2017 14th International Multi-Conference on Systems, Signals & Devices (SSD)

Fuzzy logic controller for mobile robot navigation to avoid dynamic and static obstacles

Maïssa BOUJELBEN, Dorra AYEDI, Chokri REKIK and Nabil DERBEL
Control & Energy Managment Laboratory (CEM-Lab)
University of Sfax, Sfax Engineering School, BP W,1173-3038 Sfax, Tunisia.
e-mail: boujelben.maissa@gmail.com , chokri.rekik@enis.rnu.tn, nabil.derbel@ieee.org

Abstract— **This paper addresses the problem of mobile robots navigation in static and dynamic environment. The proposed navigation method is inspired from the VFH (Vector Field Histogram) method: it adopts the principle of safe sectors, but simplifies the configuration space representation. It consists on directing the robot heading towards a desired direction allowing to reach the target or to avoid obstacles. The proposed control law uses a SI|SO fuzzy logic system. This method is characterized by its simplicity to implement and its low computational time. The validity of this approach is proved through simulation results.**

Keywords— **Mobile robots navigation; static and dynamic environment; VFH; fuzzy logic system.**

I. Introduction

For years, researchers paid a big attention to solve the problem of autonomous mobile robot navigation. The robot must be able to find the path between its initial position to the desired one without human intervention and without colliding obstacles. Several works addressed mobile robot navigation problem in static environments. Among these methods, we find Voronoi diagrams [1], artificial potential fields [2] and fuzzy systems [3], [4]. The achievement of this task is more complicated especially when the environment is dynamic, ie. when one or several obstacles change positions over time. References [5], [6], [7] describe some well known techniques of dynamic obstacle avoidance method.

In [8], authors have used hierarchical fuzzy systems for the robot control. The advantage of these hierarchical fuzzy systems compared to standard fuzzy systems is the reduction of fuzzy rule number [9].

VFH method [10] was created by combining potential fields method [2] with occupancy grids. It consists on finding safe intervals for the robot motion called candidate valleys. The middle of the closest candidate valley to the target direction is chosen as the direction of motion. It is a robust approach as it allows to generate smooth trajectories and to solve navigation problem in narrow corridors. Nevertheless, it suffers from the environment complex representation. In fact, to determine safe sectors, it is necessary to proceed by a two-dimensional Cartesian histogram grid then by the polar histogram. In addition, it is not suitable for environments containing moving obstacles.

The purposes of this paper is to produce a reliable and a smooth trajectory in static and dynamic environments. For that, we have adopted the principle of safe sectors in the VFH method with a simplified configuration space representation [11]. Indeed, we have combined the world model simple description used by Savkin and Wang [12] with a Single Input Single Output (SI|SO) fuzzy logic controller. It consists on directing the robot heading towards a desired direction allowing to reach the target or to avoid obstacles. Compared to reference [8], this approach uses a very reduced fuzzy rules number. Therefore, the developed approach is efficient and very easy to implement. In addition, it require a low computational time.

This paper is divided into six sections. Section II, presents the problem formulation. In section III, we describe the proposed navigation algorithm. The developed control law is given in section IV. Simulation results are presented in section V. Finally, a conclusion is given.

978-1-5386-3175-1/17/$31.00 ©2017 IEEE

II. Problem formulation

Khepera II robot is a miniature mobile robot having two independent driving wheels. These wheels are responsible for controlling and orienting the mobile robot by acting on the speed of each wheel. The schematic model of this robot is represented by Fig. 1.

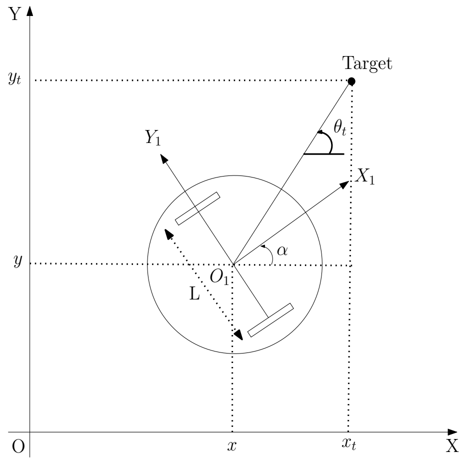

Fig. 1. The schematic model of the mobile robot Khepera II.

The kinematic model of Khepera II robot is given by:

$$\begin{cases} \dot{x} = \frac{V_R+V_L}{2} \cos\alpha \\ \dot{y} = \frac{V_R+V_L}{2} \sin\alpha \\ \dot{\alpha} = \frac{V_R-V_L}{L} \end{cases} \quad (1)$$

where (x, y) are the robot position in the absolute reference frame $A = (O, X, Y)$, α is the robot orientation with respect to the X-axis, V_R and V_L are respectively the robot right and left wheel velocities and L is the distance between two wheels.

III. Navigation algorithm

In this section, we introduce the proposed navigation algorithm for mobile robot navigation in static and dynamic environment. Firstly, we suppose that the robot is moving with a constant linear velocity v taking into account a maximum angular velocity $w_{max} = \dot{\alpha}_{max}$ (which translates the robot limits).

The next step consists on defining a virtual circle C of diameter D_c (see Fig. 2) allowing to give information about the local environment of the robot. The point of coordinates (x_c, y_c) represents the circle center. This point is determined at each iteration step and located in the front of the robot (in the direction of its current heading α). It is situated at a distance $\frac{D_c}{2}$ from the robot center with:

$$D_c = r_r + s_r \quad (2)$$

r_r is the robot radius and s_r is the sensor range.

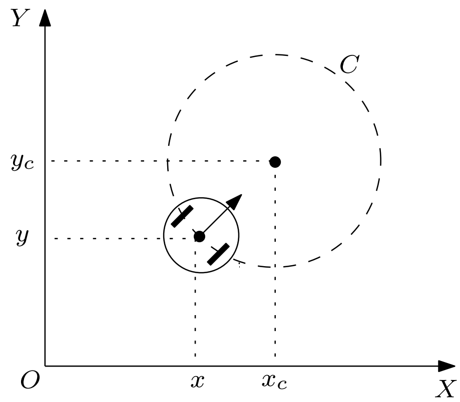

Fig. 2. Geometrical representation of the circle C

In this work, we assume that obstacles shape is circular. Then, we should determine the desired direction α_d allowing to reach the target position and to prevent constrained obstacles. For that, it is necessary to determine the intersection points between the circle C and the obstacles. So, two cases are presented.

First case: there is no intersection points between the circle C and the obstacles. This case treats the situation when the robot doesn't sense obstacles in front itself. So, the robot should turn toward the target direction. The desired direction is then:

$$\alpha_d = \theta_t \;=\; \arctan\frac{y_t - y}{x_t - x} \quad (3)$$

with θ_t is the angle between the X-axis and the line that connects between the robot center and the target position (see Fig. 1). (x_t, y_t) are the target coordinates.

Second case: there is at least one obstacle which intersects the circle C. In this case, we affirm the presence of obstacles which can disturb the robot motion. Then, to determine the desired direction, proceed by the following steps:

1) Define the interval $[A_1, A_k]$ in order to cover the entire front area of the robot, with:

$$A_1 = \alpha - \frac{\pi}{2} \quad (4)$$

$$A_k = \alpha + \frac{\pi}{2} \quad (5)$$

where $k = 2(n+1)$, n is the obstacle number and α is the robot orientation with respect to the X-axis.

2) Determine, for each detected obstacle " i ", the intersection points $P_i = (x_{P_i}, y_{P_i})$ and $Q_i = (x_{Q_i}, y_{Q_i})$ between the circle C and the obstacles (see Fig. 3).

3) Compute, from these intersection points, the intervals of collision danger $[A_j,\ A_{j+1}]$ such that:

$$A_j = \arctan \frac{y_{P_i} - y}{x_{P_i} - x} \quad (6)$$

$$A_{j+1} = \arctan \frac{y_{Q_i} - y}{x_{Q_i} - x} \quad (7)$$

with j is even and varies from 2 to $2n$.

For example $n = 3$: i.e. there is three obstacles that may provide a collision danger to the robot (see Fig. 3). The different obtained intervals are illustrated in Fig. 4. We find three intervals of collision danger and four vacant intervals in which obstacles are far enough from the robot current orientation.

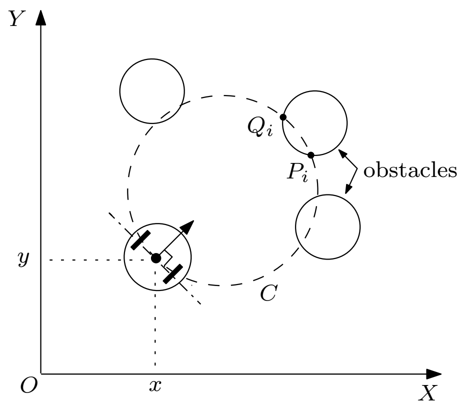

Fig. 3. Environment representation in presence of obstacles

Now, two situations are presented according to the robot orientation with respect to these intervals:

- 1^{st} situation: $\alpha \in [A_i, A_{i+1}]$ with i is odd. This means that the robot is oriented toward a vacant interval which constitute a safe direction. The desired robot orientation α_d is the middle of this interval. It is expressed as follows:

$$\alpha_d = \frac{A_i + A_{i+1}}{2} \quad (8)$$

- 2^{nd} situation: $\alpha \in [A_i, A_{i+1}]$ with i is even. In this situation, the robot is oriented toward an interval which constitute a collision danger. Let j the index of the vacant interval closest to the robot current orientation. The desired robot orientation α_d is the middle of the interval $[A_j, A_{j+1}]$. e.g. case of Fig 4, $j = 5$.

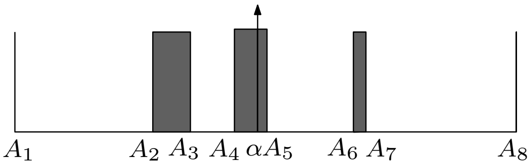

Fig. 4. Illustration of different obtained intervals

It is noted that the robot size is taken in consideration when selecting the interval of motion direction. In fact, a comparative test between the interval width and the robot diameter is performed in the simulations. If the interval doesn't allow the robot to move, so the next vacant interval should be tested.

IV. Control Law

The objective of this part is to apply a control law allowing to guide the robot to the desired direction α_d. For that, we propose to use a SI|SO fuzzy controller. The fuzzy input is the angle φ defined as the difference between the desired robot orientation and its current orientation ($\varphi = \alpha_d - \alpha$) and the fuzzy output is the robot angular velocity w. Here, it's noted that this controller is simple to understand and to implement. In addition, it requires a low computational time.

For the angle φ, we have used seven fuzzy subsets which are: NL: Negative Large; NM: Negative Medium; NS: Negative Small; Z: Zero; PS: Positive Small; PM: Positive Medium; PL: Positive Large. Membership functions of this angle are gaussian and they are represented in Fig. 5. Fuzzy inference table is represented in Table 1. In this table, seven fuzzy

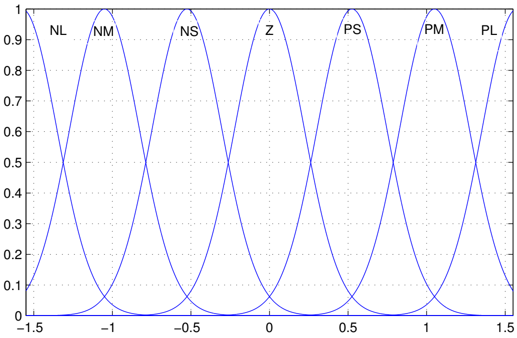

Fig. 5. Membership functions for the angle φ (in rad).

rules (of type if-then rule) are developed. With this reduced number of fuzzy rules, the implementation of this controller becomes easier and decreases the computational time compared to reference [8].

Table 1. Fuzzy inference table

φ	NL	NM	NS	Z	PS	PM	PL
w	NB	NB	NB	Z	PB	PB	PB

Fuzzy controller output is expressed as follows:

$$w = \frac{\sum_{i=1}^{7} \alpha_i w_i}{\sum_{j=1}^{7} \alpha_j} \quad (9)$$

with α_i is the activation level of rule i and w_i is the consequence of rule i. Fig. 6 shows the numerical value attributed to each linguistic variable of fuzzy rule consequences (which are NB: Negative Big, Z; Zero, PB: Positive Big).

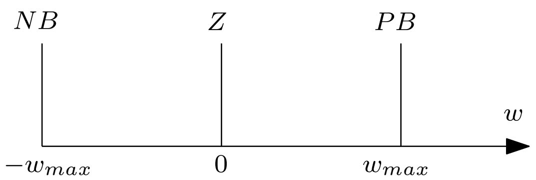

Fig. 6. Numerical values attributed to each linguistic variable.

The two velocities applied to the left and right wheels are expressed as follows [13]:

$$V_R = v + \frac{Lw}{2} \quad (10)$$

$$V_L = v - \frac{Lw}{2} \quad (11)$$

V. SIMULATION RESULTS

This part is devoted to evaluate the performances of the proposed approach. Firstly, we have considered an arbitrary static environment. Fig. 7 presents the robot trajectory in a complex static environment. In this simulation case, the robot start from the position (x_0, y_0)=(0, 0) and should reach a target located at the position $(x_T,\ y_T) = (30,\ 30)$ taking into account the presence of static obstacles. Considering different initial orientations ($\alpha_0 = 0, \frac{\pi}{4}$ and $\frac{\pi}{2}$), the robot complete successfully its task. In addition, the proposed navigation algorithm allow to produce reliable and smooth trajectories while reaching the target. Indeed, according to its current configuration, the robot pass around the obstacle or between two obstacles through the vacant intervals defining safe motion directions. Fig. 8 shows the evolution of the robot right and left wheel velocities in the case of figure 7, where $(x_0,\ y_0,\ \alpha_0) = (0,\ 0,\ \frac{\pi}{4})$.

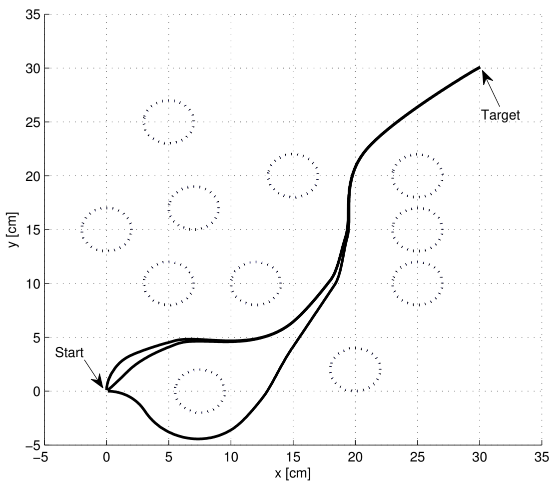

Fig. 7. Robot trajectory for complex static environment.

The next simulation is performed to demonstrate the efficiency of this approach in a dynamic environment. In Fig. 9, the threat of two moving obstacles is

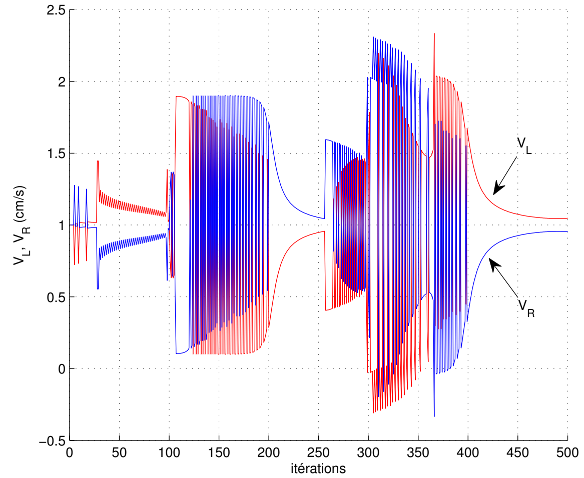

Fig. 8. Evolution of the robot right and left wheel velocities in the case of figure 7, where $(x_0,\, y_0,\, \alpha_0) = (0,\, 0,\, \frac{\pi}{4})$.

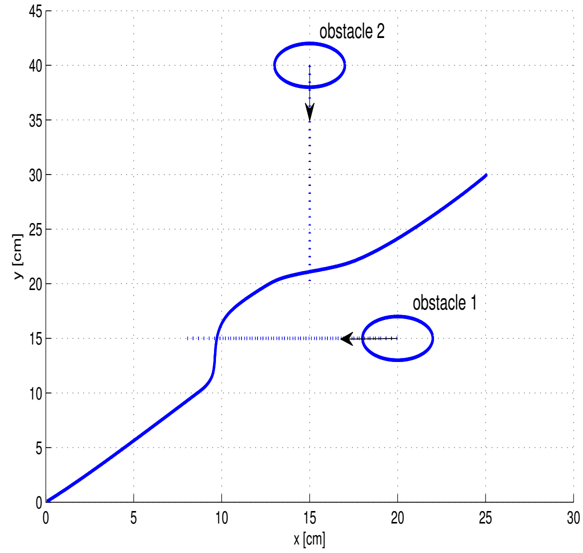

Fig. 9. Complex dynamic environment.

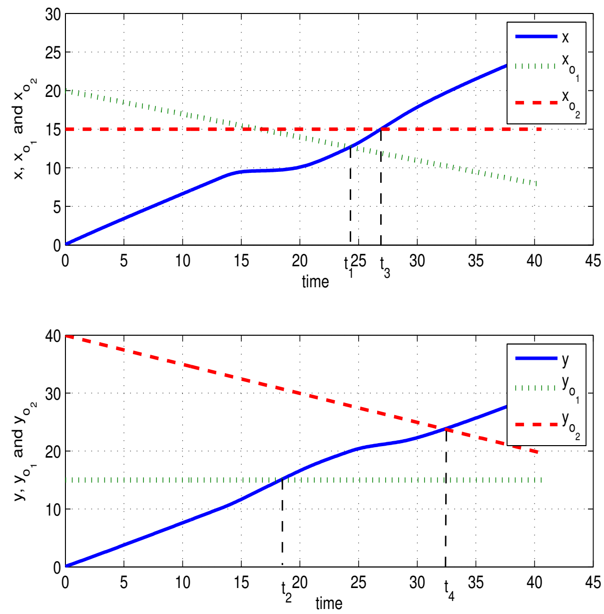

Fig. 10. Evolution of the robot and the obstacles coordinates.

taking into account. The two obstacles are assumed to move with different constant velocities. Their initial positions, trajectories and directions of motion are represented by circles, dashed lines and arrows, respectively. Fig. 10 is carried out to prove that navigation is performed safely. It illustrates the evolution of the robot and obstacles coordinates over time. In fact, collision is happened if there is an instant t in which the robot and obstacles coordinates are equal. To demonstrate that there is no collision, let's consider t_1 (respectively t_2) the intersection time between the x-position (respectively y-position) of the robot and the obstacle 1 (see Fig. 9). According to Fig. 9, it is clear that t_1 and t_2 are different. So, we can conclude that the robot is able to move away from this obstacle without collision. The same reasoning is done for the obstacle 2.

It is noted that the dimensions of the robot and the obstacles are taken into account when avoiding obstacles (the robot radius is equal to 2.75 cm and the obstacle radius is equal to 2 cm). In fact, according to Fig. 10, the robot and the obstacle 1 have the same ordinates at the instant t_2 ($y = y_{o_1} = 15$ cm). At this moment, their x-positions are distant of about 4.8 cm ($x = 9.7$ cm and $x_{o_1} = 14.5$ cm). This allow their motion, considering their circular shapes, without collision.

VI. Conclusion

In this paper, we have developed a reactive approach for mobile robot control in static and dynamic environment. We have adopted the principle of safe sectors in the VFH method with a simple configuration space representation. The proposed navigation method consists on determining a desired direction allowing to reach a target and to escape from obstacles. Then, using a SI|SO fuzzy controller, the robot must follow the desired direction. This method is characterized by its simplicity and its robustness in static and dynamic environments. In addition, it reduces fuzzy rules number compared to the method which uses hierarchical fuzzy controller.

REFERENCES

[1] J.C. Latombe, "Robot Motion Planning", Kluwer Academic Publishers, Boston, MA, (1991).

[2] O. Khatib, "Real time obstacle avoidance for manipulators and mobile robot", Int. J. of Robotics and Research, Vol.5, No.1, pp.90–98, (1986).

[3] G. Antonelle and S. Chiaverini, "Experiments of fuzzy lane following for mobile robots", American Control Conf., pp.1079–1084, (2004).

[4] O. Motlagh, T.S. Hong and N. Ismail, "Development of a new minimum avoidance system for a behavior-based mobile robot", Fuzzy Sets and Systems, vol.160, No.13, pp.1929–1946, (2009).

[5] P. Fiorini and Z. Shiller, "Motion planning in dynamic environment using velocity obstacles", Int. J. of Robotics Research, vol.17, No.7, pp.760–772, (1998).

[6] F. Large, C. Lauger, and Z. Shiller, "Navigation among moving obstacles using the NLVO: principles and applications to intelligent vehicles", Auton. Robots, Vol.19, No.2, pp.159–171, (2005).

[7] Y. Zhu, T. Zhang, J. Song, X. Li, and M. Nakamura, "A new method for mobile robots to avoid collision with moving obstacles", Artif. Life Robot, vol.16, No.4, pp.507–510, (2012).

[8] M. Boujelben, C. Rekik, C. and N. Derbel, "A multi-agent architecture with hierarchical fuzzy controller for a mobile robot", Int. J. of Robotics and Automation, Vol.30, No.3, pp.289–298, (2015).

[9] M. Brown, K.M. Bossley, D.J. Mills and C.J. Harris, "High dimensional neurofuzzy systems: overcoming the curse of dimensionality", Proc. IEEE Internat. Conf., pp.2139–2146, (1995).

[10] J. Borenstein and Y. Koren, "The vector field histogram: fast obstacle avoidance for mobile robots", IEEE J. Robotics Autom., vol.7, No.3, pp.278–288, (1991).

[11] M. Boujelben, C. Rekik, C. and N. Derbel, "A Reactive Approach for Mobile Robot Navigation in Static and Dynamic Environment using Fuzzy Logic Control", Int. J. of Modelling, Identification and Control, accepted (2016).

[12] A.V. Savkin and C. Wang, "Seeking a path through the crowd: Robot navigation in unknown dynamic environments with moving obstacles based on an integrated environment representation', *Robotics and Autonomous Systems*, vol.62, No.10, pp.1568–1580, (2014).

[13] 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", Int. J. of Control, Automation, and Systems, Vol.7, No.3, pp.429–435, (2009).