

Universal Approximation of Mamdani Fuzzy Controllers and Fuzzy Logical Controllers

Bo Yuan* and George J. Klir⁺

*NASA Center for Autonomous Control Engineering, Department of Engineering, New Mexico Highlands University, Las Vegas, New Mexico 87701

⁺Center for Intelligent Systems, Department of Systems Science and Industrial Engineering, Binghamton University, Binghamton, New York 13902

Abstract

In this paper, we first distinguish two types of fuzzy controllers, Mamdani fuzzy controllers and fuzzy logical controllers. Mamdani fuzzy controllers are based on the idea of interpolation while fuzzy logical controllers are based on fuzzy logic in its narrow sense, i.e., fuzzy propositional logic. The two types of fuzzy controllers treat IF-THEN rules differently. In Mamdani fuzzy controllers, rules are treated disjunctively. In fuzzy logic controllers, rules are treated conjunctively. Finally, we provide a unified proof of the property of universal approximation for both types of fuzzy controllers.

1 Introduction

The study of universal approximation of a fuzzy controller was first initiated by Kosko [7], Wang [8] and Wang and Mendel [10]. It was an important contribution to fuzzy control theory, since it provided a theoretical foundation for applying fuzzy controllers. It was shown that for a given continuous function defined on a compact domain, one always can design a fuzzy controller to approximate the function to any given precision, Kosko proved the result for his adaptive fuzzy system in [7], Wang and Mendel provided a proof for a special case of Mamdani fuzzy controllers in [8] and [10]. Buckley proved the same result for Sugeno type fuzzy controllers in [2]. Ying presented a proof for a general Mamdani fuzzy controller in [13]. Castro provided another proof for a general Mamdani fuzzy controller in [3]. Klawonn and Novák proved the same result for fuzzy controllers based on fuzzy logical implications in [4].

Fuzzy controllers based on the idea of interpolation and those based on the idea of logical inference are often not distinguished in the literature. More specifically, some t-norms, such as *min* and *product*, are often treated as fuzzy implications. The differences between the two types of fuzzy controllers are clearly stated in [4] and also can be found in [5]. In this paper, fuzzy controllers based on t-norms are called *Mamdani fuzzy controllers*, while fuzzy controllers based on fuzzy implications are called *fuzzy logical controllers*.

In Section 2, we first review the structure of a fuzzy controller. The differences between Mamdani fuzzy controllers and fuzzy logical controllers are examined in Section 3. In Section 4, we present a unified proof of the property of universal approximation for both types of fuzzy controllers.

2 Fuzzy Controllers: A Brief Overview

To build a fuzzy logic controller, one needs to follow the following four steps:

Step 1. Identifying state variables and control variables of a system to be controlled.

In this step, one has to determine relevant state and control variables, as well as the range of each of these variables. For instance, in the simple inverted pendulum example, the state variables are the angle of the pole, θ , and the rate of change of the angle, $\dot{\theta}$. The control variable is the force, f , applied to the cart carrying the pole. The range θ may, for example, be $(-\frac{\pi}{2}, \frac{\pi}{2})$.

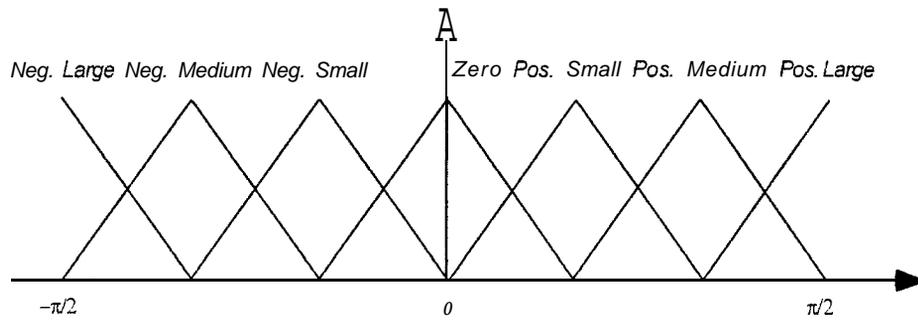


Figure 1: A fuzzy partition of the interval $[-\frac{\pi}{2}, \frac{\pi}{2}]$

Step 2. Dividing the range of each variable into several level.

The purpose of this step is to make it easier for domain experts to summarize their knowledge. The levels, which correspond to several linguistic terms, are used to describe the states of the system and control strategies. Usually, it is done by generating a fuzzy partition of the domain using fuzzy sets with triangular membership functions or, more generally, trapezoidal membership functions. Fuzzy sets with Gaussian membership functions, splines are also often used in many applications of neural-fuzzy type control [1] and [9]. For example, let us use fuzzy sets with triangular membership functions to form a fuzzy partition of the range of variable θ . We divide the interval $(-\frac{\pi}{2}, \frac{\pi}{2})$ into seven levels, which are labeled linguistically as negative large, negative medium, negative small, approximately zero, positive small, positive medium, positive large, respectively (Fig. 1).

Step 3. Specifying IF-THEN inference rules for the controller.

These IF-THEN rules express domain experts' knowledge regarding the control task. Each rule describes a control action that should be taken if the system is in one state. The general form of a rule in a rule base is

$$\text{If } x \text{ is } A \text{ then } y \text{ is } B$$

Intuitively, we should have IF-THEN rules that cover all possible states of the system that are described by the linguistic terms generated in Step 2. Some researchers argue that fuzzy controllers with less IF-THEN rules have sufficiently good performance [12]. In these cases, however, membership functions have to be carefully designed so that every state of the system is covered by some rules in the rule base.

Step 4. Selecting a defuzzification method.

A defuzzification method is a mapping $d: \mathcal{F}(\mathbb{R}) \rightarrow \mathbb{R}$, where $\mathcal{F}(\mathbb{R})$ denotes the fuzzy power set of \mathbb{R} . It maps a fuzzy set to a real number. There are many defuzzification methods in the literature. All defuzzification functions must satisfy the property

$$d(x) = x$$

for any $x \in \mathbb{R}$. One commonly used defuzzification method is called the *center of gravity method*. It takes the center of gravity of a fuzzy set as its defuzzification value. In this case, the fuzzy set is considered as the area surrounded by its membership function and the x-axis. This defuzzification method assumes that the membership function of the fuzzy set is integrable, which is true in most cases. Suppose $A(x)$ is the membership function of a fuzzy set A . Then the defuzzification value of fuzzy set A is

$$d(A) = \frac{\int xA(x)dx}{\int A(x)dx} \quad (1)$$

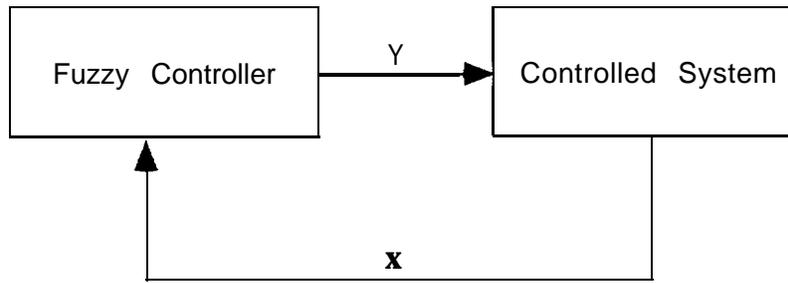


Figure 2: A general diagram of a fuzzy control system

for the continuous case, or

$$d(A) = \frac{\sum_{i=1}^n \mu_i A(x_i)}{\sum_{i=1}^n A(x_i)} \quad (2)$$

for the discrete case.

After one completes Step 1-4, he/she is ready to apply the fuzzy controller. A fuzzy controller generally has the structure shown in Fig. 2. Given the current state of a controlled system or plant, the fuzzy controller generates a value of the control variable. Then a control action is applied to the system, and the system, in turn, changes its state according to this control action. Therefore, a fuzzy controller can be considered as a function with the state of the system represented by its independent variables and the control state by its dependent variables. Suppose $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$ denotes the n -dimensional state vector of the system, y denotes the control variable, then a fuzzy controller is a function

$$\begin{aligned} FS &: D \rightarrow \mathbb{R} \\ y &= FS(\mathbf{x}) \end{aligned} \quad (3)$$

where \mathbb{R} is the real number set; D is a compact subset of n -dimensional space \mathbb{R}^n , which is specified in Step 1. Here, only one control variable is considered for the simplicity of discussion. For the case of several control variables, it can be easily decomposed into several fuzzy controllers with one control variable.

The function FS is determined by the following procedure.

2.1 Fuzzification

At this stage, a fuzzy set is generated on the basis of current state vector $\mathbf{x}^t = (x_1^t, x_2^t, \dots, x_n^t)^T$. There are many ways to generate this fuzzy set. One way is through generating symmetric triangular fuzzy numbers for all state variables $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$

$$A(\mathbf{x}) = \bigwedge_{i=1}^n A_i(x_i) \quad (4)$$

where \wedge denotes a t-norm [5] and A_i is determined by

$$A_i(x) = \begin{cases} 1 - \frac{|x - x_i^t|}{d_i} & \text{if } x \in [x_i^t - d_i, x_i^t + d_i] \\ \mathbf{0} & \text{if otherwise;} \end{cases} \quad (5)$$

d_i is a constant which is monotonically related to the fuzziness of the fuzzification for each $i \in \{1, 2, \dots, n\}$. The larger d_i , the fuzzier the resulting fuzzy set A_i . Constants d_i are specified by the designer of the controller. In most recent applications, d_i is assumed to be zero for any $i \in \{1, 2, \dots, n\}$. That is, this fuzzification step is ignored. [3] and [10].

2.2 Inference

At this stage, Zadeh's compositional rule of inference [6] and [11], is applied to calculate a fuzzy value of the control variable based on the fuzzy set obtained by fuzzification and the rule base specified in Step 3. The resulting fuzzy set B is calculated by

$$B(y) = \bigvee_{x \in D} i(A(x), R(x, y)), \quad (6)$$

where \bigvee is a t -conorm and i is a t -norm [5]. R is a fuzzy relation defined on $D \times \mathfrak{R}$, which is determined by

$$R(x, y) = \diamond_{j=1}^m R_j(x, y), \quad (7)$$

where m is the number of rules in the rule base; \diamond is either a t -norm or a t -conorm depending on the way R_j is calculated. R_j represents the j th rule in the rule base.

$$R_j(x, y) = f(A_j(x), B_j(y)), \quad (8)$$

where $f : [0, 1]^2 \rightarrow [0, 1]$ is a binary function that is either a t -norm or a fuzzy implication [5]. When f is a t -norm, the fuzzy controller functions as interpolation; \diamond must be a t -conorm. In this case, we call the fuzzy controller, FS , a Mamdani fuzzy controller. When f is a fuzzy implication, the fuzzy controller is based on logical inference, and \diamond must be a t -norm. In this case, we call the fuzzy controller, FS , a fuzzy logical controller [4]. Here a fuzzy implication is required to satisfy at least the following two conditions

$$\begin{aligned} f(0, x) &= 1 \\ f(1, x) &= x \end{aligned} \quad (9)$$

for any $x \in [0, 1]$. Both R-implications and S-implications satisfy these conditions. For a detail discussion of fuzzy implications see [5].

2.3 Defuzzification

The fuzzy set B obtained by the inference described in Sec. 2.2 is defuzzified into a real number. The output of the fuzzy controller is the defuzzified value of that is

$$FS(x) = d(B).$$

3 Differences Between Mamdani Fuzzy Controllers and Fuzzy Logical Controllers

Let us consider a Mamdani fuzzy controller $FS_1 : D \rightarrow \mathfrak{R}$, with $\bigvee = \diamond = \sup$, $i = f = \text{at-norm}$. According to Eq. (6)-(8), we have

$$\begin{aligned} B(y) &= \sup_{x \in D} i(A(x), R(x, y)) \\ &= \sup_{x \in D} i(A(x), \max_{j=1}^m i(A_j(x), B_j(y))) \\ &= \sup_{x \in D} \max_{j=1}^m i(A(x), i(A_j(x), B_j(y))) \\ &= \sup_{x \in D} \max_{j=1}^m i(i(A(x), A_j(x)), B_j(y)) \\ &= \max_{j=1}^m i(\sup_{x \in D} i(A(x), A_j(x)), B_j(y)) \end{aligned}$$

Therefore, for a Mamdani fuzzy controller, rules in the rule base are considered disjunctively. One can apply each rule in a rule base to generate a result, then take the maximum of all results as the resulting fuzzy set.

Now, let us consider a fuzzy logical controller $FS_2 : D \rightarrow \mathbb{R}$, with $\vee = \text{sup}$, $\circ = \text{min}$, $i =$ a t -norm and $f =$ a fuzzy implication. According to Eq. (6)-(8), we have

$$\begin{aligned} B(y) &= \sup_{x \in D} i(A(x), R(x, y)) \\ &= \sup_{x \in D} i(A(x), \min_{j=1}^m f(A_j(x), B_j(y))) \end{aligned} \quad (10)$$

Suppose A is a crisp point x_0 in \mathbb{R}^n , that is,

$$A(x) = \begin{cases} 1 & \text{if } x = x_0 \\ 0 & \text{if } x \neq x_0 \end{cases}$$

Then, Eq. (10) becomes

$$B(y) = \min_{j=1}^m f(A_j(x_0), B_j(y)) \quad (11)$$

Therefore, we can see that a fuzzy logical controller treats rules in a rule base conjunctively. In the case of no fuzzification, one applies each rule in the rule base and takes then the minimum of all obtained results as the final resulting fuzzy set.

In conclusion, we can see that one major difference between a Mamdani fuzzy controller and a fuzzy logical controller is that the former treats rules in a rule base disjunctively while the latter treats them conjunctively. For a more detail discussion of the differences between Mamdani fuzzy controllers and fuzzy logical controllers, see Ref. [4].

4 Universal Approximation of Mamdani and Fuzzy Logical Controllers

Wang and Mendel in [10] have shown that Mamdani fuzzy controllers with $f = \text{product}$ are universal approximators. Ying [13] and Castro [3] have proven this for a general Mamdani fuzzy controller, using different approaches. Klawonn and Novák in [4] proved that fuzzy logical controllers are universal approximators. Here, we show a unified proof of the same result for both types of fuzzy controllers.

Theorem 1 *Let $g : U \rightarrow \mathbb{R}$ be a continuous function from a compact subset U of \mathbb{R}^n to \mathbb{R} . Then for any $\varepsilon > 0$, we always can find a fuzzy controller FS such that*

$$|FS(x) - g(x)| < \varepsilon \quad (12)$$

for any $x \in U$.

Proof. Since g is continuous, for any $\varepsilon > 0$, and any $u \in U$, there exists $d(u, \varepsilon) > 0$, such that for any $x \in B(u, d(u, \varepsilon)) = \{x \in U \mid |x - u| < d(u, \varepsilon)\}$,

$$|g(x) - g(u)| < \varepsilon. \quad (13)$$

Since $U \subseteq \bigcup_{u \in U} B(u, d(u, \varepsilon))$ and U is a compact set, there exists a finite number of elements $\{u_1, u_2, \dots, u_m\}$ such that

$$U \subseteq \bigcup_{i=1}^m B(u_i, d(u_i, \varepsilon)) \quad (14)$$

Now we generate a finite disjoint cover of U based on Eq. (14). Let $A_1 = B(u_1, d(u_1, \varepsilon))$,

$$A_i = B(u_i, d(u_i, \varepsilon)) \setminus \bigcup_{j=1}^{i-1} A_j \quad (15)$$

for $i = 2, 3, \dots, m$. It is easy to see that

$$A_i \cap A_j = \emptyset$$

for $i \neq j$, and

$$U \subseteq \bigcup_{i=1}^m A_i \quad (16)$$

Now, we build a fuzzy controller without the fuzzification step and based on the following inference rules in the rule base

$$\text{If } x \text{ is } A_1 \text{ then } y \text{ is } B_1 = g(u_1)$$

.....

$$\text{If } x \text{ is } A_m \text{ then } y \text{ is } B_m = g(u_m)$$

For any $x \in U$, it follows from Eq. (16) that there exists a unique $i_0 \in \{1, 2, \dots, m\}$ such that $x \in A_{i_0}$ and $x \notin A_i$ for $i \neq i_0$. That is

$$A_{i_0}(x) = 1 \text{ and } A_i(x) = 0 \text{ for } i \neq i_0.$$

According to Eq. (6)-(8), the resulting fuzzy set is

$$\begin{aligned} B(y) &= \bigvee_{x' \in D} i(A(x'), R(x', y)) \\ &= R(x, y) \\ &= \diamond_{i=1}^m R_i(x, y) \\ &= \diamond_{i=1}^m f(A_i(x), B_i(y)) \end{aligned}$$

for any $y \in \mathbb{R}$.

When f is a t-norm and \diamond is a t-conorm,

$$\begin{aligned} B(y) &= f(1, B_{i_0}(y)) \\ &= B_{i_0}(y) \end{aligned}$$

Therefore

$$FS(x) = d(B) = d(B_{i_0}) = g(u_{i_0}).$$

for any type of defuzzification function d .

When f is a fuzzy implication and \diamond is a t-norm

$$\begin{aligned} B(y) &= \diamond(\diamond_{i \neq i_0} f(A_i(x), B_i(y)), f(A_{i_0}(x), B_{i_0}(y))) \\ &= \diamond(\diamond_{i \neq i_0} f(0, B_i(y)), f(A_{i_0}(x), B_{i_0}(y))) \\ &= \diamond(1, f(A_{i_0}(x), B_{i_0}(y))) \\ &= f(A_{i_0}(x), B_{i_0}(y)) \\ &= f(1, B_{i_0}(y)) \\ &= B_{i_0}(y) \end{aligned}$$

Therefore, again we have

$$FS(x) = d(B) = d(B_{i_0}) = g(u_{i_0}).$$

for any type of defuzzification function d .

Finally, according to Eq. (13), we have

$$|FS(x) - g(x)| = |g(u_{i_0}) - g(x)| < \varepsilon.$$

This completes the proof.

5 Conclusions

In this paper, we argue that two different types of fuzzy controllers, Mamdani fuzzy controllers and fuzzy logical controllers, should be distinguished. We also present a unified proof of the property of universal approximation for these two types of fuzzy controllers.

6 Acknowledgment

This research has been sponsored by NASA Center for Autonomous Control Engineering. The first author would like to thank Dr. Djuro G. Zrlic for his support of this research..

References

- [1] M. Brown and C. Harris. *Neurofuzzy adaptive modelling and control*. Prentice Hall, New York, 1994.
- [2] J. J. Buckley. Sugeno type controllers are universal controllers. *Fuzzy Sets and Systems*, 53(3):299–304, 1993.
- [3] J. Castro. Fuzzy logic controllers are universal approximators. *IEEE Trans. on Systems, Man and Cybernetics*, 25(4):629–635, 1995.
- [4] F. Klawonn and V. Novak. The relation between inference and interpolation in the framework of fuzzy systems. *Fuzzy Sets and Systems*, 81(3):331–354, 1996.
- [5] G. J. Klir and B. Yuan. *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice Hall PTR, Upper Saddle River, NJ, 1995.
- [6] G. J. Klir and B. Yuan, editors. *Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers by Lotfi A. Zadeh*. World Scientific, Singapore, 1996.
- [7] B. Kosko. Fuzzy systems as universal approximators. *Proc. of First IEEE Intern. Conference on Fuzzy Systems*, H: 1153–1 162, 1992.
- [8] L. X. Wang. Fuzzy systems are universal approximators. *Proc. of First IEEE Intern. Conference on Fuzzy Systems*, II: 1163–1170, 1992.
- [9] L. X. Wang. *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*. Prentice Hall, Englewood Cliffs, NJ, 1994.
- [10] L. X. Wang and J. M. Mendel. Fuzzy basis functions, universal approximation, and orthogonal least-squares learning. *IEEE Trans. on Neural Networks*, 3(5):807–814, 1992.
- [11] R. R. Yager, S. Ovchinnikov, R. M. Tong, and H. T. Nguyen, editors. *Fuzzy Sets and Applications: Selected Papers by L. A. Zadeh*. Wiley - Interscience, New York, 1987.
- [12] T. Yamakawa. A fuzzy inference engine in nonlinear analog mode and its application to a fuzzy logic control. *IEEE Trans. on Neural Networks*, 4(3):496–522, 1993.
- [13] H. Ying. Sufficient conditions on general fuzzy systems as function approximators. *Automatica*. 30(3):521–525, 1994.