Journal of Applied Sciences

ISSN 1812-5654

Journal of Applied Sciences 6 (8): 1838-1844, 2006
ISSN 1812-5654

Takagi-Sugeno Based Controller for Mobile Robot Navigation

S. Kermiche, M.L. Saidi, H.A. Abbassi and H. Ghodbane
Automatic and Signal Laboratory, Faculty of Engineering,
Annaba University, B.P.12 Annaba, Algeria

Abstract: This study presents a solution for the problem of learning and controlling a mobile robot in the presence of fixed obstacle. The objective is to move the robot from an initial position (source) to a final position (target) without collision. Thus, we propose an approach based on potential fields principle, we define the target as an attractive pole (given as a vector directly calculated from the target position) and the obstacle as a repulsive pole (a vector derived by using fuzzy logic techniques). The linguistic rules, the linguistic variables, the membership functions… etc are the parameters to be determined for the fuzzy controller conception. A learning method based on gradient descent for the self tuning of these parameters is introduced. Therefore, it is necessary to have an expert person for moving the robot manually. During this operation of training, the robot moves and memorizes the data (inputs and outputs). This operation is used to find the controller parameters in order to reach the desired outputs for given inputs.

Key words: Robot mobile, fuzzy logic control, obstacle avoidance method, descent method, self tuning

INTRODUCTION

The control of mobile robot involves several tasks such as perception, path planning and path tracking. The estimation of the mobile robot position can be performed using several sources of data including the perception system. The general purpose of a robot path planner is to find a trajectory from a start position to a goal position with no collisions while minimizing a cost measure. The objective is to follow a previously defined path by taking into account the actual position and the constraints imposed by the robot architecture and the environment. Therefore, the autonomous navigation of the mobile robot imposes a perfect knowledge of the environment in order to detect and avoid the eventual obstacles. This knowledge data base is collected during the teaching operation.

The fundamental problem toward of mobile robot control is to guide the motion until a desired target with no collision. Some methods for generating collision-free paths are adapted from mobile robots (Faverjon, 1987; Sharir and Schorr, 1984; Lozano-Perez, 1983; O'Dùnlaing and Yap, 1985).

The robot motion control is based on the priori known information or determined at each step.

NAVIGATION PROCEDURE

In a sufficiently large empty space, a mobile robot can be driven to any position with any orientation; hence the robot's configuration space has four dimensions, two for translation, one for rotation and one for the steering angle.

Let (x, y, Φ, θ) denote the configuration of the robot, parameterized by the location of the front wheels. The kinematic model of the mobile robot can be represented by Yang *et al.* (2004), Baturone *et al.* (2004), Zhao and Bement (1992)

$$\begin{aligned} \dot{x} &= v\cos\theta \\ \dot{y} &= v\sin\theta \\ \dot{\theta} &= \frac{v}{l}\tan\phi \end{aligned} \tag{1}$$

Where v corresponds to the forward velocity for the vehicle body with respect to the horizontal line is θ, the steering angle with respect to the vehicle body is Φ, (x, y) is the location of the center point of the front wheels and l is the length between the front and the rear wheels.

The steering angle of the front wheel is adjusted by using a fuzzy controller.

We distinguish three types of wheels:

- Fixed wheel, its position is characterised by four constants (α, β, l, r) and the rotation angle Φ (t).

The two components of the contact speed are given by the following constraints:

On the wheel plan (Kang *et al.*, 1999; Lallemand and Zeghloul, 1994):

Corresponding Author: H.A. Abbassi, Automatic and signal Laboratory, Faculty of Engineering, Annaba University, B.P.12 Annaba, Algeria

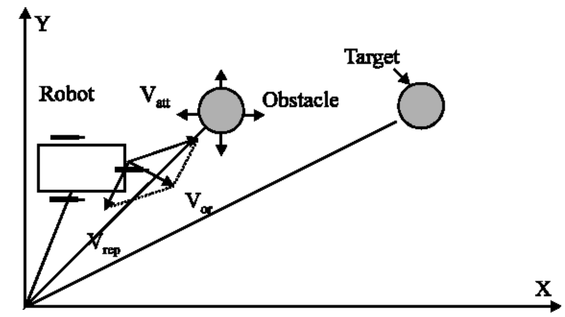

Fig. 1: The mobile robot navigation space

$$\left(-\sin(\alpha+\beta) \quad \cos(\alpha+\beta) \quad l\cos\beta\right)R(\vartheta)\dot{\zeta} + r\dot{\varphi} = 0$$

Orthogonal to the wheel plan:

$$\left(\cos(\alpha+\beta) \quad \sin(\alpha+\beta) \quad l\sin\beta\right)R(\vartheta)\dot{\zeta} = 0 \qquad (2)$$

- Steering wheel, its position is characterised by three constants (l, α, r) and the movement by two variables β(t) and ϕ(t)
- Castor wheel.

The method used here consists to control the robot motion from an initial position (source) to a final position (target) with no collision.

We suppose in our work that:

- The mobile robot, the obstacle and the target can take any position in the workspace;
- Only one obstacle exists in the robot environment;
- The robot is controlled by its orientation.

The mobile robot moves in a field of forces, an attractive force (target) and a repulsive force (obstacle), the resultant of these forces determines the robot orientation (Khatib, 1986).

The Fig. 1 shows the mobile robot, the target position and the obstacle in the workspace.

V_{att} : attractive vector
V_{rep} : repulsive vector
V_{or} : orientation vector

COLLISION AVOIDANCE STRATEGY

A fuzzy system is a system based on the concepts of approximate reasoning: linguistic variables, fuzzy propositions, linguistic if-then rules.

The goal is to realize a fuzzy controller able to evaluate the repulsive force (vector) V_{rep} characterizing the actual relative position of the obstacle (Khatib, 1986; Kermiche and Abbassi, 2004).

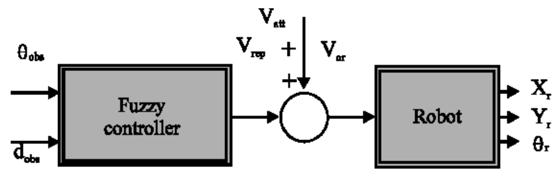

Fig. 2: Controller+Robot

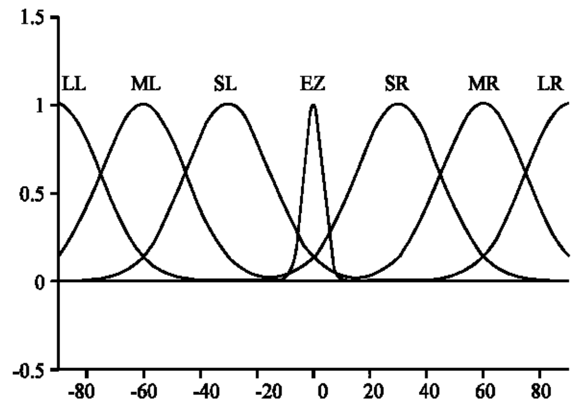

Fig. 3: Membership function for the 1st input variable θ_{obs}
LL: Large Left; ML: Middle Left; SL: Small Left
EZ: Zero Environment; SR: Small Right; MR: Middle Right, LR: Large Right

The controller has two inputs and one output, the inputs are the observation angle θ_{obs} and the distance d_{obs} towards the obstacle, the output is the repulsive vector V_{rep}. (Fig. 2).

The orientation angle depending on V_{or} is the input of the robot and its outputs are the coordinates (x_r, y_r) and the direction θ_r.

Fuzzification: The fuzzification module performs two tasks:

- Input normalisation, mapping of input values into normalised universes of discourse,
- Transformation of the crisp process state values into fuzzy sets, in order to make them compatible with the antecedent parts of the linguistic rules that will be applied in the fuzzy inference engine.

Fuzzification of the angle θ_{obs}: We suppose that the robot can perceive an obstacle in a direction inside the interval [-90° 90°]. The membership function is represented by seven fuzzy subsets of Gaussian ash shown in Fig. 3.

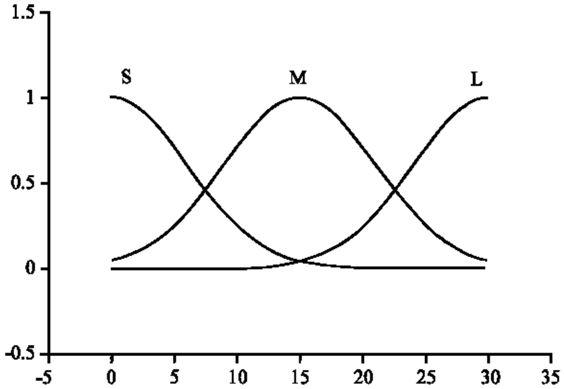

Fig. 4: Membership functions for second input variable The membership function is expressed by three fuzzy subsets: S: Short; M: Medium; L: Long

Fuzzification of the distance d_{obs}: We admit that the robot can detect an obstacle from a distance of 30 units (Fig. 4).

Fuzzification of the repulsive angle θ_{rep}: The membership function of the repulsive angle has a constant form belonging to the interval [-135°135]

Inference: Let $x_1, x_2, ..., x_m$ be linguistic variables on the input space $X = X_1 \times X_2 ... \times X_m$ and y be a linguistic variable (or a real variable) on the output space Y; then two forms of fuzzy inference rules by the fuzzy "IF... THEN..." rule model can be described as follow (Shi and Mizumoto, 2000):

Form (1): Fuzzy Inference Rules by Product-Sum-Gravity Fuzzy Reasoning Method.

The fuzzy inference rules are defined as:

Rule 1: IF x_1 is A_{11} and x_2 is A_{21} and... and x_m is A_{m1} then y is B_1 (3)

Rule 2: IF x_1 is A_{12} and x_2 is A_{22} and... and x_m is A_{m2} then y is B_2 (4)

Rule n: IF x_1 is A_{1n} and x_2 is A_{2n} and... and x_m is A_{mn} then y is B_n (5)

Where A_{ji} (j = 1, 2,..., m; i = 1, 2,..., n) and B_i are fuzzy subsets of X_j and Y, respectively and the subscript i corresponds to the ith fuzzy rule.

Form (2): Fuzzy Inference Rules by Simplified Fuzzy Reasoning Method.

The fuzzy inference rules are defined as:

Rule 1: IF x_1 is A_{11} and x_2 is A_{21} and... and x_m is A_{m1} then y is y_1 (6)

Rule 2: IF x_1 is A_{12} and x_2 is A_{22} and... and x_m is A_{m2} then y is y_2 (7)

Table 1: Fuzzy rule table

θ_{rep}		θ_{obs}						
		LL	ML	SL	EZ	SR	MR	LR
	S	APP	MP	AGP	TGN	AGN	MN	APN
d_{obs}	M	TPP	APP	MP	GN	MN	APN	TPN
	L	EZ	TPP	APP	PN	APN	TPN	EZ

Rule n: IF x_1 is A_{1n} and x_2 is A_{2n} and... and x_m is A_{mn} then y is y_n (8)

Where A_{ji} (j = 1, 2,..., m; i = 1, 2,..., n) is a fuzzy subsets of X_j and y_i is a real number on Y (Faerjon, 1987).

For examples the representations of these rules would then be constructed as folloows:

IF (θ_{ods} is LL and d_{ods} is S) then θ_{rep} is APP OR
IF (θ_{ods} is LL and d_{ods} is M) then θ_{rep} is TPP OR
IF (θ_{ods} is LR and d_{ods} is L) then θ_{rep} is EZ.
The rules are shown in Table 1.

DEFFUZZIFICATION

The defuzzification module performs the conversion of the union of modified fuzzy sets into a crisp output value followed by the denormalisation of this value.

The height method is the simplest and fastest one because only peak values of the modified fuzzy sets are taken into consideration. The resulting crisp output is the weighted sum of the peak values with respect to the heights of the modified fuzzy sets.

It is interesting to notice that for this type of defuzzification, we do not need to define the widths of the membership functions. It follows that a set of output membership functions can be defined as shown in Fig. 5. This type of membership functions is called singletons. This definition corresponds to the special case of Takagi and Sugeno's controller.

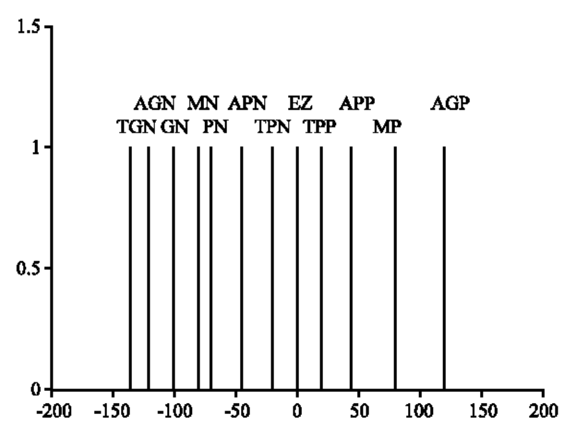

Fig. 5: Membership function of the repulsive angle θ_{rep}

ADJUSTABLE FUZZY CONTROLLER

The controller is based on Sugeno inference method and the defuzzification uses height method. The controller's output is function of linguistic variables (Foulloy and Galichet, 1995; Buhler, 1997).

Linguistic rules: We suppose that the controller has M+K linguistic variables, M inputs, K outputs and N linguistic rules (6, 7 and 8).

Learning method: It is used to determine the controller parameters values in order to reach the desired outputs for given inputs (Fig. 6).

General algorithm: The model is described by the following equations (Godjevac, 1997):

The parameters estimation consists to minimize the criterion V:

$$V = E\left\{\left(\vec{e}(t)\right)^2\right\} \tag{9}$$

Or

$$V = E\left\{\frac{1}{N}\sum_{t=1}^{N}\left(\vec{e}(t)\right)^2\right\} \tag{10}$$

E: means
N: number of iterations
e(t): learning error vector

In order to minimise the learning error, we will find the minimum of the criterion function V. It can be found by solving:

$$-\nabla_z V = \left[-\frac{\partial \nabla}{\partial z_1},\ldots\ldots,\frac{-\partial \nabla}{\partial z_p}\right] = 0 \tag{11}$$

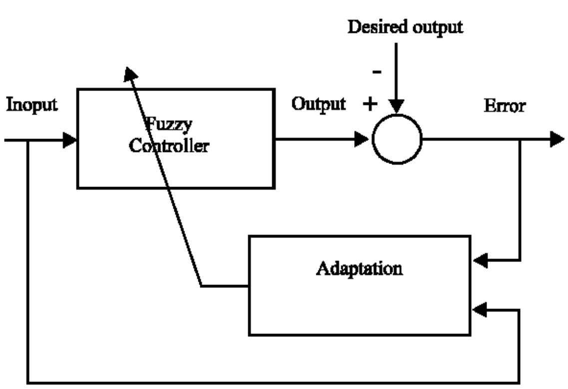

Fig. 6: Training scheme

$-\nabla_z V$: Gradient of V
P: number of adjustable parameters

$$z_p(t+1) = z_p(t) - \Gamma_p \nabla_z V\left[z_p(t)\right] \tag{12}$$

Γ_p: is the predefined constant named learning rate.

Adaptation of parameters of Takagi and Sugeno's controller: The Takagi and Sugeno's fuzzy controller has three types of parameters to adapt:

- Center values
  $a = (a_{11},\ldots,a_{nm},\ldots,a_{NM})^T$,
- Width values
  $b = (b_{11},\ldots,b_{nm},\ldots,b_{NM})^T$,
- Consequences values
  $c = (c_{11},\ldots,c_{nk},\ldots,c_{NK})^T$,

Then

$$\vec{Z} = (a_{11},\ldots,a_{NM},b_{11},\ldots,b_{NM},c_{11},\ldots,c_{NK})^T \tag{13}$$

The number of parameters to adapt is:

$$P = 2N\times M + K\times N$$

The vector which minimize the criterion function is given by:

$$\frac{-\partial V}{\partial a_{11}},\ldots,\frac{-\partial V}{\partial a_{NM}},\frac{-\partial V}{\partial b_{11}},\ldots,\frac{-\partial V}{\partial b_{NM}},\frac{-\partial V}{\partial c_{11}},\ldots,\frac{-\partial V}{\partial c_{NK}} = 0$$

And the recursive (learning) rules:

$$a_{nm}(t+1) = a_{nm}(t) - \Gamma_a \frac{\partial V(z)}{\partial a_{nm}} \tag{14}$$

$$b_{nm}(t+1) = b_{nm}(t) - \Gamma_b \frac{\partial V(z)}{\partial b_{nm}} \tag{15}$$

$$c_{nk}(t+1) = c_{nk}(t) - \Gamma_c \frac{\partial V(z)}{\partial c_{nk}} \tag{16}$$

If the membership functions of the controller are Gaussians, then the partial derivatives of the criterion V are:

$$\frac{\partial V}{\partial a_{nm}} = \sum_{k=1}^{K}(y_k - y_{dk})\frac{u_n}{\sum_{n=1}^{N}u_n}(c_{nk} - y_k)\frac{x_m - a_{nm}}{b_{nm}^2}$$

$$\frac{\partial V}{\partial b_{nm}} = \sum_{k=1}^{K}(y_k - y_{dk})\frac{u_n}{\sum_{n=1}^{N} u_n}(c_{nk} - y_k)\frac{(x_m - a_{nm})^2}{b_{nm}^3}$$

$$\frac{\partial V}{\partial c_{nk}} = (y_k - y_{dk})\frac{u_n}{\sum_{n=1}^{N} u_n} \tag{17}$$

The adaptation of the parameters of the Gaussians and weights is done by

$$a_{nm}(t+1) = a_{nm}(t) - \Gamma_a \frac{u_n}{\sum_{n=1}^{N} u_n}\frac{x_m(t) - a_{nm}(t)}{b_{nm}(t)^2}$$

$$\sum_{k=1}^{K}(y_k(t) - y_{dk}(t))(c_{nk}(t) - y_k(t)),$$

$$b_{nm}(t+1) = b_{nm}(t) - \Gamma_b \frac{u_n}{\sum_{n=1}^{N} u_n}\frac{(x_m(t) - a_{nm}(t))^2}{b_{nm}(t)^3}$$

$$\sum_{k=1}^{K}(y_k(t) - y_{dk}(t))(c_{nk}(t) - y_k(t));$$

$$c_{nk}(t+1) = c_{nk}(t) - \Gamma_c \frac{u_n}{\sum_{n=1}^{N} u_n}(y_k(t) - y_{dk}(t))$$

Linguistic rules extraction: The problem of the linguistic rules extraction is to convert the parameters (a_{nm}, b_{nm}, c_{nk}) at the end of the adjustment to linguistic values.

To solve this problem we compare the membership functions defined by a_{nm} et b_{nm} with preset ones whose linguistic terms belong to a set of trajectories provides by an expert person.

The controller parameters are identified by an off-line procedure based on the memorized data and the initial parameters.

- Number of inputs M = 2
- Number of outputs K = 1
- Number of rules N = 21

SIMULATION

The source, the target and the obstacle positions are specified.

Test:

Before training the robot moves from a start configuration to a goal configuration without collision (Fig. 7).

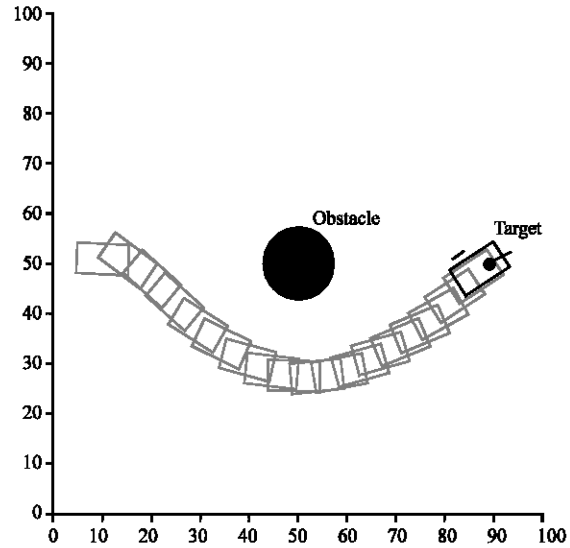

Fig. 7: Before training

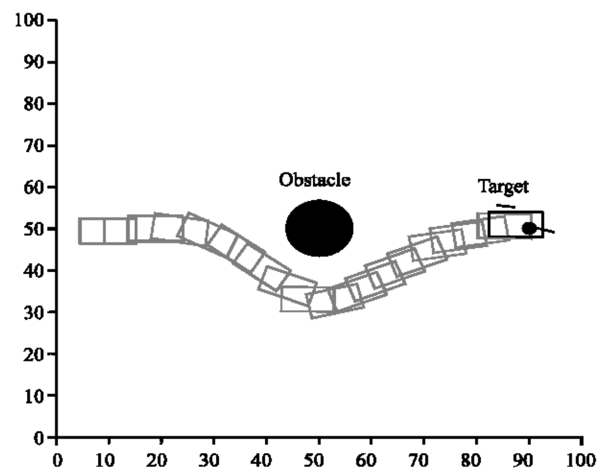

Fig. 8: The training

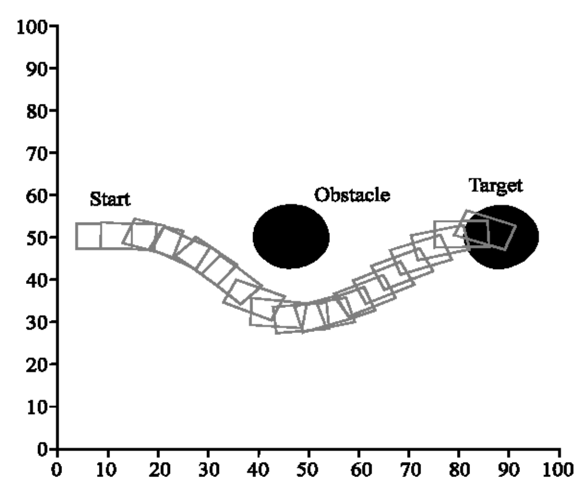

Fig. 9: After training

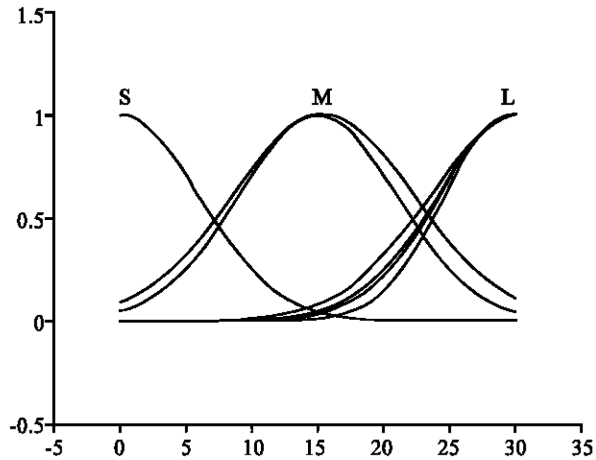

Fig. 10: Crisp membership functions for the second input

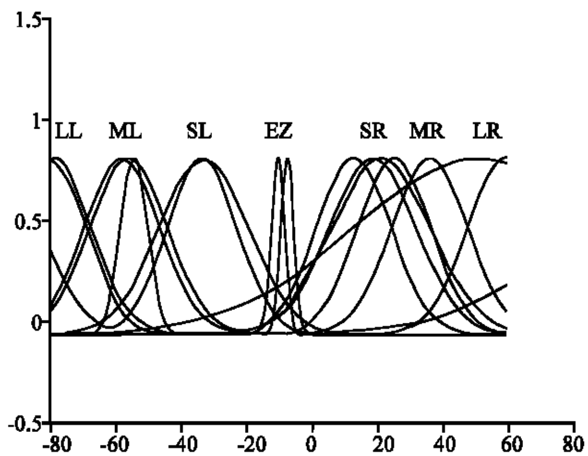

Fig. 11: Crisp membership functions for the first input

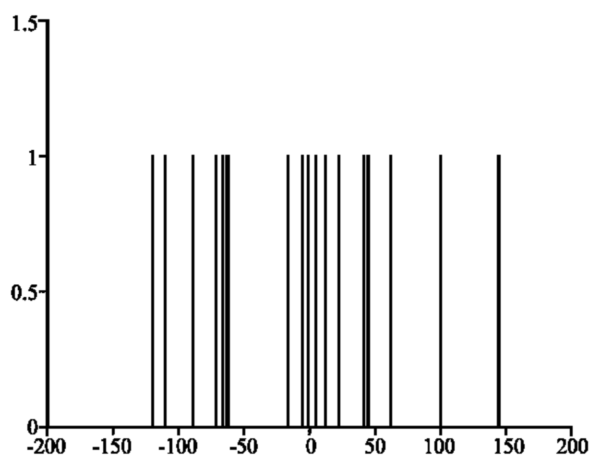

Fig. 12: Crisp membership functions for the output

Figure 8 shows the trajectory specified by the operator from the initial position to the final position (training).

Figure 9 describes the robot trajectory with collision avoidance after training.

The Fig. 10 and 11 represent the adjustment of the center and the width of each membership function after the training.

We note that the trajectory after training is optimal compared to the one before training (Fig. 11).

We have observed that after training, the concentration of singletons is located on left side. This is due to the avoidance of obstacle is done on the right side (Fig. 12).

CONCLUSIONS

In this study, we have presented a solution to the problem of trajectory tracking without collision.

The robot motion depends on the potential field approach.

The robot moves in a field of forces where the goal position is an attractive pole and where the obstacle is a repulsive pole. The attractive force is calculated from the goal position and the repulsive force is determined by a fuzzy logic.

The robot has to follow a trajectory specified by the operator from a start configuration to a goal configuration, which goes through a fixed obstacle.

When a potential collision with the obstacle is detected, the collision avoidance redirects the robot motion thanks to the repulsive force in order to generate a new collision free path.

The method has been tested on robot and the results are very satisfactory.

REFERENCES

Baturone, I., F.J. Moreno-Velo, S. Sànchez-Solano and A. Ollero, 2004. Automatic design of fuzzy controllers for car-like autonomous robots. IEEE Trans. Fuzzy Sys., 12: 447-465.

Buhler, H., 1994. Réglage par logique flou. Presses Polytechniques et Universitaires Romandes.

Faverjon, B., 1987. Obstacle avoidance using octree in the configuration space of a manipulator. In Proceedings of the IEEE International Conference on Robotics and Automation, Atlanta, GA.

Foulloy, L. and S. Galichet, 1995. Typology of Fuzzy Controllers, Teoretical Aspects of Fuzzy Control, John Wiley and Sons, Inc., New York.

Godjevac, J., 1997. Neuro-fuzzy Controllers, Design and Application. French University Polytechnic Presses.

Kang, W., N. Xi and J. Tan, 1999. Analysis and design of noontime based motion controller for mobile robots. In Proc. IEEE Intl. Conf. Robotics Automation, Detroit, MI, pp: 2964-2969.

Kermiche. S. and H.A. Abbassi, 2004. Fuzzy logic control of robot manipulatotion in presence of obstacle. The Fifth International Conference on: Computational Aspects and Their Applications in Electrical Engineering. CATAEE.

Khatib, O., 1986. Real time obstacle avoidance for manipulators and mobile robots. Intl. J. Robotics Res., 5: 90-98.

Lallemand, J.P. and S. Zeghloul, 1994. Robotique Aspects Fondamentaux, Masson.

Lozano, T.P., 1983. Spatial planning: A configuration space approach. IEEE Trans. Computers, C-32: 108-120.

O'Dùnlaing, C. and C.K. Yap, 1985. A retraction method for planning the motion of a disk. J. Algorithms, 6: 104-111.

Sharir, M. and A. Schorr, 1984. On shortest paths in polyhedral spaces. In: Proceedings of the 16th ACM Symposium on Theory of Computing, Washington, DC.

Shi, Y. and M. Mizumoto, 2000. Some considerations on conventional neuro-fuzzy learning algorithms by gradient descent method. ELSEVIER Fuzzy Sets Systems, pp: 112.

Yang, S.X., H. Li, M.Q.H. Meng and P.X. Liu, 2004. An embedded fuzzy controller for a behaviour-based mobile robot with guaranteed performance. IEEE Trans. Fuzzy Sys., 12: 436-446.

Zhao, Y. and S.L. Bement 1992. Kinematics, dynamics and control of wheeled mobile robot. In Proc. IEEE Intl. Conf. Robotics and Automation, Nice, France, pp: 91-96.