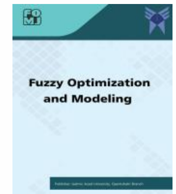
FUZZY OPTIMIZATION AND MODELLING 1(1) (2020) 11-31
E-ISNN: 2676-7007

Contents lists available at FOMJ

Fuzzy Optimization and Modelling

Journal homepage: http://fomj.qaemiau.ac.ir/



Paper Type: Original Article

A Hybrid Algorithm for Fault Diagnosis using Fuzzy Clustering Tools

Adrián Rodríguez Ramos[a], **Pedro Juan Rivera-Torres**[b], **Antônio José da Silva Neto**[c], **Orestes Llanes-Santiago**[a,*]

[a] *Departamento de Automática y Computación, Universidad Tecnológica de la Habana José Antonio Echeverría, CUJAE, La Habana, Cuba.*
[b] *Departamento de Ciencias de Computos, Universidad de Puerto Rico, Recinto de Río Piedras, San Juan, Puerto Rico.*
[c] *Instituto Politécnico da Universidade do Estado do Rio de Janeiro (IPRJ/UERJ), Nova Friburgo, RJ, Brazil.*

A R T I C L E I N F O

Article history:
Received 1 May 2020
Revised 26 July 2020
Accepted 25 August 2020
Available online 20 September 2020

Keywords:
Automatic learning
Online detection
Fuzzy clustering tools
Optimal parameters

A B S T R A C T

In this paper, a hybrid algorithm using fuzzy clustering techniques is proposed for developing a robust fault diagnosis platform in industrial systems. The proposed algorithm is applied in a fault diagnosis scheme with online detection of novel faults and automatic learning. The hybrid algorithm identifies the outliers based on data density. Later, the outliers are removed, and the clustering process is performed. To extract the important features and improve the clustering, the maximum-entropy-regularized weighted fuzzy c-means is used. The use of a kernel function allows achieving a greater separability among the classes by reducing the classification errors. Finally, a step is used to optimize the parameters m (regulation factor of the fuzziness of the resulting partition) and   (bandwidth, and indicator of the degree of smoothness of the Gaussian kernel function). The proposed hybrid algorithm was validated using the Tennessee Eastman (TE) process benchmark. The results obtained indicate the feasibility of the proposal.

1. Introduction

Nowadays, there is a marked necessity in industrial plants to efficiently produce goods with higher quality, while complying with environmental and industrial safety regulations [29, 60]. However, faults in industrial devices can have an unfavourable impact in the availability of systems, the environment, and operator safety. For such reasons, faults need to be detected and isolated; being these tasks associated to fault diagnosis systems [60].

Within fault diagnosis methods, two main categories are distinguished; those that are based on models [6, 10-12, 15, 60] and those based on historical process data [5, 23, 44, 55]. In the first approach, the use of models to represent process behaviour is needed. The tools used in these approaches are based on residue generation obtained from the difference between measurable signals collected from the real process, and values obtained from the model. This implicates an elevated knowledge about the characteristics of the processes, their parameters, and operation zones. However, this is usually very difficult to achieve due to the complexity of industrial plants. On the other hand, approaches based in historical data do not require considerable prior knowledge of the process parameters [62]. These characteristics constitute an advantage for complex systems, where relationships among

[*] Corresponding author
E-mail Address: adrian.rr@automatica.cujae.edu.cu (Adrián Rodríguez Ramos), pedro.rivera1@upr.edu (Pedro Juan Rivera-Torres), ajs_neto@uol.com.br (Antônio José da Silva Neto), orestes@tesla.cujae.edu.cu (Orestes Llanes-Santiago).

variables are nonlinear, and therefore, it is very difficult to obtain an analytical model that efficiently describes the dynamics of the process.

Most classifiers used in fault diagnosis systems based in historical data are trained for classifying process states that are determined by experts. However, with the decrease of the useful life of automation technical devices such as sensors, actuators and pumps, among others, due to their continued use, the probability of occurrence of new faults increases. In this situation, the diagnostic systems will not classify new faults correctly, which will cause an erroneous decision-making. For this reason, topics related to automatic identification of new patterns have gained great importance in the area of fault diagnosis [16, 20, 28, 66]. This is because fault diagnosis systems should be able to detect new faults and to incorporate them, after an identification process, in their knowledge base. This allows fault diagnosis systems to incorporate automatic learning mechanisms, which enables adaptating to new conditions.

By analyzing the different mathematic and computational tools recently developed for control and fault diagnosis tasks, there is a significate increase in the use of the fuzzy clustering techniques [4, 9, 30, 46, 51, 53, 74]. They deal with the uncertainty and vagueness that can be found in a wide variety of applications, as for example: image processing, pattern recognition, object recognition, and modelling and identification [33, 35, 52, 72]. The main focus of all fuzzy clustering techniques is to improve the clustering by avoiding the influence of noise and outlier data present in industrial systems.

The Fuzzy C-Means (FCM) algorithm is one of the most widely used for clustering due to its robust results for overlapped data [7]. Unlike the K-means algorithm, data points in the FCM algorithm may belong to more than one cluster center. The FCM algorithm obtains very good results with noise-free data, but is highly sensitive to noisy data and outliers [27]. Other similar techniques, such as Possibilistic C-Means (PCM) [34] and Possibilistic Fuzzy C-Means (PFCM) [43] interpret clustering as a possibilistic partition and work better than FCM in the presence of noise. PCM, however, fails to find optimal clusters in the presence of noise [27], and PFCM does not yield satisfactory results when the data set consists of two clusters, which are highly different in size and have outliers [27, 31]. The Noise Clustering (NC) [17, 18], Credibility Fuzzy C-Means (CFCM) [14], and Density Oriented Fuzzy C-Means (DOFCM) [32] algorithms were proposed specifically to work efficiently with noisy data. Recently, the DOFCM algorithm has been applied with excellent results [48, 49].

A linear relationship between the variables is a typical consideration made by many methods currently applied to data analysis, such as clustering techniques. This assumption, however, is rarely satisfied in industrial processes, where the relationship between variables is highly nonlinear. For these cases, data analysis should be performed by using algorithms that take into account such relationships. Among the algorithms that can be applied to achieve this objective are kernel methods [71]. The main characteristic of these methods is the use of kernel functions for projecting the data to greater dimensionality spaces, in order to find linear relationships between the variables for applying conventional methods of classification, grouping, and estimation of data [26]. Recent papers show the possibility of reducing classification errors by using kernel functions [47, 50].

Many attribute-weighted fuzzy clustering methods have been proposed in the last times. In [64], the weighted Euclidean distance is used to replace the general Euclidean distance in FCM. On the other hand, [8] carries out clustering on the selected subspace, instead of the full data space by directly assigning zero weights to features which have little information. Recently, [19] presented an enhanced soft subspace clustering (ESSC) algorithm by employing both within-cluster and between-cluster information [19]. In [41], a novel subspace clustering technique has been proposed by introducing feature interaction using the concepts of fuzzy measures and the Choquet integral. In [56], a survey of weighted clustering technologies is given. Finally, in [73], a Maximum-Entropy-Regularized Weighted Fuzzy C-means (EWFCM) algorithm is proposed, to extract the important features and improving the clustering. In EWFCM algorithm, the attribute-weight entropy regularization is defined in the new objective function to achieve optimal distribution of attribute weights. Therefore, it is possible simultaneously minimize the dispersion within clusters and maximize the entropy of attribute weights to stimulate important attributes for contributing to the identification of clusters. Moreover, the kernel based EWFCM (KEWFCM) clustering algorithm is executed for clustering the data with non-spherical shaped clusters.

Another problem usually presents in fuzzy clustering methods is the significative dependency of their performance to the adequate selection of their parameters [6, 54].

In order to overcome the problems above mentioned, a hybrid algorithm, named Density Oriented Kernel Maximum Entropy Regularized Weighted Fuzzy C-means (DOKEWFCM), using fuzzy clustering techniques is proposed which constitutes the main contribution of this paper.

The novel hybrid algorithm developed in this paper presents the following characteristics:

- It identifies the outliers based in data density for improving the clustering process.
- It extracts the variables with greater contribution to the clustering process using the Maximun Entropy Regularized Weighted Fuzzy C-means method for improving the classification process.
- It uses a kernel function for improving the classification process.
- It analyses the observations classified as noise in a window of time, and determines if they form a new class.

The algorithm is applied in a fault diagnosis scheme with online detection of novel faults and automatic learning, with the ability to analysing observations classified as noise in a window of time, to determine if they belong to a new class, which constitutes another contribution of this paper.

The fault diagnosis scheme based on the proposed algorithm is robust for the noise and disturbances affecting the plant, decreasing the false alarm rate (FAR) and increasing the fault detection rate (FDR) indexes.

The performance of this proposal is evaluated in two case studies: the UCI machine learning datasets and Tennessee Eastman (TE) process.

The organization of the paper is as follows: in Section 2, a description of the hybrid algorithm DOKEWFCM based on fuzzy clustering techniques is presented. Next, in Section 3, a description of the proposed algorithm used in a fault diagnosis scheme with online detection and automatic learning is displayed. Section 4 shows the case study used to validate the proposed novel algorithm, the design of experiments as well as an analysis of the results. Finally, the conclusions are presented.

2. The hybrid algorithm DOKEWFCM

2.1. Kernel-based DOEWFCM (DOKEWFCM)

The DOKEWFCM algorithm is intended as a hybrid algorithm which uses the ability of the DOFCM algorithm [32] to detect and eliminate outliers in a dataset, and the capacity of KEWFCM algorithm [77] to extract important features and improve the clustering process. The use of kernel functions is performed for clustering the data with non-spherical shaped clusters. This allows achieving greater separability among the classes for reducing classification errors. Figure 1 shows the procedure performed by the proposed DOKEWFCM algorithm.

The algorithm creates $c + 1$ clusters with c good clusters and one cluster of noise. This algorithm identifies outliers before the construction of the clusters, based on a density analysis of the dataset.

The neighbourhood of a given radius of each point in a dataset has to contain at least a minimum number of other points. DOKEWFCM defines a density factor, called neighbourhood membership, which expresses the density measure of an object in relation to its neighbourhood.

The neighborhood membership of a point i in the dataset X is defined as:

$$M^{i}_{neighborhood} = \frac{\eta^{i}_{neighborhood}}{\eta_{max}} \tag{1}$$

where $\eta^{i}_{neighborhood}$ is the number of points in the neighborhood of point i; η_{max} is the maximum number of points in the neighborhood of any point in the dataset X.

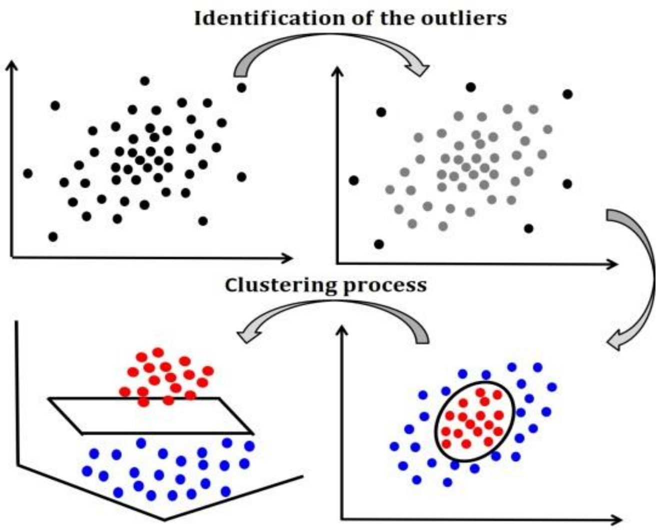


Figure 1: Procedure performed by the DOKEWFCM algorithm

If the point q is in the neighborhood of point i, q will satisfy:

$$q \in X \mid dist(i,q) \leq r_{neighborhood} \tag{2}$$

where $r_{neighborhood}$ is the radius of neighborhood, and $dist(i,q)$ is the distance between points i and q.

Neighborhood membership of each point in the dataset X is calculated using Eq. (1). The threshold value α is selected from the complete range of neighborhood membership values, depending on the density of points in the dataset. The point will be considered as an outlier if its neighborhood membership is less than α. Let i be a point in the dataset X, then:

$$\begin{cases} M^{i}_{neighborhood} < \alpha & then\, i\; is\, an\, outlier \\ M^{i}_{neighborhood} \geq \alpha & then\, i\; is\, an\, non-outlier \end{cases} \tag{3}$$

The parameter α can be selected from the range of $M^{i}_{neighborhood}$ values after observing the density of the points in the dataset and it should be close to zero. Ideally, a point will be classified as outlier only if no other point is present in its neighborhood, i.e., when neighborhood membership is zero or threshold value α =0. However, in this scheme, a point is considered as an outlier when its neighborhood membership is less than α, where α is a critical parameter to identify the outlier points. Its value will depend upon the nature of dataset, i.e., taking into account the density of the dataset, then, its value will vary for different datasets.
After the outliers are identified, the clustering process is performed. In this case, the objective function is defined as:

$$J = \sum_{i=1}^{c+1}\sum_{k=1}^{N}\left(\mu_{ik}\right)^{m}\sum_{l=1}^{M} w_{il}\Phi\left(x_{kl}\right) - \tilde{v}_{il}^{\,2} + \gamma^{-1}\sum_{i=1}^{c+1}\sum_{l=1}^{M} w_{il}\log\left(w_{il}\right) \tag{4}$$

subject to $0 \leq \sum_{i=1}^{c}\mu_{ik}$ and $\sum_{l=1}^{M} w_{il}$ for $0 \leq w_{il} \leq 1$, where $\mathbf{U} = [\mu_{ik}]_{cxN}$ is the membership degree matrix in the original space and $\mathbf{W} = [w_{il}]_{cxM}$ is the attribute weight matrix in the original space. $\widetilde{\mathbf{V}} = [\tilde{v}_{il}]_{cxM}$ is the cluster center matrix in the kernel space, Φ is the non-linear mapping from the original feature space to the kernel space. In this case, the

Gaussian kernel is used, $K\left(x_{kl}, \tilde{v}_{il}\right) = e^{-\left\|x_{kl} - \tilde{v}_{il}\right\|^2 / \sigma^2}$. The matrices **V** and **W** are updated according to the Eqs. (5) and (6), respectively. For this case, in Eq. (5) must be observed that: $i = 1, \ldots, c$.

$$\tilde{v}_{il} = \frac{\sum_{k=1}^{N} \left(\mu_{ik}\right)^m K\left(x_{kl}, \tilde{v}_{il}\right) x_{kl}}{\sum_{k=1}^{N} \left(\mu_{ik}\right)^m K\left(x_{kl}, \tilde{v}_{il}\right)} . \tag{5}$$

$$w_{il} = \frac{\exp\left(-\gamma \sum_{k=1}^{N} (\mu_{ik})^m \|\boldsymbol{\Phi}(\boldsymbol{x_{kl}}) - \tilde{\boldsymbol{v}}_{\boldsymbol{il}}\|^2\right)}{\sum_{s=1}^{M} \exp\left(-\gamma \sum_{k=1}^{N} (\mu_{ik})^m \|\boldsymbol{\Phi}(\boldsymbol{x_{kl}}) - \tilde{\boldsymbol{v}}_{\boldsymbol{il}}\|^2\right)} \tag{6}$$

Membership function μ_{ik} is modified as:

$$\mu_{ik} = \begin{cases} \sum_{j=1}^{c} \left(\sum_{l=1}^{M} w_{il} \|\boldsymbol{\Phi}(\boldsymbol{x_{kl}}) - \tilde{\boldsymbol{v}}_{\boldsymbol{il}}\|^2 \Big/ \sum_{l=1}^{M} w_{jl} \left\|\boldsymbol{\Phi}(\boldsymbol{x_{kl}}) - \tilde{\boldsymbol{v}}_{\boldsymbol{jl}}\right\|^2 \right)^{-\frac{1}{m-1}} & if\ non\ outlier \\ 0 & if\ outlier \end{cases} \tag{7}$$

The DOKEWFCM algorithm is presented in Algorithm 1.

The stopping criteria implemented in this algorithm are:

- Criterion 1: Maximum number of iterations (*Itr_max*).
- Criterion 2: $U_t - U_{t-1} < \varepsilon$ (ε is a tolerance limit prescribed a priori, and *t* is an iteration counter).

2.2. An illustrative example: UCI machine learning datasets

To validate the new algorithm were used a benchmark dataset from the UCI Machine Learning Repository [3], such as Iris, Glass, Ionosphere, Haberman and Heart. New observations were added to the original datasets and evenly distributed among the classes in order to represent all the possible outliers for each class. Table 1 gives an overview of the datasets modified.

Table 1: Description of the datasets modified

Dataset	No. of elements	No. of variables	No. of classes	Elements in each classes
Iris	198 (48 outliers)	4	3	(66, 66, 66)
Glass	279 (65 outliers)	9	6	(90, 96, 22, 18, 14, 39)
Ionosphere	421 (70 outliers)	34	2	(156, 265)
Haberman	366 (66 outliers)	3	2	(265, 101)
Heart	320 (50 outliers)	13	2	(180, 140)

Algorithm 1 DOKEWFCM

Input: data, c, $\varepsilon > 0$, $m > 1$, $\gamma > 0$

Output: data without outliers **Xp, U, V, W**

Identification of the outliers (Step 1):

Calculate neighborhood radius.

Calculate $\eta^{i}_{neighborhood}$ according to Eq. (2).

Select η_{max}.

Calculate $M^{i}_{neighborhood}$ according to Eq. (1).

With the value of α, identify outliers according to (3).

Clustering process (Step 2):

Initialize **U** to random fuzzy partition

Initialize **W** of each attribute.

for *l = 1 to l = Itr_max*

  Update **V** according to Eq. (5)

  Update **U** according to Eq. (7)

  Update **W** according to Eq. (6)

  Verify stopping criterion: $U_t - U_{t-1} < \varepsilon$

end for

To evaluate the performance of the proposed algorithm (DOKEWFCM), the KEWFCM algorithm described in [73] was selected to perform a comparative analysis. In addition, other recent algorithms (GAKFCM [21], FC-PFS [59]) with excellent results were selected to make this comparison. The values of the common parameters for these algorithms are: *Itr_max* = 100, $\varepsilon = 10^{-5}$, *m* = 2. The specific parameters are:

- KEWFCM: γ = 0.05 and σ = 10.
- GAKFCM: σ = 10, crossover rate p_{co} = 0.6 and mutation rate p_{mo} = 0.001.
- FC-PFS: α = 0.6 (where $\alpha \in (0,1]$ is an exponent coefficient used to control the refusal degree in picture fuzzy sets).

Each algorithm was executed on each dataset 10 times. In order to make the comparative analysis, the classification rate was used as a performance metric. The classification rate is a measure used to determine how well clustering algorithms perform on a given dataset with a known clustering structure [73]. It can be measured by Eq. 8, which is expressed as a percentage in this paper. For 10 times, the average classification rate (ACR) is used in the experiments.

$$CR = \frac{\sum_{i=1}^{c} d_i}{N} \tag{8}$$

where d_i is the number of objects correctly identified in the c^{th} cluster, and N is the number of all objects in the dataset. Table 2 shows the results of the comparison where the proposed algorithm obtains the best results.

Table 2: Comparison of various clustering algorithms with the one proposed here based on their respective ACR

Dataset	GAKFCM	FC-PFS	KEWFCM	DOKEWFCM
Iris	72.73	85.02	89.81	97.33
Glass	44.58	47.61	40.97	57.75
Ionosphere	60.25	62.87	67.33	79.20
Haberman	64.02	63.38	66.46	77.45
Heart	55.58	60.22	59.89	73.78

Figure 2 shows the classification process for the Iris dataset. The DOKEWFCM algorithm is able to identify the outliers (shown in black color). Later, the algorithm classifies the observations after the outliers were eliminated (Fig. 3).

Table 3 shows the attribute weight assignment performed by DOKEWFCM algorithm on Iris dataset. It is clear that attributes 3 and 4 contributed much more than the other two attributes in the clustering process, since the algorithm assigns higher weights to these two attributes.

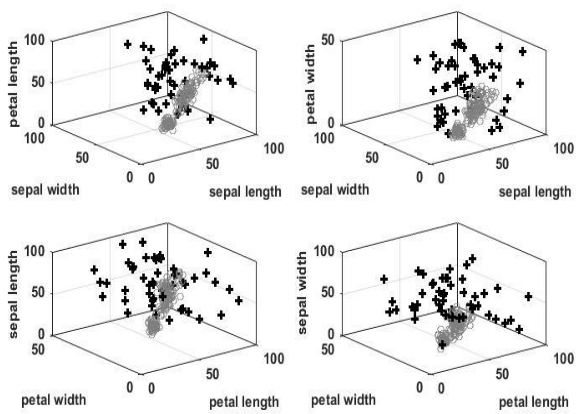


Figure 2: Identification of the outliers with DOKEWFCM algorithm

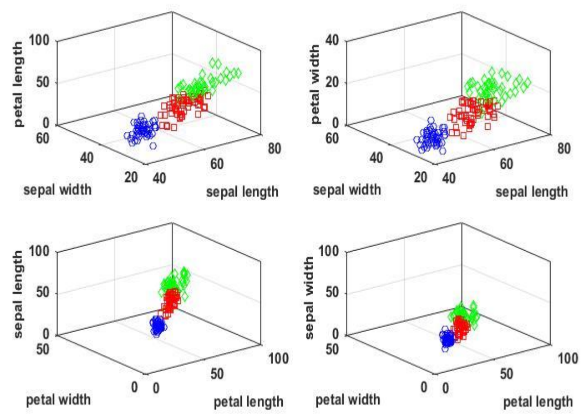


Figure 3: Classification process with DOKEWFCM algorithm

Table 3: Attribute weight assignment of DOKEWFCM algorithm on Iris dataset.

	Attribute 1	Attribute 2	Attribute 3	Attribute 4
Cluster 1	0.0006	0.0018	**0.8557**	**0.1419**
Cluster 2	0.0002	0.0004	**0.5849**	**0.4145**
Cluster 3	0.0012	0.0004	**0.6766**	**0.3218**

2.2.1. Statistical tests

However, to know if there are significant differences between them, it is necessary to apply statistical tests [24, 25, 39]. First, the non-parametric Friedman test is applied in order to find if there are significant differences among the results obtained with each algorithm. Later on, if there are significant differences among the results obtained by each algorithm, it is necessary to make a comparison in pairs to determine the best algorithm. For this, the non-parametric Wilcoxon test is applied.

Friedman Test:

The results using the Iris dataset are shown below. In this case, for four experiments (*k = 4*) and 10 datasets (*N = 10*), the value of statistical Friedman, $F_F = \frac{270}{0} \to \infty$, was obtained. With *k = 4* and N = *10*, F_F is distributed according to the *F* distribution with 4 – 1 = 3 and $(4-1)(10-1)=27$ degrees of freedom. The critical value of *F* (3, 27) for a level of significance α = 0.05 is 2.9604, so we reject the null-hypothesis (*F* (3,27) << F_F) which means that at least the average performance of at least one algorithm is significantly different from the other algorithms. For the remaining datasets (Glass, Ionosphere, Haberman and Heart) the same results were obtained when applying the Friedman test.

Wilcoxon Test:

Table 4 shows the results of the comparison in pairs of algorithms (1: GAKFCM, 2: FC-PFS, 3: KEWKFCM, 4: DOKEWFCM) using the Wilcoxon test for the Iris dataset. The first two rows contain the values of the sum of the positive (R^+) and negative (R^-) ranks for each comparison established. The next two rows show the statistical values *T* and the critical value of *T* for a significance level α = 0.05. The last row indicates which algorithm was the winner in each comparison. The summary in Table 5 shows the times that each algorithm was the winner using all datasets. These results validate the main purpose of this paper, i.e. the proposal of a new hybrid fuzzy clustering algorithm.

Table 4: Results of the Wilcoxon test for the iris dataset.

	1 vs 2	**1 vs 3**	**1 vs 4**	**2 vs 3**	**2 vs 4**	**3 vs 4**
$\sum R^+$	0	0	0	0	0	0
$\sum R^-$	55	55	55	55	55	55
T	0	0	0	0	0	0
T α **= 0.05**	8	8	8	8	8	8
Winner	2	3	4	3	4	4

Table 5: Final result of the comparison between test algorithms

	Iris		**Glass**		**Ionosphere**		**Haberman**		**Heart**		**Final result**	
Algorithm	No. Wins	Ranking	No. Wins	Ranking	No. Wins	Ranking	No. Wins	Ranking	No. Wins	Ranking	No. Wins	Ranking
GAKFCM	0	4	1	3	0	4	1	3	0	4	2	4
FC-PFS	1	3	2	2	1	3	0	4	2	2	6	3
KEWFCM	2	2	0	4	2	2	2	2	1	3	7	2
DOKEWFCM	3	1	3	1	3	1	3	1	3	1	**15**	**1**

3. Novel fault diagnosis scheme with online detection and automatic learning

In Fig. 4, the scheme of the fault diagnosis approach with online detection of novel faults and automatic learning using the DOKEWFCM algorithm proposed in this paper is presented. It presents an offline training stage and an online recognition stage. In the training stage the historical data of the process are used to train (modelling the functional stages through the clusters) a fuzzy classifier. After the training, the classifier is used online (recognition) in order to process every new sample taken from the process. In this stage, the observations which are not classified in the known classes and belonging to a window of time are analysed to determine if they constitute a new class representative of a new fault or if they are noise. If a new class is identified, experts characterize the new fault and it is incorporated to the knowledge base of known faults and the classifier is trained again.

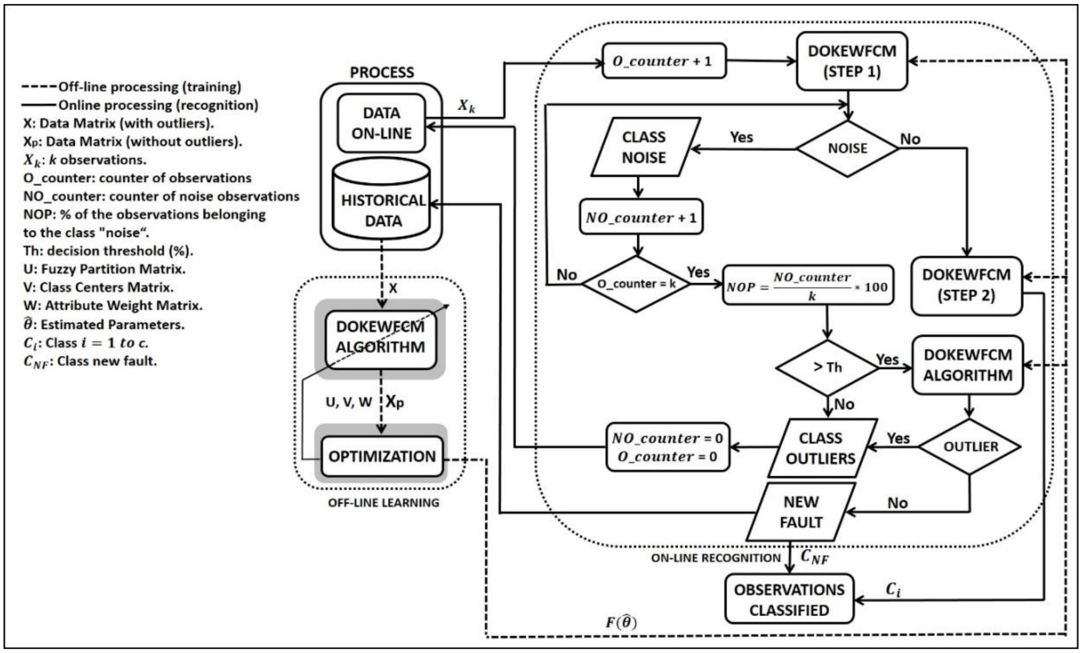


Figure 4: Classification scheme proposed using fuzzy clustering (similar to the architecture proposed by the authors in [75])

The clustering methods group the data in different classes based on a measure of similitude. In the processes, the data are acquired by means of a SCADA (Supervisory Control and Data Acquisition) system, and the classes can be associated to functional states. In the case of statistical classifiers, each sample is compared with the center of each class by means of a measure of similitude to determine to which class the sample belongs. In the case of the fuzzy classifiers, the comparison is made to determine the membership degree of the sample to each class. In general, the higher membership degree determines the class to which the sample is assigned, as it is shown in (9).

$$C_i = \left\{ i : \max\left\{ \mu_{ik} \right\}, \forall i, k \right\} \tag{9}$$

The operation of the proposed fault diagnosis scheme is presented below.

3.1. Offline training

Initially, the center of the known classes is determined by using a historical dataset representative of the different operation states of the process. In the proposed technique, a set of N observations (data points) are classified into $c+1$ classes using the DOKEWFCM algorithm. The c classes represent the normal operation conditions (NOC) of the process, and the faults to be diagnosed.

Following on, a step to optimize the parameters of the DOKEWFCM algorithm is implemented. In this step, the parameters m and σ are estimated to optimize a validity index using an optimization algorithm. This will allow obtaining an improved **U** partition matrix, and therefore, a better position of the center of each class that characterize the different operation states of the system. Later, the estimated values of m and σ in Eq. (4) will be used during the online recognition, and it will contribute to improve the classification of the samples obtained by the data acquisition system from the system.

The validity measures are indexes for evaluating quantitatively the result of a clustering method and compare its behavior when its parameters vary. Some indexes evaluate the resulting **U** matrix, while others are focused on the geometry of the resulting structure. The partition coefficient (*PC*) [36, 42, 65], which measures the fuzziness degree of the partition **U**, is used as validity measure in this case. Its expression is shown in the Eq. (10)

$$PC = \frac{1}{N}\sum_{i=1}^{c}\sum_{k=1}^{N}\left(\mu_{ik}\right)^2 \tag{10}$$

If the partition U is less fuzzy, the clustering process is better. Being analyzed in a different way, it allows the measuring of the degree of overlapping among the classes. In this case, the optimum comes up when *PC* is maximized, i.e., when each pattern belongs to only one group. Likewise, minimum comes up when each pattern belongs to each group.

Then, the optimization problem is defined as**:**

$$\begin{aligned} &\max\{PC\} = \frac{1}{N}\sum_{i=1}^{c}\sum_{k=1}^{N}\left(\mu_{ik}\right)^2 \\ &s.t. \\ &\qquad m_{min} < m \le m_{max}\ , \\ &\qquad \sigma_{min} \le \sigma \le \sigma_{max}. \end{aligned}$$

Taking into account the last definition, a range of values of *m* and σ should be defined. It is known that $1 < m < \infty$, but from the practical point of view in many applications, the value of *m* does not exceed two [33, 35, 59, 72], therefore in this case: $1 < m \le 2$. The parameter σ is called bandwidth, and it indicates the degree of smoothness of the function. If the value of σ is overestimated, the function tends to show a linear behavior and its projection in high-dimensional space loses its ability to separate non-linear data. Meanwhile, if the value of σ is underestimated, the result will be highly sensitive to the noise present in the data. Then, the search space of the algorithm must be large, so that during the exploration, small and large values will be considered. In this paper, a group of experiments were developed, and it was found that an appropriate range was: $0.25 \le \sigma \le 20$.

In many scientific areas, and in particular the fault diagnosis field, bio-inspired algorithms have been widely used with excellent results [13, 37, 38] to solve optimization problems. They can efficiently locate the neighborhood of the global optimum in most opportunities, with an acceptable computational time. There is a large number of bio-inspired algorithms, in their original and improved versions. Some examples are Genetic Algorithm (GA), Differential Evolution (DE), Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO) among others. In this proposal, the DE algorithm [1, 57] will be used to obtain the optimum values of the parameters *m* and σ due to its easy implementation and excellent outcomes [47-50].

3.2. Online recognition

In this stage, the fuzzy clustering algorithm is modified and the update of the center of each class is not developed. The principal reason of doing this modification is to avoid the incorrect displacement of the center of each class due to an unknown fault of small dimensions with a high latency time.

In this stage, a window of time with *k* observations and the parameter *Th* are established by experts. The value of *k* is related with the characteristics of process, and it represents the quantity of sampling times that the experts consider enough for analyzing the presence of a possible new fault. *Th* represents the percentage of observations established by the experts to decide if the group of observations classified as noise in *k* sampling times should be analyzed to determine if they represent a new fault. When an observation x_k arrives, the DOKEWFCM algorithm (Step 1: Identification of the outliers) classifies it as *noise* or as *good* taking into account the results of the training. If the observation is classified as a *good* sample, the DOKEWFCM algorithm (Step 2: Clustering process) identifies to which of the known classes C_i it belongs to. However, if the observation is classified as noise, it is stored and a counter of noise observations (*NO*) is incremented. This procedure is repeated until the window of time of *k* observations has been completed.

After the *k* observations are classified, the percentage classified as noise (*NOP=NO*100/k*) is determined. If *NOP* < *Th*, the noise observations are not considered representative of a new class, and the *NO* counter is restarted to begin a new cycle. If *NOP* > *Th*, the *NO* observations are analyzed to determine whether they constitute a new class or if they are outliers. The possibility that the new class represents a new operation condition is not considered here because it should be known by the technical operators and the diagnosis system should be updated before to the system begins to work in that new operation condition.

To analyze the noise observations, the DOKEWFCM algorithm is used. It is based on the fact that the outliers are dispersed data with low density and they do not form a cluster. However, when a new fault occurs, the data will be concentrated (high density) by forming a cluster which characterizes a new state. The DOKEWFCM algorithm is applied to the noise observations to determine, based on the density of the data, whether they are outliers or represent the pattern of a new class. If the noise observations constitute a new class, the experts should identify whether the pattern corresponds to a single fault or the pattern is the result of several single faults acting simultaneously (multiple fault). After identifying and characterizing the pattern, it will be store in the historical database used in the training stage. Later on, the classifier should be trained again, and the procedure of online recognition will be repeated systematically.

The procedure explained for the online stage represents a mechanism of online detection of novel faults with automatic learning for a fault diagnosis system. It is described in Algorithm 2.

4. Benchmark case study: Tennessee Eastman process

4.1. Online recognition

In order to apply the proposed classification scheme to fault diagnosis, the Tennessee Eastman (TE) chemical process benchmark was selected. This process is widely used as a chemical benchmark to evaluate the performance of new control and monitoring strategies [5, 63, 68]. The chemical process plant consists of five major units: a reactor, a condenser, a recycle compressor, a separator, and a stripper; all interconnected. The control objectives and general features of the process simulation are described in the papers [22].

The TE process contains 21 preprogrammed faults and one normal operating condition dataset. The datasets from TE are generated along 48 hours of operation, with the inclusion of faults after 8 simulation hours. Each historical dataset contains 52 variables (41 measured variables, plus 11 manipulated variables) with a sampling time of 3 min and Gaussian noise incorporated in all measurements. For the study performed in this paper, only the 33 variables available online are considered, as shown in Table 6.

Table 6: Monitored variables in the Tennessee Eastman process

No.	Variable	No.	Variable
1	A feed	18	Stripper temperature
2	D feed	19	Stripper steam flow
3	E feed	20	Compressor work
4	Total feed	21	Reactor cooling water outlet temperature
5	Recycle flow	22	Separator cooling water outlet temperature
6	Reactor feed rate	23	D feed flow valve
7	Reactor pressure	24	E feed flow valve
8	Reactor level	25	A feed flow valve
9	Reactor temperature	26	Total feed flow valve
10	Purge rate	27	Compressor recycle valve
11	Product separator temperature	28	Purge valve
12	Product separator level	29	Separator pot liquid product flow valve
13	Product separator pressure	30	Stripper liquid product flow valve
14	Product separator underflow	31	Stripper steam valve
15	Stripper level	32	Reactor cooling water flow
16	Stripper pressure	33	Condenser cooling water flow
17	Stripper underflow		

A description of simulated faults is shown in Table 7. All datasets used in this paper can be downloaded from http://web.mit.edu/braatzgroup/TE\underline\process.zip. Furthermore, 2100 new observations evenly distributed among the classes were added to this dataset in order to represent the possible outliers of each class.

Table 7: Description of faults in TE process

Fault	Description	Type
1	A/C feed ratio, B composition constant	Step
2	B composition, A/C ratio constant	Step
3	D feed temperature	Step
4	Reactor cooling water inlet temperature	Step
5	Condenser cooling water inlet temperature	Step
6	A feed loss	Step
7	C header pressure loss-reduced availability	Step
8	A, B, and C feed composition	Random variation
9	D feed temperature	Random variation
10	C feed temperature	Random variation
11	Reactor cooling water inlet temperature	Random variation
12	Condenser cooling water inlet temperature	Random variation
13	Reaction kinetics	Slow drift
14	Reactor cooling water valve	Sticking
15	Condenser cooling water valve	Sticking
16-20	Unknown	Unknown
21	The valve was fixed at the steady-state position	Constant position

4.2. Analysis and discussion of results

A very important step in the design of a fault diagnosis system is to verify the quality of its performance. The most used criterion for this analysis is the confusion matrix (CM) [47]. The confusion matrices using a cross validation process were obtained [49]. Fig. 6 shows the cross-validation process for four partitions of the data set. In the experiments implemented in the TE process, a cross-validation with 10 partitions of the training dataset was performed.

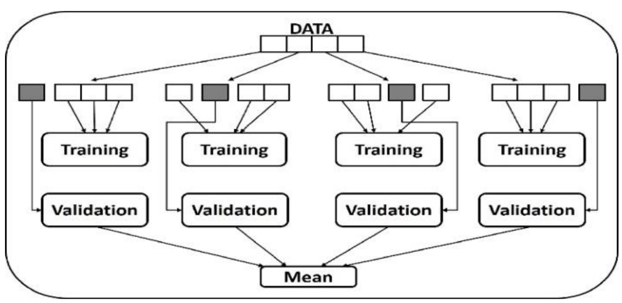


Figure 6: Cross validation process

4.2.1. Offline training stage

In this subsection, the results of the offline training stage are shown. The principal objectives in this stage are: the trainning of the classification algorithm, obtaining the parameters m and σ, and the selection of the variables with greater contribution in the clustering for the analyzed classes.

The values used for beginning the training were: Number of iterations =100, $\varepsilon = 10^{-5}$ and initial values of $m = 2$ and $\sigma = 1$.

The diagnostic system was not trained to recognize faults 16-20 with the objective of using them to test the online detection of new faults. These faults were only simulated in the online recognition stage.

To estimate the best values for m and σ parameters as was mentioned in subsection 3.1, the DE algorithm was used due to its advantages, specifically its simple structure, higher speed, and robustness [13]. The control parameters in DE are the size of the population Z, the crossover constant C_R and the scaling factor F_S. The values of the parameters for the DE algorithm, considering a search space $1 < m \le 2$ and $0.25 \le \sigma \le 20$ where, $C_R = 0.5$, $F_S = 0.1$, $Z = 10$, *Eval_max = 100* and *PC > 0.9999*. The behavior of the objective function (PC) is shown in Fig. 7 where it is possible to appreciate how the DE algorithm rapidly converges. The best parameters were obtained from iteration 12: m = 1.0389 and σ = 17.8870.

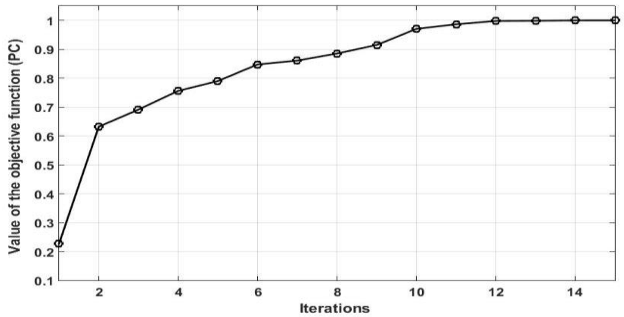


Figure 7: Value of the objective function (PC)

Table 8 shows the variables or attributes with the greater contribution to clustering (higher weight values) for the analyzed classes (operating states). To obtain these attributes, a parameter called weight threshold (Tw) must be selected from the expert criterion. If the weight of the attribute l is greater than Tw, then the attribute l is selected. Figure 8 shows an example of selection of the attributes considering the NOC (Tw = 0.03). Later, the attributes of greatest contribution are used to perform online recognition of the operating states that were considered (NOC, faults F1-F15, F21).

4.2.2. Recognition stage

In this subsection, the principal results in the recognition stage are displayed. In this sense, two objectives are addressed. First, the improved results in the classification process when only the variables with greater contribution in the clustering process are used. Second, the ability of the diagnosis system to analyze the data classified as noise in a window of time and to determine if they form a new class that represents a new fault.

In this stage, Algorithm 2 was applied to perform online recognition. In the first experiment, the operating states used in the training stage (NOC, faults F1-F15, F21) were considered. Later, in the second experiment, faults 16-20 were used to test the online detection of new faults by the algorithm.

Table 8: Results of the training stage in TE process

Operation mode	**Variables with greater contribution**
NOC	2, 3, 5, 6, 8, 9, 12, 14, 15, 17, 21, 22, 26, 30, 32
F1	1, 3, 4, 7, 8, 10, 11, 13, 16, 18-23, 25, 27, 29, 30, 31, 33
F2	3, 4, 6, 7, 10, 11, 13, 16, 18-20, 22, 24, 28, 30, 31, 33
F3	7, 10, 11, 13, 16, 18-20, 25, 27, 29, 31, 33
F4	1, 2, 4, 7, 9, 10, 11, 13, 16, 18-20, 25, 29-31, 33
F5	1, 3, 4, 7, 8, 10, 11, 13, 16, 18-23, 25, 27, 29, 31
F6	3, 4, 6, 7-11, 13, 16, 18-23, 25, 28, 29, 31
F7	1, 3, 4, 7, 8, 10, 11, 13, 16, 18-23, 25, 27, 29, 30, 31, 33
F8	1, 3, 4, 7, 8, 10, 11, 13, 16, 18-25, 27-31, 33
F9	1, 3, 7, 11, 13, 16, 18-20, 25, 29, 31, 33
F10	1, 3, 4, 7, 10, 11, 13, 16, 18-20, 23, 25, 29-31, 33
F11	1, 7, 11, 13, 16, 18-20, 25, 29-31, 33
F12	1, 3, 4, 6-8, 10, 11, 13, 16, 18-21, 23-25, 27, 29-31, 33
F13	1, 3, 4, 7, 8, 11, 13, 16, 18-21, 23, 27-31, 33
F14	1, 3, 4, 7, 10, 11, 13, 16, 18-20, 25, 27, 31, 33
F15	1, 3, 7, 11, 13, 16, 18-20, 23, 25, 29, 31, 33
F21	1, 7, 8, 10, 11, 13, 16-20, 22, 25, 31, 33

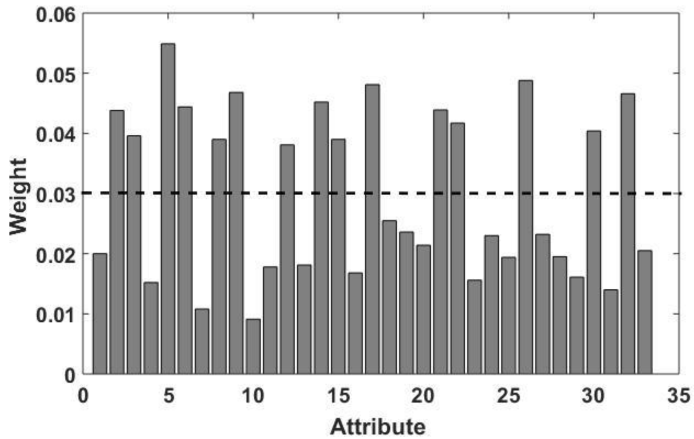


Figure 8. Attribute weight assignment of DOKEWFCM algorithm for Normal Operation Condition (NOC).

In order to detect a new fault early, 120 samples were evaluated. This implies a time window size of k = 120 equivalent to 120 seconds. In the case of the decision threshold, a value of Th = 60% was determined to establish an adequate level of majority of samples classified as noise. It is important to highlight that the criteria to select these parameters depends on the type of process, therefore, the opinion of the experts is very important.

Table 9 shows a comparison between the results of the classification using the 33 variables (Case 1) and using the variables with greater contribution (Case 2) determined in the training stage. The results show that a better classification (%) of the different operating states is obtained. by using the variables with greater contribution in the clustering obtained during the training stage. It is confirmed when the Wilcoxon test is applied to compare the results of both cases. Table 10 shows the results of that comparison. The use of a minor number of variables also simplifies the computational complexity.

In the second experiment the unknown faults F16-20 were analyzed. First, the fault 16 was considered, which was identified as a new class. After a new class is identified, experts should characterize the fault and updating the historical database used in the training of the algorithm adding the new pattern. The same experiment was performed for faults 17-20. Table 11 shows the results obtained for unknown faults F16-20. The last column reflects the variables with greater contribution to the identification of faults.

Table 9: Results of the recognition stage in the TE process (Experiment 1)

	Classification (%)	
Operation mode	**Case 1**	**Case 2**
NOC	96.67	98.33
F1	95	97.50
F2	100	100
F3	95.83	96.67
F4	94.17	95
F5	92.50	94.17
F6	100	100
F7	98.33	100
F8	100	100
F9	90.83	93.33
F10	89.17	91.67
F11	93.33	95
F12	99.17	100
F13	97.50	99.17
F14	96.67	98.33
F15	87.50	90
F21	86.67	88.33

Table 10: Results of the Wilcoxon test for the Case 1 and Case 2

	Case 1 vs Case 2
$\sum R^{+}$	0
$\sum R^{-}$	55
T	0
T^{α} = 0.05	8
Winner	Case 2

Table 11: Results of the recognition stage in the TE process (Experiment 2)

Operation mode	**Classification (%)**	**Variables with greater contribution**
F16	81.67	4, 7, 11, 13, 16, 18-20, 25, 29, 31-33
F17	85.83	1, 3, 7, 9, 11, 13, 16, 18- 21, 25, 29, 31
F18	95	1-4, 6-8, 10, 11, 13, 16, 18-21, 23-25, 27, 29-33
F19	77.50	5, 7, 13, 16, 18-21, 25, 31, 33
F20	91.67	1, 7, 11, 13, 16, 18-22, 25, 29, 31, 33

4.3. Analysis of the number of false and missing alarms

In order to evaluate the quality of the fault detection process, usually the number of false and missing alarms is analyzed. According to [71], these indicators called False Alarm Rate (FAR) and Fault Detection Rate (FDR) can be calculated by:

$$FAR = \frac{No.ofsamples\ (J > J_{lim} \mid f = 0)}{total\ samples\left(f = 0\right)} \tag{11}$$

$$FDR = \frac{No.ofsamples\ (J > J_{lim} \mid f \neq 0)}{total\ samples\left(f \neq 0\right)} \tag{12}$$

where J is the output for the used discriminative algorithms by considering the fault detection stage as a binary classification process, and J_{lim} is the threshold that determines whether one sample is classified as a fault or normal operation. The improved results are obtained for the small FAR values and large FDR values.

The results obtained for the faults F1-F15, F21 (Experiment 1) are summarized in Figs. 9 and 10. In both indicators, the improved results were obtained when the classification algorithm uses the variables with greater contribution in the clustering process (Case 2).

Figure 11 shows the results of the FAR and FDR indicators for the unknown faults (Experiment 2). In this case, only the results obtained with the algorithm using the variables with greater contribution in the clustering process are shown.

Table 12 shows the results of the FDR indicator for the algorithms' Principal Component Analysis (PCA), Dynamic PCA (DPCA), Independent Component Analysis (ICA), Modified ICA (MICA), Fisher Discriminant Analysis (FDA), Partial Least Squares (PLS), Total Projection to Latent Structure (TPLS), Modified Approach (MPLS), Subspace Aided Approach (SAP) used in [67] and the algorithm proposed in this paper. This comparison shows that the DOKEWFCM algorithm achieves better results for almost all the analyzed faults. Also, it is necessary to highlight the excellent results obtained by DOKEWFCM algorithm for faults 3, 9 and 15, where other algorithms used for the comparison obtain very low FDR values.

Finally, another comparison was made with the LAMDA (Learning Algorithm for Multivariate Data Analysis) algorithm, presented in [4] because it is an unsupervised algorithm. In this case the percent of correct classification was used as indicator. Table 13 shows the better results obtained by the DOKEWFCM algorithm proposed in this paper.

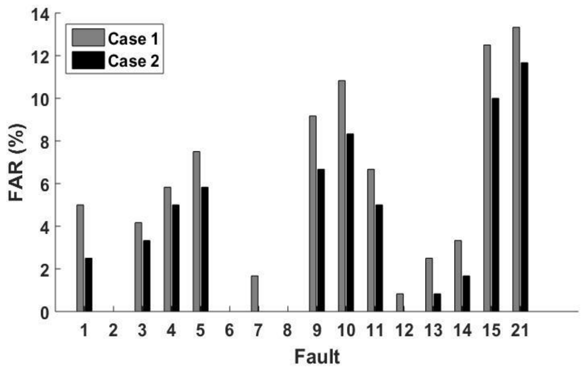


Figure 9: False Alarm Rate (%) obtained for the faults F1-F15, F21

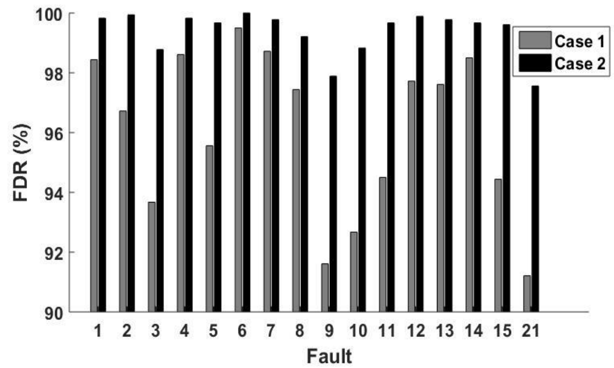


Figure 10: Fault Detection Rate (%) obtained for the faults F1-F15, F21.

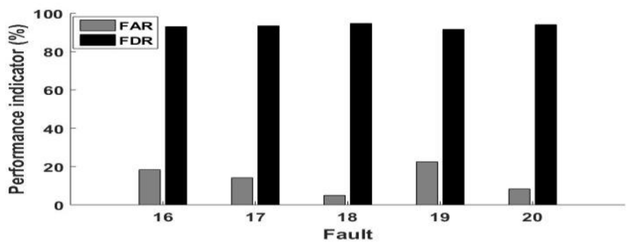


Figure 11: Performance indicator (%) obtained for the unknown faults F16-F20

Table 12: Results of the comparison of FDR indicators for the test algorithms with the one proposed in this work. (Bold values indicate the best performance)

Fault	**PCA**	**DPCA**	**ICA**	**MICA**	**FDA**	**PLS**	**TPLS**	**MPLS**	**SAP**	**DOKEWFCM**
1	99.88	99.88	**100**	99.88	**100**	99.88	99.88	**100**	99.63	99.83
2	98.75	99.38	98.25	98.25	98.75	98.63	98.88	98.88	97.88	**99.94**
3	12.88	12.25	4.5	14.25	7	14.25	24.25	18.75	6.38	**98.78**
4	**100**	**100**	**100**	87.63	**100**	99.5	**100**	**100**	99.88	99.83
5	33.63	43.25	**100**	**100**	**100**	33.63	**100**	**100**	**100**	99.67
6	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
7	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	99.88	99.78
8	98	98	98.25	97.63	98.13	97.88	98.5	98.63	95.88	**99.21**
9	8.38	12.88	4.75	8.88	6.25	14.5	23.5	12.13	0.88	**97.89**
10	60.5	72	89.25	85.88	87.13	82.63	91	91.13	95.5	**98.83**
11	78.88	91.5	78.88	61.63	73.38	78.63	83.13	83.25	84.75	**99.87**
12	99.13	99.25	99.88	99.88	99.75	99.25	99.63	99.88	99.88	**99.89**
13	95.38	95.38	95.25	95	95.63	95.25	96.13	95.5	94.88	**99.78**
14	**100**	**100**	**100**	99.88	**100**	**100**	**100**	**100**	97.63	99.67
15	14.13	19.75	7.75	10.75	12.63	23	29.88	23.25	29.5	**99.61**
16	55.25	67.38	92.38	83.38	83.25	63.38	90.75	94.28	94.88	**95.04**
17	95.25	**97.25**	96.88	93	96.63	94.25	96	97.13	97.13	96.46
18	90.5	90.88	90.5	89.75	90.75	90.75	91.88	91.25	91	**94.71**
19	41.13	87.25	92.88	80.25	87.88	26	82.88	94.25	88.5	**94.63**
20	63.38	73.75	91.38	86	81.88	62.75	78.38	91.5	83.75	**94.08**
21	52.13	61	56.38	70.75	52.75	59.88	66.38	72.75	38.63	**97.56**
AVG	71.29	77.19	80.82	79.17	79.61	73.29	83.38	83.93	80.78	**98.33**

Table 13: Results of the comparison among test algorithms based on classification percentage

Fault	**DOKEWFCM**	**LAMDA [4]**
1	**97.50**	91.77
2	**100**	96.67
4	95	95.38
7	**100**	87.69
8	**100**	84.08
14	**98.33**	89.33
AVG	**98.47**	90.82

5. Conclusions

In the present paper, a hybrid algorithm using fuzzy clustering techniques is proposed. The algorithm is applied in a fault diagnosis scheme with online detection of novel faults and automatic learning. The proposed algorithm conjugates three important characteristics, which permit it to obtain very satisfactory percentages of correct classification in the presence of noise, outliers and nonlinearities in the data. These characteristics are: a) capacity to detect and eliminate outliers, b) identification of the variables with greater influence in the clustering process using the Maximum-Entropy-Regularized Weighted method, and c) the use of a kernel function to achieve greater separability among the classes and reducing classification errors.

The fault diagnosis scheme proposed here has two stages. In the first stage (offline training stage), the algorithm is trained, and the best values of its configuration parameters (m and σ) are obtained as well as the variables with greater influence in the clustering of the data, which represent the different states of the process. In second stage (online recognition), the proposed algorithm analyzes each observation obtained from the process and classified it in one of the known classes of the process, or as noise. Furthermore, the algorithm analyzes the observations belonging to a window of time that are not classified in the known classes, and determines if they constitute a new class or if they are outliers.

If a new pattern is detected and the experts determine that it represents a new fault, the pattern should be added to the training database and the proposed algorithm should be trained again. The latter constitutes a mechanism of automatic learning.

It is important to remark that a possible disadvantage of the proposed fault diagnosis approach is the computational cost associated with the data driven methods. However, in general form, the dynamics of the complex processes present in the large industries are slow, which permits the application of data driven methods.

For future research, an interesting idea is to analyze the behavior of the proposed algorithm when the observations from the process have missing variables and it is necessary to develop an on-line imputation procedure.

Conflict of interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors acknowledge the financial support provided by FAPERJ, Fundacao Carlos Chagas Filho de Amparo á Pesquisa do Estado do Rio de Janeiro; CNPq, Consehlo Nacional de Desenvolvimento Científico e Tecnológico; CAPES, Coordenaçao de Aperfeiçoamento de Pessoal de Nível Superior, research supporting agencies from Brazil; UERJ, Universidade do Estado do Rio de Janeiro and CUJAE, Universidad Tecnológica de La Habana José Antonio Echeverría, and the help of Dr. Patricia Ortiz-Bermúdez (University of Puerto Rico-Mayagüez).

References

1. Aguitoni, M.C., Pavão, L.T., Siquiera, P.H., Jimenez, L., Da Silva Sã, M.A. Ravagnani. (2018), Heat exchanger network synthesis using genetic algorithm and differential evolution, *Computers and Chemical Engineering* 117, 82-96.
2. Akbulut, Y., Sengur, A., Guo, Y., Polat, K. (2017), KNCM: Kernel Neutrosophic c-Means Clustering, *Applied Soft Computing* 52, 714 – 724.
3. Asuncion, A., Newman, D. UCI machine learning repository. (2007), University of California, School of Information and Computer Science, Irvine, CA. http://archive.ics.uci.edu/beta/ Accessed 2007.
4. Bedoya, C., Uribe, C., Isaza, C. (2012), Unsupervised feature selection based on fuzzy clustering for fault detection of the tennessee eastman process, *Advances in Artificial Intelligence*, Springer-Verlag LNAI 7637, 350 – 360.
5. Bernal de Lázaro, J. M., Prieto Moreno, A., Llanes Santiago, O., Silva Neto, A.J. (2015), Optimizing kernel methods to reduce dimensionality in fault diagnosis of industrial systems, *Computers & Industrial Engineering* 87, 140 – 149.
6. Bernal de Lázaro, J.M., Llanes Santiago, O., Prieto Moreno, A., Knupp, D.C., Silva-Neto, A.J. (2016), Enhanced dynamic approach to improve the detection of small-magnitude faults, *Chemical Engineering Science* 146, 166 – 179.
7. Bezdek, J.C. (1981), *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum, New York.
8. Borgelt, C. (2008), Feature weighting and feature selection in fuzzy clustering, Proc. IEEE Conf. *Fuzzy Syst*. 1, 838 – 844.
9. Botia, J., Isaza, C., Kempowsky, T., LeLann, M.-V., Aguilar-Martín, J. (2013), Automation based on fuzzy clustering methods for monitoring industrial processes, *Engineering Applications of Artificial Intell*igence 26, 1211 – 1220.
10. Brizuela Mendoza, J.A., Astorga Zaragoza, C.M., Zavala Río, A., Canales Abarca, F. (2016), Active Fault Tolerant Control: Sensor Fault Estimation and Accomodation applied to a Rider less Bicycle LPV Model. RIAI: *Revista Iberoamericana de Automática e Informática Industrial* 13, 174 – 185.
11. Camps Echeverría, L., de Campos Velho, H.F., Becceneri, J.C., Silva Neto, A. J., & Llanes Santiago, O. (2014). The fault diagnosis inverse problem with ant colony optimization and ant colony optimization with dispersion. *Applied Mathematics and Computation* 227, 687 – 700.
12. Camps Echevarría, L., Llanes Santiago, O., Hernández Fajardo, J.A., Silva Neto, A. J., & Jímenez Sánchez, D. (2014). A variant of the particle swarm optimization for the improvement of fault diagnosis in industrial systems via faults estimation. *Engineering Applications of Artificial Intelligence* 28, 36 – 51.
13. Camps Echevarría, L., Llanes Santiago, O., Silva Neto, A.J. (2010), An approach for fault diagnosis based on bio-inspired strategies, *Studies in Computational Intelligence* 284, 53 – 63.
14. Chintalapudi, K.K., Kam, M. (1998), A noise resistant fuzzy c-means algorithm for clustering, In: *IEEE Conference on Fuzzy Systems Proceeding* 2, 1458 – 1463.
15. Costa Castelló, R., Puig, V., J. Blesa. (2016), Introduction to Model based Fault Diagnosis using Project based Learning. RIAI: *Revista Iberoamericana de Automática e Informática Industrial* 13, 186 - 195.
16. D'Angelo, M.F.S.V., Palhares, R., Filho, M.C.O.C., Maia, R.D., Mendes, J.B., Ekel, P.Y. (2016), A new fault classification approach applied to tennessee eastman benchmark process, *Applied Soft Computing* 49, 676 – 686.
17. Dave, R.N. (1991), Characterization and detection of noise in clustering, *Pattern Recognition Letters* 12, 657 – 664.
18. Dave, R.N., Krishnapuram, R. (1997), Robust clustering methods: A unified view. *IEEE Transactions on Fuzzy Systems* 5, 270 – 293.
19. Deng, Z., Choi, K.S., Chung, F.L., Wang, S. (2010), Enhanced soft subspace clustering integrating within-cluster and between-cluster information, *Pattern Recognition* 43, 767 – 781.
20. Detroja, K., Gudi, R., Patwardhan, S. (2006), A possibilistic clustering approach to novel fault detection and isolation. *Journal of Process Control* 16, 1055 – 1073.
21. Ding, Y., Fu, X. (2016), Kernel-based fuzzy c-means clustering algorithm based on genetic algorithm, *Neurocomputing* 188, 233 – 238.
22. Downs, J.J., & Vogel, E.F. (1993), A plant-wide industrial process control problem, *Computers Chemical Engineering* 17, 245 – 255.
23. Fan, J., Wang, Y. (2014), Fault detection and diagnosis of nonlinear non-gaussian dynamic processes using kernel dynamic independent component analysis, *Information Sciences* 259, 369 – 379.

24. García, S., Herrera, F. (2008), An extension on statistical comparisons of classifiers over multiple data sets for all pairwise comparisons, *Journal of Machine Learning Research* 9, 2677 – 2694.
25. García, S., Molina, D., Lozano, M., Herrera, F. (2009), A study on the use of non-parametric tests for analyzing the evolutionary algorithms behaviour: a case study on the cec 2005 special session on real parameter optimization, *Journal of Heuristics* 15, 617 – 644.
26. Graves, D., Pedrycz, W. (2010), Kernel-based fuzzy clustering and fuzzy clustering: A comparative experimental study, *Fuzzy Sets and Systems* 161, 522 – 543.
27. Gosain, A., Dahika, S. (2016), Performance analysis of various fuzzy clustering algorithms: A review, 7th International Conference on Communication. *Computing and Virtualization* 79, 100 – 111.
28. Hu, D., Sarosh, A., Dong, Y.F. (2012), A novel kfcm based fault diagnosis method for unknown faults in satellite reaction wheels, *ISA Transactions* 51, 309 – 316.
29. Hwang, I., Kim, S., Kim, Y., Seah, C. (2010), A survey of fault detection, isolation, and reconfiguration methods, *IEEE Transactions on Control Systems Technology* 18, 636 – 656.
30. Jahromi, A.T., Er, M.J., Li, X., Lim, B.S. (2016), Sequential fuzzy clustering based dynamic fuzzy neural network for fault diagnosis and prognosis, *Neurocomputing* 196, 31 – 41.
31. Kaur, P., Soni, A., Gosain, A. (2013), Robust kernelized approach to clustering by incorporating new distance measure, *Engineering Applications of Artificial Intelligence* 26, 833 – 847.
32. Kaur, P. (2011), A density oriented fuzzy c-means clustering algorithm for recognizing original cluster shapes from noisy data, *International Journal of Innovative Computing and Applications* 3, 77 – 87.
33. Kesemen, O., Tezel, O., Ozkul. E. (2016), Fuzzy c-means clustering algorithm for directional data (fcm4dd), *Expert Systems with Applications* 58, 76 – 82.
34. Krishnapuram, R., Keller, J.M. (1993), A possibilistic approach to clustering, *IEEE Transactions on Fuzzy Systems* 1, 98 – 110.
35. Leski., J.M. (2016), Fuzzy c-ordered-means clustering, *Fuzzy Sets and Systems* 286, 114 – 133.
36. 36. Li, C., Zhou, J., Kou, P., Xiao, J. (2012), A novel chaotic particle swarm optimization based fuzzy clustering algorithm, *Neurocomputing* 83, 98 – 109.
37. Liu, Q., Lv, W. (2009), The study of fault diagnosis based on particle swarm optimization algorithm, *Computer and Information Science* 2, 87 – 91.
38. Lobato, F., Steffen Jr., F., Silva Neto, A.J. (2009), Solution of inverse radiative transfer problems in two-layer participating media with Differential Evolution, *Inverse Problems in Science and Engineering* 18, 183 – 195.
39. Luengo, J., García, S., Herrera, F. (2009), A study on the use of statistical tests for experimentation with neural networks: Analysis of parametric test conditions and non-parametric tests, *Expert Systems with Applications* 36, 7798 – 7808.
40. Modha, D.S., Spangler, W.S. (2003), Feature weighting in k-means clustering, *Machine Learning* 52, 217 – 237.
41. Ng, T.F., Pham, T.D., Jia, X. (2012), Feature interaction in subspace clustering using the Choquet integral, *Pattern Recognition* 45, 2645 – 2660.
42. Pakhira, M., Bandyopadhyay, S., Maulik, S. (2004), Validity index for crisp and fuzzy clusters, *Pattern Recognition* 37, 487–501.
43. Pal, N.R., Pal, K., Keller, J.M., Bezdek, J.C. (2005), A possibilistic fuzzy c-means clustering algorithm, *IEEE Transactions on Fuzzy Systems* 13, 517 – 530.
44. Pang, Y.Y., Zhu, H. P., Liu, F. M. (2014), Fault diagnosis method based on KPCA and selective neural network ensemble, *Advanced Materials Research* 915, 1272 – 1276.
45. Parsons, L., Haque, E., Liu, H. (2004), Subspace clustering for high dimensional data: a review, *SIGKDD Explorati*ons 6, 90 – 105
46. Qin, S.J. (2012), Survey on data-driven industrial process monitoring and diagnosis, *Annual Reviews in Control* 36, 220 – 234.
47. Rodríguez Ramos, A., Bernal De Lázaro, J.M., Prieto Moreno, A., Silva Neto, A.J., Llanes Santiago, O. (2017), An approach to robust fault diagnosis in mechanical systems using computational intelligence, *Journal of Intelligent Manufacturing*, 1 – 15, DOI:10.1007/s10845-017-1343-1.
48. Rodríguez Ramos, A., Bernal De Lázaro, J.M., Silva Neto, A.J., Cruz Corona, C., Verdegay, J.L., Llanes-Santiago, O. (2017), An Approach to Fault Diagnosis Using Fuzzy Technique, In: Kacprzyk J., Szmidt E., Zadrożny S., Atanassov K., Krawczak M. (eds) *Advances in Fuzzy Logic and Technology* 2017. EUSFLAT 2017, IWIFSGN 2017. Advances in Intelligent Systems and Computing, vol 643. Springer, Cham

49. Rodríguez Ramos, A., Llanes-Santiago, O., Bernal de Lázaro, J.M., Cruz Corona, C., Silva Neto, A.J, Verdegay, J.L. (2017), A novel fault diagnosis scheme applying fuzzy clustering algorithms, *Applied Soft Computing* 58, 605-619.
50. Rodríguez Ramos, A., Prieto Moreno, A., Silva Neto, A.J., Llanes-Santiago, O. (2017), A Proposal of Online Detection of New Faults and Automatic Learning in Fault Diagnosis. *Soft Computing Based Optimization and Decision Models, Studies in Fuzziness and Soft Computing*, Springer.
51. Rodríguez Ramos, A., Silva Neto, A.J., Llanes-Santiago, O. (2018), An approach to fault diagnosis with online detection of novel faults using fuzzy clustering tools, *Expert Systems with Applications* 113, 200-212.
52. Saltos, R., Weber, R. (2016), A rough-fuzzy approach for support vector clustering, *Information Sciences* 339, 353 – 368.
53. Seera, M., Lim, C.P., Loo, C.K., Singh, H. (2015), A modified fuzzy min-max neural network for data clustering and its application to power quality monitoring, *Applied Soft Computing* 28, 19 – 29.
54. Silva Filho, T.M., Pimentel, B.A., Souza, R.M., Oliveira, A.L.I. (2015), Hybrid methods for fuzzy clustering based on fuzzy c-means and improved particle swarm optimization, *Expert Systems with Applications* 42, 6315 – 6328.
55. Sina, S., Sadough, Z.N., Khorasani, K. (2014), Dynamic neural network-based fault diagnosis of gas turbine engines, *Neurocomputing* 125, 153–165.
56. Tang, C.L., Wang, S.G., Xu, W. (2010), New fuzzy c-means clustering model based on the data weighted approach, *Data Knowledge Engineering* 69, 881 – 900.
57. Tarajo Buba, A., Soon Lee, L. (2018), A differential evolution for simultaneous transist network design and frequency setting problem, *Expert Systems with Applications* 106, 277-289.
58. Thanh Ngo, L., Hop Dang, T., Pedrycz, W. (2018), Towards Interval – Valued Fuzzy Set-based Collaborative Fuzzy Clustering Algorithms, *Pattern Recognition,* doi: 10.1016/j.patcog.2018.04.006.
59. Thong, P.H., Son, L.H. (2016), Picture fuzzy clustering for complex data, *Engineering Applications of Artificial Intelligence* 56, 121 – 130.
60. Venkatasubramanian, V., Rengaswamy, R., Kavuri, S.N. (2003), A review of process fault detection and diagnosis, part 1: Quantitative model-based methods, *Computers and Chemical Engineering* 27, 293 – 311.
61. Venkatasubramanian, V., Rengaswamy, R., Kavuri, S.N. (2003), A review of process fault detection and diagnosis, part 2: Qualitative models and search strategies, *Computers and Chemical Engineering* 27, 313 – 326.
62. Wang, J., Hu, H. (2009), Vibration-based fault diagnosis of pump using fuzzy technique, *Measurement* 39, 176 – 185.
63. Wang, H., Khan, F., Ahmed, S., Imtiaz, S. (2016), Dynamic quantitative operational risk assessment of chemical processes, *Chemical Engineering Science* 142, 62 – 78
64. Wang, X.Z., Wang, Y.D., Wang, L.J. (2004), Improving fuzzy c-means clustering based on feature-weight learning, *Pattern Recognition Letter* 25, 1123 – 1132.
65. Wu, K., Yang, M. (2005), A cluster validity index for fuzzy clustering, *Pattern Recognition* 26, 1275–1291.
66. Yan, K., Ji, Z., Shen, W. (2017), Online fault detection methods for chillers combining extended kalman filter and recursive one-class SVM, *Neurocomputing* 228, 205 – 212.
67. Yin, S., Ding, S.X., Haghani, A., Hao, H., Zhang, P. (2012), A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process, *Journal of Process Control* 22, 1567 – 1581.
68. Yin, S., Kaynak, O. (2015), Big data for modern industry: challenges and trends, *Proc. IEEE* 103(2), 143 – 146.
69. Zhang, D., Ji, M., Maulik, S., Yang, J., Zhang, Y., Xie, F. (2014), A novel cluster validity index for fuzzy clustering based on bipartite modularity, *Fuzzy Sets and Systems* 253, 122 – 137.
70. Zhang, D.Q., Chen, S.C. (2003), Clustering incomplete data using kernel based fuzzy c-means algorithm, *Neural Process Letters* 18, 155 – 162.
71. Zhang, D.Q., Chen, S.C. (2004), A novel kernelized fuzzy c-means algorithm with application in medical image segmentation, *Artificial Intelligence in Medicine* 32, 37 – 50.
72. Zhang, L., Lu, W., Liu, X., Pedrycz, W., Zhong., C. (2016), Fuzzy c-means clustering of incomplete data based on probabilistic information granules of missing values, *Knowledge - Based Systems* 99, 51 – 70.
73. Zhou, J., Chen, L., Philip Chen, C.L., Zhang, Y., Li, H.L. (2016), Fuzzy clustering with the entropy of attribute weights, *Neurocomputing* 198, 125 – 134.

74. Zhou, J., Lai, Z., Miao, D., Gao, C., Yue, X. (2018), Multigranulation Rough-Fuzzy Clustering Based on Shadowed Sets, *Information Sciences*, doi: 10.1016/j.ins.2018.05.053.