BULETINUL Universităţii Petrol – Gaze din Ploieşti	Vol. LXV No. 2/2013	41 - 48	Seria Tehnică

An Overview on Fuzzy Control for Autonomous Systems

Cristina Popescu

Universitatea Petrol-Gaze din Ploieşti, Bd. Bucureşti 39, Ploieşti
e-mail: cristinap@upg-ploiesti.ro

Abstract

Adaptive fuzzy systems and neural networks are aiming at the same objective, namely, emulating the human brain operations. One might consider that the artificial neural networks try to emulate the hardware of the human brain, while trying to perform systems adaptive fuzzy software emulations of the human brain. Thus, a multi-layer perceptron neural network is simulating the basic structure of the brain, while the fuzzy systems attempt to simulate the human brain in terms of making high-level input-output systems. This paper presents an overview on fuzzy control for a mobile robot.

Key words: *autonomous systems, fuzzy control, mobile robot.*

Introduction

Reasons for increased interest in the study and application of fuzzy control are:

- ease of synthesis and implementation of multiple expert knowledge;
- the possibility of coordinating several objectives;
- achieving simplicity, providing flexibility and easy adaptation process operating conditions;
- the sturdiness.

Actions produce fuzzy control systems using a set of fuzzy rules based on fuzzy logic, which are different from conventional logic predicates. In conventional logic, the statements are *false* or *true* about the world and there is nothing between them. Fuzzy logic provides a different perspective, allowing variables to take values determined by the degree to which they belong to a particular fuzzy set (defined by a membership function). These fuzzy variables are expressed as linguistic variables that do not have a rigid meaning (e.g. quickly, slowly, far, close, etc).

Structure of a Fuzzy Controller

Designing a fuzzy controller starts with choosing linguistic variables, such as the process state variables, input and output variables. The next step is choosing a set of linguistic rules and fuzzy reasoning process type. Deffuzification strategy is determined when the significance of fuzzy rule set, obtained by inference, is a fuzzy set and this generates an output value.

The block diagram of a fuzzy controller is shown in Figure 1 and consists of the following four modules [1]:

- fuzzification interface performing the transformation of numeric input fuzzy sets;
- knowledge base that provides the information necessary to the inference engine and interface deffuzification to function correctly;
- decision-making unit or inference engine that determines the significance of the set of linguistic rules;
- interface that transforms union deffuzification fuzzy sets (individual contribution of each rule in the rule base) in a numerical output.

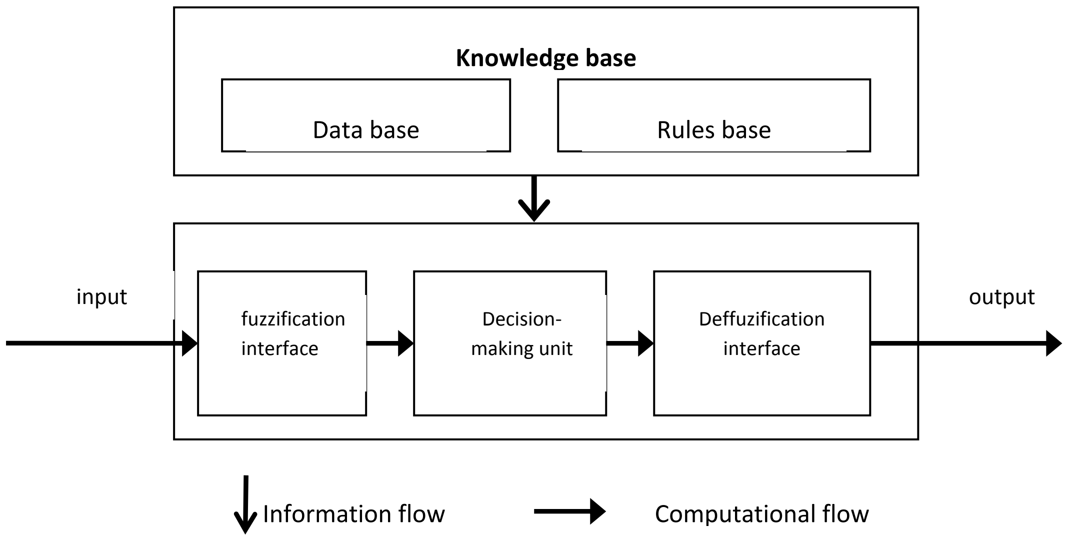

Fig. 1. Block diagram of a fuzzy controller

a. Fuzzification

The fuzzification module accomplishes two tasks:

- normalization of entry;
- transform the actual values of the process state into fuzzy sets, in order to make them compatible with linguistic rules to be applied in the fuzzy inference engine.

b. Knowledge base

The knowledge base is formed by a database and a rule base. The database contains the normalization factors and parameters that determine the significance of fuzzy sets of linguistic values of linguistic variables. Rule base management strategy is a form of linguistic rules *if-then* type. Each rule is an interpretation of an act of an expert. The rule is a causal description of the state of a system which effect is the action to be performed by the system operator.

The database design consists of:

- selection of scaling factors. These factors play a role similar to that of the gain of a conventional regulator;
- choice of linguistic variables, fuzzy crucial for leadership. Usually, linguistic variables represents states, error status and error status derivatives;

- defining sets of terms for linguistic variables. A set of terms consists of a finite number of linguistic values that a linguistic variable can take. The size of this set determines the granularity of control action and the number of rules;
- choice of membership functions. This includes the choice of forms, distribution and parameters of membership functions for each linguistic variable.

The design rule base consists of choosing the source and content of the set of rules. In literature there are four methods to obtain the linguistic rules:

- modeling control knowledge engineer. Most general knowledge involves an introspective verbal human experience;
- modeling human operator actions and experience. This method includes a query operators well-organized experiment using a questionnaire;
- driven fuzzy modeling system. The process is a fuzzy linguistic description of its dynamic characteristics. Based on this model there are generated fuzzy driven rules;
- method of learning. Fuzzy controller is based on two rules: one is general and the other, of supervisor type, has the role to create and modify the general rules on the desired performance of the system.

c. Unit decision

The task of this module is to determine the significance of the set of linguistic rule, i.e. the total value of the control output variable based on the individual contribution of each rule. Output fuzzyfication interface is implemented according to the casual of each rule and sets a starting point for each rule. Based on the linguistic value of the variable, it changes the control output from the rule, that is the effect, and thus one obtains a fuzzy set limited or reduced (down-converted**).** The significance of this set of rules is determined as the union of all control values for limited rules or scalar enabled. The resulting set is the total value of the value of the output fuzzy controller.

The design features of the inference engine are:

- choosing the operator that determines the significance of the meaning of a single rule;
- selecting the mix of fuzzy reasoning rules. The methods used are Min-Max, Takagi and Sugeno.

d. Deffuzification

The deffuzification module converts the modified fuzzy sets in a real output value, followed by the denormalisation of this value.

Suppose a controller that has two inputs *x*, *y* and an output *z* (fig. 2).

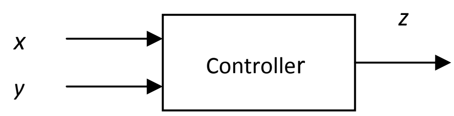

Fig. 2. Input and output variables of the controller.

The defined set of linguistic rules is:

R1: If *x* is A1 and *y* is B1, then *z* is C1 or

R2: If *x* is A2 and y is B2, then *z* is C2 or

…

Rn: If x is An and y is Bn, then z is Cn,

where A_i, B_i and C_i ($i=1,...,n$) are linguistic values defined in the universes U, V and W, for x, y or z.

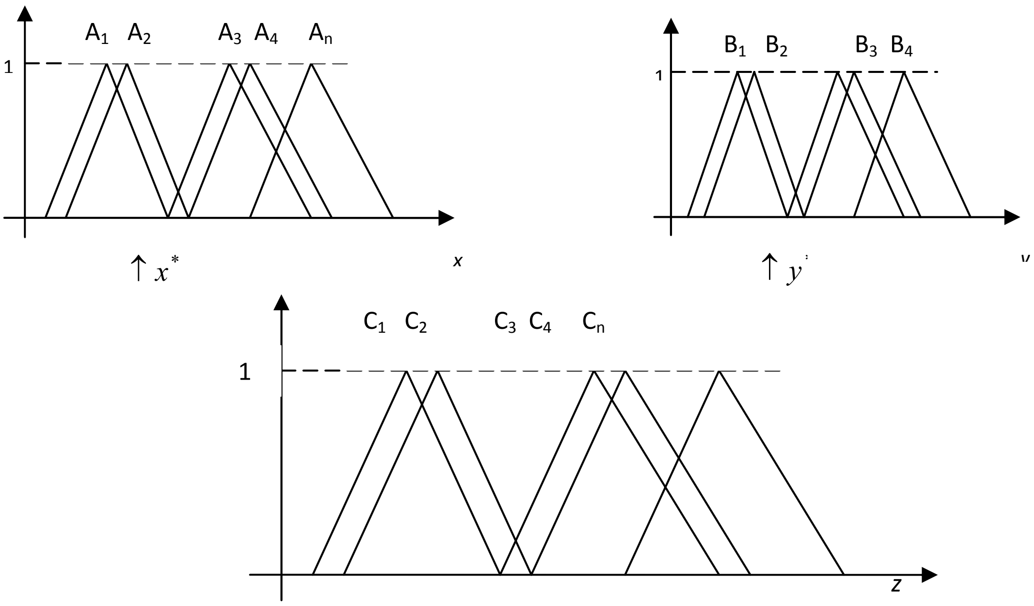

Fig. 3. Membership functions for x, y, z

It is obvious that not all rules can be activated at the same time. It is assumed that the given value of $x=x^*$ belongs to the fuzzy sets A_1 and A_2, with grades different from zero. Also, it is assumed that the degree of affiliation of the data value $y=y^*$ for the fuzzy sets B_1 and B_2 are different from zero. This means that they are activated only following two rules:

R1: If x is A1 and y is B1, then z is C1

R2: If x is A2 and y is B2, then z is C2.

The significance of the other rules is zero.

There are different types of fuzzy reasoning. Among these, the best known are Max-Dot, Min-Max, Tsukamoto, Takagi and Sugeno. If using Max-Dot reasoning method, the linear modulation produces C_1S and C_2S sets. The significance of the set of rules is obtained by **meeting of the two sets**. With the Min-Max method, modeling limiting (threshold level) produces two sets modified fuzzy C_1C and C_2C. The significance of the rules set will be the meeting of two sets. According to the Tsukamoto method, output membership function must be monotonically increasing. The significance of the set of rules is the weighted average of each actual output:

$$z = \frac{w_1 z_1 + w_2 z_2}{w_1 + w_2} \tag{1}$$

Takagi and Sugeno method assumes that the rule in effect is a function of the input variables. It is assumed that it is defined as a linear function. The general form of Takagi and Sugeno language rule is [6]:

RTS: If x is A and y is B, then z is $f(x,y)$

It is assumed that the two active rules are the following:

RTS1: If x is A_1 and y is B_1, then z is $f_1(x,y) = a_1+c_1+b_1y$

RTS2: If x is A_2 and y is B_2, then z is $f_2(x,y) = a_2+c_2+b_2y$.

The significance of the set of rules is a weighted average as shown in equation (1).

Example

The purpose of this example is driving a mobile robot (fig. 4) using a simple algorithm.

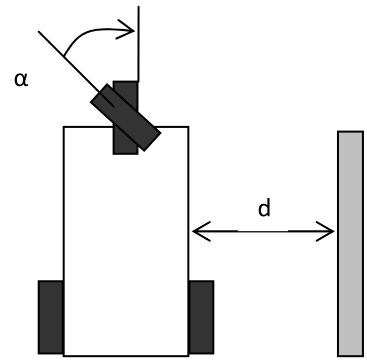

Fig. 4. Mobile robot
d – distance between the robot and the obstacle; *α* – angle of the back wheel robot

The robot is equipped with a single sensor that informs about the distance *d* between the robot and an obstacle to the right of it [2].

The first step in designing the controller is to define linguistic variables (inputs and outputs). In this example, we consider a controller with one input and one output. The input is the distance between the robot and the obstacle, *d*, and the output is the angle of the back wheel robot, α.

One has established the following set of linguistic rules [4]:

R1: If the distance between the robot and the obstacle is less than 10 cm, then turn -10 degrees;

or

R2: If the distance between the robot and the obstacle is less than 10 cm, then turn +10 degrees;

or

R3: If the distance between the robot and the obstacle is 10 cm, then keep the direction.

These rules define the behavior of a "following a wall"- type. Universe definition for these two linguistic variables is shown in Figure 5.

For the fuzzy controller language there were defined the following rules:

RF1: If the distance between the robot and the obstacle is less than 10 cm, then back to the left (α is positive);

or

RF2: If the distance between the robot and the obstacle is less than 10 cm, then turn to the right (α is negative);

or

RF3: If the distance between the robot and the obstacle is approximately 10 cm, then keep the direction (α is approximately 0).

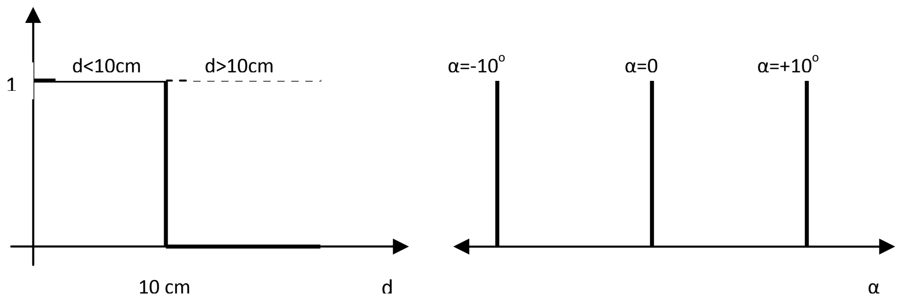

Fig. 5. Universe definition for distance and turn angle linguistic variables.

With these rules, one defined three linguistic values for the variable “distance”: more than 10 cm, about 10 cm (optimal distance), and less than 10 cm. For the variable “angle back”, one chose three linguistic values: negative for right turn, zero forward, positive for left turn.

Universe definition for distance and angle of turn is shown in Figure 6.

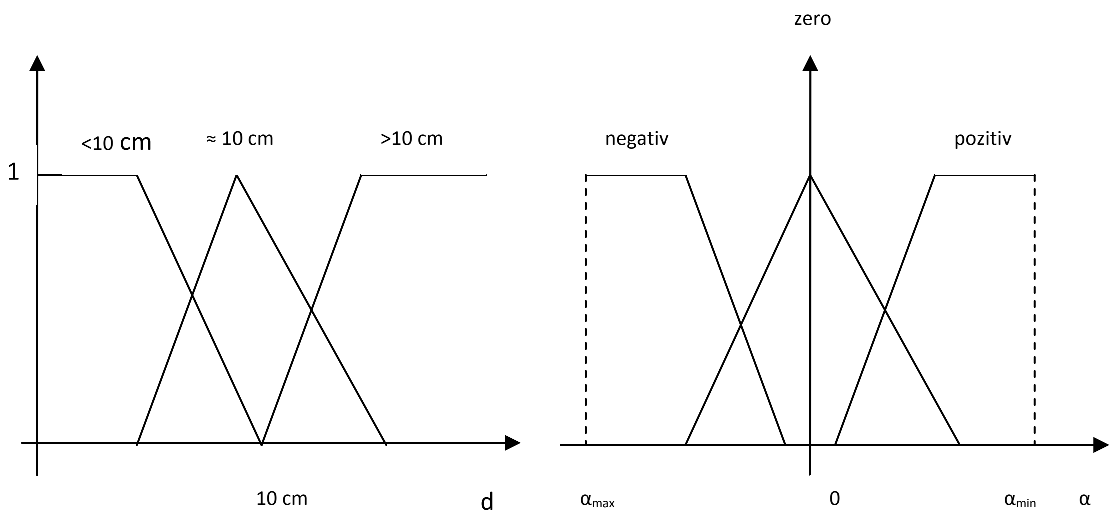

Fig. 6. Universe definition for distance and turn angle

Command value can be calculated by applying one of the inference methods. Assume that the sensor reports that the distance is 5 cm. This value belongs to the set of fuzzy distance “ less than 10 cm” with the degree of membership 0.7 and one can assume that the weight is 0.7 for RF1 rule. Meanwhile, with the weight of 0.3, it can be considered that this distance belongs to the fuzzy set “about 10 cm”, as shown in Figure 7.

The first part of the RF1 rule is satisfied with the degree 0.7. This value represents the weight of this rule. This means that the fuzzy set “Negative” must be modulated by 0.7.

For the second rule, the process is the same. The distance is “about 10 cm”, but only to the degree of truth 0.3. The robot must keep direction with this degree of truth. The first part of the third rule is not satisfied, because the degree of truth is 0. It can be concluded that the robot should turn left with weight 0.7 and retain executives with weight 0.3. Figure 8 illustrates the modulation by cutting and scaling. The real value of the order can be found by applying on a deffuzification methods. Centroid method and mediated maxima method are presented.

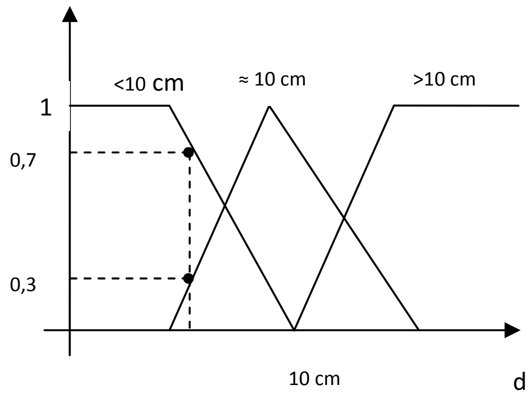

Fig. 7. Fuzzification.

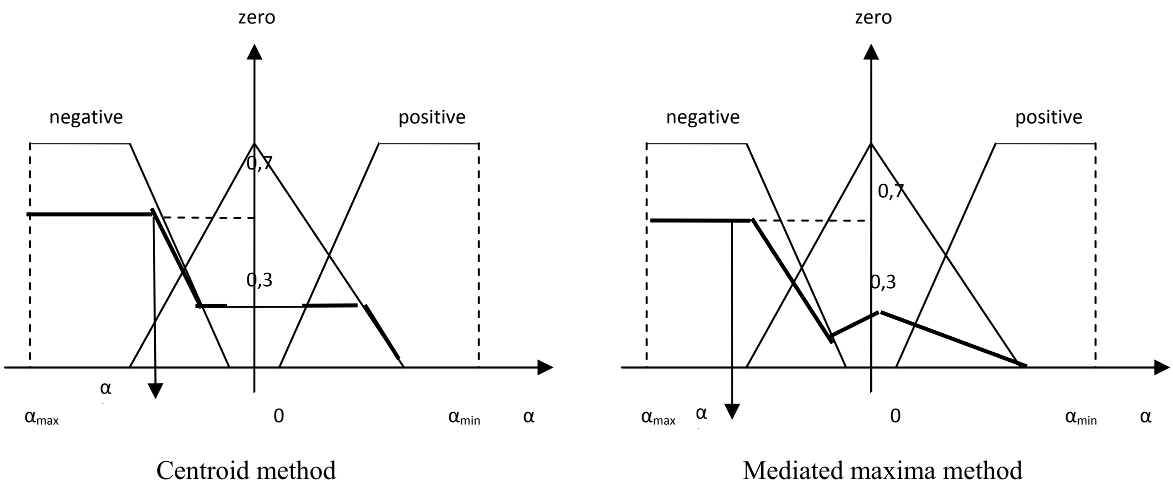

Fig. 8. Fuzzy inference and deffuzification.

This example leads to the conclusion that the level of complexity for driving the robot increases the number of inputs and outputs of the controller, as well as the number of possible rules [3].

Conclusion

From the presented studies, it was concluded that the design and construction of a fuzzy controller is a complex task, as many parameters are required. It was considered that the designer has to choose parameters for the regulator by establishing linguistic rules, and parameters for linguistic variables (for membership functions), etc.

It is obvious that methods of inference Min-Max and Max-Dot, combined with centroid method for deffuzification or maxima mediated method, do not facilitate the determination of the functional dependence between the inputs and outputs of a fuzzy controller. Further researches would prove that the Takagi and Sugeno controller is much simpler from the point of view of mathematical modeling. Also, it is at great rates due to the fact that the time of computation is the significantly reduced as compared to time of the calculation of a controller using the method the center of gravity for deffuzification.

This paper also offers a number of possible directions of future work. First, the use of supervised learning techniques can be studied. These techniques could avoid learning from a supervisor and completely automate the design procedure. Second, more complicated tasks for mobile robot should be investigated.

References

1. Godjevac, J. – *Neuro-Fuzzy Controllers Design and Application*, Presses Polytechniques et Universitaires Romandes, Laussane, 1997.
2. Bucur, G. – *Senzori, traductoare, măsurare*, Editura UPG Ploiesti, 2011.
3. Popescu, C., Cangea, O. – Implementation of neuro-fuzzy controller on Khepera III mobile robot, *Petroleum-Gas University of Bulletin, Technical Series*, 2008.
4. Popescu, C. – *Neuro-Fuzzy Control for Mobile Robot*, Ph.D. Thesis, Ploiesti, 2008.
5. Steele, N., Godjevac, J. – Adaptive Radial Basis Function Neural Networks and Fuzzy Systems, *Conference on Computational Engineering in Systems Applications, CESA'96*, Lille, France, pp.143-148.
6. Sugeno, M., Murakami, K. – An Experimental Study on Fuzzy Parking Control Using a Model Car, In: *Industrial Applications of Fuzzy Control*, Sugeno, M. (Ed.), North-Holland, 1985, pp.125-138.

Aspecte generale despre conducerea fuzzy a sistemelor autonome

Rezumat

În prima parte a acestei lucrări au fost prezentate motivele care conduc la utilizarea conducerii fuzzy pentru sistemele autonome, precum si structura generala a unui regulator fuzzy, cu prezentarea detaliata a blocurilor componente. In partea a doua, a fost realizat un studiu de caz pentru conducerea fuzzy a unui robot simplu, echipat cu un senzor care furnizează informații asupra distanței dintre robot şi obstacolele întâlnite. Cu un set minim de reguli este implementat comportamentul de urmărire a unui perete.