Mathware & Soft Computing 5 (1998) 279-289

Fuzzy Controller for Obstacle-Avoidance with a Non-Holonomous Mobile Robot

Juan Pedro Uribe and Joseba Urzelai
Department of Artificial Intelligence.
IKERLAN. Applied Research Centre
e-mail: jpuribe@ikerlan.es & robot_movil@ikerlan.es

Abstract

This paper describes the design and development of a sensor based navigation system which makes it possible for a non-holonomous mobile robot to avoid obstacles using information on its environment picked up by a belt of ultrasonic sensors. To control the robot no preliminary information regarding its environment is required, the robot adapts to them through the information gathered on the spot by the ultrasonic sensors and the information released from a spatial memory. The controller, which allows the generation of the signal to the robot's motors, has been developed based on the concept of general perception, and the principles of fuzzy logic have been applied. The platform on which the navigation system has been tested, is a prototype mobile robot (VEA-II) developed at IKERLAN, which is highly manoeuvrable, and capable of moving both in a holonomous and non-holonomous configurations.

Key words: Mobile robot, local navigation, spatial memory, fuzzy logic.

1 Description of the mobile robot

The basic aim of the project which includes the navigation system we are dealing with, is to build and control a vehicle equipped with an arm capable of operating in an autonomous and teleoperated manner with a view to its possible utilisation in telemanipulation applications in indoor and outdoor environments.

This project enables new technical and scientific solutions to be sought in the field of mobile robotics, in conjunction with telemanipulation and offering a platform for testing the different subsystems which are involved.

The prototype developed has been divided into three clearly differentiated parts: a control station (responsible for aspects of control which do not need a real-time response), a manipulation arm installed in the vehicle (a PUMA 560 robot installed in the front end of the vehicle) and a vehicle. The arm-vehicle assembly is equipped with a number of sensors (ultrasonics, vision cameras, laser telemeter) which supply information about its environment, enabling the unit to react to unexpected events (obstacles in its course, changes in ground inclination, etc.) A description of each of the three parts is given below.

Figure 1. The mobile robot VEA-II.

The mobile vehicle shown in the previous figure has a pseudo-rectangular base and measures 1400 x 800 mm.

The locomotion system is based on four wheels each of which has independent traction and steering. In developing this subsystem, emphasis has been given to the fact that the robot must not only be able to function on level ground but also go up and down slopes and operate on rough ground, etc.

Trajectory control allows the vehicle to travel in any direction, and to make all the necessary manoeuvres to reach any accessible configuration in its working environment.

The sensors on the vehicle are, on the one hand, a 24-ultrasonic sensor belt providing information used by the controller to detect and avoid unexpected obstacles and to follow the contours of objects [8]. On the other hand, a laser telemeter is used to locate the vehicle and to model free spaces in which the vehicle may travel.

As mentioned earlier, the study and development of navigation techniques have been approached on the basis of sensorial information. These techniques enable the vehicle to move within its environment, following collision-free paths, adapting the pre-programmed path to any possible unexpected obstacles, avoiding them and it can also plan movements by taking into consideration the successive modelling of the environment carried out by the vehicle's sensorial systems.

2 General perception and distributed perception

The limitations of ultrasonic sensors as regards directional resolution are notorious. Although these can determine the distance to a given object with a high degree of precision, the object may be located at any point within the sensor cone. Furthermore, the angle of the cone depends on the surface of the object, on the distance and angle of incidence, and thus it is difficult to make a model of the environment in order to guide the robot. The concept of both general perception and distributed perception avoid modelling of the environment, and attempt to build a system of vectors to supply information on these.

In case of distributed perception, these vectors are known as *distributed perception vectors*, since there is one vector for every side of the vehicle. On the other hand, in general perception there is a single vector called *general perception vector* [2, 3, 8].

In order to compose each perception distributed vector P_d , the set of sensorised points is replaced by means of a straight line. The direction of the distributed perception vector P_d will be perpendicular to the straight line calculated earlier, with a module equal to the maximum perception on that side.

$$p_d = \frac{d_{max} - d_{d,min}}{d_{max} - d_{min}} \tag{1}$$

where d_{min} and d_{max} represent, respectively, the minimum and maximum distance the sensor is able to measure. $d_{d,min}$ is the distance measured by the sensor detecting the nearest obstacle. p_d is between the limits 0 and 1, and thus:

$$p_d = \begin{cases} 0 & \text{si } d_{d,min} > d_{max} \\ 1 & \text{si } d_{d,min} < d_{min} \end{cases} \tag{2}$$

In this way the rectangular robot's four distributed perception vectors are made up.

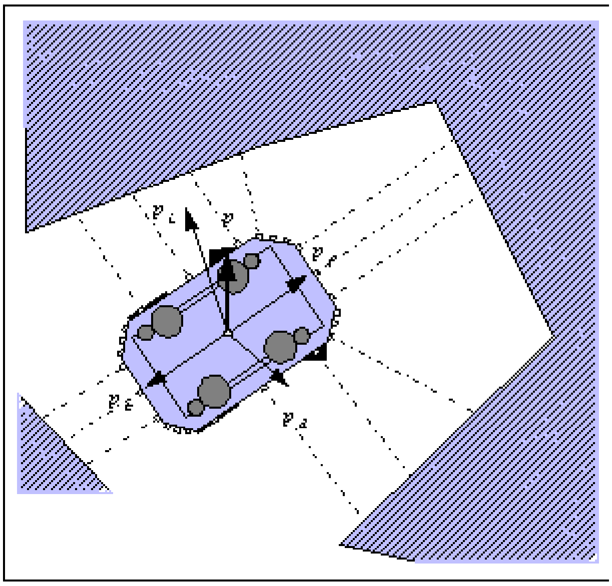

Figure 2: The distributed perception vectors and the general perception vector.

The general perception vector pg is made up of the distributed perception vectors. The direction corresponds to the weighted vectorial sum of the distributed vectors and the module is equal to the maximum value of the distributed perception modules:

$$p_g = p_{d,max} \frac{\sum k_d p_d}{|\sum p_d|} \tag{3}$$

The term kd in the vectorial sum weighs the distributed vectors, in accordance with the size of the vehicle, since this has considerable influence on its behaviour.

3 Spatial Memory

Sometimes, the vehicle loses some perception of its environment due to poor orientation (sensors at excessive angles with respect to the obstacle). However, assuming these are static obstacles, the robot may recognise their location through the information received from previous movements, that is, the robot could memorise previous perceptions (transferred odometrically) and use them to cover sensing losses.

The set of points perceived by the vehicle through the sensors is replaced by a straight line: this straight line is characterised by a slope $m = tg\alpha$ and the nearest point p. As can be seen in the figure, any movement of the vehicle with respect to the wall is equivalent to a turning-transfer movement of the straight line with respect to the vehicle.

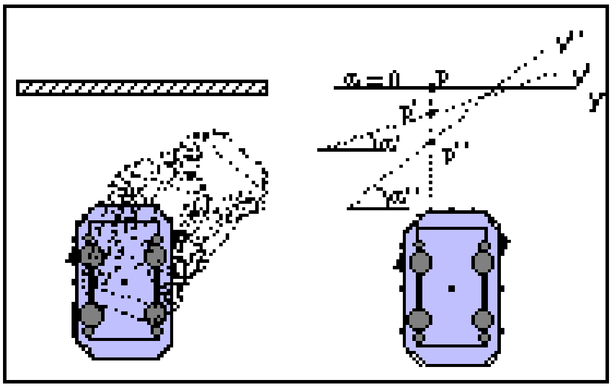

Figure 3: The information detected by the sensors by means of a straight line characterised by m and p.

Therefore, the robot will memorise the parameters of each step (m_i, p_i) to calculate the new points in the following iteration. The actual perception (m_i, p_i) always has priority over memorised perceptions (m_{im}, p_{im}). The robot obtains the memorised points by calculating the intersection of the sensor beams with the straight line defined by (m_{im}, p_{im}).

There are a number of restrictions with reference to the memorised parameters (m_{im}, p_{im}). In case of front perception, when the memorised slope m_{in} is greater than the maximum slope m_{max}, the information can be transferred to the lateral and the points are deleted from the memory, since otherwise the robot would follow a fictitious straight line indefinitely, and be unable to execute convex curves. Another restriction occurs when the minimum slope is smaller than the angle in the sensor cone (this angle depends on the surface of the object). If $\{p_{11}, p_{21}, ...p_{n1}\}$ stands for the points corresponding to the n sensors on side l, the following analysis is made:

If for all p_{i1} it is assumed that $p_{i1} > p_{i+1,1}$ or $p_{i1} < p_{i+1,1}$, then it is a straight wall which the sensors should be able to detect, and so the points are deleted from the memory.

If, on the other hand, it is assumed that $p_{i1} < p_{j1} > p_{k1}$ or $p_{i1} > p_{j1} < p_{k1}$ when $i < j < k$, then the obstacle cannot be detected by the robot (due to an excessive angle of incidence) even though the resulting inclination remains within the sensoring cone and, therefore, the memorised points are maintained.

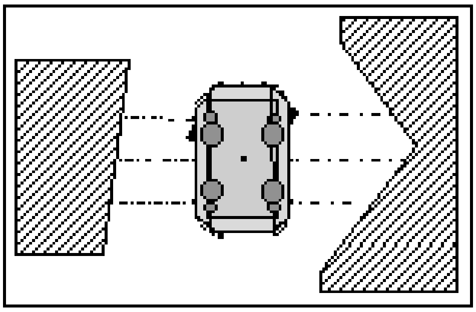

Figure 4: On the left, a situation is shown in which the memorised points must be eliminated because the wall should be detected by the sensors. On the right, however, the robot is unable to detect the wall because the angle between the two walls is too acute; in this case, the robot must take the memorised points into account.

Another important aspect of spatial memory is the transfer of information from the front memory to lateral memories, from front to back with respect to the direction of movement, as the orientation of the obstacle changes with regard to the vehicle.

Let $\{p'_{11}, p'_{21}, ...p'_{n1}\}$ be the points obtained for the n lateral sensors from the points stored by the m front sensors $\{p_{1f}, p_{2f}, ...p_{mf}\}$, and let $\{p_{11,m}, p_{21,m}, ...p_{n1,m}\}$ be the points memorised by the robot from the lateral sensors. The minimum operator is used to establish the final lateral points:

$$\text{For all of } i, p_{il} = \textbf{min}\ (p'_{il}, p_{il,m}) \tag{4}$$

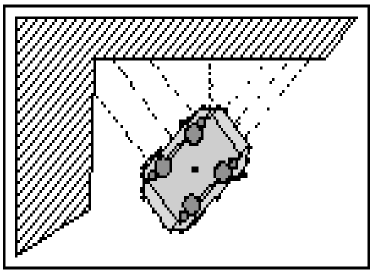

Figure 5: Thanks to memory transfer the robot is able to detect the wall ahead with its lateral sensors before it reaches the scope of its sensor cone.

Memory transfer becomes extremely useful in situations such as walls forming right-angles (Fig. 5). When the information from all the lateral sensors matches the memorised information transferred from the front sensors, the information from the memory of the front sensors will be eliminated since this information is now wholly present in the lateral sensors.

Finally, it should be mentioned that thanks to spatial memory the number of situations in which the robot loses perception altogether is reduced and what is more, the transfer of perception from a front to a lateral location allows information to be supplied even before the lateral sensors are able to detect an obstacle.

4 Fuzzy Controller

When the mobile robot executes a path planning calculated mission, it often finds obstacles in its path. In these cases, the mobile robot has to avoid these obstacles and go to a point on the path planner´s trajectory.

The aim of this controller is to reach an end goal, avoiding the obstacles on the path. The mobile robot is led by information acquired with ultrasound sensors and it always knows the location of the end goal.

In this paper only the controller based on the general perception is developed.

4.1 Inputs

4.1.1. General Perception

The representation of inputs is based on general perception vectors and the angle of the mobile robot with respect to the end goal. The input variables of the controller are general perception p, general perception angle α and the angle of the mobile robot related to the end goal β. Fuzzy variables are called *modP, angP and angGoal* respectively.

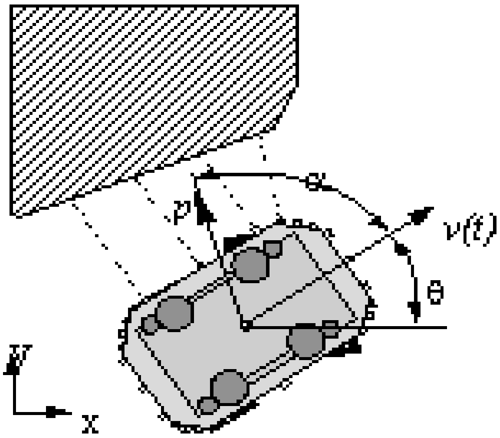

Figure 6: Representation of the antecedents and consequents of the general perception controller.

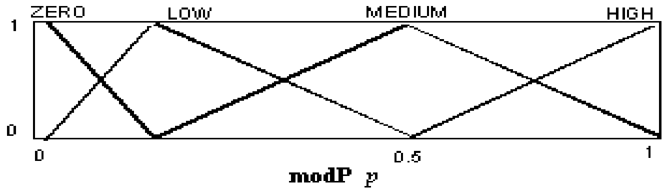

Figure 7: Adjectives of the General Perception.

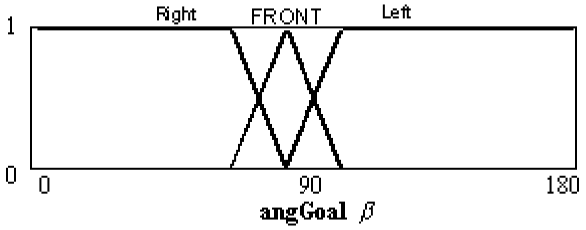

Figure 8: Adjectives of the General Perception angle.

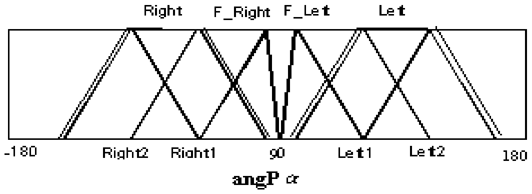

Figure 9: Adjectives of the angle of the mobile robot with respect to the end goal.

4.2 Outputs

Output variables are the variation in vehicle orientation θ and vehicle acceleration a. The corresponding fuzzy variables are called *incTeta* and *Acel*, respectively.

The membership functions of the fuzzy output variable adjectives are of the *Singleton* type.

The variable *incTeta* represents the variation in vehicle orientation (not of the wheels). The adjective membership functions are shown in Figure 10.

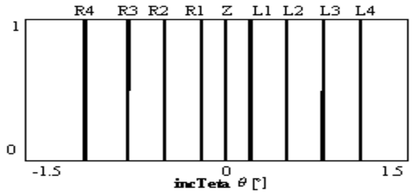

Figure 10: Singleton adjectives represent the change in vehicle orientation.

The variable *Acel* is defined by means of five adjectives which allow the vehicle to be accelerated and braked, and an adjective EM_BRAKE which produces emergency braking of the robot in case of danger. Figure 11 shows the membership functions of the adjectives.

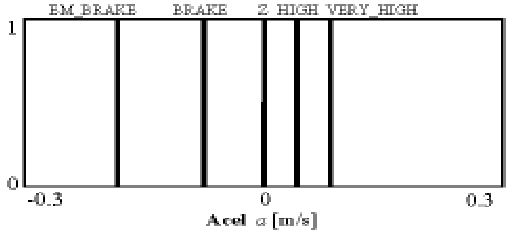

Figure 11: Singleton adjectives represent the acceleration of the autonomous mobile robot

4.3 Rule Base

The aim of this module is to move the mobile robot from the initial position to the end goal. The end goal produces an attractive force on the mobile robot and the possible obstacles on the path produce a repulsion force. It tries to get a balance among these forces to move the mobile robot on the free spaces onto the end goal.

Taking into account this idea, two groups of rules for orientation control have been defined. The first one represents the attractive force of the goal point over the mobile robot. The second one follows the repulsion forces exercised by the obstacles. These rules constrain the mobile robot to move away from perceived obstacles.

The speed control is built in the repulsion rules group, since if the mobile robot detects no obstacles, it must accelerate to reach maximum speed. The mobile robot must brake when avoiding close obstacles.

It must be pointed out that the repulsion forces have priority over the attractive ones, as a collision would be intolerable.

Evaluation of rules is made by the product and sum combination method and defuzzification is made by the centroid method.

a. Direction Control

This rules base consists of twenty two rules and its inputs are the general perception module p, the general perception angle and the angle of the mobile robot with respect to the goal point. As output it generates variations in vehicle direction.

It consists of two groups of rules, the first one heads the mobile robot for the goal when the way is free, guaranteeing obstacles remoteness (modP) and turns to the goal provided that there is no obstacle in that direction (angP and andGoal).

In the table 1, the rules are preceded by:

If modP is NOT HIGH AND modP is NOT MEDIUM AND...

Or

If modP is ZERO OR modP is LOW AND...

α/β	LOW	MEDIUM	HIGH
NOT RIGHT	R4	–	–
NOT F_RIGHT	–	Z	–
NOT F:LEFT	–	Z	–
NOT LEFT	–	–	R1

Table 1: Rule base controlling vehicle direction (L: Left, R: Right, Z: Zero).

The second group of rules solves the conflict among goal attraction and repulsion of some obstacles on the way. The strategy is: if there is an obstacle between the goal and the robot and both are in front of it (FRONT), the obstacle avoidance has the priority so the robot will avoid the obstacle and reach the goal following another way. If the obstacle is on one side and somewhat in front of the robot (RIGHT1, LEFT1), it has to avoid it but without much turning. Finally, if the obstacle is on one side and the robot is overtaking it (RIGHT2, LEFT2), the robot keeps moving towards the goal. In these three groups there are different turning levels, depending on how far the obstacle is from the robot.

b. Speed Control

This rule base is made up of twelve rules and the inputs are the general perception module p and the general perception angle α. Its output is the value for vehicle acceleration.

Speed is limited between ν_{max} and ν_{min} where ν_{min} is calculated as:

$$v_{min} = \text{Min}(0.05, 1 - p)$$

The idea is that if there are not obstacles or they are far away, the mobile robot accelerates, but if they are closer: it stops if the obstacle is located frontally, it brakes if it is somewhat in the front but somewhat on the right (left) and it accelerates if it is on one side (changing the level depending on the perception, MEDIUM or HIGH):

α/p	ZERO	LOW	MEDIUM	HIGH
RIGHT2	VH	VH	VH	VH
RIGHT1	VH	VH	H	B
F_RIGHT	VH	VH	Z	E
F_LEFT	VH	VH	Z	E
LEFT1	VH	VH	H	B
LEFT2	VH	VH	VH	VH

Table 2: Rule base controlling the vehicle speed (H: High, E: Emergency, Z: Zero).

5 Results and conclusions

After tuning the membership functions with genetics algorithms [2], the results obtained in both the simulator and the robot VEA-II (Experimental Autonomous Vehicle II) developed at IKERLAN are very satisfactory.

Similar to wall following robot [8], the next step is to develop the distributed perception controller to achieve a smoother control and a better solution when moving in environments with high density of obstacles.

It is worth noting the robustness given by the spatial memory system.

The next figure shows a typical example of obstacle avoidance:

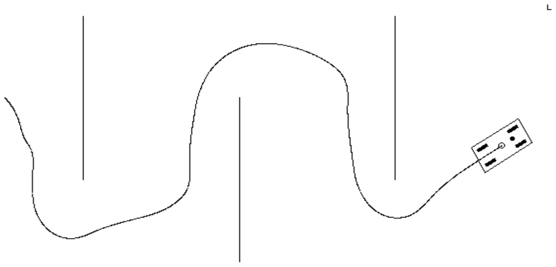

Figure 12: Obstacles avoidance

Hereafter, the next figures show, on the one hand, the AVOIDANCE module dropping down local minima and afterwards a mixed strategy where if the goal is left behind the WALL FOLLOWING controller is activated as far as it confronts the goal.

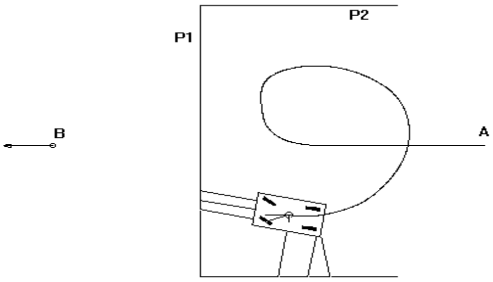

Figure 13: Local Minima in obstacles avoidance.

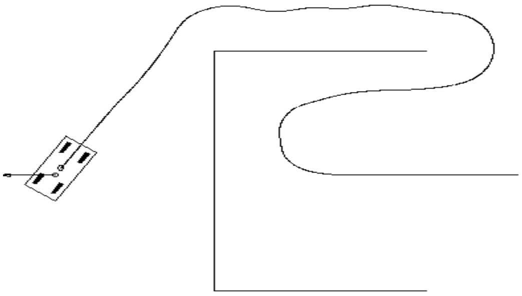

Figure 14: Mixed strategy of obstacles avoidance and wall following.

References

[1] Borenstein J., Koren Y., The Vector Field Histogram - Fast Obstacle Avoidance for Mobile Robots. IEEE Transactions on Robotics and Automation, vol. 7, n. 3, June 1991, pp. 278-288.

[2] Braunstingl Reinhard, Jokin Mujika, Juan Pedro Uribe, A Wall Following Robot With A Fuzzy Logic Controller Optimised By a Genetic Algorithm. FUZZ-IEEE/IFES ´95 FUZZY ROBOT COMPETITION, YOKOHAMA, MARCH, 1995.

[3] Braunstingl R., Sanz P. and Ezkerra J.M., Fuzzy Logic Wall Following of a Mobile Robot Based on the Concept of General Perception. ICAR´95, Seventh International Conference on Advanced Robotics, Catalonia, September 1995.

[4] Crowley J.L., Navigation for an Intelligent Mobile Robot. IEEE Journal of Robotics and Automation, vol. RA-1, n. 1, march 1985, pp. 31-41.

[5] Kathib O., Real-Time Obstacles Avoidance for Manipulators and Mobile Robots. The International Journal of Robotics Research, vol. 5, n. 1, Spring 1986, pp. 90-98

[6] Li W., Fuzzy Logic-based reactive behaviour of an autonomous mobile system in unknown environments. Engng applic. Artif. Intell., 1994, 11(8), pp. 743-760.

[7] Reignier P. Pilotage Reactif d'un Robot Mobile. Etude du Lien entre la Perception et l'Action. Ph. D. Thesis, Institut National Polytechnique de Grenoble, December 1994.

[8] Urzelai J, Uribe J P , Ezkerra J M, Fuzzy Controller for wall-following with a non holonomous mobile robot. FUZZ-IEEE/IFES ´97, BARCELONA, July, 1997.