International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 6, June - 2014

Full Parameterization Process for Singleton Fuzzy Logic Controllers: A Computing Algorithm

Osama A. Awad
College of Information Engineering.
Nahrain University
Baghdad, Iraq

Rami A. Maher
Dept. of Electrical Engineering
Isra University
Amman, Jordan

***Abstract*—Although fuzzy systems demonstrate their ability to solve different kinds of problems in various applications, there is an increasing interest on developing solid mathematical implementations suitable for control applications such as that used in fuzzy logic controllers (FLC). It is well known that, wide range of parameters is needed to be specified before the construction of a fuzzy system. To simplify in a systematic way the design and construction of a general fuzzy system, and without loss for generality a full parameterization process for a singleton type FLC is proposed in this paper. The presented methodology is very helpful in developing a universal computing algorithm for a standard fuzzy like PID controllers. An illustrative example shows the simplicity of applying the new paradigm.**

Keywords—parameterization;fuzzy logic controller(FLC); singletone FLC ; PID

I. INTRODUCTION

Since Zadeh introduced the basics of fuzzy sets [1] in 1965, and the fuzzy logic concepts [2] in 1968; fuzzy logic has been successfully applied to a wide range of applications in various fields. Mamdani and Assilian [3] first applied the fuzzy logic control in to the control field, and since then fuzzy logic controllers have attracted a great deal of interest among many researchers. Later on, fuzzy logic controller is proven to be an effected way in control engineering applications.

There are mainly two types of a ruled base fuzzy system. One is the Mamdani type FLC [4], and the other is the Takagi-Sugeno (TS) [5]. Structure for the both types are the same, the only difference is related to the definition of the output in the consequent field of the rule base. TS type uses a crisp values for the output in the rule base, where it is a fuzzy linguistic in the case of Mamdani type.

Another type gaining a wider acceptance in control and industrial applications, which is called a singleton fuzzy controller [6] will be adopted and focused on by this paper. Although it defines a singleton membership function over the output, it is actually uses a constant real value called a singleton of the rule output, representing the position of the trivial output MF. With this type several activation, accumulation and defuzzification methods yield identical results [7].

As the field of fuzzy computing is an active research field, many methodologies are developed for constructing and computing the FLC. The designer of a fuzzy controller for certain control application is faced with the many design choices that the fuzzy set theory provides. Fundamental comparisons and suggestions are found in the literature, and they are well presented by [8-11,16-18]. These computing approaches are not unique; it is mainly due the lack of having a general good mathematical formulation for the fuzzy system construction algorithm.

A solution for this problem may be solved if good parameterization process is developed. The parameterization of a fuzzy system is insufficiently addressed in the literature. This work is a trial to solve this problem, and mainly devoted to present a fuzzy system for control applications, whether used in construction of the FLC, or in fuzzy system modeling or used in design and tuning of the FLC itself.

II. PROPLEM FORMULATION

Fuzzy logic system (of which FLC is a special application) is a natural extension of fuzzy set theory to relations between fuzzy sets and rules. A FLC is characterized by four modules: [fuzzifier, inference engine, knowledge base, and defuzzifier]. A schematic representation of FLC is presented in Fig. 1.

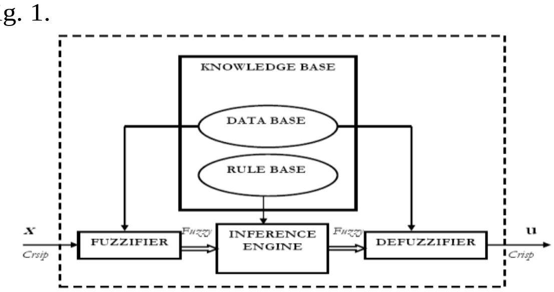

Fig. 1. Basic structure of a FLC.

The parameters of an FLC can be classified into four categories [12]: logical, structural, connective, and operational as can be shown in Table I.

A. Parameterization Process of the FLC

In the following, we will discuss the suggested FLC parameterization methodology specified for a singleton type fuzzy system, from different aspects according to the classification of parameters summarized by Table I.

B. Parameterization of Input Membership Functions

Consider the fuzzy system as shown in Fig. 2. which is a simplified form of Fig. 1 with input x and output u, where $x \in \Re^{n_x}$ is the fuzzy system input variables, $u \in \Re^{n_u}$ is the fuzzy system output variables, and { n_x, n_u } are the dimension of input and output variables.

TABLE I. PARAMETER CLASSIFICATION OF A FLC

CLASS	PARAMETERS
LOGICAL	REASONING MECHANISM, FUZZY OPERATORS, MEMBERSHIP FUNCTIONS TYPE, DEFUZZIFICZTION METHOD
STRUCTURAL	RELEVANT VARIABLES, NUMBER OF MEMBERSHIP FUNCTIONS, NUMBER OF RULES
CONNECTIVE	ANTECEDENT PART OF THE RULE, CONSEQUENT PART OF THE RULE, RULE WEIGHTS
OPERATIONAL	MEMBERSHIP FUNCTION VALUES

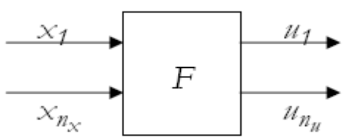

Fig. 2. A simplified input-output fuzzy block.

The input membership functions are parameterized by:

$$\theta_{in} \in \Re^{n_x \times n_i} \qquad (1)$$

where: $$n_i = c \times \max(m_{x_i}; i = 1,2,......n_x) \qquad (2)$$

and c represents the standard number of parameters that define a certain type of MF, m_{x_i} is the number of fuzzy sets assigned for input x_i. The $i\,th$ row of θ_{in} includes all the parameters that characterize the MFs which are related to the $i\,th$ input (shape, no. of sets, width of the fuzzy set, spacing and overlapping between the sets…).

Example 1:

For a fuzzy system with two inputs, each defined with three triangular MFs, then $n_x = 2$; $m_{x_1} = 3$; $m_{x_2} = 3$ and $c = 3$.

A triangular MF is normally defined by three points (a, b, c). Therefore θ_{in} will be defined by

$$\theta_{in} \in \Re^{2\times 9}$$

$$\theta_{in} = \begin{bmatrix} a_{11}b_{11}c_{11}a_{12}b_{12}c_{12}a_{13}b_{13}c_{13} \\ a_{21}b_{21}c_{21}a_{22}b_{22}c_{22}a_{23}b_{23}c_{23} \end{bmatrix} \qquad (3)$$

As the triangular MF is characterized by its core (most fuzzy systems employ a normalized fuzzy sets to specify the entire partition of a fuzzy variables), a great number of reduction in the number of parameters defining the input MF can be introduced. Without loss of generality, this implementation is adopted in the work. Hence (3) will be reduced to:

$$\theta_{in} = \begin{bmatrix} b_{11}b_{12}b_{13} \\ b_{21}b_{22}b_{23} \end{bmatrix} \qquad (4)$$

C. Parameterization of Membership Degree

In FLC computing, only the degree of membership is further processed. Often information is lost during this procedure, although it is not required that the MFs are normalized (i.e. their sum is equal to one for all x), this property is called fuzzy partition and often is employed because it makes the interpretation and computation easier [7]. The degree of membership is obtained for the current input vector by:

$$\delta_{x_i} \in \Re^{m_{x_i}} \qquad (5)$$

and,

$$\delta_{x_i} = \mu(x_i; \theta_{in}^{(i)}, m_{x_i}), i = 1,2,3......, n_x. \qquad (6)$$

where $\mu(.;.)$ is a generalized MF producing degrees of membership for all fuzzy sets related to the input x_i and

$$\delta_{x_i} = [\, \mu_1, \mu_2, ..., \mu_k, ...\mu_{m_{x_i}} \,] \qquad (7)$$

is a vector of dimension m_{x_i}, each element represents the degree of membership for fuzzy subset (k) associated with input x_i and evaluated using the input parameter $\theta_{in}^{(i)}$ at a given point in the range of the relative input.

Example 2:

Consider the previous example; membership degree μ can be evaluated for input x_1 at a measured value cx_1 by using (7) as follows:

$$\delta_{x_1} = \mu_T(cx_1; \theta_{in}^{(1)}, m_{x_1} = 3)$$
$$\delta_{x_1} = [\mu_1 \mu_2 \mu_3] \qquad (8)$$

where the index T is put as an indicator targeting for using the triangular function in calculating the membership degree.

$$\mu_1 = \mu_T(cx_1; [a_{11}b_{11}c_{11}]), k = 1$$
$$\mu_2 = \mu_T(cx_1; [a_{12}b_{12}c_{12}]), k = 2 \qquad (9)$$
$$\mu_3 = \mu_T(cx_1; [a_{13}b_{13}c_{13}]), k = 3$$

The membership degree of each fuzzy set $(k = 1,2,3)$ for a triangular MF is evaluated by:

$$\mu_k(cx_1) = \begin{bmatrix} 0 & ; \; cx_1 \le a_{1k} \\ \dfrac{cx_1 - a_{1k}}{b_{1k} - a_{1k}} & ; \; a_{1k} \le cx_1 \le b_{1k} \\ \dfrac{c_{1k} - cx_1}{c_{1k} - b_{1k}} & ; \; b_{1k} \le cx_1 \le c_{1k} \\ 0 & ; \; c_{1k} \le cx_1 \end{bmatrix} \quad (10)$$

at least one value of the vector δ_{x_1} is not zero, otherwise the value of the input x_1 is not represented by any of the MFs (fuzzy sets) defined over the relative UOD.

For an acceptable (50%) overlapping between MFs, it is sufficient to parameterize the inputs by their cores only as shown in Fig. 3.

D. Parameterization of Rule Base Premise

Without loss of generality, completeness of the rule

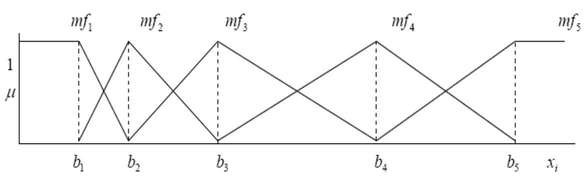

Fig. 3. Triangular input MF parameters represented by their cores.

base (RB) will be assumed, by taking all the possibilities encountered by the predefined MFs over the input variables. This approach, which is adopted by the presented methodology, will cover both reduced and full RB representation by letting the certainty of each undefined rule to be zero, and set one otherwise.

Also, most of FLCs used in control application are assumed to have fuzzy propositions connected with fuzzy *and* connectives only. Thus the structure of the rules premise is defined by:

$$\theta_{rules} \in Z^{n_R \times n_x} \quad (11)$$

where n_R denotes the total number of rules in the RB, and n_x represents the number of inputs used in constructing the FLC. Noting that n_R is defined by:

$$n_R = \prod_{i=1}^{n_x} m_{x_i} \quad (12)$$

a row of θ_{rules}, connects the index of the input fuzzy sets for each input variable defined by each rule, and hence reflects the degree of membership δ_{x_i} function to be taken into considerations for evaluating the premise certainty (truth value) of the specified rule, defined some times in the literature as degree of fulfillment (dof) or firing strength.

Accordingly, the firing strength of the rule is performed using the generalized T-norm or T-conorm function given by:

$$\Phi^{(j)} = T(\delta_{x_1}, \delta_{x_2}, \ldots, \delta_{x_i}, \ldots \delta_{x_{nx}}; \theta^{(j)}_{rules}), \; j = 1,2,..n_R \quad (13)$$

where (13) operates the T-norm or T-conorm between the elements of the vectors defined by θ_{rules}. This operation represents the aggregation stage of the inference engine.

Example 3

Consider the same previous example, for which the premise part of the complete RB will be constructed as follows:

$$\theta_{rules} = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 & 3 & 3 & 3 \\ 1 & 2 & 3 & 1 & 2 & 3 & 1 & 2 & 3 \end{bmatrix}^T \forall \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (14)$$

where the dimension of the RB according to (12) is $n_R = 9$. Then, to compute the dof for rule $j = 6$, we will proceed as follows:

1st -Access (14) given j equal six, then get the index for the MF defined for each input variable $\Rightarrow k = 2$ for input x_1, and $k = 3$ for input x_2.

2nd-Apply the T-norm on the degree of memberships (δ_{x_1} and δ_{x_2}) as given by (6), choosing product operator (or minimum) to represent the T-norm, then (13) gives the following:

$$\Phi^{(6)} = T(\delta_{x_1}, \delta_{x_2}; \theta^{(6)}_{rules}) \quad (15)$$

$$\Phi^{(6)} = \mu_2(x_1) \bullet \mu_3(x_2) \quad (16)$$

Hence, certainty of the 6th rule is evaluated by the product of the 2nd MF value (defined by mf_2) for input 1 (calculated at the measurement value of input x_1) and the 3rd MF value (defined by mf_3) for input 2 (calculated at the measurement value of input x_2).

E. Parameterization of the Rule Base Output

After the *dof* has been calculated for all the rules, and, for the inference engine process to be completed, it is required now to consider the outputs of the RB.

Now, consider the output parameters vector by defining:

$$\theta_{out} \in \Re^{n_R \times n_u} \quad (17)$$

where n_u represents the number of controller outputs, n_R represents the dimension of the rule base.

The i th row within θ_{out} is defined by:

$$\theta^{(i)}_{out} = [h_{i1} \ldots h_{ij} \ldots h_{in_u}]; \; j = 1,2,\ldots.n_u \quad . \quad (18)$$

where h_{ij} is a constant real value representing the place of the singleton MF selected from a number of m_{u_j} fuzzy sets defined over the output u_j at rule i.

Since the output membership value is always one at the core and zero elsewhere, hence the outputs in the RB are always defined by their singletons which are represented by:

$$h_{ij} = c_k \quad ; \quad k = 1,2,3 \; \ldots \; m_{u_j} \qquad (19)$$

Example 4

For a singleton FLC structure with one output ($n_u = 1$) defined over five MFs ($m_{u_1} = 5$), then the RB consequence of the previous example could be represented by the following parameterization vector:

$$\theta_{out} = [\; h_{11} \quad h_{21} \; h_{31} \; . \; . \; . \; . \quad h_{81} \quad h_{91}]^T \qquad (20)$$

and, for five singleton MFs defined by:

$$[\, c_1 \quad c_2 \quad c_3 \quad c_4 \quad c_5 \,]$$

then θ_{out} could be set as follows:

$$\theta_{out} = [\, c_1 \; c_2 \; c_3 \; c_2 \; c_3 \; c_4 \; c_3 \; c_4 \; c_5 \,]^T \qquad (21)$$

for an arbitrary values of the output singleton MF cores, as given by $[\; -1 \quad -0.7 \quad 0 \quad 0.7 \quad 1 \;]$

then, the output RB parameterization vector could be set by:

$$\theta_{out} = [-1 \; -0.7 \;\; 0 \;\; -0.7 \;\; 0 \;\; 0.7 \;\; 0 \;\; 0.7 \;\; 1]^T \qquad (22)$$

F. Parameterization of the FLC Output

As a final stage for calculating the crisp output out of the FLC, a defuzzificztion stage is mandatory for this purpose. Many defuzzificztion formulas are developed [11], each is suitable for a certain application. For control applications it is found that using the center of area (COA), and a well known version named the center of gravity (COG) are highly recommended [8].

The output U of the FLC is evaluated by:

$$U = \frac{\theta_{out}^T \cdot \Phi}{\sum_{i=1}^{n_R} \Phi^{(i)}} \qquad (23)$$

where Φ and θ_{out} are defined by (13) and (18) respectively.

The generalized form of the defuzzified output can be written in the following form:

$$U = D(\Phi, \theta_{out}) \qquad (24)$$

where D can be any defuzzification formula applied to evaluate the crisp output U.

G. Parameterization of Input and Output Variables

In closed loop systems as shown in Fig. 4., there are several signals which should be taken into consideration when the control signal is calculated. The error signal between the set point y^* and the measurement output

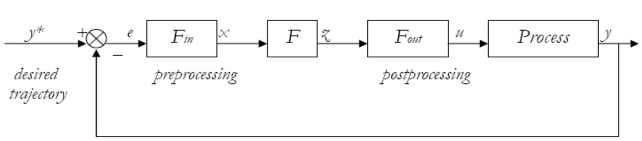

Fig. 4. Typical FLC in a closed loop control system.

variable y is observed, and given by:

$$e(k) = y^*(k) - y(k) \qquad (25)$$

The main control objective is to keep the error signal as small as possible. Also, the rate of change in the error signal is given by:

$$\Delta e(k) = e(k) - e(k-1) \qquad (26)$$

with those two inputs $\{ e(k), \Delta e(k) \}$ the FLC can perform the PD or PI type control depending on whether the output is taken to be the pure control signal $u(k)$ or the change in the control signal $\Delta u(k)$. Different PID like FLC structures can be generated using the above concepts in various forms.

H. Parameterization of the Fuzzy Operators

There are multiple choices for representing the premise fuzzy conjunction *and* , and fuzzy disjunction *or* operators. A common choice is the (*min-max*) composition [13] and the (*product-sum*) composition [14]. It is found that representing the T-norm by any operator other than the *product* operator will introduce un-adjustable nonlinearities [11]. In this work *product* operator is used to implement both and conjunction and the T-implication.

I. Parameterization of the scaling factors

Although scaling factors is not part of the parameters for a fuzzy system, but practically, is highly acceptable to be part of the FLC structure. Those scaling factors related to the inputs g_X ($g_e, g_{\Delta e}$) and the outputs g_U ($g_u, g_{\Delta u}$) are playing an important role in tuning the fuzzy controller and for normalization the input and output UODs.

$$UOD_{x_1} = [-\alpha, \alpha] \Rightarrow g_{x_1} = [1/\alpha]$$
$$UOD_{x_2} = [-\beta, \beta] \Rightarrow g_{x_2} = [1/\beta]$$
$$UOD_{u_1} = [-\gamma, \gamma] \Rightarrow g_{u_1} = [1/\gamma] \qquad (27)$$

Actually they are part of the pre-processing and post-processing stages.

III. FLC COMPUTING ALGORITHM

Back to Table 1, a complete set of FLC parameters are defined and connected by systematic mathematical formulation suitable for the computing algorithm, and summarized as follows:

1) The inference mechanism uses the singleton fuzzy system as given by the general RB structure:

$$IF \;\; (\theta_{rules}) \;\; THEN \, (\theta_{out}) \qquad (28)$$

2) Relevant inputs are $\{ e_1 \,, \Delta e_1 \}$ and outputs are $\{ u_1, \Delta u_1 \}$.
3) Number of MFs defined for each input and output is given by m_{x_i} and m_{u_j}.
4) Input MFs characteristics θ_{in} are defined by (3), and output MFs are defined by singletons (19).
5) The fuzzy operators are defined to use the *product*

operator for calculating the rule premise certainty as given by (13) and for calculating the implication as given by numerator part of (23).

6) Number of rules is calculated by (12).
7) Antecedents of the rules are defined by θ_{rules} (11).
8) Consequents of the rules are defined by θ_{out} (18).
9) Membership values are calculated by (7).
10) Firing strength of the rule (*dof*) is defined by (14).
11) Defuzzifications are performed using (24).

Based on the parameterization process proposed above for a singleton FLC, the output of the FLC is given by:

$$U = F(X, \theta_{in}, \theta_{rules}, \theta_{out}, g_X, g_U) \qquad (29)$$

Fig. 5. shows the main components constructing the FLC generated according to the presented parameterization methodology.

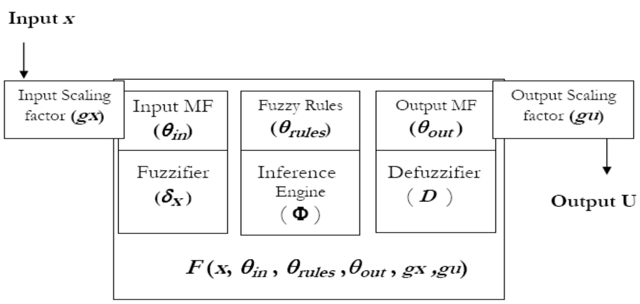

Fig. 5. Components of a FLC computing algorithm.

IV. ILUSTRATIVE EXAMPLE (INVERTED PENDULUM PROBLEM)

Consider the problem of balancing an inverted pendulum on a cart [15] as shown in Fig. 6. for which the dynamic equation is given by:

$$\ddot{y} = \frac{9.8\sin(y) - \cos(y)\left[\dfrac{u + 0.05\dot{y}^2\sin(y)}{1.1}\right]}{0.5\left[\dfrac{4}{3} - \dfrac{0.1}{1.1}\cos^2(y)\right]} \qquad (30)$$

where y is the angle of the pendulum with respect to the vertical line and $\dot{y}$ is the pendulum angular velocity and u is the applied control force.

To evaluate the presented methodology, a PD like fuzzy controller is to be constructed for the non-linear control system given by (30), using the developed singleton FLC computing algorithm. We will discuss the regulator problem, i.e. keeping the inverted pendulum balanced in a vertical position with reference to different initial positions of $y(0)$.

A. FLC construction

We will show in a systematic way how the presented parameterization methodology is applied for computing the control action u using (29) as follows:

1) The FLC relevant inputs are chosen to be $x_1 = e = y$ and $x_2 = \dot{e} = \dot{y}$ with u as the relevant only output, hence $n_x = 2;$ and $n_u = 1$.
2) The active UODs for e and $\dot{e}$ are chosen to be $[-\frac{\Pi}{2}, \frac{\Pi}{2}]$ and $[-\frac{\Pi}{4}, \frac{\Pi}{4}]$ respectively, and for the output u_1 is chosen to be $[-20, 20]$ (i.e. the control signal boundaries).
3) The input variables are to be partitioned into five triangular MFs satisfying symmetricity feature with 50% overlapping. While the output variable is chosen to have nine equidistant singleton MFs.

Hence:

$$m_{x_1} = 5; m_{x_2} = 5; m_{u_1} = 9$$

And according to (2) for a triangular MF it gives: $c = 3$ and $n_i = 15$.

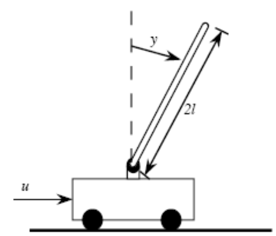

Fig. 6. Inverted pendulum on a cart.

4) θ_{in} is evaluated by (1), and its reduced version (4) gives:

$$\theta_{in} = \begin{bmatrix} -1 & -0.5 & 0 & 0.5 & 1 \\ -1 & -0.5 & 0 & 0.5 & 1 \end{bmatrix}$$

with normalization scaling factors evaluated by (27) as:

$$g_{x_1} = [2/\Pi] \quad ; g_{x_2} = [4/\Pi] \quad ; g_{u_1} = [1/20]$$

5) θ_{rules} reflects the indices of the MFs involved in the construction of the rule premise, which is identified by the two inputs x_1 and x_2 respectively. $n_R = 25$ is calculated by (13), and RB premise is generated by (12) as:

$$\theta_{rules} = \begin{bmatrix} 1111122222333334444455555 \\ 1234512345123451234512345 \end{bmatrix}^T$$

6) The consequence θ_{out} of the rule base is constructed according to (18), using the summation formula defined by passino [8], gives the following:

$$\theta_{out} = [\,1 \;\; .75 \;\; .5 \;\; .25 \;\; 0 \,|\, .75 \;\; .5 \;\; .25 \;\; 0 \;\; -.25 \,|\; .5 \;\; .25 \;\; 0 \;\; -.25 \; -.5 \,|\, .25 \;\; 0 \;\; -.25 \;\; -.5 \;\; -.75 \,|\; 0 \;\; -.25 \;\; -.5 \;\; -.75 \;\; -1\,]^T$$

1

where the output MFs are assumed to have the same partitions of the input MFs, that is identified by:

$\{mf_1, mf_2, mf_3, mf_4, mf_5\}$ = {NL, NS, Z, PS, PL}

If normalization is used, it will be in the form of:

{-1,-0.5, 0, 0.5, 1}

The rule base table generated by θ_{rules} and θ_{out} could be summarized as shown by Table II.

7) The firing strength Φ^i is accomplished using (13), taking into consideration the choice of a triangular form MFs, hence (10) is used to evaluate the membership value at the given measurement points

TABLE II.

RULE-BASE TABLE FOR THE INVERTED PUNDULUM PROBLEM

$\dot{e}$ / e	NL	NS	Z	PS	PL
NL	1	.75	.5	.25	0
NS	.75	.5	.25	0	-.25
Z	.5	.25	0	-.25	-.5
PS	.25	0	-.25	-.5	-.75
PL	0	-.25	-.5	-.75	-1

cx_1 and cx_2 for x_1 and x_2 respectively, for the given θ_{in}.

8) After calculating the vectors Φ and θ_{out}, the COG formula as given by (23) is used for defuzzification, which will calculate the final crisp value for the control action u_1 at instants k based on a the measurement values cx_1 and cx_2, knowing that $cx_1 = e(k)$ and $cx_2 = \Delta e(k)$.

B. Simulation and results

Complete set of programs are written to simulate the presented computing algorithm using MATLAB 7.3. The differential equation for the inverted pendulum problem is solved by using the fourth order Runge-Kutta algorithm. The sampling time is taken to be 0.02 seconds.

The closed loop system output which is identified by the angular position y is examined for different initial conditions of $y(0)$. The designed FLC shows a high stability regulation for the inverted pendulum system over the entire UOD defined for the controller input variables. The behavior of the controlled system is summarized by Fig. 7. It is clearly shown that the effective margin of stability for the proposed simple FLC controller is capable to control back the pendulum to its vertical position from different initial conditions bounded by $y(0) = [-42.97, 42.97]$.

VI. CONCLUSION

A full parameterization process for a singleton fuzzy system is developed. It presents a systematic methodology for developing a singleton fuzzy logic controller for control applications. The assumptions made by the parameterization process is highly simplified the FLC computing algorithm. A well-known inverted pendulum problem is chosen for evaluating the capability of the

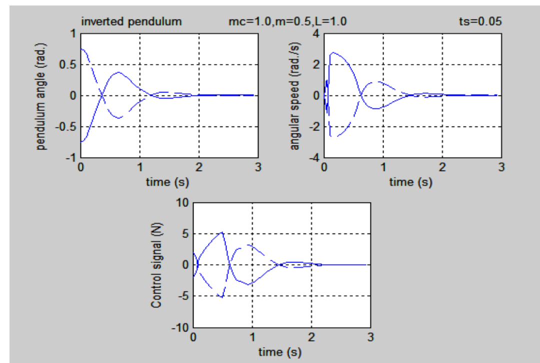

Fig. 7. Inverted pendulum for different initial conditions using the proposed singleton FLC.

proposed approach. The obtained result shows the effectiveness of the developed approach in terms of simplicity and transparency in setting the solution.

Although, the proposed methodology is tested for constructing the FLC, a current research now is initiated for developing a structural design methodology using soft-computing controller based on the presented parameterization process. Both FLC structure and FLC parameters are to be designed and tuned in one single phase simultaneously.

REFERENCES

[1] L. A. Zadeh, “Fuzzy sets,” *Inform. Control*, vol. 8, pp. 338–353, 1965.

[2] ——, “Fuzzy algorithm,” *Inform. Control*, vol. 12, pp. 94–102, 1968.

[3] E. H. Mamdani and S. Assilian, “Applications of fuzzy algorithms for control of simple dynamic plant,” in *Proc. Inst. Elect. Eng*., vol. 121,, pp. 1585–1588, Dec. 1974.

[4] Mamdani, “Advances in the linguistic synthesis of fuzzy controllers,” *Int. Journal of Man-Machine Studies*, vol. 8(6), pp.669–678, 1976.

[5] T. Takagi and Sugeno, “Fuzzy identification of systems and its applications to modeling and control,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 15, pp. 116–132, Jan. 1985.

[6] P. Viljamaa, “Fuzzy gain scheduling and tuning of multivariable fuzzy control-methods of fuzzy computing in control systems,” Ph.D. thesis, Automatic Control Institute, Tampere University of Technology, Tampere, Finland, 2002.

[7] O. Nelles, *Nonlinear System Identification: from Classical Approaches to Neural Networks and Fuzzy Models*, Springer-Verlag Berlin Heidelberg, 2001.

[8] K. M. Passino and S. Yurkovich, *Fuzzy Control*, Addison-Wesley, Melno Park, CA, 1998.

[9] C. Lee, “Fuzzy logic in control systems: Fuzzy logic controller—Part 1; Part 2,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 20, pp. 404–435, Mar./Apr. 1990.

[10] R. Babuska, *Fuzzy Modeling for Control*, Kluwer Academic Publisher, Boston, 1998.

[11] R. Jager, “Fuzzy Logic in Control,” Ph.D. thesis, Technische Universiteit Delft, June 1995.

[12] C. A. Pe˜na-Reyes and M. Sipper,”Fuzzy CoCo: A cooperative-coevolutionary approach to fuzzy modeling,” *IEEE Transactions on Fuzzy Systems*, vol. 9, pp. 727-737, October 2001.

[13] L. A. Zadeh, “Outline of a New Approach to the Analysis of Complex Systems and Decision Processes”, *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 3, pp. 28–44, 1973.

[14] W. Bandler, and L. J. Kohout, “Fuzzy Power Sets and Fuzzy Implication Operators”, *Fuzzy sets and Systems*, vol. 4, pp. 13-30, (1980).

[15] L. X. Wang, “A Supervisory Controller for 'Fuzzy Control Systems that Guarantees Stability,” *IEEE Transactions on Automatic Control*, vol. 39, pp. 1845-1847, Sep. 1994.

[16] A.-Lopez , A. Aguilar-martin ,J.A ,"Simplified Fuzzy Controller : The Natural Logic Control ", vol. 14,no.1,pp.16-30,2006.

[17] Prem Kumar P.I Kar and I Behera, "Variable Gain Controller for Nonlinear Systems Using T-S Fuzzy Model, IEEE Trans on SMC, 2009.

[18] A.G. Aissaoui, M. Abid, A. Tahour and A. C. Megherbi, "A Fuzzy Logic and Variable Structure Control for Permanent Magnet Synchronous Motors", International Journal of Systems Control, vol.,no.1, pp.13-21, 2010.