidance for mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 278–288, 1991.
- [9] B. You, J. Qui, and D. Li, "A novel obstacle avoidance method for low-cost household mobile robot," in *Proceedings of the 2008 IEEE International Conference on Automation and Logistics (ICAL)*, pp. 111–116, Qingdao, China, September 2008.
- [10] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics and Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.
- [11] O. Brock and O. Khatib, "High-speed navigation using the global dynamic window approach," in *Proceedings of the 1999 IEEE International Conference on Robotics and Automation, ICRA99*, pp. 341–346, May 1999.
- [12] J. Hong and K. Park, "A new mobile robot navigation using a turning point searching algorithm with the consideration of obstacle avoidance," *The International Journal of Advanced Manufacturing Technology*, vol. 52, no. 5–8, pp. 763–775, 2011.
- [13] W. G. Wu, H. T. Chen, and Y. J. Wang, "Global trajectory tracking control of mobile robot," *Acta Automatica Sinica*, vol. 27, no. 3, pp. 326–331, 2001.
- [14] I. Kolmanovsky and N. H. McClamroch, "Developments in nonholonomic control problems," *IEEE Control Systems Magazine*, vol. 15, no. 6, pp. 20–36, 1995.
- [15] M. Boujelben, C. Rekek, and N. Derbel, "Mobile robot navigation using fuzzy-sliding mode control in a cluttered environment," in *Proceedings of the 2nd World Congress On Computer Applications and Information Systems (WCCAIS'15)*, Hammamet, Tunisia, 2015.
- [16] J. H. Lee, C. Lin, H. Lim, and J. M. Lee, "Sliding mode control for trajectory tracking of mobile robot in the RFID sensor space," *International Journal of Control, Automation and Systems*, vol. 7, no. 3, pp. 429–435, 2009.
- [17] R. Rojas and A. G. Förster, "Holonomic Control of a robot with an omnidirectional drive," in *Proceedings of the 2006 IEEE 3rd Latin American Robotics Symposium*, pp. 41–46, IEEE, Santiago, Chile, October 2006.
- [18] D. P. Atherton and S. Majhi, "Limitations of PID controllers," in *Proceedings of the 1999 American Control Conference (99ACC)*, pp. 3843–3847, San Diego, CA, USA, June 1999.

- [19] F. Cherni, Y. Bouterraa, C. Rekik, and N. Derbel, "Path planning for mobile robots using fuzzy logic controller in the presence of static and moving obstacles," in *Proceedings of Engineering and Technology*, pp. 503–509, 2016.
- [20] C. B. Butt and M. K. Rahman, "Limitations of simplified fuzzy logic controller for IPM motor drive," in *Proceedings of the Conference Record of the 2004 IEEE Industry Applications Conference; 39th IAS Annual Meeting*, pp. 1891–1898, Seattle, WA, USA, October 2004.
- [21] R. Solea, A. Filipescu, and U. Nunes, "Sliding-mode control for trajectory-tracking of a wheeled mobile robot in presence of uncertainties," in *Proceedings of the 7th Asian Control Conference (ASCC '09)*, pp. 1701–1706, Hong Kong, August 2009.
- [22] H. Surmann, J. Huser, and L. Peters, "Fuzzy system for indoor mobile robot navigation," in *Proceedings of the 1995 IEEE International Conference on Fuzzy Systems. Part 1 (of 5)*, pp. 83–88, Yokohama, Japan, March 1995.
- [23] H. Lu and C. Chuang, "The implementation of fuzzy-based path planning for car-like mobile robot," in *Proceedings of the 2005 International Conference on MEMS, NANO and Smart Systems (ICMENS'05)*, pp. 467–472, Banff Alta, Canada, 2005.

