International Conference on Control, Engineering & Information Technology (CEIT'13)
Proceedings Engineering & Technology - Vol.4, pp. 55-61, 2013
Copyright - IPCO

Nonlinear system identification using allied fuzzy c-means algorithm and particle swarm optimization

TROUDI Ahmed[#1], BOUZBIDA Mohamed[*2], HOUCINE Lassad[*3], CHAARI Abdelkader[#4]

University of Tunisia, Higher School of Sciences and Techniques of Tunis (Research unit C3S)
5 Avenue Taha Hussein Tunis 10008, Tunisia
[1]troudi.ahmed@yahoo.fr
[4]assil.chaari@esstt.rnu.tn
University of Tunisia, Higher School of Sciences and Techniques of Tunis (Research unit C3S)
5 Avenue Taha Hussein Tunis 10008, Tunisia
[2]bouzbida_mohamed@hotmail.fr
[3]houcinelassad@yahoo.fr

***Abstract*— several clustering algorithms have been proposed in literature to identify the parameters involved in the Takagi-Sugeno fuzzy model, we can quote as an example the Fuzzy C-Means algorithm (FCM), the Possibilistic C-Means algorithm (PCM) and the Allied Fuzzy C-Means algorithm (AFCM). In this paper a new clustering algorithm called AFCM-PSO is proposed. This algorithm is a combination between Allied Fuzzy C-Means (AFCM) algorithm and Particle Swarm Optimization (PSO). Indeed, AFCM algorithm can deal with noisy better than FCM and PCM but it does not solve the problems of convergence and the computation time. The AFCM-PSO algorithm is more robust than FCM, PCM, and AFCM algorithms. The effectiveness of this algorithm is tested on a nonlinear system and on an electro-hydraulic system.**

***Keywords*— Fuzzy clustering, fuzzy c-means, possibilistic c-means, allied fuzzy c-means, fuzzy identification.**

I. INTRODUCTION

Fuzzy clustering algorithms are one of the best techniques allowing the identification of the parameter intervening in the TS fuzzy model. In the literature, many clustering algorithms have been proposed [5], [6], [7], [8], [9], [12], [13], [14], [15]; we can quote as an example the Fuzzy C-Mean algorithm (FCM) which is proposed by Dunn [4] in 1974 and generalized by Bezdek [3] in 1981. However, FCM algorithm is sensitive to noises or outliers because the probabilistic constraint used by FCM. To overcome these disadvantages Krishnapuram and Keller have proposed in 1996 another algorithm called the Possibilistic C-Means (PCM) algorithm [10] by abandoning the constraint of FCM and constructing a novel objective function. The PCM can deal with noisy data better than FCM. In 2006 a novel fuzzy clustering algorithm, called Allied Fuzzy C-Means (AFCM) clustering [18], has been proposed by Wu and Zhou to deal with noisy data. AFCM can produce memberships and possibilities simultaneously and it overcomes the noise sensitivity shortcoming of FCM and the coincident clusters problem of Possibilistic C-Means (PCM) [6]. The AFCM clustering algorithm becomes more robust than FCM and PCM algorithms; however, this algorithm does not solve the problems of convergence and the computation time. To overcome this problem, several solutions have been proposed in the literature. The idea of these techniques is to combine the clustering algorithms with other optimization techniques such as genetic algorithm [2] and particle swarm optimization [1], [11], [17]. Furthermore, we are presenting in this paper another algorithm called AFCM-PSO for the identification of highly nonlinear systems and operating in a stochastic environment. These approaches make it possible to combine the AFCM algorithm with the particle swarm optimization (PSO) algorithm. Indeed the particle swarm optimization is a global optimization technique. Thus the incorporation of local research capacity of clustering algorithms and the global optimization ability of PSO algorithm can give very good results. The effectiveness of this algorithm is tested on a nonlinear system and on a level control system.

This paper is organized as follows:

In section 2, we introduce the TS fuzzy model. The AFCM algorithm is introduced in section 3. The PSO for optimization of the TS fuzzy model is presented in section 4. The AFCM-PSO algorithm is introduced in section 5. The simulations results are introduced in section 6. The validation results are presented in section 7. And finally section 8 conclude the paper.

II. TAKAGI-SUGENO FUZZY MODEL

The TS fuzzy models consist of linguistic if-then rules that can be expressed by the following form.

$$R_i \ : \text{If } x_k \text{ is } A_i \text{ then } y_i = a_i^T x_k + b_i \qquad (1)$$

Where k is the current iteration number,

The ″ if ″ rule function defines the premise part, while the ″ then ″ rule function constitutes the consequent part of the TS fuzzy model. $A_i \in R^n$ is a multidimensional antecedent fuzzy set, defined by its membership function $\mu_{A_i}(x_k)$.

$x_k = \left[x_{k_1}, x_{k_2}, ..., x_{k_n}\right] \in R^n$, is the input vector of the premise; $a_i \in R^n$, $b_i \in R$: are the polynomial coefficients that form the consequent parameters of the i[th] rules, and $i = 1,...,c$ (c : denotes the number of rules in the rule base). $y_i \in R$: Is the rule output variable.

The output of the general nonlinear system is calculated as the average of output corresponding to the rules multiplied by the degree of fulfillment of the antecedent γ_i of the form:

$$\hat{y} = \frac{\sum_{i=1}^{c} \gamma_i(x_k) y_i}{\sum_{i=1}^{c} \gamma_i(x_k)} \quad (2)$$

With:

$$\gamma_i = \mu_{i1}(x_{k_1}).\mu_{i2}(x_{k_2}). \dots .\mu_{in}(x_{k_n}) \quad (3)$$

Introduce λ_i : the degree of achievement standard described by the following expression:

$$\lambda_i = \frac{\gamma_i(x_k)}{\sum_{i=1}^{c} \gamma_i(x_k)} \quad (4)$$

The estimated output of the Takagi-Sugeno fuzzy model can be expressed by:

$$\hat{y} = \sum_{i=1}^{c} \lambda_i(x_k)\left[a_i^T x_k + b_i\right] \quad (5)$$

III. ALLIED FUZZY C-MEANS ALGORITHM

The Possibilistic C-Means algorithm (PCM) overcomes the problem of sensitivity to the noise better than FCM but it still sometimes generates identical clusters. The AFCM algorithm can produce memberships and possibilities simultaneously and it overcomes the noise sensitivity shortcoming of FCM algorithm and the coincident clusters problem of Possibilistic C-Means (PCM) akgorithm. AFCM algorithm combines the benefits of FCM algorithm and PCM algorithm and it overcomes theirs defects. Given an unlabeled data set $X = \{x_1, x_2, ..., x_N\}$, the AFCM algorithm finds the partition of data vectors $x_k \in X$ into c clusters. The certainty of assignment of a vector $x_k \in X$ for the various clusters is measured by membership functions $\mu_i(x_k) = \mu_{ik}$:

$$\text{With} \begin{cases} 0 \le \mu_{ik} \le 1 \;\forall i, k \\ 0 < \sum_{k=1}^{N} \mu_{ik} < N \end{cases} \quad (6)$$

The AFCM algorithm is developed to solve the minimization of the following criterion:

$$J(U,t,V) = \sum_{i=1}^{c} \sum_{k=1}^{N} \left(a\mu_{ik}^m + bt_{ik}\right) D_{ik}^2 + \sum_{i=1}^{C} \eta_i \sum_{k=1}^{N} \left(t_{ik} \log\left(t_{ik}\right) - t_{ik}\right) \quad (7)$$

Where N is the total number of observations, $D_{ik}^2 = \left\|x_k - v_i\right\|^2$ is the Euclidian distance from simple point. x_k to the cluster center v_i , c is the number of clusters, $U = \left[\mu_{ik}\right]$ is a $C \times N$ matrix , denoted a fuzzy partition matrix, t_{ik} is the typicality of x_k in class i. $V = \left\{v_1, v_2, ..., v_c\right\}$; $v_i \in R^n$; $1 \le i \le c$ a set of c cluster centers, m is a weighting exponent, $m > 1$ and η_i is a suitable positive numbers. It should be noted that the first term demands that the distance between x_k to v_i be as low as possible, however the second term forces μ_{ik} to be as large as possible. In the following, we present the stages of the AFCM algorithm to determine the cluster centers V , the typicality matrix t_{ik} and the fuzzy partition matrix U .

Step 1: Initialization l=0

Calculate the fuzzy partition matrix, the cluster centers from the FCM algorithm

Choose the number of clusters c, $1 < c < N$.

Choose weighting exponent m , $m > 1$.

Choose a and $b>0$

Choose the tolerance of the end of the algorithm δ , $\delta > 0$

Step 2 : Compute η_i

$$\eta_i = k \frac{\sum_{k=1}^{N} \mu_{ik}^m D_{ik}^2}{\sum_{k=1}^{N} \mu_{ik}^m} \quad , k > 0 \quad (8)$$

Repeat l=l+1:

Step 4: Compute the cluster centers v_i :

$$v_i = \frac{\sum_{k=1}^{N} (a\mu_{ik}^m + bt_{ik}) x_k}{\sum_{k=1}^{N} (a\mu_{ik}^m + bt_{ik})}, \forall i \quad (9)$$

Step 5: Update the fuzzy partition matrix U

$$\mu_{ik} = \left[\sum_{j=1}^{C} \left(\frac{D_{ik}}{D_{jk}}\right)^{\frac{2}{m-1}}\right]^{-1} \quad (10)$$

Step 5: Update the typicality matrix t_{ik}

$$t_{ik} = \exp\left(-\frac{bD_{ik}^{2}}{\eta_i}\right), \forall i, k \quad (11)$$

Step 6: if $\left\| U^{(l)} - U^{(l-1)} \right\| > \delta$, return to step 3, if not stop.

IV. PSO FOR OPTIMIZATION OF THE T-S FUZZY MODEL

IV.1. FUNDAMENTALS OF THE PSO APPROACH

The Particle Swarm Optimization (PSO) is a stochastic optimization technique, it was originally developed by Kennedy and Eberhart (1995), it uses a population of candidates solution to develop an optimal solution of the problem. The degree of optimality is measured by a fitness function (Eberhart and Kennedy, 1995). Similar to genetic algorithm [2] the Particle Swarm Optimization (PSO) is an optimization technique based on a population where each member of population is considered as a particle, and each particle represent a solution of the current problem [11], [17]. Each particle in the algorithm is associated to a randomized velocity which enables it to move in the research space. The PSO algorithm does not have operators, such as crossover and mutation as in the genetic algorithm, in fact the PSO algorithm does not implements the survival of the suited individual, but it implements the simulation of social behavior individuals. From the algorithm, a swarm is randomly distributed in the search space, each particle also having a position and a random velocity. Then, at each time instant, each particle is able to evaluate the quality of its position and to keep in memory its best performance. That is to say the best position it has achieved so far. Each particle in the PSO is able to query with a number of these neighbors and get each of them its own best solution noted pbest, and then choose the best of the best performances in its possession noted gbest. The optimization procedure of PSO consists of each time instant to change the velocity of each particle flying the values of pbest and gbest. Acceleration is weighted by random terms, with separate random numbers being generated by acceleration toward of pbest and gbest locations, respectively. The implementation procedure of the PSO algorithm is summarized by the following steps:

Step 1: Initialize a population of particles with random positions and velocities using a uniform probability distribution.

Step 2: Compute the fitness value of each particle.

Step 3: Compare the fitness of each particle's with pbest, if the current value is better than pbest, then set the pbest value equal to the current value.

Step 4: Compare the fitness of each particle's with *gbest*, if the current value is better than *gbest,* then set the *gbest* value equal to the current value.

Step 5: update the position and velocity of the particle according to the "(12)" and "(13)".

$$v_d(k+1) = w\, v_d(k) + \rho_1\left(p_d(k) - x_d(k)\right) + \rho_2\left(p_g - x_d(k)\right) \quad (12)$$

$$x_d(k+1) = x_d(k) + v_d(k+1) \quad (13)$$

Where k is the current iteration number, $x_d = \left[x_{d_1}, x_{d_2}, x_{d_3}, ..., x_{d_k}, ..., x_{d_N}\right]^T$ represents the position of the i[th] particle, $v_d = \left[v_{d_1}, v_{d_2}, v_{d_3}, ..., v_{d_k}, ..., v_{d_N}\right]^T$ represents the velocity of the i[th] particle and $p_d = \left[p_{d_1}, p_{d_2}, ..., p_{d_N}\right]^T$, represents the best previous position (the position of which can give the best fitness value) of the i[th] particle. Index g: represents the index of the best particle in the population who can provide the best solution to the problem. ρ_1 and ρ_2 : represents two random variables defined as follows:

$$\begin{cases} \rho_1 = r_1 c_1 \\ \rho_2 = r_2 c_2 \end{cases}$$

r_1 and r_2 are two random variables between 0 and 1, c_1 and c_2 are two positive constants satisfying the following relationship: $c_2 + c_1 \leq 4$

w : represents the factor of inertia proposed by Shi and Eberhart. This factor sets the ability to explore each particle which aims at improving the convergence of the method. Note that the size of this factor directly influences the size of the search space. Shi and Eberhart have shown that for $w \in [0.8, 1.2]$, can have a better convergence of the problem. The chosen of this factor also depends on the type of the intended application and the desired performance.

Step 6: until reaching the stopping criterion of the problem. It should be noted that the convergence of the algorithm towards the global optimal solution is not always guaranteed. For this reason it is necessary to define a stopping criterion for the algorithm. The stopping criteria used in most literature is the following:

i. The maximum number of iterations $nbIter_{\max}$ is reached.

ii. The change of speed is very low.

iii. The value of the fitness function is reached.

The position of particle, and its initial velocity must be chosen randomly following the uniform law, but to avoid the rapid movement of particle from one region to another in the search space, we fix a maximum speed $v_{\max}$ and we assume that the velocity of particle p_d at time k is equal $v_d(k)$, so that these two velocities satisfying the following conditions:

$$\begin{cases} v_d(k) = v_{\max} & si \;\; v_d(k) > v_{\max} \\ v_d(k) = -v_{\max} & si \;\; v_d(k) < -v_{\max} \end{cases} \quad (14)$$

In a subsequent a combination between the particle swarm optimization algorithm (PSO) and the AFCM algorithm is used to build another approach called AFCM-PSO to identify the premise parameters involved in the Takagi - Sugeno fuzzy model.

V. AFCM-PSO CLUSTERING ALGORITHM

The optimal position is measured with said fitness function defines the following optimization problem. This according to the following fitness function:

$$f(x_k) = \frac{H}{J(U,t,V)} \quad (15)$$

Where H : It is a positive constant.

AFCM-PSO algorithm

Given a data set $X = \{x_1, x_2, ..., x_N\}$, AFCM-PSO algorithm is described by the following steps:

Step 1: Initialization

Set the number of clusters c, $1 < c < N$

Set the weighting exponent m, $m > 1$

Set the tolerance of the end of the algorithm ε , $\varepsilon > 0$

Give ρ_1 and ρ_2

Set the weight of inertia: ω

Set the size of the search space: D

Initialize the 1^{st} particle generation.

Initialize the position and velocity of each particle.

Initialize the fitness function $f(x_k)$

Compute the fuzzy partition matrix, the clusters centers from the FCM algorithm

Repeat l=l+1

Step 2: Compute η_i "(8)".

Step 3: Compute the cluster centers v_i "(9)".

Step 4: Update the fuzzy partition matrix $U = [\mu_{ik}]$ "(10)".

Step 5 : Update the typicality matrix t_{ik} "(11)".:

Step 6: Compute the new value of the fitness function for each particle "(15)".

Step 7: Update the position and the velocity of each particle with "(12)" and "(13)"

So: get the stability of the fuzzy partition, that is to say $\left\| v_k^{(l+1)} - v_k^{(l)} \right\| < \varepsilon$

If this condition is satisfied, stop iteration and find the best solution in the last generation. If not go to 2nd stage.

CONSEQUENCE PARAMETERS IDENTIFICATION

Since the defuzzification method used in the Takagi-Sugeno fuzzy model is linear consequent parameters a_i and b_i Therefore, using the recursive weighted least squares (RWLS) method can be used to estimate these parameters.

$\theta_i^T = [a_i^T; b_i]$: Represent the parameters vector of the ith fuzzy rule (ith cluster).

$x_e = [x;1]$: represent an extension of x (regression vector)

The recursive weighted least squares (RWLS) algorithm is Summarized by the following steps:

Step 1: initialization of the algorithm

$\theta_i(0) = [a_i^T, b_i] = 0, \; P(0) = \sigma I$

Where P : is the gain matrix value

Step 2: Compute $G(k)$ at each simple time

$$G(k) = \frac{P(k-1)x_e^T(k)}{\frac{1}{\mu_{ik}(k)} + x_e^T(k)P(k-1)x_e(k)} \quad (16)$$

Where $x_e^T(k)$ is the observation matrix, and $0 < \mu_{ik}(k) < 1$ is a forgetting factor

Step 3: Update the gain matrix

$$P(k) = \frac{1}{\lambda}\left[P(k-1) - \frac{P(k-1)x_e^T(k)x_e(k)P(k-1)}{\frac{1}{\mu_{ik}(k)} + x_e^T(k)P(k-1)x_e(k)} \right] \quad (17)$$

Step 4: Update the consequent parameter

$$\hat{\theta}(k) = \hat{\theta}(k-1) + G(k)\varepsilon(k)y(k) \quad (18)$$

Where

$$\varepsilon(k) = y(k) - x_e^T \theta(k-1) \quad (19)$$

VI. SIMULATION RESULTS

In this section, we will present an example of nonlinear system difficult to be described by the ordinary method, so the fuzzy model presented in this paper is adopted.

Example 1: This system is described by the following equation:

$$y(k+1) = \frac{y(k)\,y(k-1)\,y(k-2)\,u(k\text{-}1)\big(y(k-2)-1\big)+u(k)}{1+y^2(k-1)+y^2(k-2)} + e(k)$$

Where $y(k)$, $u(k)$ are the output and the input of the system respectively?

$e(k)$ is a linear noise given by the recurrent equation.

$$e_1(k+1) = \cos(\beta)e_1(k) + \sin(\beta)e_2(k)$$

$$e_2(k+1) = -\sin(\beta)e_1(k) + \cos(\beta)e_2(k)$$

$$e(k) = 0.5e_1(k)$$

$$\beta = \frac{\pi}{6}$$

In this case, we present the simulation results concerning the identification of the algorithms we have introduced previously.

i. There exists the system by a random binary signal given in Fig. 1.

ii. For another input, the simulation results given by the FCM algorithm, AFCM algorithm, and AFCM-PSO algorithm are givens in Fig. 2, Fig. 3 and Fig. 4.

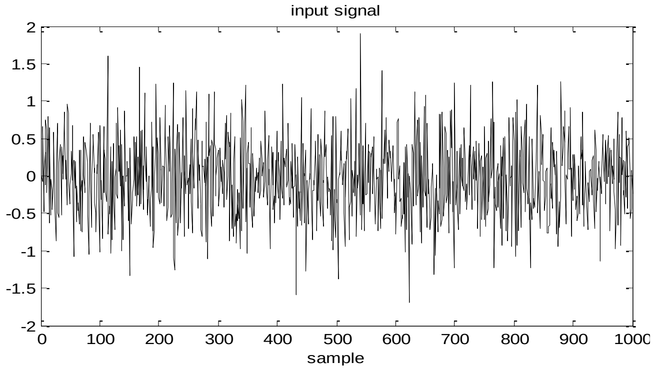

Fig. 1 *Sequences of input-output*

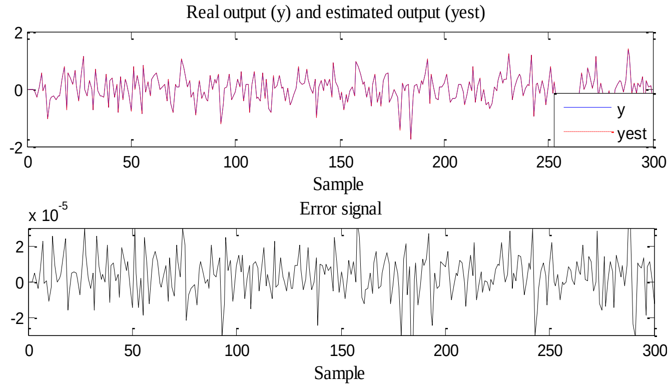

Fig. 2 *Identification result for the FCM algorithm*

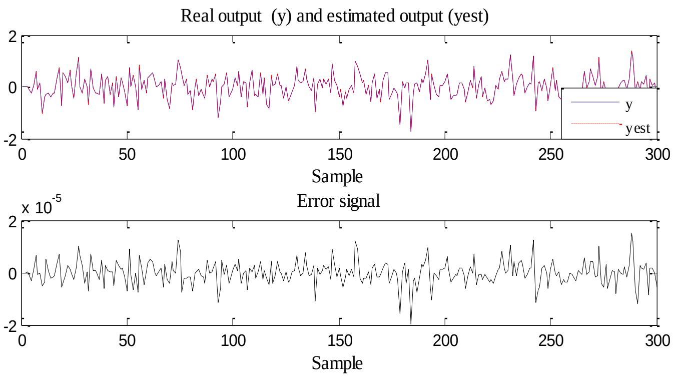

Fig. 3 *Identification result for the AFCM algorithm*

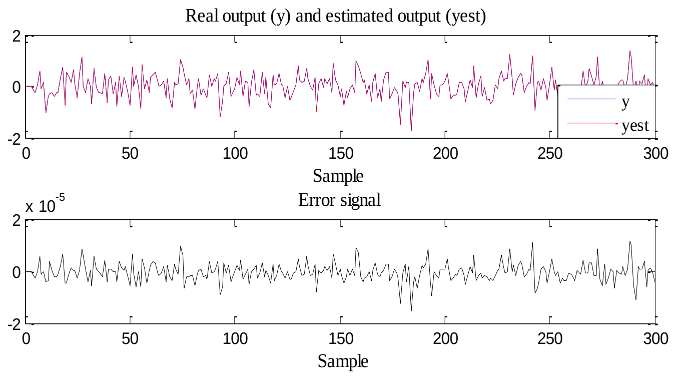

Fig.4 Identification *result for the AFCM-PSO algorithm*

Example 2: APPLICATION TO AN ELECTRO-HYDRAULIC SYSTEM

The effectiveness of the AFCM-PSO identification algorithm we proposed in this paper is tested on an electro-hydraulic system described by the schematic diagram in Fig. 5.

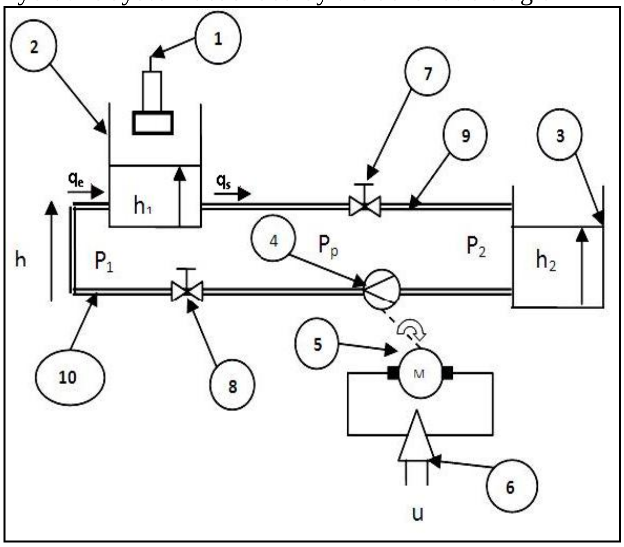

Fig. 5 B*loc diagram*

1 : ultrasonic level sensor
2 : Tank 1
3 : Tank 2
4 : centrifugal pump
5 : DC motor
6 : Variable speed
7 : manual valve v1
8 : manual valve v2
9 : Pipe 1
10 : Pipe 2

IDENTIFICATION OF SYSTEM PARAMETERS

To identify the parameters of this system, we applied a KAFCM-PSO clustering algorithm. The set of observations we have taken is illustrated in Fig. 6.

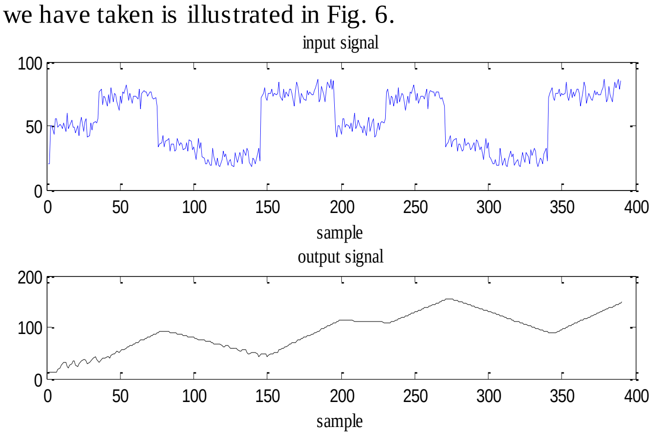

Fig. 6 *Sequences of input-output*

For another sequences of input-output, the simulation result given by the proposed algorithm is given as follows:

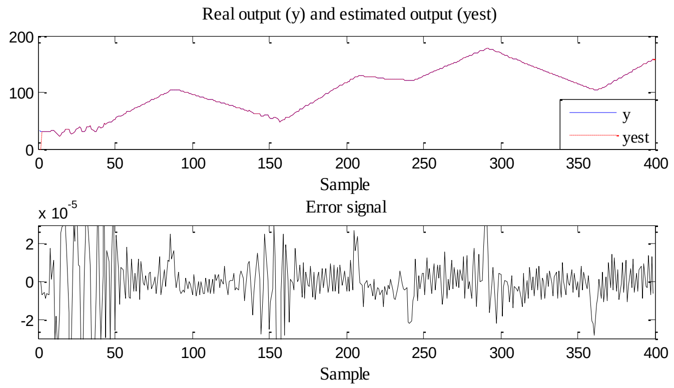

Fig.7 *Identification results for the FCM algorithm*

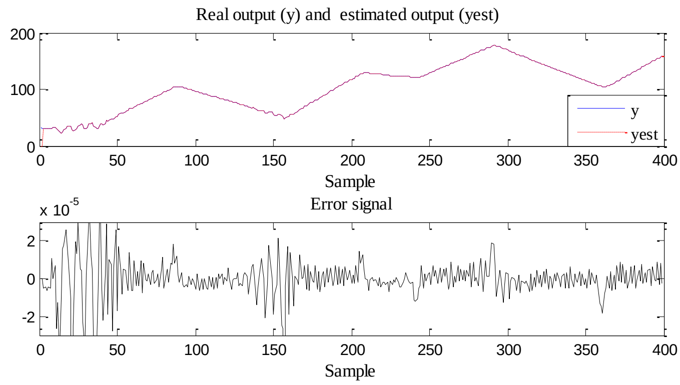

Fig. 8 *Identification result for the AFCM algorithm*

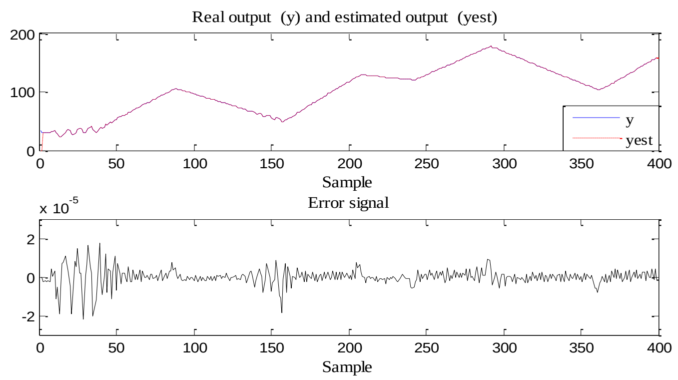

Fig. 9 *Identification result for the AFCM-PSO algorithm*

VII. VALIDATION MODEL

Therefore, to ensure that the model obtained from the estimation it is compatible with other forms of inputs to represent correctly system operating to identify it. It we present, in this paragraph, statistical tests to validate a prediction model based on the RMSE test and the VAF test.

VII.1. RMSE (Root Mean Square Error)

This test calculates the mean squared error between the measured output and model output.

$$RMSE = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(y(k)-\hat{y}(k)\right)^2} \qquad (20)$$

When the model output and actual output are are very near, the test tends to zero.

VII.2. VAF (Variance Accounting For)

This test calculates the percentage standard deviation of the variance between the measured output and model output. It is defined this way:

$$VAF = 100\%\left[1-\frac{\operatorname{var}\left(y(k)-\hat{y}(k)\right)}{\operatorname{var}(y(k))}\right] \qquad (21)$$

TABLE I. VALIDATION RESULTS (EXAMPLE 1)

	RMSE (10^{-6})	VAF (%)	Time computing
FCM	6.62	99.990	8.86
AFCM	4.25	99.991	10.12
AFCM-PSO	3.13	99.992	6.68

The simulation results (table 1) show that the proposed algorithm AFCM-PSO can effectively solve the problem of the others algorithms. The validation tests used have shown good performance of these algorithms.

TABLE II. VALIDATION RESULTS (ELECTRO-HYDRAULIC SYSTEM)

	RMSE (10^{-5})	VAF	Time computing
FCM	15.61	15.61	15.61
AFCM	9.97	9.97	9.97
AFCM-PSO	4.26	4.26	4.26

The validation results (RMSE and VAF test) show well the effectiveness of the proposed algorithm AFCM-PSO compared to the others clustering algorithms when we have proposed.

VIII. CONCLUSION

In this paper, a novel fuzzy clustering algorithm called AFCM-PSO is presented. Unlike to the others clustering algorithms which we have proposed in literature such FCM, PCM and AFCM. The proposed algorithm is a combination between Allied Fuzzy C-Means algorithm (AFCM) and Particle Swarm Optimization (PSO) algorithm. Simulation results show the effectiveness of the proposed algorithm AFCM-PSO compared to the other clustering algorithms particularly for nonlinear system operating in stochastic environment.

REFERENCES

[1] B. Biswal, P. K. Dash and B. K. Panigrahi, “Power Quality Disturbance Classification Using Fuzzy C-Means Algorithm and Adaptive Particle Swarm Optimization, “ *IEEE trans on industriel electronics*, vol. 56, no. 1, Jan .2009.

[2] D.E. Goldberg. “Genetic Algorithms in Search, Optimization, and Machine Learning,” *Reading. MA. Addison*. Wesley, 1989.

[3] J. C. Bezdek. “Pattern Recognition with Fuzzy Objective Function Algorithm.”*Plenum Press*, New York, 1981.

[4] J. C. Dunn. “A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters,” *J. Cybern.*, vol. 3, no. 3, pp.32–57, 1974.

[5] Jian-Qin Chen, Yu-Geng Xi and Zhong-Jun Zhang. “A clustering algorithm for fuzzy model identification.” *Fuzzy Sets and Systems 98*, pp. 319-329, 1998.

[6] Lu. Pingli, Ying Yang and Ma. Wenbo “Random sampling fuzzy c-means clustering and recursive least square based fuzzy identification”, Proceedings of the 2006 American control conference.

[7] N. Zahid, O. Abouelala, M. Limouri, A. Essaid. “Fuzzy clustering based on K-nearestn neighbors rule,” *Fuzzy Sets and Systems*, pp.73-85, 2003.

[8] R. Babuska, Fuzzy Modeling for Control. Kluwer Academic Publishers, Mass., USA, 1998.

[9] R.Diaz Robainas, M.Z.Huang, A.Zilouchian, “Fuzzy identification with clustering with clustering methods of rules construction”, *department of mechanical engineering*, Florida Atlantic University; Boca Raton, Florida 33431.

[10] R. Krishnapuram and J. Keller, “ The Possibilistic c-Means Algorithm: Insights and Recommendations”, *IEEE Trans. Fuzzy Systems*, vol. 4. no. 3, pp. 385-393, 1996.

[11] S. Leandro and M. Bruno, “Fuzzy Identification Based on a Chaotic Particle Swarm Optimization Approach Applied to a Nonlinear Yo-yo Motion System,” *IEEE trans on industrial electronics*, vol. 54, no. 6.

[12] T. Ahmed, C. Abdelkader G. Moncef. “Algorithmes de clustering pour l’identification d’un modèle flou ,” *JTEA*,2008, pp. 1085-1094.

[13] T. Ahmed T, H. Lassad, C. Abdelkader C and G. Moncef, “Optimization problem in Model Based Predictive Control (MBPC),” *11th Inter conf STA*, 2010.

[14] T. Ahmed, H. Lassad and C. Abdelkader, “New Extented Possibilistic C-Means algorithm for identification of an Electreo-hydraulic system,” *12th. Inter. Conf. STA* 2011, ACS. 1695, pp. 1-10.

[15] T. Ahmed, H. Lassad and C. Abdelkader. “Nonlinear system identification using clustering algorithm and Particle Swarm optimization, “ *Scientific Research and Essays*, vol. 7, no 13, pp. 1415-1431, April. 2012.

[16] T. Takagi and M. Sugeno, “Fuzzy identification of systems and its application to modelling and control,” *IEEE Trans. On systems*, Man and Gybernetics, vol. 15, pp.116-132, 1985.

[17] W.S. Jang, H.I. Kang, B.H. Lee, K.I. Kim, D.I. Shin and S.C. Kim, “Optimized Fuzzy Clustering By Predator Prey Particle Swarm Optimization,” *IEEE*, 2007.

[18] X. H. WU and J.J. ZHOU, ‘‘Allied fuzzy c-means clustering model,’’*Transaction of Nanjing University of Aeronautics and Astronautics*, vol. 23, no. 3, pp. 208-213, 2006.