10.31341/jios.45.1.12
Open Access

UDC 004.838.2
Original Scientific Paper

Dissimilarity Clustering Algorithm for Designing the PID-like Fuzzy Controllers

Essam Natsheh *dr_natsheh@hotmail.com*
Dep. of Informatics Engineering, College of Engineering
AMA International University, Bahrain

Abstract

Fuzzy logic controller is one of the most prominent research fields to improve efficiency for process industries, which usually stick to the conventional proportional-integral-derivative (PID) control. The paper proposes an improved version of the three-term PID-like fuzzy logic controller by removing the necessity of having user-defined parameters in place for the algorithm to work. The resulting non-parametric three-term dissimilarity-based clustering fuzzy logic controller algorithm was shown to be very efficient and fast. The performance study was conducted by simulation on armature-controlled and field-controller DC motors, for linguistic type and Takagi-Sugeno-Kang (TSK) models. Comparison of the created algorithm with fuzzy c-means algorithm resulted in improved accuracy, increased speed and enhanced robustness, with an especially high increase for the TSK type model.

Keywords: Three-term controller, PID controller, fuzzy systems, fuzzy logic controller, clustering algorithms

1. Introduction

Fuzzy logic, as opposed to classical true-false digital logic, is based on the idea that variables may take values continuously from 0 to 1. Values in regular Boolean logic can be only true or false (1 or 0). Fuzzy logic allowed creating machine control systems that calculated solutions on a scale rather than discretely on or off. This way, a machine could express behaviors that are different from “do” and “not do”, such as “do a little”, “do moderately”, etc. In many cases this approach allowed to build effective systems that operated smoothly.

Being applied in many industries, research on fuzzy logic controllers (FLCs) resulted in algorithms improving more and more over the years. The research is still ongoing, and mathematic tools have improved drastically since the advent of the fuzzy logic controller. We, specifically, are interested in the three-term version of it, which combines the benefits of the almost century-old PID (or proportional–integral–derivative controller) with fuzzy logic. The regular PID controller algorithms are to this day the most widely used in process industries, such as chemicals, pharmaceuticals, petroleum, coal, plastics, etc. PID-like FLC, on the other hand, is on the rise, and every year research produces increasingly complex and useful algorithms. One of the main benefits of the three-term controller is that it better emulates how humans think, therefore allowing staff with much less training in control to design a particular control system.

Clustering algorithms are used to group objects into clusters with high similarity between objects in the same cluster, and low similarity between objects in different clusters. For the purpose of this study, the dissimilarity measure was chosen. Therefore, the former statement could be reformulated as follows: these algorithms will group objects into clusters with low dissimilarity between objects in the same cluster, and high between objects in different clusters. The problem is that predefined user parameters affect the clustering and could be detrimental to the performance of the FLC.

The main objective of this paper is to derive a clustering algorithm that, without parameters defined by the user, would generate a rule-base for a proportional-integral-derivative-like FLC. The numerical data will be clustered based on the dissimilarity of the object's "position".

2. Literature Review

A lot of work has been put into the research of the FLC since Mamdani and Assilian introduced the concept in [1]. They applied their novelty controller to a simulated model of an industrial plant – a steam engine. Resulting from their study was a ground-breaking conclusion that fuzzy set theory is applicable and useful in the design of controllers. They proved that an unstructured set an efficient algorithm could be derived from an unstructured set of heuristics.

Esposito et al. in [2] reviewed different dissimilarity measures for symbolic descriptions, including Boolean and probabilistic descriptions. They provide an overview and mathematical basis for different approaches in calculating the dissimilarity measure in multivariate data using ASSO (analysis system of symbolic official data) software. The DISS module of the ASSO software was used to obtain dissimilarity measure with different methods. The VDISS module was used "for the visualization of dissimilarities by means of two-dimensional scatterplots and line graphs".

Gowda and Diday in [3], [4] propose symbolic clustering using their new dissimilarity measure. It is "non-parametric, hierarchical and agglomerative in nature", and therefore is invaluable for our current research. They selected the least dissimilar mutual pair of objects by finding the ones with the smallest possible increase in the sum of squared error.

The paper [5] of Appice et al. proposed a symbolic objects' classification (clustering) method SO-NN as an extension of the k-nearest neighbor method, as a lazy learning approach. Their research resulted in a conclusion that the SO-NN method was able to show adaptive behavior by approximating the symbolic object's unknown class value locally, while the prediction accuracy remained high. They also were able to compare different measures for dissimilarity, for both Boolean and probabilistic symbolic objects. Finally, they suggest that by weighting each variable separately to compute the dissimilarity between objects, it is possible to significantly improve the predictive capability of the proposed symbolic objects nearest neighbor algorithm as compared to the conventional k-nearest neighbor classification.

Malerba et al. in [6] were able to compare different dissimilarity measures and obtained controversial results. First, they established that a comparative investigation has never been done on dissimilarity measures to check their performance on real data. Then, they conducted the study with a data set for which there was a defined expected property. Among seven dissimilarity measures only two were found to exhibit stable behavior when applied to a real data set with understandable properties. They concluded that more extensive experimentation is required in this case.

Djamal Ziani proposed an improved algorithm Minset-Plus in [7]. This algorithm was used to select discrimination variables from a set of symbolic objects. They conducted experiments on both real and generated databases, and compared the discrimination power and discrimination by extent between the two algorithms. The obtained results indicated good cluster selection with strong discrimination for any types of data.

In [8], El-sonbaty and Ismail suggested a modified fuzzy c-means algorithms for clustering of symbolic objects. They tested it on real and simulated data sets and compared the results with those that could be found in the literature. Their new algorithm was said to surmount almost all of the drawbacks that come with clustering symbolic data using hierarchical algorithms. It was achieved through representing the clustering of symbolic objects as an optimization problem, where the function was constrained by specific boundary conditions.

Ng et al. in [9] looked into updating the k-modes categorical data clustering algorithm (which is the k-means algorithm that uses a matching dissimilarity measure, frequency-based method and clusters' modes instead of means) by using a new dissimilarity measure. Their results suggest that their improved version of the k-modes is more efficient in clustering categorical data sets, which is proved using both mathematical means and experimentation. In particular, they found that the convergence and clustering efficiency is better with the new dissimilarity measure.

3. Non-parametric dissimilarity-based clustering for designing a FLC

In our previous study [10], various clustering algorithms that could be used to design a PID-like FLC were discussed. These algorithms require the determination of some parameters that affect the number of clusters to be generated. In next subsections, an algorithm based on Gowda and Diday's dissimilarity measure is proposed.

3.1. Composite objects

Successive merging lies at the heart of agglomerative clustering methods. Merging is the process of gathering together, on the basis of the similarity or dissimilarity measures, two samples and assigning them a same-cluster membership or a label for further clustering [3], [4]. For example, let A and B be two objects; Gowda and Diday proposed to compose these two objects by consideration their minimum interval which include both A and B. Due to non-interval data, the composite object O resulting from the merging of A and B is proposed as:

$$O = \text{mean}\,(A, B) \tag{1}$$

3.2. Mutual pair

The concept of mutual pair consists of two elements that are part of two different groups of data. One element or object of one group X_i and a second element or object of a second group X_j, which share the same boundary, may have the highest similarity or the lowest dissimilarity. If one of the two previous conditions occurs effectively, it means that it constitutes a mutual pair.

The two concepts are not technically the same condition, but similar to the mathematical concept of limit, the two conditions must occur for mutual pair condition.

The concept of *mutual nearest neighborhood*, introduced by Gowda and Krishna [11], [12], is not exclusive for computer sciences. Still, it applies in biology with the condition of two or more cells that are physically one side to another, sharing similar chemical and biological characteristics. In computer sciences, there are common characteristics such as learning, editing, condensed nearest-neighbor rule, desegregate clustering, and error correction.

3.3. Dissimilarity and similarity measures

The relative positions of two points, named A and B, can be analyzed from a dissimilarity and similarity, considering the similar characteristics of two points of differences. The dissimilarity, named as D, is represented mathematically as a difference between the points "A" and point "B." The dissimilarity is as follows:

$$D\,(A, B) = \| A - B \| \tag{2}$$

The symbol "‖" shows an inner product norm metric, according to the Euclidian norm. It could be rewritten using the square root of a sum, as-is:

$$D\,\left(A, B\right) = \sqrt{\sum_{i=1}^{n} \left(A_i - B_i\right)^2} \tag{3}$$

The dissimilarity (D) is inside the norm, between 0 and 1, using the following:

$$D' = \frac{D - D_{\min}}{D_{\max} - D_{\min}} \tag{4}$$

The calculation considers D's value for a specific position, and its transformation considers the maximum and minimum value of dissimilarity.

3.4. Dissimilarity measure-based algorithm (DMBA)

The calculation of the dissimilarity measure-based algorithm (DMBA) can be developed in four steps, according to the following procedure [3], [4]:

1. It is necessary to identify the system's different objects, using the letter X with the identification of the initial number of clusters N equal to number of objects. For example, X_1, X_2, …, X_N. As was told previously, the number of clusters is a discrete value; and each cluster has a specific weight of 1.
2. Calculate between the different points of the data the weighted dissimilarity D_w. The calculation considers the dissimilarity of a specific (X_i, X_j) object and the total amount of objects. The mathematical equation is the following:

$$D_w(X_i, X_j) = D(X_i, X_j).\sqrt{\frac{n_i.n_j}{n_i + n_j}} \tag{5}$$

   n_i and n_j represent the cluster weights for X_i and X_j, and the dissimilarity is given by D (X_i, X_j) that calculated using equation (3). The calculation provides a weighted measurement because the ratio is below a square root calculation.

   The number of clusters is reduced to 1. According to previous definitions, the pair of individuals is merged using a composite object using the lowest dissimilarity.
3. Steps 2 must be repeated until the number of clusters is equal to 1.
4. Calculate the cluster indicator (*CI*). The calculation considers three stages: "*p*", "*p*+1" and "*p*-1". The recommendation is to calculate the value of R_p for each stage and later determine the value of CI

   It is necessary to use the minimum dissimilarity for the current stage, the previous stage, and the forward stage in each stage. The higher this value, the more improper it is to merge the two objects of the mutual pair at this stage [3]. The use of a calculation loop is useful when there are a large number of clusters to calculate. The *CI* equation is the following:

$$CI = \frac{R_p}{R_{p+1}} \tag{6}$$

   where

$$R_p = \frac{D_{min} \text{ at } (p)}{D_{min} \text{ at } (p+1) + D_{min} \text{ at } (p-1)} \tag{7}$$

4. Developing the rule-base of the PID FLC using clustering algorithms

Fuzzy system models basically fall into two categories [13]. The first category is linguistic models (LMs) that contains list of IF–THEN rules with vague predicates. The second category of fuzzy models called Takagi-Sugeno-Kang (TSK) method [14] where the rules have a fuzzy antecedent part and a functional consequent; essentially, they are combination of fuzzy and non-fuzzy models. This section is a discussion of how both kinds of models can be applied in the design of a three-term FLC.

4.1. Linguistic models as tools for three-term FLC representation

Starting with the observed data pairs (e_k, de_k, se_k, U_k) the clustering methods provide a collection of clusters and their centers (e_{Ci}, de_{Ci}, se_{Ci}, U_{Ci}). Each center can be viewed as a prototypical fuzzy point in the relationship between input and output, as shown in Figure 1.

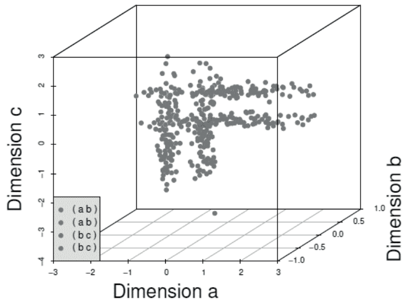


Figure 1: Three-dimensional data clustering for rule determination.

It is proposed that the rules in the PID-like FLC rule-base are formed linguistically as follows:

IF error is close to *cluster i* AND error-change is close to *cluster i* AND error-sum is close to *cluster i*, THEN control action is close to *cluster i*.

$$\text{For } i = 1 \ldots C \tag{8}$$

where C is the number of clusters. The cluster centers (e_{Ci}, de_{Ci}, se_{Ci}, U_{Ci}) of each fuzzy variable are considered to be the peaks of their membership functions.

To achieve a systematic method of defining the membership functions of the antecedent fuzzy sets, the use of the Gaussian curve membership function is proposed:

$$\mu_i(y) = exp\left(\frac{-(y - y_i^*)^2}{2\sigma^2}\right) \tag{9}$$

where y is the input vector, y^*_i is the center of the cluster i, and σ is the width of the cluster i. To compute the initial value of σ for the clusters in the fuzzy variable X (e, de, se, and U), the following equation is proposed:

$$\sigma = \frac{\text{Maximum point in } X - \text{Minimum point in } X}{\text{Number of clusters}} \tag{10}$$

To refine the σ value for each variable, the algorithm 1 is proposed. The use of a decrease_ratio of 0.7 and an increase_ratio of 1.05 is suggested.

Algorithm 1: optimizing σ values

```
Input: initial σ value for each variable
Output: optimized σ value for each variable
Use eq. (10) to get the initial σ value to each variable;
Initialize SSE (sum-squared error) to large value;
For i=1 to maximum number of epochs to refine all σ
  If SSE < sse_goal, break, end if
   For j=1 to minimum no. of epochs to refinement one σ
      Run the experiment and get new_sse;
      If (new_sse ≤ SSE)
         SSE = new_sse;
         Save σ;
         σ = σ × increase_ratio;
      else
        σ = σ × decrease_ratio;
     end if
   end for
end for
```

4.2. Takagi-Sugeno-Kang models as tools for three-term FLC representation

A definition of the TSK rules in the PID-like FLC rule-base is proposed as follows:

IF error (y_1) is close to *cluster i* AND error-change (y_2) is close to *cluster i* AND error-sum (y_3) is close to *cluster i*, THEN control action is $a_{i1}y_1 + a_{i2}y_2 + a_{i3}y_3 + a_{i4}$

$$\text{For } i = 1 \ldots C \tag{11}$$

This subsection contains a description of Chiu's method [15], [16] for fuzzy model identification. His method for fuzzy model identification from data is based on the use of a cluster estimation method to create the rules and their initial parameters and then on the application of optimization algorithms to tune these parameters. Chiu uses a *recursive least squares* estimation algorithm to optimize the fuzzy model. The issue with this algorithm is that it is iterative and slow. To optimize the fuzzy model, the use of a *singular value decomposition* method is proposed. This is a non-iterative algorithm capable of obtaining the parameter estimates very quickly and reliably.

4.2.1. Fuzzy model identification

Consider a set of C cluster centers $\{c_1, c_2, ..., c_C\}$ in an M dimensional space. Let the first N dimensions correspond to input variables and the last M-N dimensions correspond to output variables. Each vector c_i is decomposed into two components y^*_i and z^*_i, where y^*_i contains the first N elements of c_i (i.e., the coordinates of the cluster center in input space) and z^*_i contains the last M-N elements (i.e., the coordinates of the cluster center in output space). Given an input vector y, the degree to which rule i is fulfilled is defined as [15], [16]:

$$\tau_i = \exp\left(-\|y - y_i^*\|^2\right) \tag{12}$$

The output vector z can be computed via:

$$z = \frac{\sum_{i=1}^{C} \tau_i z_i^*}{\sum_{i=1}^{C} \tau_i} \tag{13}$$

To systematically define the membership functions of the antecedent fuzzy sets, the use of the Gaussian curve membership function as defined by (9) is proposed. Equation (10) should be used to compute the σ value for the Gaussian curves.

4.2.2. Optimizing the fuzzy model

The use of different variables increases the complexity of the calculation. It is possible to optimize the fussy model with the use of a set of clusters by a direct way of calculation. The calculation requires the optimization of rules using several variables' linear function, better than using only a simple constant. That is,

$$z^*_i = G_i\, y + h_i \tag{14}$$

where G_i is a matrix of $(M\text{-}N) \times N$ constant elements. To be consistent, h_i is a vector with $(M\text{-}N)$ constant elements. The z^*_i expression is a linear function that allows it to calculate with low computational complexity.

The optimization process of the equation is possible using the principle of the *least-squares* estimation procedure [14]. The equation for the calculation is:

$$\rho_i = \frac{\tau_i}{\sum_{j=1}^{C} \tau_j} \tag{15}$$

Equation (13) can then be rewritten as

$$z = \sum_{i=1}^{C} \rho_i\, z_i^* = \sum_{i=1}^{C} \rho_i\,(G_i y + h_i) \tag{16}$$

or

$$z^T = [\rho_1 y^T \quad \rho_1 \quad \dots \rho_C y^T \quad \rho_C] \begin{bmatrix} G_1^T \\ h_1^T \\ \vdots \\ G_C^T \\ h_C^T \end{bmatrix} \tag{17}$$

where z^T and y^T are row vectors. With "n" input data points in the form of y_1, y_2, ..., y_n; the result of the matrix operation is the following:

$$\begin{bmatrix} z_1^T \\ \vdots \\ z_n^T \end{bmatrix} = \begin{bmatrix} \rho_{1,1} y_1^T & \rho_{1,1} & \dots \rho_{C,1} y_1^T & \rho_{C,1} \\ & & \vdots & \\ \rho_{1,n} y_1^T & \rho_{1,n} & \dots \rho_{C,n} y_n^T & \rho_{C,n} \end{bmatrix} \begin{bmatrix} G_1^T \\ h_1^T \\ \vdots \\ G_C^T \\ h_C^T \end{bmatrix} \tag{18}$$

The value of $\rho_{i,j}$ is evaluated in the value of y_j. In the matrix operation, the product is confirmed by two sides. The left hand of the product is constant, while the second side of the matrix contains all the optimized parameters.

The estimation of least-squares can be calculated with the following procedure:

$$AX = B \tag{19}$$

B's value is the output calculation; A is a constant matrix, and X is the parameters matrix to be calculated. Let us choose X in such a way that the following objective function J is minimized [17], [18]:

$$J = \|B - AX\|_2^2 \equiv (B - AX)^{-T}(B - AX) \tag{20}$$

To carry out the minimization, J is differentiated with respect to X and the result is equated to zero. Thus,

$$\frac{\partial J}{\partial X} = -2A^T B + 2A^T X = 0 \tag{21}$$

from which X can be solved as

$$X = (A^T A)^{-1} + A^T B \tag{22}$$

In practice, the most reliable method of computing the pseudo-inverse of a matrix is the *singular value decomposition* (SVD). For an overview of the SVD, its theory, and numerical details, the reader is referred to [19].

5. Performance analysis of the proposed design method

In the next sections, the use of two performance measures with two simulated systems is proposed. The objective of the simulation is to demonstrate the feasibility of the proposed three-term design method when applied to second-order systems. A comparison of its performance with the performance of other clustering methods is carried out.

5.1. Performance study

To test the models, two performance measures have been chosen, which will be used to analyze the performance of the proposed methods for designing a FLC. They are:

1. *Accuracy:* To design a PID-like FLC with clustering algorithms, the rise-time, overshoot, and settling-time performance measures will be omitted, because a teacher signal is used with these algorithms as a reference model. To validate the results, it is proposed that an accuracy criterion be employed. Accuracy means the correctness of the answer. In order to measure it, the use of the sum of squares for error is proposed. Thus, the smaller the error, the better the accuracy, and the larger the error, the worse the accuracy will be. The error here is the error between the output of the system under analysis and its reference model.
2. *Robustness:* To measure robustness, it is proposed that the defuzzification method parameter be varied. During the design of the FLC, center of area (COA) was chosen as a defuzzification method. To measure the robustness of this controller, the use of bisector of area (BOA) as a defuzzification method [20] is suggested. The procedure used to implement this method shown in [24].

Two types of direct current (DC) motors are analyzed to examine the performance of proposed design methods: *armature-controlled* with fixed field and *field-controlled* with fixed armature current [21]. The same details and parameters as described in our previous studies [22], [23], [24], [25] were used for these two systems.

For the clustering technique, the reference model with inputs $[e, de, se]^T$ and output U is used to designate the desired performance. To design a PID-like FLC using the *fuzzy c-means* (FCM) algorithm, the weighting exponent parameter q for the membership functions matrix M was chosen as 2.0 [26], [15]. To design a PID-like FLC with a subtractive algorithm, the cluster radius r_a as 0.5 for all data dimensions, squash factor r_b as 1.5, accept ratio $\bar{\varepsilon}$ as 0.5, and reject ratio $\underline{\varepsilon}$ as 0.15 [15] were chosen.

5.2. Simulation results

The performance of the PID-like FLC design methods is examined by analyzing the transient response and accuracy in subsection 5.2.1 and robustness in subsection 5.2.2.

5.2.1. Transient response and accuracy

The following subsection focuses on the performance of the armature-controlled DC motor, while subsection 5.2.1.2 focuses on the performance of the field-controlled DC motor.

5.2.1.1 Armature-controlled DC motor system

The *CI* for the armature-controlled DC motor system is shown in Figure 2. This figure shows that the DMBA generates only 5 clusters from 40 sampling points.

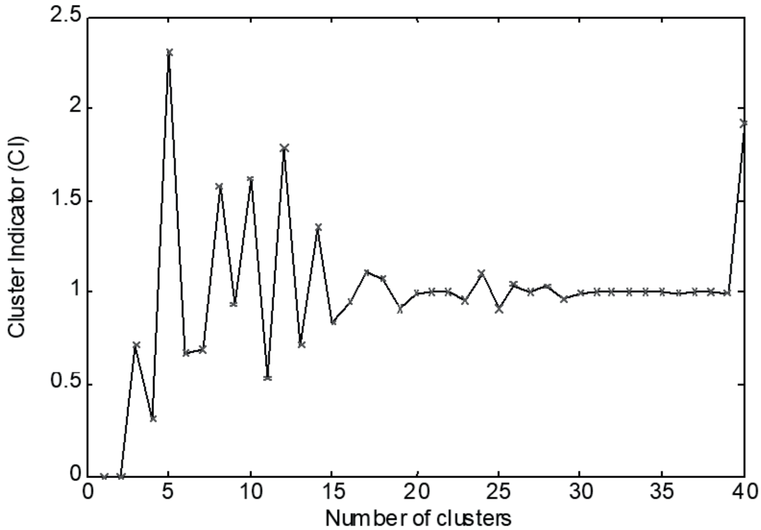
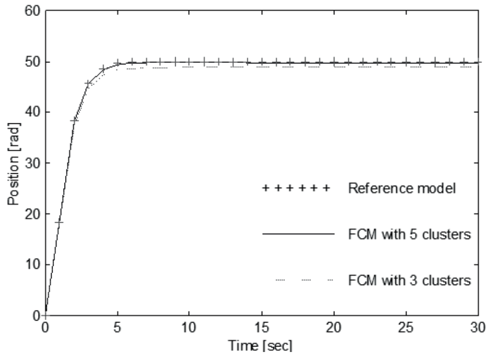


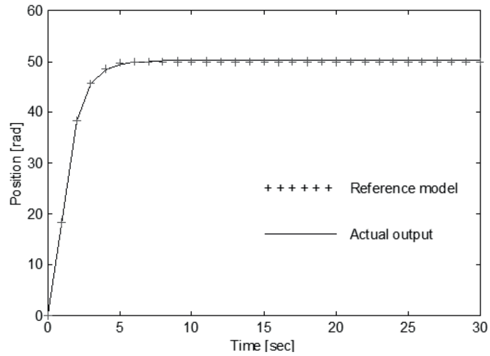
Figure 2: Cluster indicator for armature-controlled DC motor system

Figure 3 shows the step responses and accuracy of the armature-controlled DC motor system. The FCM algorithm used to generate 3 clusters and 5 clusters for comparison. It can be seen how the controller output gets close to the reference model as more clusters are considered. The subtractive algorithm generates only one cluster for this system, so it cannot be used with this model. A comparison between the SSE for the linguistic-type DMBA and other controller shows that they are comparable. Note that SSE for the DMBA is significantly smaller than the SSE for the other controller. Thus, we can conclude that no over-transient response occurs with DMBA method.

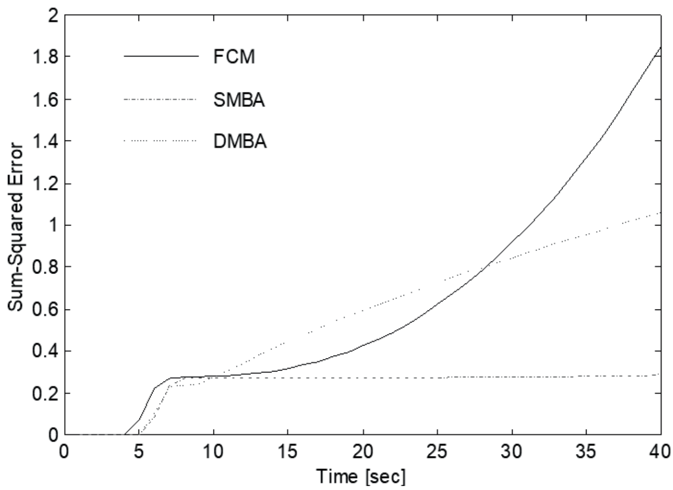
Figure 4 shows the step responses and accuracy of the armature-controlled DC motor system using the FCM algorithm (used to generate 5 clusters), the subtractive algorithm, and the DMBA for the TSK-type model of the PID-like FLC. In figure 4-B, the values of SSE for FCM and DMBA are multiplied by ten to the power of negative seven which mean they are clearly more accurate than the subtractive clustering algorithm.



(A-1) FCM with 3 clusters and 5 clusters



(A-2) Proposed non-parametric DMBA.



(B) Accuracy of clustering algorithms.

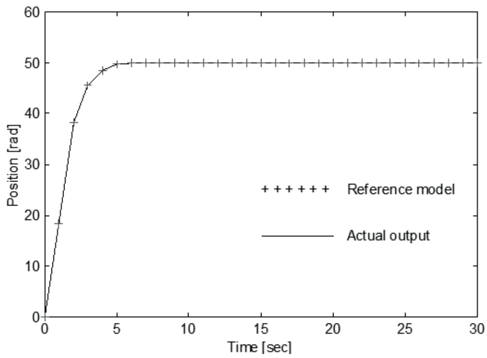
Figure 3: Step responses (A) and accuracy (B) of armature-controlled DC motor system using clustering algorithms used to develop linguistic type model PID-like FLC.

Actually, in that very small range of values for SSE we can say the accuracy of the two algorithms is almost the same and that is clear from the figure 4-A.

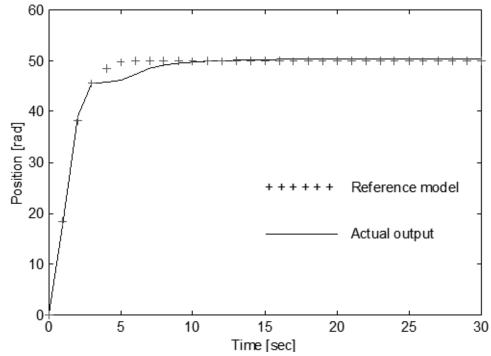
5.2.1.2 Field-controlled DC motor system

The cluster indicator (CI) for the field-controlled DC motor system using DMBA is shown in Figure 5. This figure shows that the DMBA generates 5 clusters from 40 sampling points.

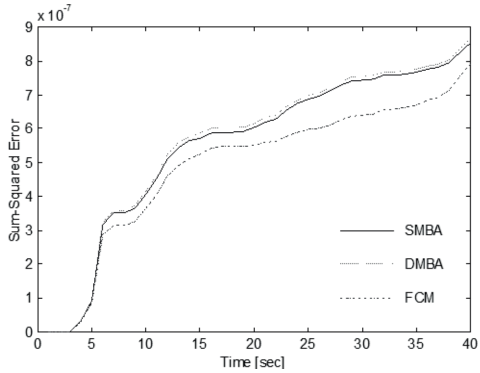
Figure 6 shows the step responses and accuracy of the field-controlled DC motor system using the FCM algorithm (used to generate 5 clusters) and the DMBA to develop a linguistic-type model of the PID-like FLC.



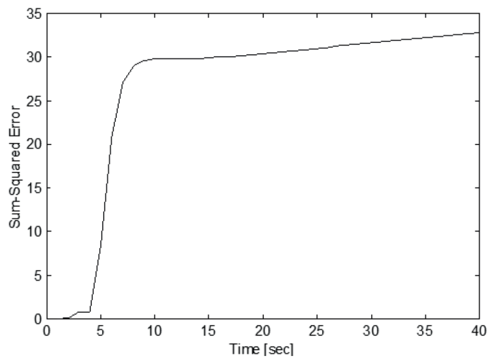
(A-1) FCM and DMBA.



(A-2) Subtractive clustering algorithm.



(B-1) FCM and DMBA.



(B-2) Subtractive clustering algorithm.

Figure 4: Step responses (A) and accuracy (B) of armature-controlled DC motor system using clustering algorithms used to develop TSK type model PID-like FLC.

The subtractive algorithm generates only one cluster for this system, so it cannot be used to develop a linguistic-type model of the PID-like FLC.

The system obtained with the controller responds to load disturbances and measurement noise. This is observed from Figure 6-A. The overshoot for setpoint changes is smallest for DMBA, which is the case to avoid large transients in the control signal due to sudden changes in the setpoint. Even though it generates higher SSE values for DMBA compared with FCM, as shown in Figure 6-B, the transient response for both algorithms is almost the same which is clearly shown in Figure 6-A.

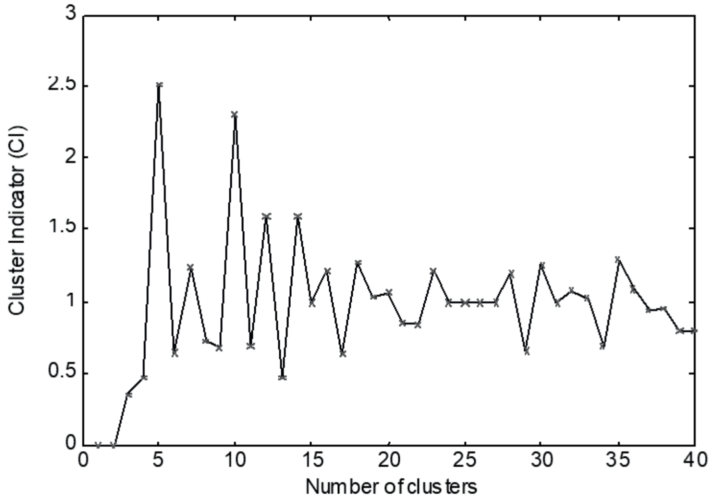
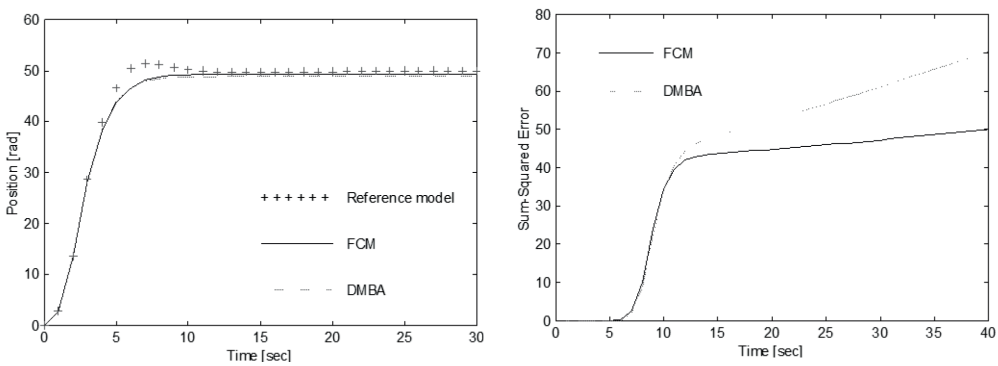


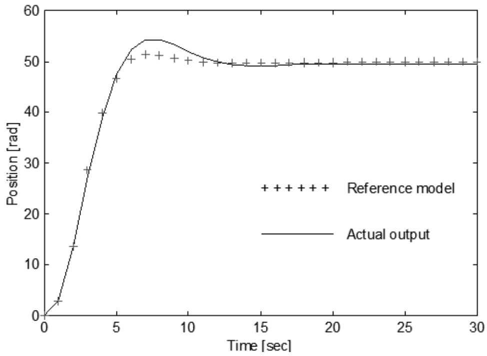
Figure 5: Cluster indicator for field-controlled DC motor system using DMBA



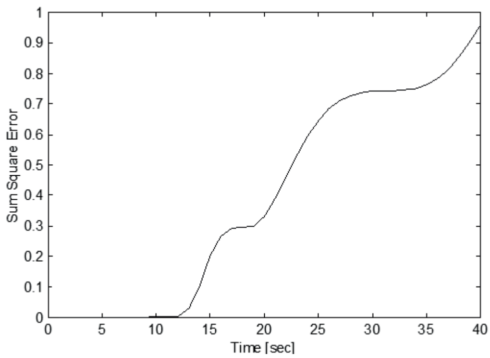
(A) FCM and DMBA. (B) FCM and DMBA.

Figure 6: Step responses (A) and accuracy (B) of field-controlled DC motor system using clustering algorithms used to develop linguistic type model PID-like FLC.

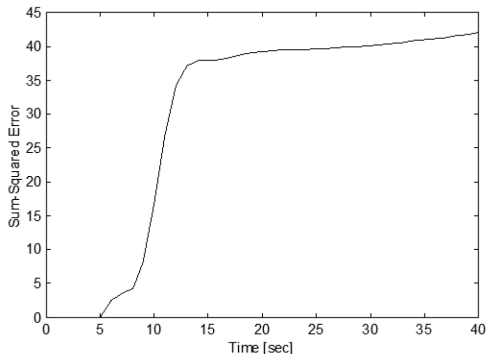
Step responses of field-controlled DC motor system using FCM and DMBA used to develop TSK type model PID-like FLC is similar to figure 4-A. It can be seen that the step response results by these methods is better than the subtractive clustering algorithm shown in figure 7-A. Where, zero overshoot achieved in FCM and DMBA methods. The figure 7-B shows that the proposed DMBA tuning method works more precisely than other tested controllers.



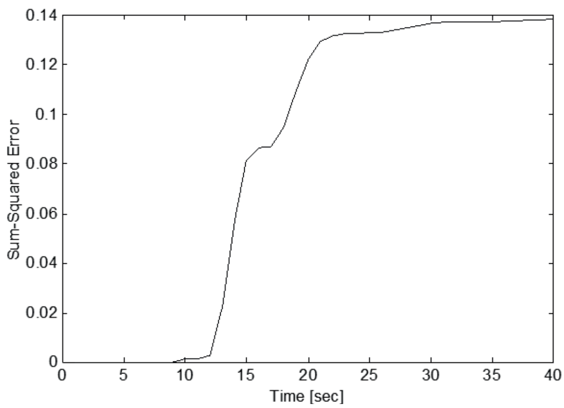
(A) Subtractive clustering algorithm.



(B.1) FCM clustering algorithm.



(B.2) Subtractive clustering algorithm.



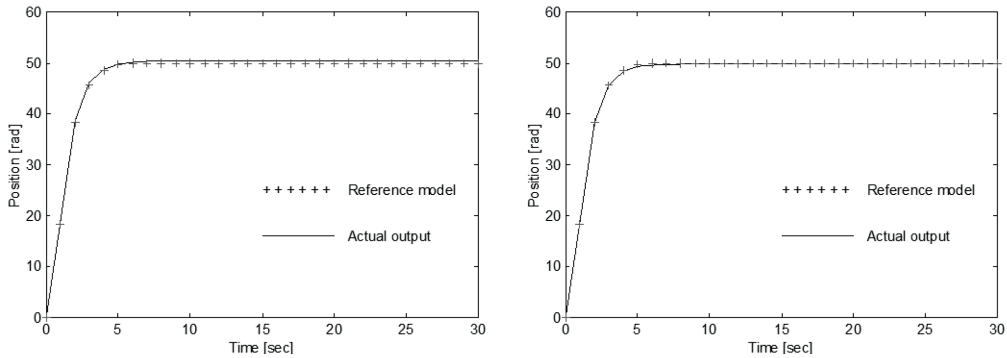
(B.3) Proposed non-parametric DMBA.

Figure 7: Step responses (A) and accuracy (B) of field-controlled DC motor system using clustering algorithms used to develop TSK type model PID-like FLC.

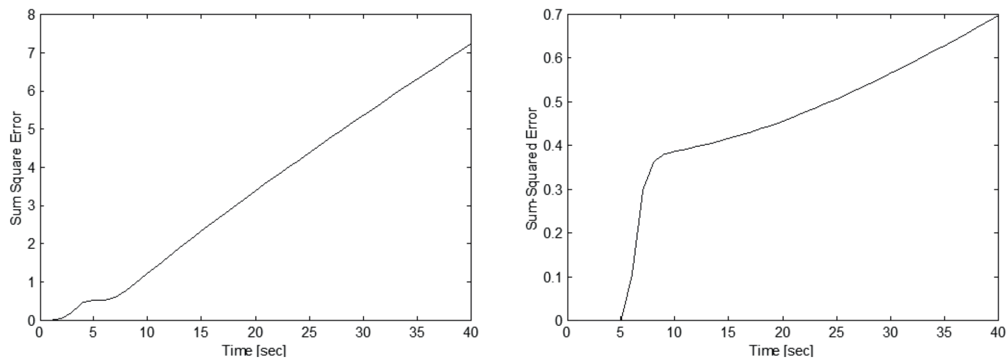
5.2.2. Robustness test

The following subsection analyzes the robustness of the PID-like FLC design methods when varying the defuzzification method from center of area (COA) to bisector of area (BOA). This test cannot be used with the TSK-type model. The only defuzzification method that can be used with this model is the *weighted average* method, because the fuzzy output does not have a geometric shape.

Figures 8 and 9 show the step responses and accuracy of the armature-controlled DC motor and the field-controlled DC motor respectively, using the FCM algorithm (used to generate 5 clusters) and the DMBA to develop a linguistic-type model of the PID-like FLC. It is shown that there is a substantial improvement in the time domain specification in terms of lesser rise time, settling time and overshoot using DMBA algorithm. Hence this method is a robust design method for determining the PID controller parameters.

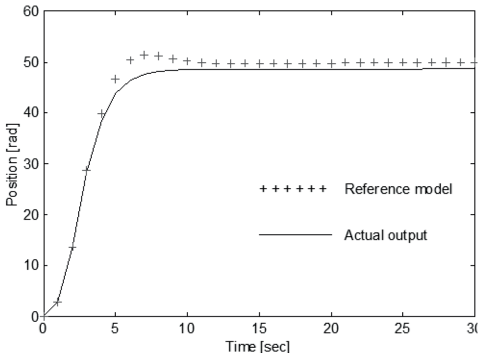


A.1 FCM clustering algorithm. A.2 Non-parametric DMBA

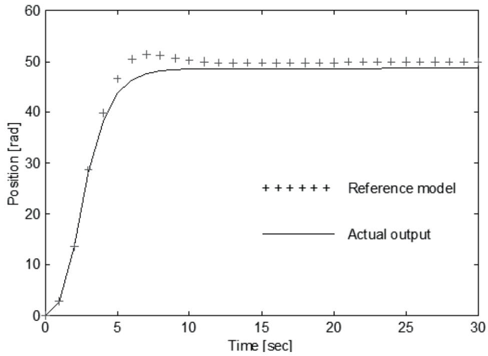


B.1 FCM clustering algorithm. B.2 Non-parametric DMBA

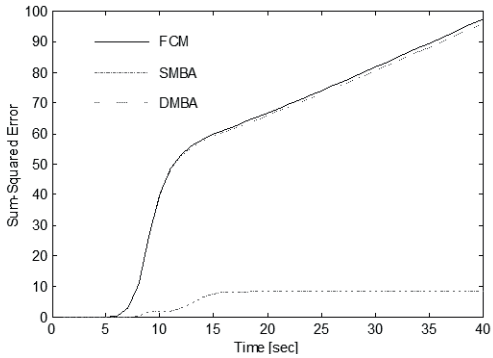
Figure 8: Robustness test to varying of defuzzification method for armature-controlled DC motor system: (A) Step responses and (B) Accuracy test.



A.1 FCM.



A.2 Proposed non-parametric DMBA.



B. Accuracy of clustering algorithms.

Figure 9: Robustness test to varying of defuzzification method for field-controlled DC motor system using clustering algorithms used to develop linguistic type model PID-like FLC.

5.2.3. Discussion

The graphs in the Section 5 provide an outlook at the effectiveness of the proposed non-parametric DMBA applied to create linguistic type and Takagi-Sugeno-Kang (TSK) type models of three-term fuzzy logic controllers. In graphs for the armature-controlled DC motor system with linguistic type model, the following results were obtained. The sum of squared error is increasing exponentially with time for the FCM and subtractive algorithms, while for the DMBA it is linear. Therefore, DMBA in this case is superior in terms of accuracy. As for the TSK type model used for the armature-controlled DC motor system, the sum of squared error is also lower for the DMBA, proving its increased correctness.

The simulation graphs for the robustness test show that non-parametric DMBA is more accurate and robust when different defuzzification methods are applied. Also, from the graphs we can conclude that TSK type model shows better accuracy than the linguistic type model.

The transient response for FCM and DMBA is almost the same for many cases studied in this paper. The main problem of supervised FCM clustering algorithms is determination of the number of clusters that satisfactorily represent the system. Besides that, FCM have another problem. It requires determination of some parameters that will affect (increase/decrease) the number of generated clusters. Unsupervised DMBA clustering algorithms solve these problems.

6. Conclusions

Three-term PID-like fuzzy logic controllers better simulate the decision-making process of humans. Therefore, engineers are not required to understand everything in the complicated mathematical apparatus to set up a control system. On the other hand, a problem arises that there are parameters that a user has to define manually for the clustering algorithm to start working. This step affects the amount of generated clusters and could be deemed unnecessary and excessive. As an attempt to find a solution, we proposed a non-parametric dissimilarity clustering algorithm.

Resulting from the performance study, we identified that the dissimilarity measure based clustering algorithm is more accurate than regular FCM and subtractive clustering algorithms. Together with singular value decomposition the proposed dissimilarity measure based algorithm has been shown to be fast and reliable in terms of construction of a fuzzy model from numerical data input. The algorithm shows improved accuracy, increased speed and enhanced robustness as compared to the FCM and subtractive clustering algorithms for both linguistic type and Takagi-Sugeno-Kang type models. We also conclude that the latter perform better than the former.

Acknowledgements

The author would like to express his appreciation to College of Engineering at AMA International University Bahrain for supporting this research.

References

[1] E. H. Mamdani and S. Assilian, “An experiment in linguistic synthesis with a fuzzy logic controller”, *International journal of man-machine studies*, Vo.*7*, No.1, pp. 1-13, 1975

[2] F. Esposito, D. Malerba, & A. Appice, “Dissimilarity and matching”, *Symbolic Data Analysis and the SODAS Software*, pp. 61-66, 2008

[3] K. C. Gowda and E. Diday, "Symbolic Clustering using a New Dissimilarity Measure", *Pattern Recognition*, Vol.24, No. 6, pp. 567-578, 1992

[4] K. C. Gowda and E. Diday, "Symbolic Clustering Using a New Similarity Measure", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-22, No. 2, pp. 368-378, 1992

[5] A. Appice, C. d'Amato, F. Esposito, & D. Malerba, “Classification of symbolic objects: A lazy learning approach”, *Intelligent Data Analysis*, Vol.*10*, No.4, pp. 301-324, 2006.

[6] D. Malerba, F. Esposito, V. Gioviale, & V. Tamma, “Comparing dissimilarity measures for symbolic data analysis”, *Proceedings of Exchange of Technology and Know-how and New Techniques and Technologies for Statistics*, Vol.*1*, pp. 473-481, 2001.

[7] D. Ziani, “Feature selection on Boolean symbolic objects”, *arXiv preprint arXiv:1405.0647*, 2014.

[8] Y. El-Sonbaty and M. A. Ismail, “Fuzzy clustering for symbolic data”, *IEEE Transactions on fuzzy systems*, Vol.*6, No.*2, pp. 195-204, 1998

[9] M. K. Ng, M. J. Li, J. Z. Huang, & Z. He, “On the impact of dissimilarity measure in k-modes clustering algorithm”, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, No.3, pp. 503-507, 2007

[10] E. Natsheh, "Taxonomy of Clustering Methods Used in Fuzzy Logic Systems", *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, vol.4, no.1, pp. 65-72, 2012

[11] K. C. Gowda and G. Krishna, "Disaggregative Clustering Using the Concept of Mutual Nearest Neighborhood", *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. SMC-8, No. 12, pp. 888-895, 1978

[12] K. C. Gowda and G. Krishna, "The Condensed Nearest Neighbor Rule Using the Concept of Mutual Nearest Neighborhood", *IEEE Transaction on Information Theory*, Vol. 25, No. 4, pp. 488-490, 1979.

[13] R. R. Yager and D. P. Filev, "*Essentials of Fuzzy Modeling and Control*", Wiley India, 2002.

[14] T. Takagi and M. Sugeno, "Fuzzy Identification of Systems and Its Applications to Modeling and Control", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-15, No. 1, pp.116-132, 1985

[15] S. Chiu, "Fuzzy Model Identification Based on Cluster Estimation", *J. of Intelligent & Fuzzy Systems*, Vol. 2, No. 3, pp. 267-278, 1994.

[16] S. Chiu, "Selecting Input Variables for Fuzzy Models", *J. of Intelligent & Fuzzy Systems*, Vol. 4, No. 4, pp. 243-256, 1996

[17] J. Yen and R. Langari, "Fuzzy Logic: Intelligence, Control and Information", Prentice-Hall, 1999.

[18] J. R. Jang, C. Sun, and E. Mizutani, "Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligent", Prentice-Hall, 1997.

[19] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, "*Numerical Recipes in C: The Art of Scientific Computing*", 3rd Edition, Cambridge University Press, 2007.

[20] J. Jantzen, "Foundations of Fuzzy Control: A Practical Approach", 2nd Edition, Wiley, 2013.

[21] M. Golob, "Decomposition of a Fuzzy Controller Based on the Inference Break-up Method", *Intelligent Data Analysis*, Vol. 3, Issue 2, pp. 127-137. 1999.

[22] E. Natsheh and K. Buragga, "Comparison between Conventional and Fuzzy Logic PID Controllers for Controlling DC Motors", *International Journal of Computer Science Issues (IJCSI)*, vol. 7, no. 5, pp. 128-134, 2010.

[23] E. Natsheh and K. Buragga, "A Performance Comparison for Various Methods to Design the Three-Term Fuzzy Logic Controller", *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 10, no. 5, pp. 142-151, 2010.

[24] E. Natsheh, "Designing and Tuning PID Fuzzy Controllers for Armature-Controlled DC Motor", *International Journal of Human and Technology Interaction (IJHaTI)*, vol.2, no.2, pp. 17-24, 2018

[25] E. Natsheh, "Designing a Fuzzy Logic Controller with a Non-Parametric Similarity-Based Clustering Algorithm", *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, vol.12, no.4, pp. 21-29, 2020.

[26] I. Gath and A. B. Geva, "Unsupervised Optimal Fuzzy Clustering", *IEEE Transactions on Pattern Analysis and Machine Intelligen*ce, Vol. 11, No. 7, pp. 773-781, 1989