*Journal of Advanced Computer Science & Technology*, 3 (1) (2014) 37-47
©Science Publishing Corporation
www.sciencepubco.com/index.php/JACST
doi: 10.14419/jacst.v3i1.1783
Research Paper

# Optimal design of T–S fuzzy controller for a nonlinear system using a new adaptive particle swarm optimization algorithm

**Omid Naghash Almasi [1]*, Ali Ahmadi Naghedi [2], Ebrahim Tadayoni [1], Assef Zare [1]**

[1] Departments Of Electrical Engineering, Islamic Azad University, Gonabad Branch, Iran
[2] Department Of Electrical Engineering, Tehran Science And Research Branch, Islamic Azad University, Tehran, Iran
*Corresponding author E-mail: o.almasi@ieee.org

## Abstract

Designing an optimal Takagi–Sugeno (T–S) fuzzy system for real–world non–linear control problems is a challenging problem. Complex non–linear system produces large fuzzy rule–based and requires large amount of memory. To overcome these problems, this paper proposes a hybrid approach to generate the optimal T–S fuzzy system. First, the Fuzzy Clustering Method (FCM) is employed to partitioning the input space and extracting initial fuzzy rule–based. Moreover, a new Adaptive Particle Swarm Optimization (APSO) technique is suggested to determine the optimal number of clusters in FCM, which is the same as the number of fuzzy rules. Finally, Recursive Least Square (RLS) method based on the Mean Square Errors (MSE) criterion is used to regulate the coefficients of the consequent part of initial fuzzy rules. Some simulations are conducted on a Non–Linear Inverted Pendulum (NLIP) system to support the efficiency of the proposed approach in designing compact and accurate T–S fuzzy systems.

***Keywords***: *Adaptive PSO, FCM, Non–Linear Systems Optimal Design; Takagi Sugeno Fuzzy System.*

## 1. Introduction

Fuzzy logic systems are important tools for incorporating human expert knowledge in complement to mathematical knowledge. Based on this feature, utilization of fuzzy knowledge–based control to deal with non–linear systems whose dynamics are not explicitly understood and whose models cannot be simply established has been increased [1], [2]. Among various types of fuzzy systems, Takagi–Sugeno (T–S) fuzzy systems have recently gained much attention due to their special rule consequent structure and their success in a function approximation [3].

In T–S fuzzy systems the consequent part of fuzzy rules is a real–valued function of the input variables instead of a fuzzy set [4]. T–S fuzzy systems have some advantage in comparing to the other fuzzy system because of their modification. They can represent the complex plants stronger and also become appropriate for application in various kinds of learning algorithms.

Generally, human knowledge can be classified into two groups [5]. The first, conscious knowledge which human experts are simply expressed their knowledge in the terms of fuzzy IF–THEN rules. In the other group, subconscious knowledge which the human experts cannot express its knowledge in the term of fuzzy IF–THEN rules. In this situation while the human experts are doing their works, they are viewed as black boxes and the inputs and the outputs data are measured and collected, i.e. input–output data pairs.

T–S fuzzy modeling involves two stages: structure identification and parameter estimation [6]. There are two challenging problems in fuzzy structure identification: first, extracting significant input variables among all possible input–output pairs and the other, determining the number of required rules, i.e. finding how many rules are necessary and sufficient to achieve the given mapping, which is called partition validation or cluster validation.

Those are usually determined by human experts in conventional fuzzy inference design with trial–and–error and based on their experience or some experiments. This study focuses on a data–driven approach since no expert is available to determine the optimal number of rules in the system.

With an increasing number of variables, the complexity of non–linear system, the number of rules is increasing exponentially, which makes more difficult for experts to define the rule set for good system performance, makes memory limitations, and will be a big obstacle to progress in the real application use of T–S fuzzy systems [6–8].

Therefore, an appropriate choice of the number of rules is important in designing the fuzzy systems, because too many rules result in a complex fuzzy system that may be unnecessary for the problem, whereas too few rules produce a less powerful fuzzy system that may be insufficient to achieve the objective. The optimal T–S fuzzy system uses one rule for one input–output pair, thus it is no longer a practical system if the number of input–output pairs is large. For these large problems, various clustering techniques, such as Chiu [9], Chung et al [10], an extension of Kohonen's self–organizing map [11], hierarchical clustering [12], can be used to cluster the input–output pairs so that a cluster can be represented by one rule.

For this purpose, unsupervised fuzzy clustering in input space is used to determine the number of rules in the real control problems. Since Fuzzy C–means (FCM) is a fuzzy version of K–means algorithm, that is widely adopted in the literature, and significantly reduces the number of rules in the system modeling, it was used to identify the structure of FIS in this work.

Two difficulties arise in the application of FCM; one is how to set the initial cluster center, and the other is the optimal number of clusters. For the former, we randomly select the data points and set them as initial centers. Various methods have been proposed to deal with determining the optimal number of clusters in FCM [13], [14].

One of the renowned approaches to determine the number of clusters properly is Particle Swarm Optimization (PSO); however, it has some problem. It convergence to local minima and also has some parameters which should be tuned for having a good search ability [15], [16]. In this study, an Adaptive Particle Swarm Optimization (APSO) is proposed to overcome those shortages. The APSO adjust its parameter adaptively by using the dynamic feedback from the success rate from the personal best updating of its particles.

In this paper, an optimal stable T–S fuzzy system as a controller for a non–linear system is proposed. Adequate input–output data are obtained from the feedback linearization control. Then the initial fuzzy rule–based is derived by the FCM as a rule extraction method and in order for simplicity of fuzzy inference systems, the APSO is proposed to find the optimal number of rules in T–S fuzzy system. Finally, RLS technique based on MSE criterion is used to tune the coefficients of the consequent part of initial fuzzy rules.

This paper is organized as follows. In section 2, APSO is introduced and it superior performance over standard PSO is proved. In section 3, the optimal procedure to design an optimal T–S is presented. In 4.1, The mathematical formulation of a non–linear pendulum (NLIP) as a classic example of non–linear systems is reviewed, 4.2 presents a nonlinear feedback linearization method to stabilize the NLIP, 4.3 presents some well–known error measurements criteria to validate the T–S fuzzy controller, and 4.4 shows the simulation results of applying the optimal T–S fuzzy controller on NLIP. Finally, conclusions are drawn in Section 5.

## 2. Adaptive particle swarm optimization

The Particle Swarm Optimization (PSO) which was first introduced by Kennedy and Eberhart, is one of the modern population–based optimization algorithms [17]. It uses swarm of particles to find a global optimum solution in the search space. Each particle represents a candidate solution to the cost function and it has its own position and velocity. Assume particle swarms are in D–dimensional search space. Let the ith particle in a D–dimensional space be represented as $x_i = (x_{i1}, \dots, x_{id}, \dots, x_{iD})$.The best previous position of the ith particle is recorded and represented as $p_{bi} = (p_{bi1}, \dots, p_{bid}, \dots, p_{biD})$, which gives the best value in the cost function and also called pbest. General best position, gbest, denoted by $p_{gb}$ is the best value of the Pbest among all the particles in the cost function.

The velocity of the ith particle is represented as $v_i = (v_{i1}, \dots, v_{id}, \dots, v_{iD})$. In each of the iterations, the velocity and the position of each particle are updated according to Eq. (1) and Eq. (2), respectively.

$$v_{id} = wv_{id} + C_1 r_1 (p_{bid} - x_{id}) + C_2 r_2 (p_{gb} - x_{id}) \tag{1}$$

$$x_{id} = x_{id} + v_{id} \tag{2}$$

Where w is an inertia weight and it is typically selected in the interval [0, 1]. $C_1$ Is a cognition weight factor, $C_2$ is a social weight factor, $r_1$and $r_2$ are generated randomly in the interval [0, 1]. In order to improve the performance of standard PSO the inertia, the cognition, and the social weight factors should be modified.

The main idea for modifying the inertia weight is inspired from $\frac{1}{5}$ success rate of Schwefel [18], [19]. Here the concept of success rate is extended to while in each iteration the Pbest is achieved the better objective function value from its previous iteration. Thus, the success rate can be formulated in Eq. (3). Then simply the percentage of success rate is calculated by using Eq. (4).

$$\mathrm{Sucess}_{\mathrm{Rate}} = \begin{cases} 1 \text{ if ObjFcn } \left(\mathrm{pbest}_i^{\mathrm{iter}}\right) < ObjFcn \left(\mathrm{pbest}_i^{\mathrm{iter}-1}\right) \\ 0 \qquad\qquad \text{Otherwise} \end{cases} \tag{3}$$

$$P_{\mathrm{Succ}} = \frac{\sum_{i=1}^{n} \mathrm{Sucess}_{\mathrm{Rate}}(i,t)}{n} \tag{4}$$

**Table 1:** Test problems used in the experiments of this paper.

| Function name | Test function | D | Search space | $x^*$ | $f(x^*)$ |
|---|---|---|---|---|---|
| Rosenbrock | $f(x) = \sum_{i=1}^{D-1} [100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2$ | 30 | $[-5,10]^D$ | [0,...,0] | 0 |
| Rastrigin | $f(x) = \sum_{i=1}^{D} ( x_i^2 - 10\cos(2\pi x_i) + 10$ | 30 | $[-5.12,5.12]^D$ | [0,...,0] | 0 |
| Ackly | $f(x) = -20\exp\left(-0.2\sqrt{\frac{1}{30}\sum_{i=1}^{D} x_i^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos 2\pi x_i\right) + 20 + e$ | 30 | $[-32,32]^D$ | [0,...,0] | 0 |

Where n is the number of particles? Now the value of $P_{Succ}$ can vary in the interval [0, 1]. It is transparent that while the value of $P_{Succ}$ increased the pbest is far from the optimum point and the opposite of this situation is true. Therefore, the Inertia weight should be function of $P_{Succ}$. Due to frequent use of linear form of the inertia weight, the authors formulate the function of the inertia weight as a linear function of $P_{Succ}$.

$$w(iter) = (w_{max} - w_{min})P_{succ} + w_{min} \tag{5}$$

The range of the inertia weight $[w_{min}, w_{max}]$ is selected to be [0.2, 0.9].

In order to control the trade–off between exploitation and exploration property of the PSO algorithm the beginning of the optimization procedure, a large value for the cognitive weight and a small value for the social weight should be chosen. Hence, the exploration property of the PSO enhanced. By contrast, near the end stage of the PSO algorithm, a small cognitive weight and a large social weight should be selected for the convergence of the algorithm to the global optimum point [20]. Therefore, it is necessary to change the cognitive weight and social weight during the optimization process adaptively. The following formula for the APSO is proposed [20], [21]:

If $C_1^{final} < C_1^{initial}$

$$C_1 = \left(C_1^{final} - C_1^{initial}\right)\left(\frac{iter}{iter_{max}}\right) + C_1^{initial} \tag{6}$$

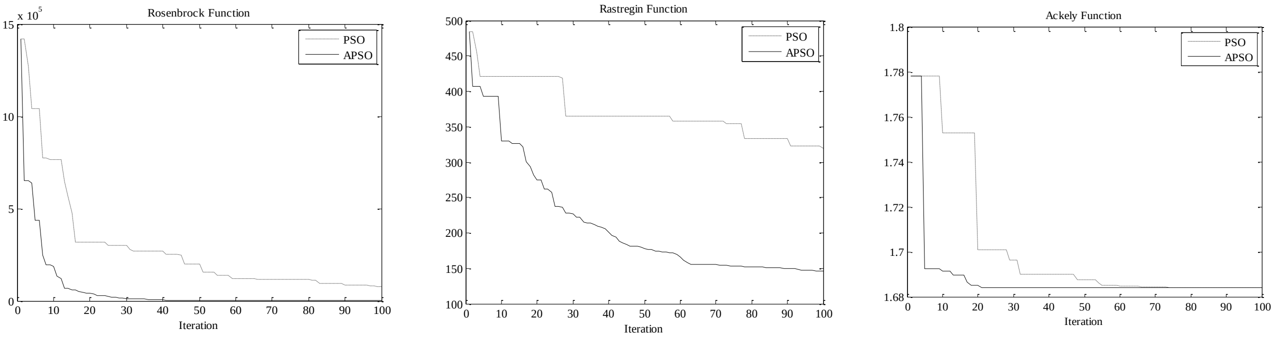
If $C_2^{final} > C_2^{initial}$

$$C_2 = \left(C_2^{final} - C_2^{initial}\right)\left(\frac{iter}{iter_{max}}\right) + C_2^{initial} \tag{7}$$

Where the superscripts "initial" and "final" indicate the initial and final values of the cognition weight and the social weight factor, respectively.

In this section, the performance of APSO is compared with standard PSO. Finally, APSO is applied to a real world optimization problem to show its performance in engineering problem.

The termination criterion of both PSO and APSO is determined by reaching a maximum iteration number. In this study, the maximum number of iterations and the number of particles are selected to be 50 and 20, respectively. In order to test and compare standard PSO and proposed APSO in this paper, three commonly used static benchmark functions are used. The test functions are used to investigate the convergence speed and solution quality of the PSO and APSO. Table 1 provides a detailed description of these problems. All the test functions are minimization problems. The first function (Rosenbrock) is unimodal function while the rest of the functions (Rastrigin and Ackly) are multimodal optimization problems. The dimension of the search space (D) is 30. For each test problem, $x*$ is the best solution to the test problem and $f(x*)$ represents the best achievable fitness for that function. Fig. 1 shows the results based on the final accuracy and the convergence speed over 100 iterations. These results demonstrate that APSO has a considerable higher performance in unimodal and multimodal optimization problems.



(a) (b) (c)

**Fig. 1:** A comparison between PSO algorithm and new APSO algorithm, a) Rosenbrock function, b) Rastregin function, c) Ackley function

# 3. Optimal design of T-S fuzzy system

T–S fuzzy systems are appropriate fuzzy rule–based systems for the approximation of many systems and function. In the T–S fuzzy systems, the consequents are taken to be either crisp numbers of linear functions of the inputs

$$\mathrm{R_i}: \text{IF x is } \mathrm{A_i} \text{ THEN } \mathrm{y_i} = \mathrm{a_i^T x + b_i, i = 1,2, \ldots, M} \tag{8}$$

Where $\mathrm{x \in R^n}$ is the input variable (antecedent) and $\mathrm{y_i \in R}$ is the output (consequent) of the i–Th rule$\mathrm{R_i}$. M Is the number of the rules. $\mathrm{A_i}$ Is the antecedent fuzzy set in the interval [0,1] in $\mathrm{R^n}$. the fuzzy antecedent in the T–S system is normally described by means of the product operator.

$$\mathrm{A_i(x)} = \prod_{\mathrm{j=1}}^{\mathrm{n}} \mathrm{\mu_{ij}(x_j)} \tag{9}$$

For the k–th input $\mathrm{x_k}$ the total output $\mathrm{y(k)}$ of the system is defined by the sum of the independence rules contributions:

$$\mathrm{y(k)} = \sum_{\mathrm{i=1}}^{\mathrm{M}} \mathrm{u_{ki} y_i(k)} \tag{10}$$

Where $\mathrm{u_{ki}}$ is the normalized degree of realization of the antecedent clause of rule $\mathrm{R_i}$ and is as follows:

$$\mathrm{u_{ki}} = \frac{\mathrm{A_i(x_k)}}{\sum_{\mathrm{i'=1}}^{\mathrm{M}} \mathrm{A_i x_k}} \tag{11}$$

By using well–known off–line training algorithms, a T–S fuzzy system is obtained from the collected input–output data. Due to the limited off–lines data at hand over a wide operating range of the plant, achieve a global T–S fuzzy system is not easy. While for successful control of non–linear a global T–S system is not essentially needed and a suitable control signal can be generated from the input–output data. In this study, FCM is used for extracting initial fuzzy rules. The number of rules is equal to the number of FCM clusters. Since choosing the optimal number of clusters in the FCM is yet rewarding problem an adaptive PSO, is employed to find the optimal number of clusters. After the maximum number of iterations is reached in APSO, global best particle represents an optimal solution which consists of the number of rules the T–S fuzzy controller.

In the rule extraction approaches based on FCM, each cluster basically categorizes a region in the data space which contains a satisfactory mass of data to support the existence of a fuzzy input–output relationship. Because a rule is generated only where there is a cluster of data, the resulting rules are scattered in the input space rather than placed according to grid–like partitions in the input space. This fundamental property of clustering–based rule extraction approaches assists avoid combinatorial explosion of rules with increasing dimension of the input space, i.e. MISO and MIMO systems.

The initial fuzzy model is not optimal because the antecedent part of the rules was achieved only based on partitioning the input–output space. Thus, the model is not able to portray the relation between input and output adequately. Several methods such as gradient descent and least mean squares by taking a measure of the output error have been used for adjusting the parameters of the fuzzy system [5]. In this study, the RLS method due to its higher speed and lower memory occupation compared to other optimizing TSK fuzzy systems is used. The summation of the number of rules (#Rules) and MSE is defined in equation (13) to use as the objective function for evaluating the performance index in the optimal designing of T–S fuzzy system.

$$\mathrm{ObjFcn} = \alpha_1.\mathrm{MSE} + \alpha_2.\#\mathrm{Rules} \tag{12}$$

Where the MSE formulates as follows:

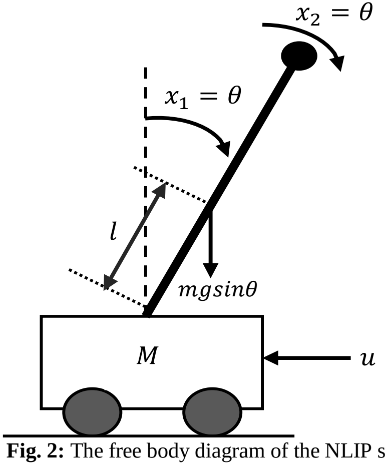


**Fig. 2:** The free body diagram of the NLIP system.

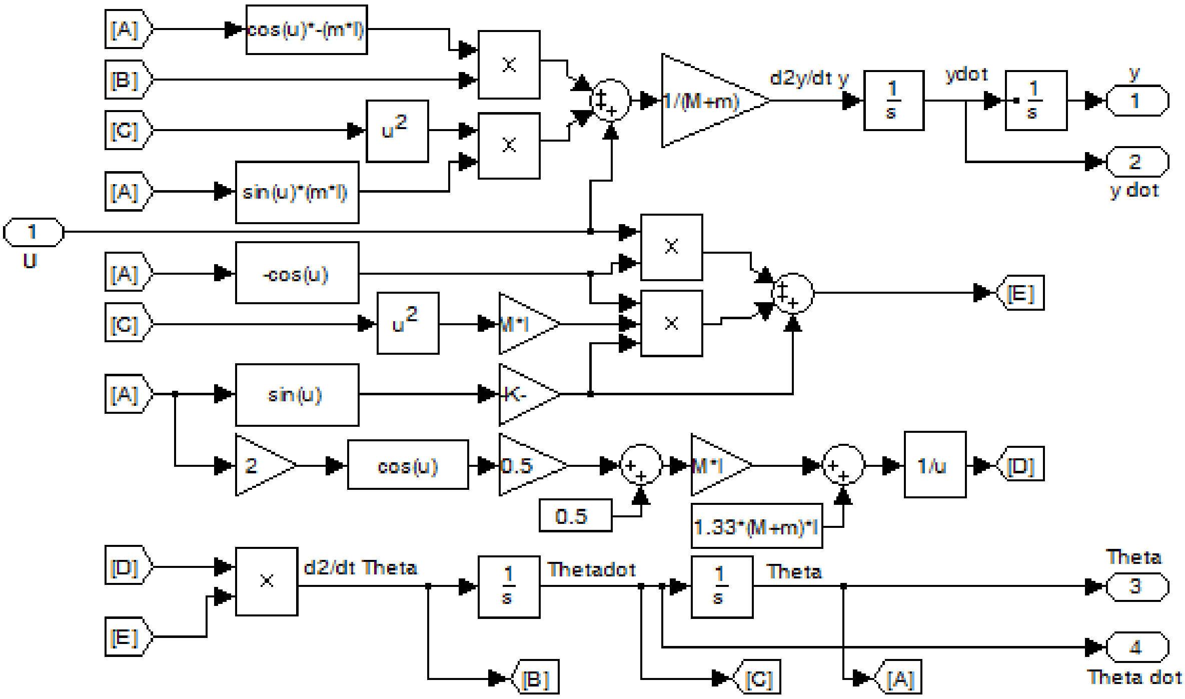


**Fig. 3:** The Simulink model of NLIP

$$\mathrm{MSE} = \frac{1}{K}\sum(y - y_d)^2 \tag{13}$$

Where, y is the output of T–S fuzzy system, $y_d$ is desired output, and K is the number of points.

# 4. Simulation

Non–Linear Inverted Pendulum (NLIP) system is a classical problem in the field of nonlinear control theory which can be observed in the many real world control problems [22, 23]. The NLIP system is an open–loop unstable system, that is, while the NLIP system simulated the pendulum drops down as fast as possible due to the gravity of the earth. Therefore, a problem encountered in designing T–S fuzzy is how to collect sufficient informative input–output data from the NLIP system. The output data from the unstable system does not show enough information or the dynamics of the system. A Feedback linearization control method is performed to stabilize the system before the rule extraction method can take place.

## 4.1. Mathematical formulation

The NLIP system consists of an inverted pole placing on a cart, which is moving in the horizontal direction, i.e. x. Fig. 2 shows the free body diagram of the NLIP system. The NLIP system has two degrees of freedom. They are horizontal displacement of the cart, i.e. x, and the angle between pendulum and its upright position, i.e. θ. The friction–free non–linear dynamical equations of an inverted pendulum system are as follows:

$$\ddot{x} = \frac{m\left(l\dot{\theta}\,\sin\theta - \frac{3}{8}g\,\sin2\theta\right) - f\dot{x} + u}{M + m(1 - \frac{3}{4}\cos^2\theta)} \tag{14}$$

$$\ddot{\theta} = \frac{3}{4l}(g\,\sin\theta - \ddot{x}\,\cos\theta) \tag{15}$$

WhereM, m are the masses of the cart and the pole, respectively, g is the gravitational force, l is half length of the pole, and f is applied force signal (control signal). In addition, the states of NLIP system are assumed to be $x_1 = x, x_2 = \dot{x}, x_3 = \theta$, $x_4 = \dot{\theta}$ for the state space representation. Fig. 3 shows the Simulink model of NLIP system. The open–loop of NLIP does not sufficiently generate informative input–output data for designing a T–S fuzzy controller, because the NLIP falls over quickly. Therefore, for gathering adequate input–output data the NLIP system must be stabilized.

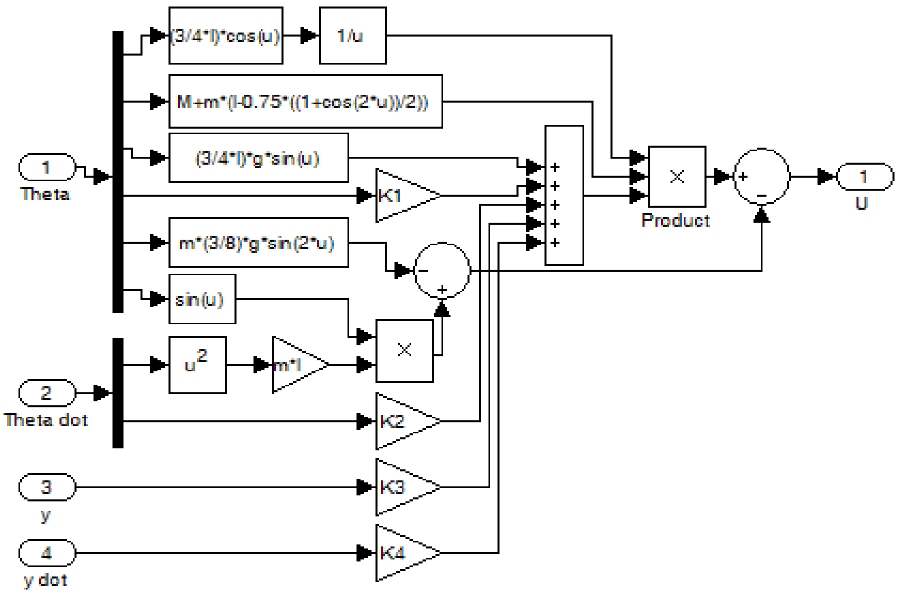


**Fig. 4:** The certainty equivalence control law

## 4.2. Design of feedback control

Feedback linearization is a celebrated control design method for nonlinear systems which attracted lots of research in the last three decades [24], [25], [26]. A feedback linearization controller is designed to stabilize control
Of the NLIP system. The main idea in feedback linearization approaches is to algebraically transform nonlinear systems dynamics into (fully or partially) linear ones. This aim is realized by eliminating the nonlinear part of the NLIP system. Hence, the closed loop system is more linear and the NLIP can be stable by positioning the closed loop poles in the stable region, i.e. in the open–left half–plane.
The four equations (16–20) are entered into the main equation to prepare the adequate control law, certainty equivalence control law, for stabilizing the NLIP system.

$$h_1 = \frac{3}{4l} g \sin\theta \tag{16}$$

$$h_2 = \frac{3}{4l} \cos\theta \tag{17}$$

$$f_1 = m\left(l \sin\theta\, \dot{\theta}^2 - \frac{3}{8} g \sin 2\theta\right) - f\dot{x} \tag{18}$$

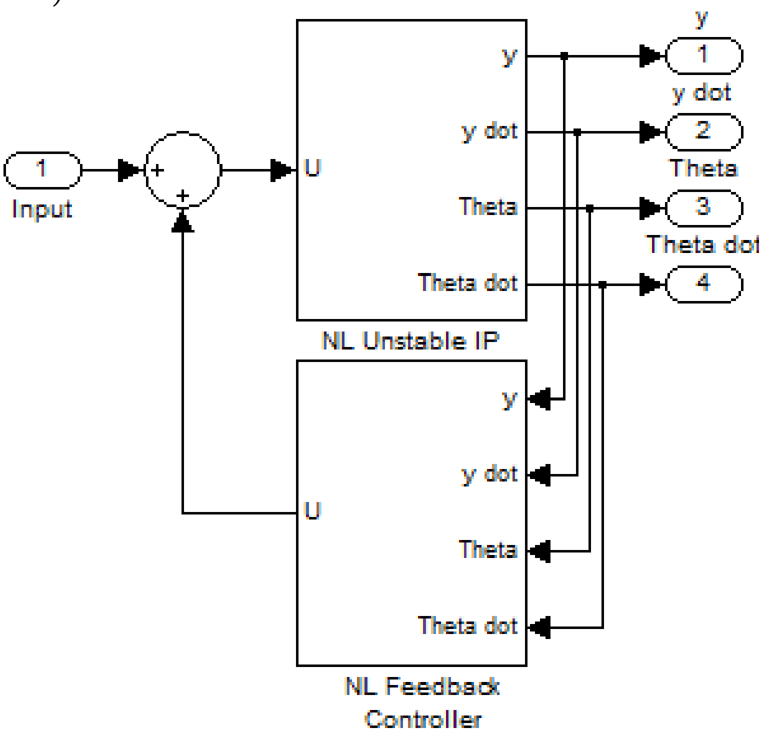


**Fig. 5:** Stable NLIP with the feedback linearization controller

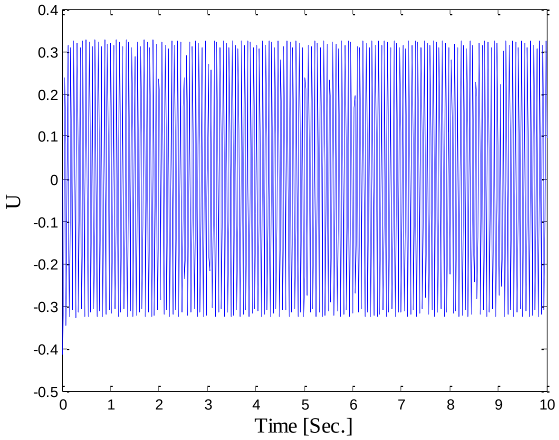


**Fig. 6:** The feedback linearization controller output

**Table 2:** The validation results of fuzzy controller

| Method | SSE | MAE | MSE | # Rules |
|---|---|---|---|---|
| Fuzzy controller | 61.2424 | 0.2502 | 0.0975 | 3 |

$$f_2 = M + m\left(1 - \frac{3}{4}\cos^2\theta\right) \quad (19)$$

$$u = \frac{f_2}{h_2}\left(h_1 + k_1(\theta - \theta_d) + k_2\dot{\theta} + k_3(x - x_d) + k_4\dot{x}\right) - f_1 \quad (20)$$

where in Eq. 20, $k_1 = 25, k_2 = 10, k_3 = 1, k_4 = 2.6$. The model of the certainty equivalence control law is presented in Fig. 4 and the Simulink model of the NLIP with its stable feedback linearization controller is shown in Fig 5. The inputs to this controller are the four–output states of the NLIP system. The accurate magnitude of control signal is derived from the control law to keep the pendulum stable. Fig. 6 shows the control signal derived by the feedback linearization controller. Now, by using the feedback controller, the input–output data will contain more information and they become proper for designing a T–S fuzzy controller based on input–output data.

## 4.3. Validation criteria

The efficiency of the fuzzy controller is evaluated by comparing the resultant control signal with output of non–linear feedback controller. Three error measurement indices, i.e. Sum of the Squared Errors (SSE), Mean Absolute Error (MAE), and MSE are used to verify the validation of the fuzzy controller.

SSE is used to measure what the difference between each error and its group mean.

$$SSE = \sum(e_i - \bar{e})^2 \quad (21)$$

Where $e_i$ is the value of the i–th error and $\bar{e}$ is the mean of all the errors.

MAE is a quantity used to measure how close the output is to the actual output. MAE is formulated as follows:

$$MAE = \frac{1}{K}\sum|y - y_d| \;\; \frac{1}{K}\sum|e_i| = \quad (22)$$

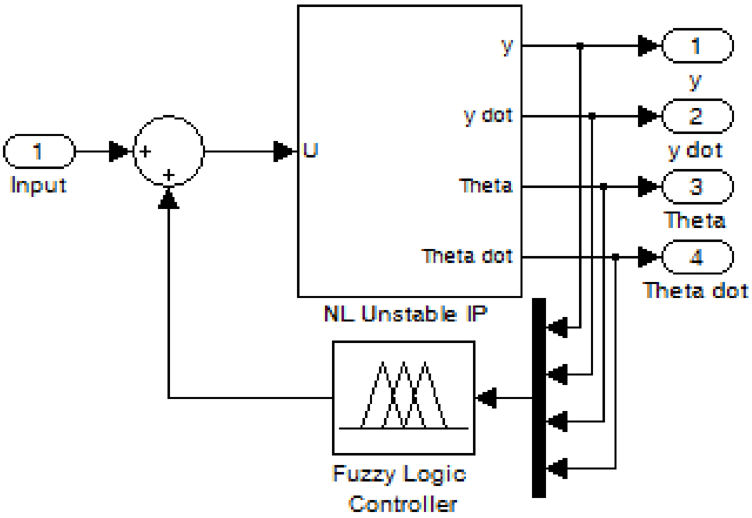


**Fig. 7:** Stable fuzzy controller NLIP system

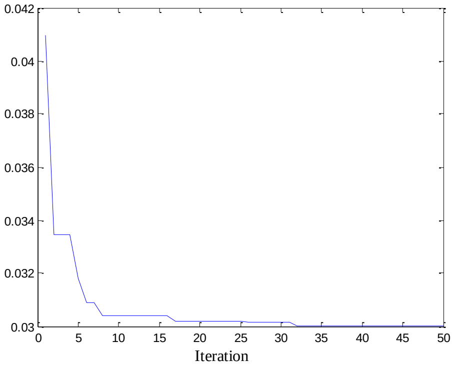
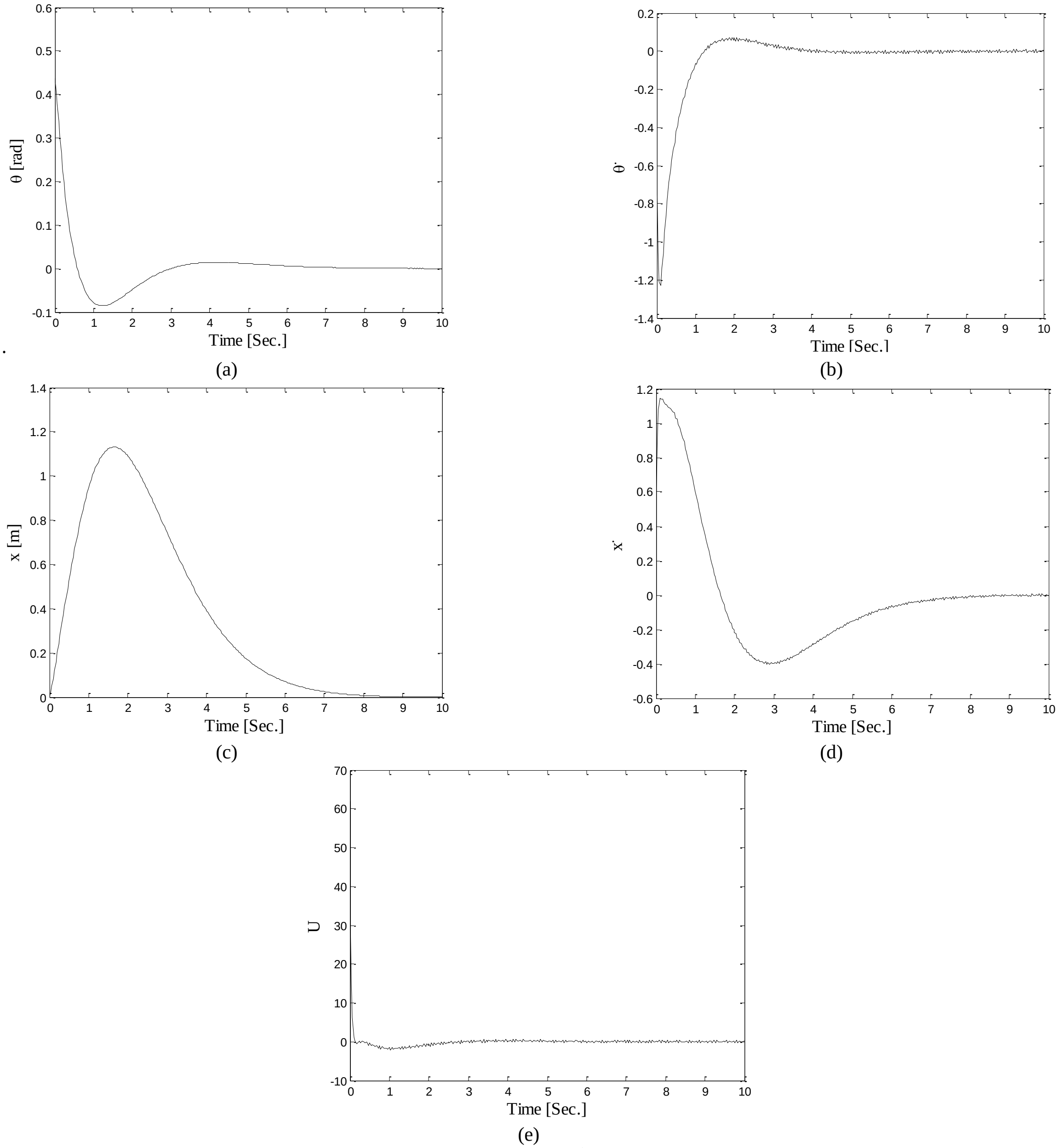


**Fig. 8:** The convergence of objective function via APSO

where ythe output of fuzzy controller is, $y_d$ is the output of non–linear feedback controller, and K is the total number of data. In brief, these measurement criteria will verify an accuracy fuzzy controller, if the values of SSE, MAE, and MSE are close to zero. Table 2 presents the validation results.



(a)

(b)

(c)

(d)

(e)

**Fig. 9:** The simulation reults, a) the angle (θ) of pendulum, b) the angle change (θ̇) of pendulum, c) the position (x) of cart, d) the position change (ẋ) of the cart, and e) control signal of fuzzy controller

## 4.4. The simulation results of optimal T–S fuzzy controller

In order to demonstrate the validity of the proposed method, the NLIP system is simulated. The NLIP parameters are assigned the following values, $\mathrm{M} = 1.2\,\mathrm{Kg}$, $\mathrm{m} = 0.5\,\mathrm{Kg}$, , $\mathrm{l} = 0.3\,\mathrm{m}$, $\mathrm{g} = 9.81\,\mathrm{m/s}$, $\mathrm{k}_1 = 25$, $\mathrm{k}_2 = 10$, $\mathrm{k}_3 = 1$, $\mathrm{k}_4 = 2.6$, $\mathrm{x_d} = 0\,\mathrm{m}$, and $\theta_\mathrm{d} = 0\,\mathrm{rad}$. The MATLAB software (R2008b) is employed in the simulation test. The T–S fuzzy controller is applied to the NLIP system. The results demonstrate that the proposed controller is stable and can properly control the NLIP system as well as the non–linear feedback linearization control method. The Simulink model of stable T–S fuzzy controller is drawn in Fig. 7.

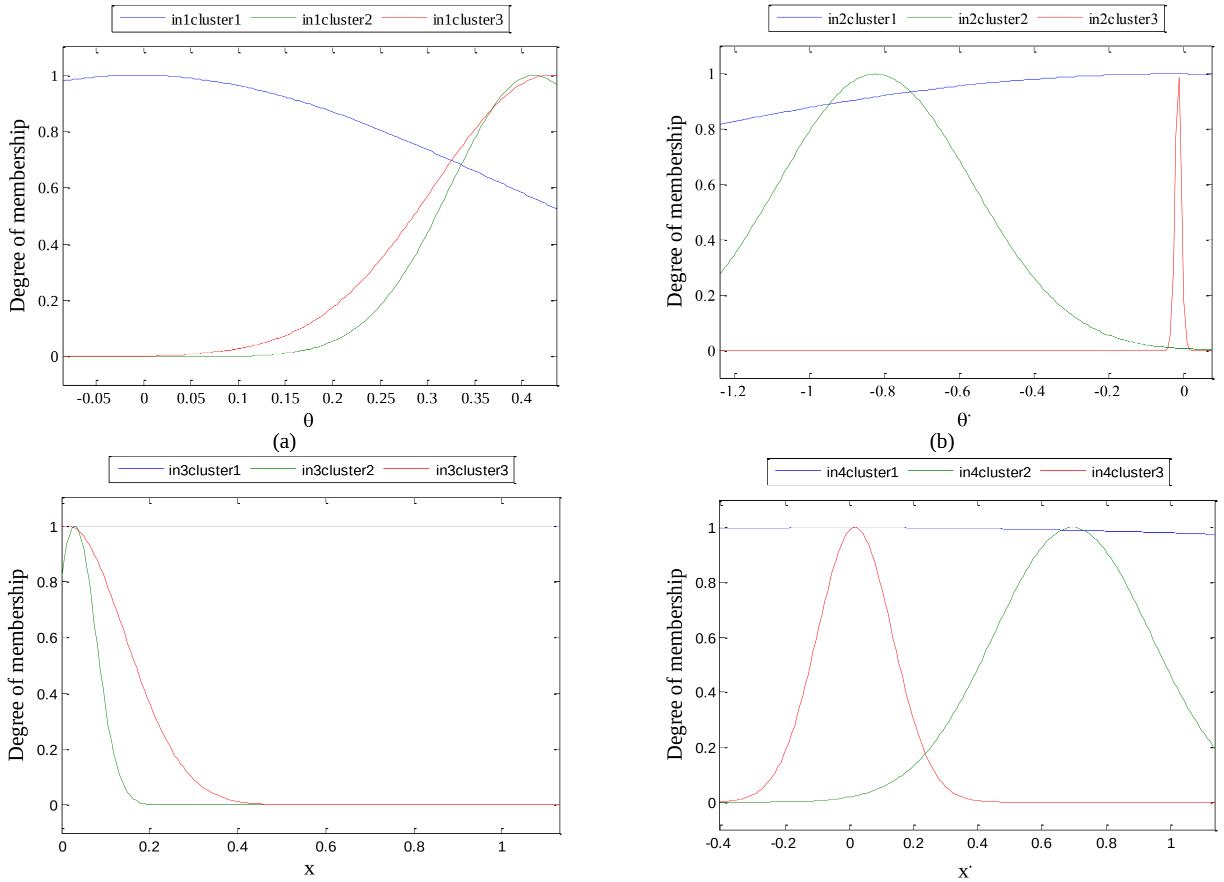
In the objective function y is considered as output of T–S fuzzy controller with different numbers of rules and $\mathrm{y_d}$ is the output of the non–linear feedback linearization. The convergence of the objective function versus the numbers of iterations during the procedure of APSO is shown in Fig. 8. The parameters of the APSO based on trial–and–error (the selected values are the best ones among several runs) is set as follows: $\mathrm{C}_1^{\mathrm{initial}} = 0.7$, $\mathrm{C}_1^{\mathrm{final}} = 1.6$, $\mathrm{C}_2^{\mathrm{initial}} = 0.9$, $\mathrm{C}_2^{\mathrm{final}} = 1.7$. The values of $\mathrm{a}_1$ and $\mathrm{a}_2$ are chosen to be 0.8 and 0.2 for this study. The obtained three optimal rules are as follow:

**Rule$_1$**: **IF** ($\theta$ is in1cluster1) and ($\dot{\theta}$ is in2cluster1) and (x is in3cluster1) and ($\dot{\mathrm{x}}$ is in4cluster1) **THEN** $\mathrm{y}= 141.9\,\theta + 51.04\,\dot{\theta} + 5.251\,\mathrm{x} + 13.42\,\dot{\mathrm{x}} -0.04644$

**Rule$_2$**: **IF** ($\theta$ is in1cluster1) and ($\dot{\theta}$ is in2cluster2) and (x is in3cluster2) and ($\dot{\mathrm{x}}$ is in4cluster2) **THEN** $\mathrm{y}= 58.99\,\theta - 1.575\,\dot{\theta} - 39.35\,\mathrm{x} - 49.28\,\dot{\mathrm{x}} + 34.87$

**Rule$_3$**: **IF** ($\theta$ is in1cluster3) and ($\dot{\theta}$ is in2cluster3) and (x is in3cluster3) and ($\dot{\mathrm{x}}$ is in4cluster3) **THEN** $\mathrm{y}= 25.23\theta + 24.85\,\dot{\theta} + 2.446\,\mathrm{x} - 17.71\,\dot{\mathrm{x}} + 50.96$

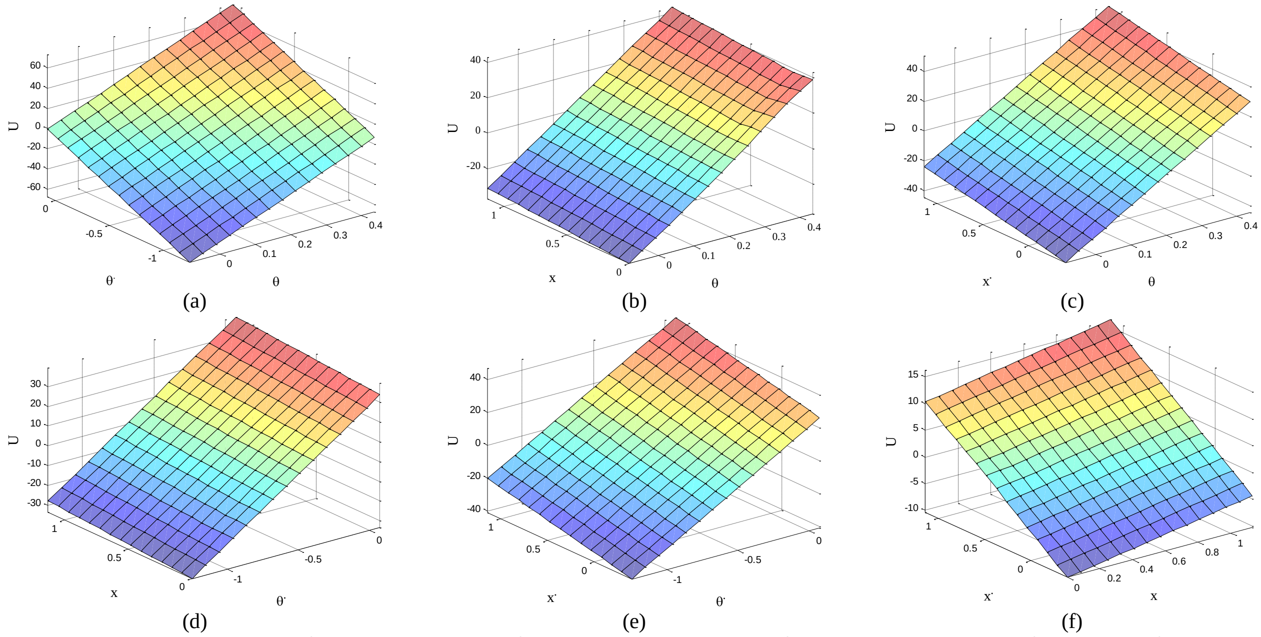
Fig. 9 shows the position and position change of the cart and the angle and angle change of pendulum to its up straight.
By doing so, the total number of rules in T–S fuzzy inference controller is dramatically reduced. Reduction in the number of rules has an acceptable error range. As a result of reduction in the number of rules, the memory requirement and model complexity are decreased. Fig. 10 represents the fuzzy membership function of the optimal T–S fuzzy controller. In Fig. 11, the input-output relations of T-S fuzzy controller are shown.



(a) (b)

(c) (d)

**Fig. 10:** The membership functions of fuzzy T–S controller, a) the angle ($\theta$) of pendulum, b) the angle change ($\dot{\theta}$) of pendulum, c) the position (x) of cart, and d) the position change ($\dot{\mathrm{x}}$) of the cart



(a) (b) (c)

(d) (e) (f)

**Fig. 11:** The input-output relation in fuzzy T–S controller, a) first input and second input, b) first input and third input, c) first input and four input, d) second input and third input, e) second input and third input, and f) third input and fourth input

# 5. Conclusion

An optimal T–S fuzzy system for controlling a nonlinear inverted pendulum was proposed. Since complex non–linear systems generate large fuzzy rule based, the FCM is used for extracting an optimal and compact FIS. To achieve that aim, a new adaptive PSO was also presented to find the optimal number of rules. The simulation results demonstrate that the proposed method can provide optimal model structures for fuzzy systems and has low complexity while the error remain in an acceptable range.

# References

[1] Y. Diao and K. M. Passino, “Adaptive Neural/Fuzzy Control for Interpolated Nonlinear Systems”, *IEEE Trans. on fuzzy systems*, Vol. 10, No. 5, pp. 583–595, Oct. 2002.

[2] G. Feng, “A survey on analysis and design of model–based fuzzy control systems”, *IEEE Trans.* on fuzzy System, Vol. 14, No. 5, pp. 676–697, oct. 2006.

[3] T. Takagi, M. Sugeno, “Fuzzy identification of systems and its applications to modeling and control”, *IEEE Trans. Systems Man Cybernet*., Vol. 15, pp. 116–132, 1985.

[4] R. Qi, M. A. Brdys, “Stable indirect adaptive control based on discrete–time T–S fuzzy model”, *Fuzzy Sets and Systems*, Vol. 159, pp. 900–925, 2008.

[5] W. L. Xin, *a Course in Fuzzy Systems and Control*. Englewood Cliffs, NJ: Prentice–Hall, 1996.

[6] D. A. Linkens, M.Y. Chen, “Input selection and partition validation for fuzzy modeling using neural network”, *Fuzzy Sets and Systems*, Vol.107, pp. 299–308, 1999.

[7] Z. Fang, N. Song, L. Wang, “Design and Implementation of a Novel Fuzzy Controller with DSP for Rotary Inverted Pendulum”, *IEEE Int. Conf. on Control and Decision*, pp.6122–6127, 2009.

[8] M. Setnes, “Supervised fuzzy clustering for rule extraction”, *IEEE Trans. fuzzy systems*, Vol. 8, No. 4, pp. 416–424, Aug. 2000.

[9] S. Chiu, “Extracting fuzzy rules from data for function approximation and pattern classification”, *Fuzzy Information Engineering: A Guided Tour of Application. D. Dubois, H. Prade and R. Yager (eds)*, John Wiley Sons, 1997.

[10] F.L. Chung and T. Lee, “A fuzzy learning method for membership function estimation and pattern classification”, *IEEE Int. Conf. on Fuzzy Systems*, pp. 426–431, 1994

[11] C. Hung and L. Huang, “Extracting Rules from Optimal Clusters of Self–Organizing Maps”, *Int. Conf. on Computer Modeling and Simulation*, pp., 2010.

[12] C. F. Juang, C. M. Hsiao, and C. H. Hsu, “Hierarchical Cluster–Based Multispecies Particle–Swarm Optimization for Fuzzy–System Optimization”, *IEEE Trans. on fuzzy systems*, Vol. 18, No. 1, pp. 14–26, Feb. 2010.

[13] G. E. Tsekouras, “On the use of the weighted fuzzy c–means in fuzzy modeling”, *Advances in Engineering Software*, Vol. 36, pp. 287–300, 2005.

[14] M. Setnes, “Supervised fuzzy clustering for rule extraction”, *IEEE Trans. fuzzy systems*, Vol. 8, No. 4, pp. 416–424, aug 2000.

[15] M. Zhang, W. Zhang, H. Sicotte, and P. Yang, “A New Validity Measure for a Correlation–Based Fuzzy C–means Clustering Algorithm”, *IEEE Conf. engineering in medicine and society*, pp. 3865–3868, 2009.

[16] F. Beloufa, M. A. Chikh, “Automatic fuzzy rules–based generation using modified particle swarm optimization”, *Int. Conf. modelling and Implementation of Complex Systems*, 2012.

[17] J. Kennedy, R. C. Eberhart, *Swarm Intelligence*, USA: Academic Press, 2001.

[18] Schwefel, H. P. Paul, *Evolution and Optimum Seeking*, John Wiley & Sons, 1993.

[19] H.G. Beyer, H.P. Schwefel, “Evolution Strategies: A Comprehensive Introduction”, *Natural Computing*, Vol. 1, pp. 3–52, 2002.

[20] X. Yuan, L. Wang, Y. Yuan, “Application of enhanced PSO approach to optimal scheduling of hydro system”, *Energy Conversion and Management*, Vol.49, pp. 2966–2972, 2008.

[21] Y. Wang, B. Li, T. Weise, J. Wang, B. Yuan, and Q. Tian, “Self–adaptive learning based particle swarm optimization”, *Information Sciences*, Vol. 181 , pp.4515–4538, 2011.

[22] H. K. Khalil, *Nonlinear Systems*, 2nd ed. Englewood Cliffs, NJ: Prentice–Hall, 1996.

[23] W. Junfeng and Z. Wang, “Research on Fuzzy Control of Inverted Pendulum”, *IEEE Int. Conf. Instrumentation Measurement Computer Communication and Control*, pp. 868–71, 2011.

[24] R. Marino and P. Tomei, “Globally adaptive output–feedback control on nonlinear systems, Part I: Linear parameterization”, *IEEE Trans. Automat. Contr.*, Vol. 38, pp. 17–32, 1993.

[25] R. Marino and P. Tomei, “Globally adaptive output–feedback control on nonlinear systems, Part I: Linear parameterization”, *IEEE Trans. Automat. Contr.*, Vol. 38, pp. 17–32, 1993.

[26] J. J. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, NJ: Prentice–Hall, 1991.