# A new approach to design fuzzy controllers for mobile robots navigation

Olivier AYCARD, François CHARPILLET & Jean-Paul HATON
CRIN/CNRS & INRIA-Lorraine
Bâtiment LORIA, Campus scientifique
B.P. 239
54506 Vandœuvre CEDEX
FRANCE
Phone: +33 3 83592084, Fax: +33 3 83413079
E-mail: {aycard, charp, jph}@loria.fr

**Keywords:** Fuzzy systems, behavior based control, reactive navigation.

**Abstract**

*This article presents a new method to design, in two levels, fuzzy controller for reactive navigation of a mobile robot in a structured unknown environnment. At the first level, adjacent sensors are grouped in areas and are used to define local behaviors. These local behaviors are then gathered at the second level in order to define a global behavior. Two experiments with different local behaviors and different mechanism of integration are presented on our Nomad200 mobile robot.*

## 1 Motivations

A key issue in research in mobile robotics is to design and to build autonomous robots able to act and move in a real environment, to achieve tasks like delivery, surveillance or cleaning. Mobile robots could also have a major role in exploration and intervention in a hostile environment (like nuclear plant, for instance). But, a mobile robot moving in a real environment has some problems to solve :

- The environment is vast and dynamic. Obstacles (or people) can move, appear or disappear. The set of all possible situations can not be computed. The mobile robot should be equipped with sensors (cameras, sonar or infrared) to have information about its local environment and act with this information.
- Due to the type of the ground and the slippage of the wheels, the actions are not completely reliable. Moreover, the values of the sensors have different reactions depending of the humidity of air, the temperature, the shape and the type of the objects included in the environment.
- As the robot moves in its environment and the environment changes, it has to react in a finite (often short) time to a new event.

The goal of research in navigation is to find a model which given the goal to reach and the data of the sensors, delivers the best action to apply to the mobile robot. As these data are noisy and the actions are not completely reliable, fuzzy control seems to be an appropriate way to control a mobile robot.
Fuzzy control [4] is an application of fuzzy logic to the control of dynamic systems. [10] defines the fuzzy logic as an extension of classic logic for representation and reasoning about approximate data. Several mobile robotic applications of fuzzy logic have been realized. For instance, [6][8] use fuzzy logic to control a mobile robot in an unknown structured indoor environment.
Unfortunately, fuzzy controller are sometimes difficult to design, because of :

- The choice of appropriate inputs to make fuzzy.
- The definition of rules, and the possibility to give more importance to rules, or to group several rules.
- The way to defuzzy outputs of the system.

Some researchers have defined methods to design fuzzy controllers. For instance, [3] gives a different weight for each rule and a method to automatically learn these weights. [7] or [5] define high level behaviors (like avoiding obstacles, following a wall...) by grouping rules with a level of activation of each behavior, and uses an algorithm to define a global behavior according to the level of each behavior.

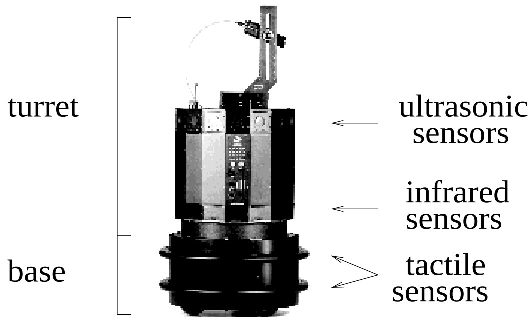


Figure 1: Our robot mobile

In this paper, we propose a new approach to design a fuzzy controller in two levels. Contrary to [7], we argue that the global behavior is the fusion of local behaviors of each physical part, instead of the fusion of high level behaviors. So in the first level, we merge adjacent sensors to define an area corresponding to the physical part of the mobile robot, and define a behavior for each area, and in a second level, we merge the different physical behaviors to provide a coherent global behavior for the mobile robot. This approach permits designing and testing each behavior in a modular way, and to test different techniques of behavior fusion.
This paper is organized as follow. In section 2, we give a short presentation of our mobile robot. In section 3, we define the fuzzy logic and its application to control. Section 4 is the description of our methodology. We present two experiments in section 5 and give some conclusions and perspectives in section 6.

## 2 Description of our robot

Our robot (figure 1) is a Nomad200 commercialized by [9]. It is composed of a base and a turret. The base is formed by 3 wheels and tactile sensors. The turret is a uniform 16-sided polygon. On each side, there is an infrared and an ultrasonic sensor. The turret can rotate independently of the base.

### 2.1 Tactile Sensors

A ring of 20 tactile sensors surrounds the base. They detect contact with objects. They are just used for the emergency cases. They are associated with low-level reflexes such as emergency stop and backward movement.

### 2.2 Ultrasonic Sensors

The angle between two ultrasonic sensors is 22.5 degrees, and each ultrasonic sensor has a beam width of approximately 23.6 degrees. By examining all 16 sensors, we can obtain a 360 degree panoramic view fairly rapidly. The ultrasonic sensors give range information from 17 to 255 inches. But the quality of the range information greatly depends on the surface of reflection and the angle of incidence between the ultrasonic sensor and the object.

### 2.3 Infrared Sensors

The infrared sensors measure the light differences between emitted light and reflected light. They are very sensitive to the ambient light, the object color, and the object orientation. As we assume that for short distances, the range information is acceptable, we just use infrared sensors for the areas shorter than 17 inches, where the ultrasonic sensors are not usable.

### 2.4 Odometry Measurements

The odometry measurement integrates the translation and rotation of the robot, and updates the position and orientation of the robot. As with all odometric systems, it accumulates errors during movements. We use it to have a coarse idea of the position and orientation of the robot.

## 3 Fuzzy logic and fuzzy control

A fuzzy set S is defined by its membership function $\mu_S(x)$. For each $x$, there exists a value $\mu_S(x) \in [0, 1]$ representing the degree of membership of $x$ to S. In fuzzy logic control, membership functions, associated with linguistic variables, are used to fuzzify physical quantities. For instance, if we use the linguistic variables defined in figure 4, a numerical value of 5 inches is very near with a degree of membership of 0.5, and near with a degree of membership of 0.5 too. Fuzzy inputs are used to form fuzzy rules. These rules characterize the relationship between fuzzy inputs and fuzzy outputs. For example, a simple fuzzy control rule relating input $v$ to output $u$ might be expressed in the condition-action form as follows :
*if $v$ is $V$ then $u$ is $U$* where $V$ and $U$ are fuzzy variables defined on the possible values of $v$ and $u$, respectively. The response of each fuzzy rule is weighted according to the degree of membership of its input conditions. The inference engine provides a set of control actions according to fuzzified inputs. Since the control actions are in fuzzy sense, hence a defuzzification method is required to transform fuzzy control actions into a precise output value of the fuzzy logic controller. A widely used defuzzification method is the centroid method :

$$u = \frac{\sum_{i=1}^{n} \mu_Y(c_i) * c_i}{\sum_{i=1}^{n} \mu_Y(c_i)} \quad (1)$$

where $u$ is an output value of the controller, $n$ is the number of control rules associated with the fuzzified inputs, and $c_i$ is the centroid of membership function associated with each linguistic variable in the output space.

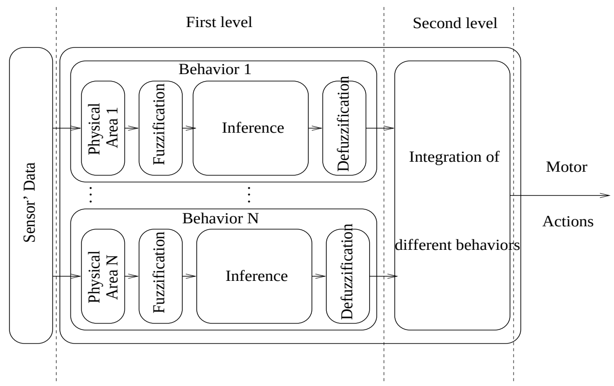


Figure 2: Our approach

# 4 Our approach

To define our fuzzy controller (figure 2), we use the 16 infrared/ultrasonic data as inputs, and we generate a motor action for the base and the turret as outputs. This generation is done in two levels :

- First, the adjacent sensors are strategically grouped to form a physical area. For each area, we choose a criterion (average, minimum) to define the fuzzified sensors' data using linguistic variables. We define rules for each physical area to give a fuzzified local motor action to the base and the turret corresponding to the desired behavior. As rules are written for each area, we do not have a conjunction in the condition part, so rules are easy to write. The corresponding local motor action is defuzzified using a classical method as the centroid method to compute a numerical rotation for the base and the turret. In fact, each behavior can be seen like a simple local fuzzy controller. At the end of the first level, we have a numerical local motor action for each physical area.
- Second, we define a mechanism to integrate all local behaviors to provide a global behavior to the mobile robot. We use two mechanisms to define the global behavior of the mobile robot :
  - A mechanism of fusion like [1], computing an average of all local actions.
  - A mechanism of inhibition like [2], where each local behavior has a level of priority and the actived (when the local motor action is not null) local behavior with the highest priority controls the mobile robot.

A combinaison of these two mechanisms is possible too.

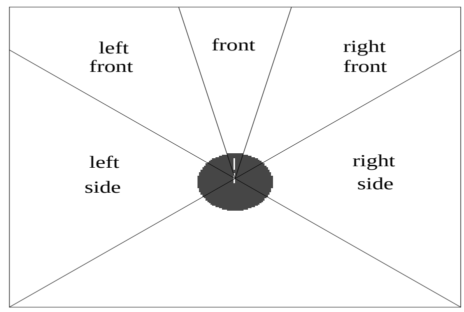


Figure 3: The 5 physical areas

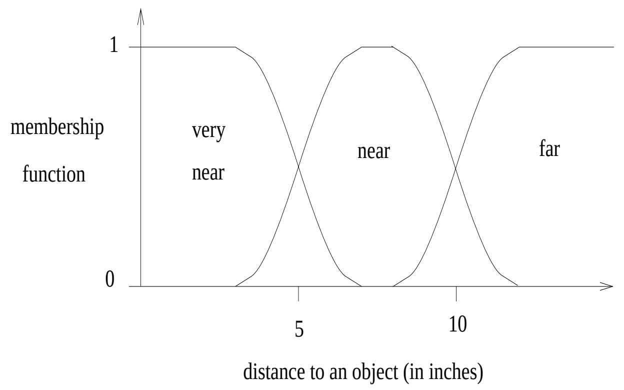


Figure 4: Linguistic Variables used for physical areas

# 5 Two experiments

## 5.1 Reactive navigation in corridor

In this first experiment, we define a fuzzy controller just to control reactively a mobile robot wandering in corridors.

### 5.1.1 Definition of physical behaviors

Of the 16 ultrasonic/infrared sensors, we use 11 sensors grouped in 5 physical areas called : front composed of 1 sensor, left front composed of 2 sensors, left side composed of 3 sensors, right front composed of 2 sensors and right side composed of 3 sensors (figure 3). For each physical area, we choose the minimum sensor value of the area as input of the behavior. This value is associated with a linguistic variable as shown in figure 4.

With these 5 physical areas, we define 5 physical behaviors (left side, left front, front, right side, left side). The left side and left front behaviors are used to avoid obstacles on the left of the mobile robot. The right side and right front behaviors are used to avoid obstacles on the right of the mobile robot. The front behaviors are used to avoid obstacles in front of the mobile robot. The side behaviors are used to avoid

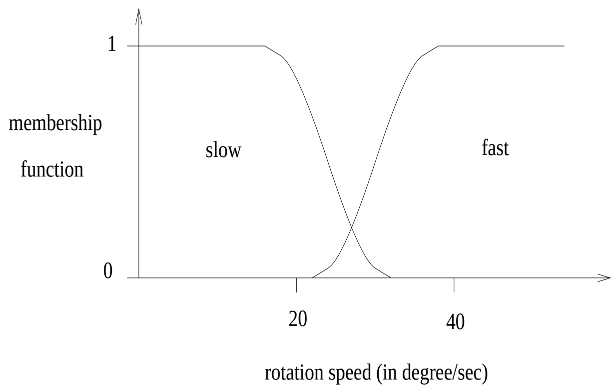


Figure 5: Linguistic Variables used for local motor actions

obstacles on the side of the mobile robot. To deliver a local motor action as the output of each physical behavior, we define linguistic variables (figure 5) associated with the action part of each rule, and use the centroid method to defuzzify it.
For the front areas, we want the following behavior :

- near or very near obstacles cause a fast rotation in the opposite direction of the obstacle for the mobile robot.
- distant obstacles do not cause any rotation for the mobile robot.

For the side areas, we want the following behavior :

- very near obstacles cause a fast rotation for the mobile robot.
- near obstacles cause a slow rotation for the mobile robot.
- far obstacles do not cause any rotation for the mobile robot.

For the front, as fast rotations are generally not sufficient to avoid obstacles in front and as our mobile robot can turn its base and its turret independently, we want the following behavior :

- very near obstacles cause a large shift of the base
- near obstacles cause a medium shift of the base
- far obstacles cause no shift of the base

The idea is that in presence of an obstacle in front of the mobile robot, the mobile robot will navigate like a crab. It means that the turret will stay facing the obstacle but the base will shift to permit to the mobile robot to go along the obstacle as it will find a passageway to avoid the obstacle. For this behavior, we need two new linguistic variables (figure 6). A shift is not a rotation speed but an angle of rotation.

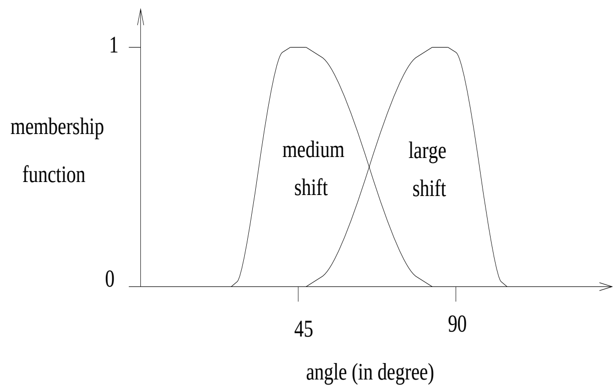


Figure 6: Linguistic Variables used for shift of the base

In the first level, we do not define the direction of the shift (left or right). This will be done in the phase of integrating local behaviors.
Given the definitions of each local behavior, the rules for each local behavior are intuitive. For example, the rule to avoid near obstacles in the left front area is :
`if left front is near then turn fast to the right`

#### 5.1.2 Integration of physical behaviors

All the local behaviors give a local motor action, and we need to integrate these local motor actions to find the best global motor action for the mobile robot.
In this experiment, all the local behaviors have the same priority so the global motor action will be a fusion of all these local motor actions. As all local behaviors are used to avoid obstacles, it makes no sense to say that some behaviors can inhibit the other because they have a higher degree of priority. As we use the local angular frame of reference of the mobile robot to define angles and rotation speeds, the rotations to the left (from the right side behavior and from the right front behavior) are positive and the rotations to the right (from the left side behavior and from the left front behaviors) are negative. To compute the global rotation speed of the mobile robot, we simply add the local rotation speeds generated by the left front, the left side, the right front and the right side behavior. This global rotation speed is the same for the base and the turret.
If a shift of the base is needed, the direction of the shift is defined by the sign of the rotation speed. The base will shift to the left when the global rotation speed is positive and to the right when the global rotation speed is negative. This can be intuitively explained by the fact that the shift will be done on the side where obstacles are furthest.

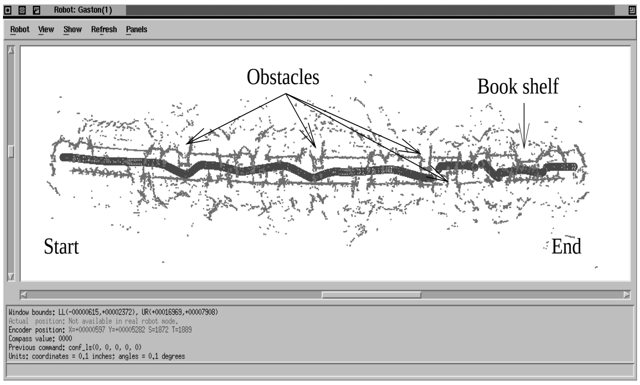


Figure 7: Following of a corridor

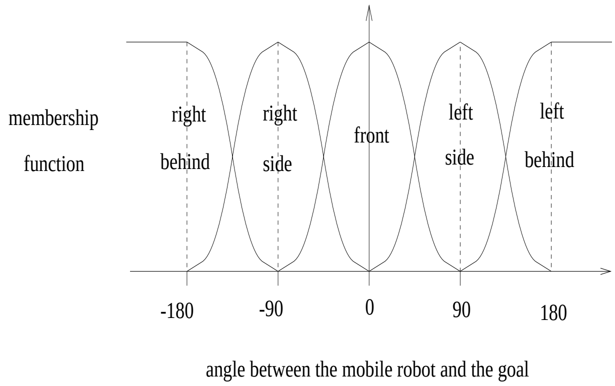


Figure 8: Linguistic Variables used to compute the angle between the robot and its goal

To compute the translation speed of the mobile robot, we simply subtract the absolute value of the rotation speed from the maximum translation speed. It means that more the robot turns the less quickly it goes and vice versa.

#### 5.1.3 Results on our mobile robot

We tested our fuzzy controller to control our mobile robot running in a corridor of approximately 30 meters. At the beginning, the mobile robot was positioned at the start of the corridor, at the same distance from the two walls, and in a direction parallel to the two walls. The mobile robot had to follow the corridor until the other end and to return to the start. This experiment was done 50 time, without any human intervention. The mobile robot had to avoid static obstacles, and people wandering in the corridor. The static obstacles were moved between the experiments. We give a trace of one run in figure 7.

The mobile robot does not really go straight, it goes from one wall to the other. But, it never touches walls and passes around static obstacles without collisions. Moreover, it avoids people wandering in the corridor.

### 5.2 Reactive navigation in a hall

In the first experiment, we saw that our mobile robot is able to navigate reactively and to avoid obstacles. But it is not able to reach a defined goal, because in the first experiment, the notion of goal to reach is unknown. In this second experiment, we add one behavior to take a goal in account, and modify the mechanism of integration.

#### 5.2.1 A new behavior to take a goal to reach in account

The goal of this behavior is to make the robot face its goal. To design this new behavior, we use odometric information to compute the angle between the mobile robot and the goal. As the odometry is not completely reliable (section 2.4), we fuzzify the angle between the robot and the goal using the linguistic variables represented in figure 8 and use the fuzzified rotation speed of figure 5. This behavior is called "reach goal behavior".

For the reach goal behavior, we want the following to occur :

- a goal behind the mobile robot causes a fast rotation speed to the goal.
- a goal situated on the side of the mobile robot causes a slow rotation speed to the goal.
- a goal situated in front of the robot does not cause rotation.

Given the definition, the rules for the reach global behavior are intuitive.

#### 5.2.2 Integration of the reach goal behavior

As in this experiment, we argue that avoiding obstacles and reaching the goal are two different things, it makes no sense to simply merge them to control the mobile robot.

To control our mobile robot, we use a combinaison of the two types of mechanism of integration :

- The 5 local behaviors of the first experiment are merged to avoid obstacles in the local environnement of the mobile robot.
- A mechanism of inhibition to choose which behavior (between the 5 local behaviors of the first experiment and the "reach goal behavior") will control the mobile robot.

We argue also that the 5 local obstacle avoidance behaviors are more important than the "reach goal be-

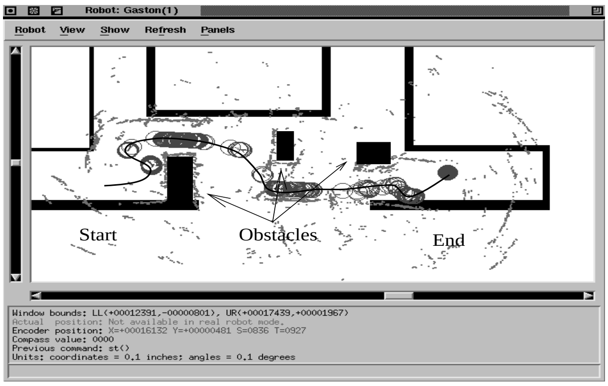


Figure 9: Navigation in a hall

havior", so we give them a higher priority. To integrate the 6 behaviors, we give the highest priority to the 5 local behaviors of the first experiment, and the lowest priority to the reach goal behavior. It means that when one the 5 highest priority behaviors gives a non null motor action, the lowest priority "reach goal behavior" is automatically disactived.
The translation speed is computed as earlier with the rotation speed of the level which controls the mobile robot.

#### 5.2.3 Results

We tested our fuzzy controller to control our mobile robot going to a predefined goal in a hall. At the beginning, the mobile robot is positioned in a known position and orientation but different for each run. It is asked to reach a predefined goal in this hall. At the beginning, when there are no obstacles in the local environment of the mobile robot, it turns to face its goal, and goes straight until it detects an obstacle. At this moment, it avoids the obstacle and turns to face its goal. We give a trace of one run in figure 9. The mobile robot goes to its goal avoiding obstacles, and people in the hall.

## 6 Conclusion

In this article, we propose a new approach to design fuzzy controller in two levels. This approach permits the design of fuzzy controller in a simpler way. In the first level, local areas and local rules are defined to design local behaviors. In the second level, a mechanism of integration of these local behaviors provides a global motor action to the mobile robot. Moreover, local behaviors can be designed and tested independently and the proposed mechanisms of integration gives a global coherence to the mobile robot.
Our approach has been extensively tested with the two scenarios, and our mobile robot rarely touches obstacles or people. With our front local behavior, the mobile robot will never do excessive rotation to avoid front obstacles, so it will never do a U-turn.
But, our mobile robot is not at the moment able to exit from a local minimum. Moreover, we do not use the ultrasonic/infrared sensor at the rear of the robot, which can be useful in backing up.

## References

[1] R.C Arkin. Motor schema based navigation for a mobile robot : an approach to programming by behavior. In *proceedings of the 1987 IEEE International Conference on Robotics and Automation*, pages 264–271, 1987.

[2] R.A Brooks. A robust layered control system for a mobile robot. *In IEEE Journal of Robotics and Automation*, 2:14–23, 1986.

[3] P Garnier. *Contrôle d'exécution réactif de mouvements de véhicules en environnement dynamique structuré*. PhD thesis, Institut National Polytechnique de Grenoble, 1994. In French.

[4] C.C Lee. Fuzzy logic in control systems : Fuzzy logic controller. *IEEE Transactions on Systems, Man and Cybernetics*, 20(2):491–418, March 1990.

[5] W Li. Perception-action behavior control of a mobile robot in uncertain environments using fuzzy logic. In *proceedings of the 1994 IEEE International Conference on Intelligent Robots and Systems*, pages 439–446, 1994.

[6] P Reignier. Fuzzy logic techniques for mobile robot obstacle avoidance. *International Journal on Robotics and Autonomous System*, 12:143–153, 1994.

[7] E.H Saffioti, A Ruspini and K Konolige. Robust control of a mobile robot using fuzzy logic. In *proceedings of the First European Congress on Fuzzy and Intelligent Technologies*, 93.

[8] K Song and J Tai. Fuzzy navigation of a mobile robot. In *proceedings of the 1992 IEEE International Conference on Intelligent Robots and Systems*, pages 621–627, 1992.

[9] Nomadic Technologies. Nomad 200 usert́s manual. Technical report, 1996.

[10] L.A Zadeh. Fuzzy sets. *Information and Control*, 8:338–365, 1965.