Anas Fattouh, FadiFouz / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 2, Issue 3, May-Jun 2012, pp.2660-2665

A Two-Stage Representation of Fuzzy Systems

Anas Fattouh*, FadiFouz*
*(Department of Computer Science, King Abdulaziz University, Saudi Arabia)

ABSTRACT

Computing the output of fuzzy systems usually passes through three stages i.e. fuzzification, inference, and defuzzification. The computation of the output requires too many operations and considerable time. This apparently discourages the use of fuzzy systems in time-critical applications. In this paper, a new two-stage representation of fuzzy systems, which reduces the operations needed to compute the output of fuzzy systemis proposed. In this representation the inference stage is embedded in the defuzzification stage. This permits computation offline, which in effect reduces the real-time computation time and therefore permits the use of fuzzy systems in real-time applications. To evaluate the performance of the proposed representation, it was used to represent the truck backer-upper control system and compared with conventional representation of fuzzy systems. The simulation results support the two-stage representation preference.

Keywords–**Defuzzification, fuzzification, fuzzy systems, inference, real-time systems.**

I. INTRODUCTION

Fuzzy systems are systems based on fuzzy set theory developed by Zadeh [1]. Since the first implementation of fuzzy systems by Mamdani [2], fuzzy systems have successfully used in many applications such as control systems, data mining, expert systems and pattern recognition [3,4,5].

The relationship between the components of fuzzy systems involvesfuzzy logical operations, i.e.union, intersection, and complement of fuzzy sets. These fuzzy set operations require many multiplication and division operations and need huge execution time [6].

Many techniques were proposed in the literature to reduce the execution time including parallelization [7], pipelining [8], using only integer operation [9] and using special hardware [10].

Most of the proposed technique keep the same structure of the fuzzy system and try to increase the speed of computation.

In this paper, we propose a new structure of fuzzy systems. We call it *"Two-Stage Representation"*, because we propose to embed the inference stage of the conventional fuzzy systems into the defuzzification stage. The objective of the proposed structure is to enhance the execution time.

The rest of this paper is organized as follows. In Section 2 the units of conventional fuzzy systems are presented. The new two-stage representation of fuzzy systems is derived in Section 3. The new representation is applied to truck backer-upper control problem in Section 4 and the results are compared with the result obtained using conventional fuzzy system. Finally a concluding remarks and future works are given in Section 5.

II. CONVENTIONAL FUZZY SYSTEMS

Conventional fuzzy systems consist of three units [11,12], membership functions of systems' inputs, rules, and membership functions of system's output. Membership functions define the fuzzy sets of inputs and outputs, while the rules define the relationships between input fuzzy sets and output fuzzy sets.

The computation of fuzzy system's output passes through three units, fuzzification unit, inference unit, and defuzzification unit as shown in Fig.1.

The process of computing the fuzzy system's output is as follows:

1. The fuzzification unit finds the value of all input membership functions at corresponding given crisp inputs. The output of this unit will be a set of fuzzy sets.
2. The inference unit has three tasks:
   a. Uses the antecedent parts of the rules to combine input fuzzy sets.

b. Uses the consequent parts of the rules with the output of previous task to imply the fuzzy sets of the output.

c. Aggregates the output fuzzy sets in one output fuzzy set.

3. The defuzzification unit finds the crisp output value at which the output membership function has the fuzzy set obtained from previous stage.

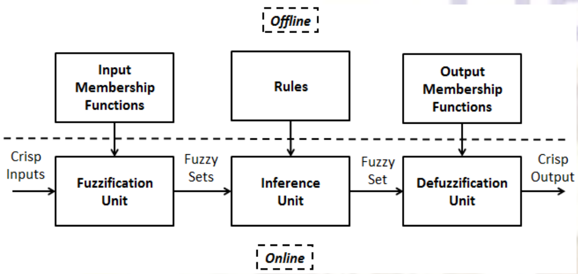

Figure 1. The units of conventional fuzzy system.

In order to illustrate the above procedure, consider the following tipping problem [13]:Given two sets of numbers between 0 and 10 (where 10 is excellent) that respectively represent the quality of the service and the quality of the food at a restaurant, what should the tip be? Fig. 2 shows a fuzzy solution to the tipping problem.

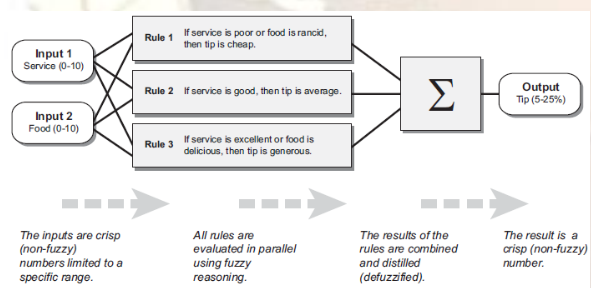

Figure 2. Fuzzy solution of the tipping problem.

Suppose the service is 3 and the food is 8 then the tip can be calculated using the fuzzy system in Fig. 2 and the procedure described above as shown in Fig. 3.

If r is the number of inputs and m is the number of rules, then the process of computing the fuzzy system's output needs: (r × m) fuzzification operations, m connection operations, m implication operations, one aggregation operation, and one defuzzification operation. In total the process needs (r × m) + 2(m + 1) operations. It should be noted that all these operations need to be done online, i.e. during the execution of the fuzzy system.

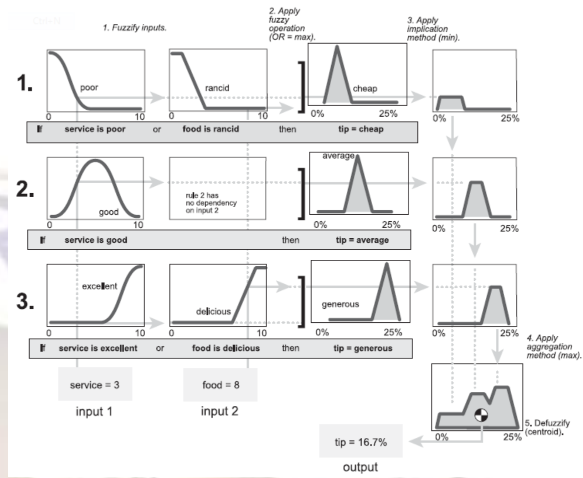

Figure 3. Output of the tipping fuzzy system for service = 3 and food = 8.

In next section, a two-stage representation of fuzzy systems is developed which reduces the number of operations needed to find the fuzzy system's output. In addition most of the needed operations can be done offline which reduces the execution time of the fuzzy system.

III. Two-Stage representation

In this section, each unit in conventional fuzzy systems shown in Fig. 1 will be analyzed and new units will be proposed in order to simplify the process of computing the fuzzy system's output.

Consider the following general form of rule$j$:

$$If\, con(x_i\, is\, MF_i^j)\, Then\, y\, is\, MF_o^j \quad (1)$$

where $x_i$is the system input number i, MF_i^j is the membership function of input i in rule$j$, $con$is the connection function of antecedent part in rule j, y is the system output, and $MF_o^j$is the membership function of output in rule$j$.

3.1 Analyzing fuzzification unit: in conventional fuzzy systems the fuzzification unit takes the crisp inputs and finds the value of corresponding membership function in each rule, i.e. the output of fuzzification unit of rule in equation (1) will be:

$$\mu_i^j = MF_i^j(x_i) \quad (2)$$

In order to simplify the functionality of fuzzification unit, the interval of input $x_i^1 \le x_i \le x_i^2$ and it corresponding membership functions in all rules MF_i^j are sampled and stored in one matrix as shown in equation (3).

$$\propto_i = \begin{bmatrix} MF_i^1 \\ MF_i^2 \\ \vdots \\ \vdots \\ MF_i^m \end{bmatrix}, i = 1, 2, \dots, r \quad (3)$$

where m is the number of rules and r is the number of system's inputs. Using equation (3) the fuzzification process consists of finding the index ofthe input x_i in the range $[x_i^{min}, x_i^{max}]$ and extract the corresponding column from matrix $\propto_i$.

3.2 Analyzing inference unit: in conventional fuzzy systems, the inference unit has three tasks.

3.2.1 Task 1. The inference unit combines μ_i^j for all inputs in each rule to get one fuzzy set for each rule as shown in equation (4).

$$\mu^j = con(\mu_i^j) \quad i = 1, 2, \dots, r \quad j = 1, 2, \dots, m \quad (4)$$

3.2.2 Task 2. The inference unit implies the output of each rule as shown in equation (5) where *imply* is an implication operator which can be the minimum or the product or other operators.

$$\overline{MF_o^j} = imply(\mu^j, MF_o^j), j = 1, 2, \dots, m \quad (5)$$

where μ^j is given in equation (4) and MF_o^j is defined by the rule in equation (1).

3.2.3 Task 3. The inference unit aggregates $\overline{MF_o^j}$ in all rules in one fuzzy set as shown in equation (6) where *aggr* is an aggregation operator which can be the maximum or the sum or the algebraic sum or other fuzzy operator.

$$MF_o = aggr(\overline{MF_o^j}), j = 1, 2, \dots, m \quad (6)$$

In order the simplify Task 2 and Task 3 of the inference unit and without loss of generality we will consider the product operator as implication operator and the sum operator as aggregation operator. In this case the two equations (5) and (6) can be combined in one equation as shown in equation (7).

$$MF_o = sum(\mu^j \times MF_o^j), j = 1, 2, \dots, m \quad (7)$$

where sum is performed element by element.

3.3 Analyzing defuzzification unit: in conventional fuzzy systems, thedefuzzification unit finds the crisp y corresponding to the fuzzy set defined in equation (7). One of the most used methods in defuzzification unit is the center of gravity (COG) method defined in equation (8).

$$y = \frac{sum\left(prod\left(MF_o, [y^{min}, y^{max}]\right)\right)}{sum\left(MF_o\right)} \quad (8)$$

where $prod$ in the numerator is matrix multiplication performed element by element.

In order to embed Tasks 2 and Task 3 in the inference unit in the defuzzificationunit, define the following matrices.

$$\propto = [\mu^1 \quad \mu^2 \quad \cdots \quad \cdots \quad \mu^m] \quad (9)$$

$$M_o = \begin{bmatrix} MF_o^1 \\ MF_o^2 \\ \vdots \\ \vdots \\ MF_o^m \end{bmatrix} \quad (10)$$

$$PM_o = \begin{bmatrix} y^{min} \times MF_o^1 \\ y^{min+1} \times MF_o^2 \\ \vdots \\ \vdots \\ y^{max} \times MF_o^m \end{bmatrix} \quad (11)$$

$$SM_o = sum(M_o) = M_o \begin{bmatrix} 1 \\ 1 \\ \vdots \\ \vdots \\ 1 \end{bmatrix} \quad (12)$$

$$SPM_o = sum(PM_o) = PM_o \begin{bmatrix} 1 \\ 1 \\ \vdots \\ \vdots \\ 1 \end{bmatrix} \quad (13)$$

where $\mu^j, j = 1, 2, \cdots, m$are given in equation (4) and $MF_o^j, j = 1, 2, \cdots, m$ are given in the rules (1).

Using equations (7), (9), (10), (11) and (12), the numerator of equation (9) can be written as follows.

$$sum\left(prod(MF_o,[y^{min},y^{max}])\right) = sum\left(prod(\propto M_o,[y^{min},y^{max}])\right) = sum(\propto PM_o) = \propto .\times SPM_o \quad (14)$$

In the same way the denominator of equation (8) can be written as follows.

$$sum(MF_o) = sum(\propto M_o) = \propto .\times SM_o \quad (15)$$

where .× in equations (14) and (15) means element by element matrix multiplication.

Substitution equations (14) and (15) in equation (8) gives equation (16).

$$y = \frac{\propto .\times SPM_o}{\propto .\times SM_o} \quad (16)$$

Notice that equation (16) embeds Tasks 2 and Task 3 in the inference unit in the defuzzification unit which simplifies the process of computing the fuzzy system's output. Notice also that matrices SPM_o and SM_o can be computed offline while $\propto$ has to be computed online.

Based on above development, the two-stage representation of fuzzy systems can be summarized as shown in Fig. 4.

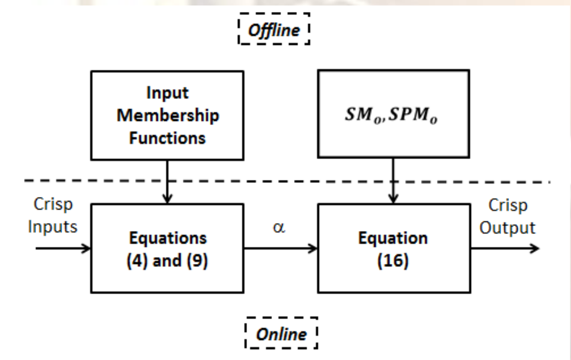

Figure 4. The units of two-stage fuzzy systems.

The process of computing the fuzzy system's output is as follows:

1. Construct offline the matrices SM_o and SPM_o using equations (10)-(13).
2. Find $\propto$ using equations (4) and (9).
3. Compute the fuzzy system's output using equation (16).

If r is the number of inputs and m is the number of rules, then the process of computing the output of the two-stage fuzzy system needs: r fuzzification operations, m connection operations, and one defuzzification operation. In total the process needs (r + m + 1) online operations.

In next section, the above procedure will be applied to truck backer-upper control system.

IV. Performance Evaluation

In order to evaluate the performance of the proposed two-stage representation, it will be used to represent the controller of truck backer-upper system. The system will be simulated and the result will be analyzed and compared with the conventional fuzzy controller.

The goal of the truck backer-upper control problem is to back a truck to a loading dock as quickly and precisely as possible. This control problem is a typical non-linear control problem that cannot be solved by the conventional control techniques [14].

Fig. 5 shows a model of a truck and a loading dock used in the truck backer-uppercontrol problem. The position of the truck is precisely determined by (x, y, φ)where φ is the angle between the truck's forward direction and the x axis. Thetracking control of the truck is done by the angleθwhich is the angle between the truck'sforward direction and the axis of wheel.

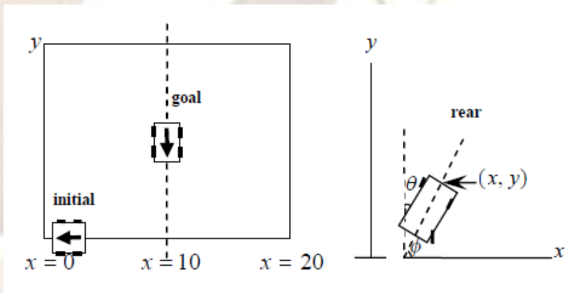

Figure 5. Truck backer-upper control system.

The approximate control dynamics of the truck backer-upper control problem is given by [14]:

$$x(t+1) = x(t) + cos\big(\varphi(t)+\theta(t)\big) + sin\big(\varphi(t)\big)sin\big(\theta(t)\big) \quad (17)$$

$$y(t+1) = y(t) + sin\big(\varphi(t)+\theta(t)\big) - cos\big(\varphi(t)\big)sin\big(\theta(t)\big) \quad (18)$$

$\varphi(t+1)=\varphi(t)-\sin^{-1}\left(2\sin(\theta(t)/b)\right)$(19)

where b is the length of truck and b is set to 4 in this work.

If the distancebetween the truck and the loading dock is sufficiently great, it has only to back thetruck straightforwardly once the truck comes close to near x = 1 and φ = 0°. Thus the variable y can be excluded from the fuzzy input variables (x, y, φ) forsimplicity.

So, the design problem of fuzzy controller for the truck backer-uppercontrol problem is reconfigured as to back the truck at a certain position (x_0, φ_0) inthe intervals {0 ≤ x ≤ 20, -90° ≤ φ ≤270°} to the loading dock at (x = 10, φ = 90°) as fast and precisely as possible.

Fig. 6 shows the membership functions of the input variables x, φ and the output variableθ.

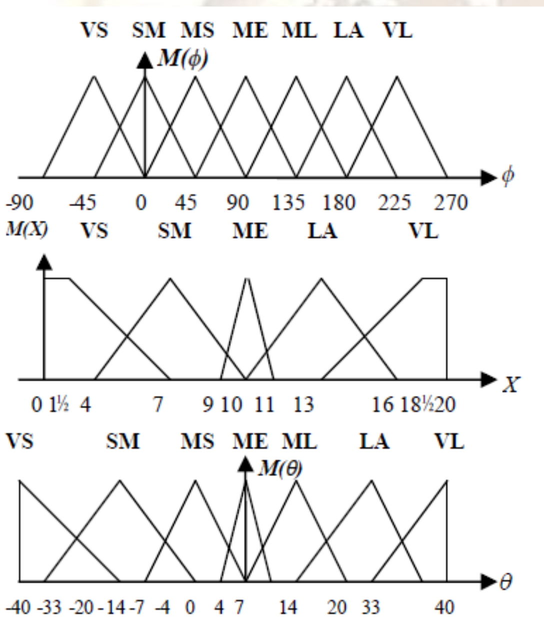

Figure 6. Membership functions of the input variables x, φ and the output variableθ.

Table 1 shows the fuzzy rule base for the truck backer-upper control problem where VS, SM, MS, ME, ML, LA, andVL stand for Very Small, Small, Medium Small, Medium, Medium Large, Large, and Very Large, respectively.

The procedure of applying the fuzzy controller (Fig. 6 and Table 1) on truck backer-up system (equations (17)-(19)) is as follows [9]:

Table 1. Fuzzy rules for the truckbacker-upper control problem.

φ\ x	VS	SM	ME	LA	VL
VS	SM	VS			
SM	SM	VS	VS	VS	
MS	ML	MS	SM	VS	SM
ME	LA	LA	ME	SM	SM
ML	LA	VL	LA	ML	MS
LA		VL	VL	VL	LA
VL				VL	LA

1. Using the value of the initial x and φ, find θ using the fuzzy system described in Fig. 6 and Table 1.
2. Using the control kinetics in equations (17) and (19), find x and φ values for the next stage.
3. Repeat steps 1 and 2 until the goal is reached.

The above procedure was applied using conventional fuzzy system and then using proposed two-stage representation. Table 2 shows the obtained simulation results using conventional fuzzy system and the proposed two-stage fuzzy representation running on the same machine.

Table 2. The simulation results.

Iteration	**Conventional Method**			**Proposed Two-Stage Method**		
-	x	φ	-θ	x	φ	-θ
0	1.00	0.00	20.00	1.00	0.00	20.00
1	1.94	10.04	20.00	1.94	10.04	20.00
2	2.86	20.08	16.14	2.86	20.08	16.91
3	3.77	28.17	10.83	3.76	28.56	10.58
4	4.63	33.59	12.69	4.63	33.85	9.46
5	5.45	39.94	9.74	5.45	38.58	8.48
6	6.20	44.82	4.34	6.22	42.83	7.04
7	6.91	46.99	7.00	6.95	46.35	7.00
8	7.59	50.49	7.00	7.63	49.85	7.00
9	8.22	53.99	7.00	8.27	53.35	7.00
10	8.80	57.49	7.00	8.86	56.85	7.00

11	9.33	60.99	16.98	9.41	60.36	17.24
12	9.80	69.51	18.95	9.88	69.00	19.44
13	10.13	79.02	21.72	10.22	78.76	20.81
14	10.31	89.93	9.81	10.40	89.21	9.05
Time [msec.]	**11.8446**			**5.8884**		
# of online operations	**110**			**30**		

From Table 2, we can see that using both conventional and two-stage fuzzy systems the truck reaches approximately the desired final state ($x = 10$, $\varphi = 90°$) in 14 iterations. However, the proposed two-stage representation needs less time and requires less number of online operations to make the truck reaching the same state.

V. Conclusion

In this paper, a two-stage representation of fuzzy systems is developed. In this representation, the inference stage is embedded in the defuzzification stage. This reduces the number of operations required to find the output of a fuzzy system and therefore reduces real-time computation. The proposed representation was used to model a fuzzy controller of a truck backer-upper system. The comparison between the computation time of the proposed representation and the conventional one showed that the proposed representation has higher performance. Future work will investigate digital and parallel implementation of the proposed representation of fuzzy systems.

Acknowledgements

Dr. Anas Fattouh is also a member of the staff at theUniversity of Aleppo, Aleppo, Syria.

References

[1] L. A. Zadeh, Fuzzy Sets, *Information and Control, 8,* 1965, 338-353.
[2] E. H. Mamdani, Application of fuzzy algorithms for control of a simple dynamic plant, *Proc. IEE 121(12),* 1974, 1585-1588.
[3] J. Yen and R. Langari, *Fuzzy Logic: Intelligence,* (Prentice Hall, Englewood Cliffs, NJ, USA, 1999).
[4] A. M. Ibrahim, *Fuzzy logic for embedded systems applications* (Elsevier Science, 2004).
[5] B. K. Bose, *Power electronics and motor drives: advances and trends* (Academic Press, 2006).
[6] Z. Miao and X. Zhao, Matrix representation and implementation of fuzzy system, *ICSC Congress on Computational Intelligence Methods and Applications,* 2005.
[7] S. G. Lee, Parallel fuzzy inference system for large volumes of remote sensing data, *Proc. of the IEEE International Symposium on Industrial Electronics,* vol. 1, 2001.
[8] G. Aranguren, et. al., Hardware implementation of a pipeline fuzzy controller, *Fuzzy Sets and Systems 128,* 2002, 61-79.
[9] S. G. Lee and John D. Carpinelli, High-speed integer operations in the fuzzy consequent part and the defuzzification stage for intelligent systems, *International Journal of Intelligent Control and Systems, 10(4),* 2005, 258-268.
[10] D. Anderson and S. Coupland, Parallelisation of Fuzzy Inference on a Graphics Processor Unit Using the Compute Unified Device Architecture, *Proc. of the 2008 UK Workshop on Computational Intelligence,* 2008.
[11] L.X. Wang and J. M. Mendel, Generating fuzzy rules from numerical data, with applications, USC-SIPI Report No. 169, University of Southern California, Los Angeles, 1991.
[12] L. X. Wang and J. M. Mendel, Generating fuzzy rules by learning from examples, *IEEE Trans. on System, Man, and Cybernetics, 22(6),* 1992, 1414-1427.
[13] *Fuzzy Logic Toolbox™ User's Guide* (The MathWorks, Inc., 2012).
[14] D. Kim and I.H. Cho, An accurate and cost-effective COG defuzzifier without the multiplier and the divider, *Fuzzy Sets and Systems, 104,* 1999, 229-244.