Appl Intell (2011) 35:269–284
DOI 10.1007/s10489-010-0219-2

Weighted and constrained possibilistic C-means clustering for online fault detection and isolation

Soheil Bahrampour · Behzad Moshiri · Karim Salahshoor

Published online: 13 March 2010

Abstract In this paper, a new weighted and constrained possibilistic C-means clustering algorithm is proposed for process fault detection and diagnosis (FDI) in offline and online modes for both already known and novel faults. A possibilistic clustering based approach is utilized here to address some of the deficiencies of the fuzzy C-means (FCM) algorithm leading to more consistent results in the context of the FDI tasks by relaxing the probabilistic condition in FCM cost function. The proposed algorithm clusters the historical data set into C different dense regions without having precise knowledge about the number of the faults in the data set. The algorithm incorporates simultaneously possibilistic algorithm and local attribute weighting for time-series segmentation. This allows different weights to be allocated to different features responsible for the distinguished process faults which is an essential characteristic of proper FDI operations. A set of comparative studies have been carried out on the large-scale Tennessee Eastman industrial challenge problem and the DAMADICS actuator benchmark to demonstrate the superiority of the proposed algorithm in process FDI applications with respect to some available alternative approaches.

S. Bahrampour (✉) · B. Moshiri
Control and Intelligence Processing Center of Excellence, School of ECE, University of Tehran, Tehran, Iran
e-mail: s.bahrampour@ece.ut.ac.ir

B. Moshiri
e-mail: moshiri@ut.ac.ir

S. Bahrampour
e-mail: s.bahrampour@ece.ut.ac.ir

K. Salahshoor
Department of Instrumentation and Automation, Petroleum University of Technology, Tehran, Iran
e-mail: salahshoor@put.ac.ir

Keywords Fault detection and isolation · Possibilistic clustering · Feature weighting

1 Introduction

Large amounts of data are collected in many industrial processes for condition monitoring and control purposes. The most important parts of condition monitoring are fault detection and isolation (FDI). Fault detection is used to determine when abnormal process behavior has occurred whereas fault isolation is used to distinguish between different faults. Hundreds of measurements are recorded with small sampling times in a typical process plant. Due to the existence of this large amount of data and the improvements in the processor technologies, data driven methods have attracted more attention in the recent research that is done on process FDI. Sensor and actuator biases as well as disturbances and parametric faults generate different patterns in the measured process variables which can be classified into either normal or abnormal operation. The monitoring task is often related to the ability to classify plant operation into 3 different categories: normal operation, belonging in one or more of the previously identified faults or belonging in a novel fault scenario that has not been identified yet.

The existing data driven methods can be generally divided into statistical and non-statistical ones. Principal component analysis (PCA) [1, 2], independent component analysis (ICA) [3], fisher discriminate analysis [4], partial least squares [1] and its modifications [5] and finally statistical pattern classifiers [6] form the major classes of statistical methods [7]. Neural network [9] and support vector machines [10] are among the most powerful non-statistical methods. A major disadvantage of all these methods, however, is their critical dependencies on the availability of a priori knowledge of the complete normal and faulty scenarios making them unable to identify the novel faults. In fact, it is usually extremely difficult in practice to have such a data set, containing patterns of all possible faults in a plant.

Clustering algorithms, in contrast, may overcome this deficiency since they are unsupervised classifications, attempting to automatically partition the operational data set into a number of clusters. Therefore, FDI can be accomplished using a multivariate time-series clustering algorithm in which values of the process variables at successive time samples provide the multivariate time-series. Time-series segmentation addresses the following data-mining problem: given a time-series, T, find a partitioning of T into C segments that are internally consistent [12]. In FDI application, the goal of segmentation is to locate stable periods of time in a multivariate time-series. This problem may be considered as clustering with a time-ordered constraint. Most recently, a modified Gath-Geva (MGG) clustering method was proposed in [13] for segmentation of multivariate time-series. In [21], the k-means clustering is used along with PCA and Fisher discriminant analysis for the FDI task. An adaptive FCM has been proposed in [22] for process monitoring in a waste water plant. Recently, a possibilistic clustering algorithm, called possibilistic Gustafson–Kessel (PGK), was proposed for novel fault detection and isolation by Detroja et al. [11]. Possibilistic clustering algorithm is a powerful technique that is similar to probabilistic clustering methods with different constraints leading to more consistent results in FDI application. In possibilistic C-means (PCM) clustering, in contrast to FCM algorithm, the number of clusters does not need to be specified accurately; they can be derived during the classification step [24]. It has also been shown that the possibilistic clustering algorithm is relatively insensitive to noise and outliers [8].

Dealing with high dimensional data set is still an important challenging area for FDI purposes in large-scale process and existing clustering algorithms suffers from it. High dimensionality poses a dual problem. First, regardless of similarity definition, the presence of irrelevant attributes reduces clustering tendency. The second problem is that data separation is somehow impractical in high dimensional space. While this outcome is also possible with low dimensional data, the likelihood of presence and number of irrelevant attributes grows with dimension. It was mathematically illustrated that these problems start to be severe for dimensions greater than 15 [14]. Two of the most well-known solutions for high dimensionality problem are dimension reduction and attribute weighting. Principal component analysis (PCA) and different modifications of it have been widely used for dimension reduction [15, 16]. However, if the data set contains an important number of irrelevant features with wide dispersion, these methods would not be efficient. In attribute weighting, weights are assigned for the features which can be regarded as degrees of relevance. Most feature weighting methods assume that feature relevance is invariant over the task's domain, and hence a single set of weights is allocated for the entire data set called as global weights [17, 18]. This assumption, however, can impose unnecessary and even destructive constraints on the learning procedure when the data set is made of different categories or classes [23]. In the FDI application especially, diverse variable are responsible for different faults. Therefore, a localized feature weighting for each fault is preferable [20]. On the other hand, most of the existing localized feature weighting methods have been presented for supervised learning or classification [19]. Simultaneous clustering and attribute discrimination (SCAD) has been recently proposed for unsupervised learning of prototypes and local feature weighting based on the FCM clustering algorithm [23]. SCAD sets the weights of the features through an optimization procedure.

The main contribution of this paper is to address the shortcomings of the existing algorithms discussed above and to develop an algorithm for unsupervised fault detection and isolation in high dimensional processes. For this purpose, the ideas of local feature weighting and PCM clustering are organized into a framework to solve FDI problem in large-scale processes more efficiently, leading to a new weighted and constrained possibilistic C-means clustering (WCPCM) method. The resulting algorithm can be effectively used for FDI purposes in both offline and online manners which outperforms some of the existing methods and solves the problem of high dimensionality in clustering algorithms. The proposed scheme is also able to isolate novel fault, i.e. contributing the cases when the archived data does not contain data for the new fault. The proposed monitoring scheme is based on existing historical data sets and updates the knowledge base as new fault situations arise during online process monitoring.

A series of simulations are conducted in this paper for investigating the performances of different FDI algorithms in isolating faults of the challenging TE process benchmark. The process simulator for the Tennessee Eastman (TE) industrial challenge problem was created by the Eastman Chemical Company to provide a realistic industrial process for evaluating process control and monitoring methods [25]. The TE process has been widely used as a benchmark process for evaluating process diagnosis methods. PCA and its different modifications [1, 2], steady state-based approach [33], support vector machines [10] are among approaches that have been applied to the TE process. Although most of the foregoing methods show good diagnostic performances, they have the crucial drawback of dependency on rich data sets, which are rarely available in practice. In this paper, the TE process benchmark is utilized to test the efficiency of the proposed unsupervised WCPCM algorithm in FDI. For this purpose, the proposed WCPCM method is compared with two other methods. First, the TE process faults are isolated through Modified Gath-Geva (MGG) algorithm, a powerful time-series segmentation method introduced in [13]. Then, DPCA is utilized as a dimension reduction technique to improve the MGG algorithm to obtain more proper results. Finally, our proposed WCPCM algorithm is exercised to the TE process plant. The simulation results demonstrate the capability of the proposed approach to detect and isolate the TE process faults in both offline and online modes while outperforming the other alternative methods. To further investigate the capability of the WCPCM algorithm, it is also employed for FDI purpose on the DAMADICS actuator benchmark.

The organization of the remainder of the paper is described below. The WCPCM algorithm is developed in Sect. 2 for FDI application after a review of the existing clustering algorithms. Two simulation studies are then presented on the basis of the TE benchmark problem and DAMADICS actuator benchmark in Sect. 3 comparatively explore performance of the proposed WCPCM algorithm with respect to a number of alternative approaches. Finally, concluding remarks will be summarized in Sect. 4.

2 Development of the WCPCM clustering algorithm

In this section, WCPCM is developed for fuzzy segmentations of time-series. Since this method is an extension of the existing clustering algorithms, a brief introduction of FCM and PCM clustering algorithms [8] will help in better understanding of this method. Section 2.1 presents this introduction. Afterwards, WCPCM is introduced for time-series segmentation.

2.1 Overview of existing clustering algorithms

The goal of any clustering analysis is to derive a partition of a set of N data points or objects based on some similarity metric so that similar data points/objects are categorized in the same cluster. A brief review of some existing clustering techniques is presented here.

Let $\mathbf{X} = \{X_j | j = 1 \dots N\}$ be a set of N feature vectors or sample points in an n-dimensional feature space. Nonhierarchical clustering algorithms try to partition these N data points into a set of C different groups. Let $\mathbf{B} = (\beta_1, \dots, \beta_c)^T$ represents a C-tuple of prototypes each of which characterizes one of the C clusters. Each β_i consists of a set of parameters which are centers of the ith cluster features $\beta_i = (c_{i1} \dots c_{in})$. Let u_{ij} represents the membership grade of feature point X_j in ith cluster. Different clustering algorithms have various techniques for obtaining β_i and u_{ij}.

2.1.1 Fuzzy C-means clustering (FCM)

The matrix $U = [u_{ij}]$, having a dimension of $C \times N$, is called a constrained fuzzy C-partition matrix if it satisfies the following conditions [24]:

$$u_{ij} \in [0, 1] \quad \forall i; \qquad 0 < \sum_{j=1}^{N} u_{ij} < N \tag{1}$$

$$\forall j; \qquad \sum_{i=1}^{C} u_{ij} = 1 \tag{2}$$

The problem of fuzzily partitioning the feature vectors into C clusters can be formulated as the minimization of an objective function $J(\mathbf{B}, \mathbf{U}; \mathbf{X})$ of the form:

$$J(\mathbf{B}, \mathbf{U}; \mathbf{X}) = \sum_{i=1}^{C} \sum_{j=1}^{N} (u_{ij})^m d_{ij}^2 \tag{3}$$

subject to the constraints in (1) and (2). In the above equation, $m \in [1, \infty)$ is a weighting exponent called the fuzzifier, and d_{ij}^2 represents the distance from a feature point X_j to the prototype β_i. Minimization of the objective function in (3) with respect to $\boldsymbol{U}$ gives [24]:

$$u_{ij} = \frac{1}{\sum_{p=1}^{C} (\frac{d_{ij}^2}{d_{pj}^2})^{1/(m-1)}} \tag{4}$$

Minimization of (3) with respect to $\boldsymbol{B}$ varies according to the choice of the prototypes and the distance measure. For example, in the fuzzy *C-means* (FCM) algorithm, the Euclidean distance is used and each of the resulting prototypes is described by the cluster center β_i, which may be updated in each iteration using [24]:

$$c_i = \frac{\sum_{j=1}^{N} (u_{ij})^m X_j}{\sum_{j=1}^{N} (u_{ij})^m} \tag{5}$$

2.1.2 Possibilistic C-means clustering (PCM)

In possibilistic clustering, the fuzzy partitioning constraint of the FCM algorithm, (2), is relaxed so as to get membership values, representing the 'degree of typicality' to a cluster. Simply relaxing this constraint produces a trivial solution, i.e., the objective function is minimized by assigning all membership values to 0. Therefore, the objective function of (3) is modified as:

$$J(\mathbf{B}, \mathbf{U}; \mathbf{X}) = \sum_{i=1}^{C} \sum_{j=1}^{N} (u_{ij})^m d_{ij}^2 + \sum_{i=1}^{C} \eta_i \sum_{j=1}^{N} (1 - u_{ij})^m \tag{6}$$

Possibilistic clustering algorithm tries to minimize the above cost function subject to the only constraint of (1). The first term in (6) minimizes the distances of data points from the cluster centers while the second term imposes the membership values to be as large as possible. The value of parameter η_i determines the distance at which the membership value of a point in a cluster becomes 0.5. Therefore, it needs to be chosen based on the desired bandwidth of the possibility distribution for each cluster. In practice, however, the following definition works well [8]:

$$\eta_i = \frac{\sum_{j=1}^{N}(u_{ij})^m d_{ij}^2}{\sum_{j=1}^{N}(u_{ij})^m} \tag{7}$$

For obtaining the parameters of the clusters $(\mathbf{B}, \mathbf{U})$ in PCM algorithm, a number of similar recursive formulas are given in [8]. The possibilistic clustering algorithm is very similar to the FCM algorithm, except for the additional parameter η_i which should be estimated from the initial partitioning matrix. However, it is not necessary to calculate η_i at all iterations. Since the parameter η_i is independent of the relative location of the clusters, the membership value u_{ij} depends only on the distance of a point from the cluster center (centroid) and this is the main advantage of PCM over FCM. In other words, in the possibilistic clustering algorithm, the membership of a point in a cluster is solely determined by how far a point is from the centroid of that cluster. In fact, PCM relies on finding meaningful clusters as defined by dense regions and each cluster is independent of the other clusters. However, in the probabilistic clustering algorithm, the membership of a point in a cluster is coupled with the relative distances of the point to the other cluster centroids.

It has been shown [30] that for a given value of η_i, each of the C sub-objective functions are minimized by choosing the centroid location such that the sum of the memberships is maximized. This makes each cluster centroid converge to a dense region. Thus, even if the true value of the number of clusters is unknown, the outcome of the algorithm will give C 'good' clusters, i.e., dense regions. Thus, PCM has self validating capability which can be very useful when C is not known a priori. When the number of clusters is more than the actual number of clusters in the dataset, PCM gives approximately coinciding clusters, indicating that the actual number of clusters is less than specified numbers of clusters. This could be interpreted accordingly and the clusters can be collapsed into a single cluster for further analysis [11].

2.2 WCPCM algorithm for time-series segmentation

In this section, a new algorithm is developed for time-series segmentation. For this purpose, PCM clustering algorithms is utilized and modified to yield a more effective algorithm for solving fault isolation problems. Two modifications will be proposed here to enhance the well-known PCM cost function for FI applications: (1) time-constrained clustering and (2) feature weighting.

2.2.1 *Time-constrained clustering*

For an isolation task, a multivariate time-series segmentation is required to be performed on the basis of successive time samples provided by the multivariate time-series record. A time-series $T = \{X_j | j = 1 \dots N\}$ is a finite set of N samples labeled by time points $t_1, \dots, t_N$ where $X_j = [x_{j1} \dots X_{jn}]$. A segment of T is a set of consecutive time points denoted by $S(a, b) = \{a \le j \le b\}, X_a, X_{a+1}, \dots, X_b$. The C-segmentation of time-series T is a partition of T to C non-overlapping segments, represented by $S_T^C = \{S_i(a_i, b_i) | 1 \le i \le C\}$, such that $a_1 = 1$, $b_C = N$ and $a_i = b_{i-1} + 1$. It should be noted that in FDI applications, the goal of segmentation is to find homogeneous periods of time in a multivariate time-series. Time-series segmentation may be defined as clustering with a time-ordered constraint. Therefore, data points should not only be grouped based on their similarity, but also satisfying the constraint that all points in a cluster must come from successive time points. Moreover, the changes in the variables of the time-series are usually vague and do not focus on any specific time point. A Gaussian membership function was proposed to represent fuzzy segments of a time-series [13]. In this paper, a Gaussian function is utilized to incorporate the sequential time constraint into the PCM clustering cost function i.e. (6). Thus, the constrained cost function for time-series segmentation becomes as follows:

$$J_{\mathrm{WCPCM}}(\mathbf{B}, \mathbf{U}, \mathbf{\Psi}, \mathbf{\Gamma}; \mathbf{X}) = \sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m \frac{1}{A_i(t_j)} d_{ij}^2 + \sum_{i=1}^{C}\eta_i \sum_{j=1}^{N}(1 - u_{ij})^m \tag{8}$$

where

$$A_i(t_j) = \exp\left(\frac{-1}{2}\frac{(t_j - \alpha_i)^2}{\sigma_i^2}\right) \tag{9}$$

is the Gaussian membership function for time constraint, $\mathbf{\Psi} = [\alpha_1, \dots, \alpha_C]^T$ and $\mathbf{\Gamma} = [\sigma_1, \dots, \sigma_C]^T$ are centers and variances of membership function, respectively. Considering Euclidean distance, greater d_{ij}^2 indicates more dissimilarity of the jth sample with the ith cluster. On the other hand, time constraint implies that larger $A_i(t_j)$ means more similarity of the jth sample with the ith cluster. This implication illustrates why d_{ij}^2 and $A_i(t_j)$ have an inverse relationship in the proposed cost function given by (8).

2.2.2 *Feature weighing*

Different variables are responsible for different faults and it is not suitable to equally weight them in isolated tasks of the various faults. In other words, features should be weighted for clustering different faults. This characteristic is essential for proper FDI operations.

Similar to SCAD algorithm, it is allowable to allocate different weights to different features in distinguished faults. In fact, the proposed algorithm is designed to search for the optimal prototype parameters B as our primary goal like PCM. However, the optimal set of feature weights, i.e. V, is also identified simultaneously. Each cluster i is allowed to have its own set of feature weights $V_i[v_{i1}, \ldots, v_{in}]$ where v_{ik} represents the significance of the kth feature in the ith cluster. The modified version of the cost function proposed in e.q. (8) is:

$$J_{\text{WCPCM}}(\mathbf{B}, \mathbf{U}, \mathbf{V}, \mathbf{\Psi}, \mathbf{\Gamma}; \mathbf{X}) = \sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m \frac{1}{A_i(t_j)} \sum_{k=1}^{n} (v_{ik})^q d_{ijk}^2 + \sum_{i=1}^{C} \eta_i \sum_{j=1}^{N} (1-u_{ij})^m \tag{10}$$

subject to the constraints described in (1) and the following constraint:

$$v_{ik} \in [0,1] \quad \forall i,k; \quad \text{and} \quad \sum_{k=1}^{n} v_{ik} = 1, \quad \forall i \tag{11}$$

In (10), q is a discriminant exponent, v_{ik} represents the relevance weight of feature k in ith cluster, and d_{ijk} is given by:

$$d_{ijk} = |x_{jk} - c_{ik}| \tag{12}$$

where x_{jk} is the kth feature value of data point X_j, and c_{ik} is the kth component of the ith cluster center vector. In other words, d_{ijk} is the projection of the displacement vector between feature point X_j and the ith class center (c_i) along the kth dimension. d_{ijk} measures the Euclidean distance of the kth feature of jth sample with kth center of the ith cluster. To minimize the cost function, greater value of d_{ijk} results in smaller values of relevant v_{ik} and vice versa. Therefore, minimization of the introduced cost function explicitly tries to find more impact and dense clusters and therefore, the variables with lower variances in a particular cluster are much more likely to have greater weights. Consequently, the resulting clusters contain points with lower variations and it is much more likely that points in one cluster belong to one regime of the process, i.e. faulty or normal regime.

To incorporate the constraint described in (11) into the cost function, Lagrange multiplier technique is utilized, resulting in the following cost function:

$$J_{\text{WCPCM}}(\mathbf{B}, \mathbf{U}, \mathbf{V}, \mathbf{\Psi}, \mathbf{\Gamma}; \mathbf{X}) = \sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m \frac{1}{A_i(t_j)} \sum_{k=1}^{n} (v_{ik})^q d_{ijk}^2 + \sum_{i=1}^{C} \eta_i \sum_{j=1}^{N} (1-u_{ij})^m - \lambda_i \left(\sum_{k=1}^{n} v_{ik} - 1\right) \tag{13}$$

where λ denotes Lagrange multipliers. The optimum values of the parameters are obtained by setting the gradients of $\bar{J}_{\text{WCPCM}}$ with respect to $\boldsymbol{B}, \boldsymbol{U}, \boldsymbol{V}, \boldsymbol{\Psi}$ and $\boldsymbol{\Gamma}$ to zero. This results in the following easily implementable algorithm.

WCPCM algorithm

Initialization

Given a time-series T, specify an initial number of clusters C, discriminant exponent q and fuzzifier m. Choose a termination tolerance $\varepsilon > 0$ and maximum acceptable iteration $l_{\max}$. Initialize the values of $c_{ik}, u_{ij}, v_{ik}, \alpha_i$ and σ_i^2 such that the constraints in (1) and (11) are satisfied.

Repeat the following calculations until $(\max|\Delta u_{ij}| < \varepsilon$ and $l < l_{\max})$

Here Δu_{ij} is the differences between previous and new values of u_{ij} and l is the number of iteration:

- Weight of the features $\boldsymbol{V}$:

$$v_{ik} = \frac{1}{\sum_{p=1}^{n} (\frac{\tilde{D}_{ik}}{\tilde{D}_{ip}})^{\frac{1}{q-1}}} \tag{14}$$

where

$$\tilde{D}_{ik} = \sum_{j=1}^{N} (u_{ij})^m \frac{1}{A_i(t_j)} d_{ijk}^2 \tag{15}$$

while $A_i(t_j)$ and d_{ijk}^2 are computed by (9) and (12) respectively.

- Centers of the clusters $\boldsymbol{B}$:

$$c_{ik} = \begin{cases} 0 & \text{if } v_{ik} = 0 \\ \frac{\sum_{j=1}^{N} (u_{ij})^m \frac{1}{A_i(t_j)} x_{jk}}{\sum_{j=1}^{N} (u_{ij})^m \frac{1}{A_i(t_j)}} & \text{if } v_{ik} \neq 0 \end{cases} \tag{16}$$

- Fuzzy partition matrix $\boldsymbol{U}$:

$$u_{ij} = \frac{1}{1 + (\frac{\sum_{k=1}^{n} v_{ik} d_{ijk}^2}{A_i(t_j)\eta_i})^{\frac{1}{m-1}}} \tag{17}$$

where

$$\eta_i = \frac{\sum_{j=1}^{N} (u_{ij})^m \sum_{k=1}^{n} v_{ik}^q d_{ijk}^2}{\sum_{j=1}^{N} (u_{ij})^m} \tag{18}$$

- Centers of the membership functions in time $\boldsymbol{\Psi}$:

$$\alpha_i = \frac{\sum_{j=1}^{N} (u_{ij})^m t_j}{\sum_{j=1}^{N} (u_{ij})^m} \tag{19}$$

- Variances of the membership functions in time $\boldsymbol{\Gamma}$:

$$\sigma_i^2 = \frac{\sum_{j=1}^{N} (u_{ij})^m (t_j - \alpha_i)^2}{\sum_{j=1}^{N} (u_{ij})^m} \tag{20}$$

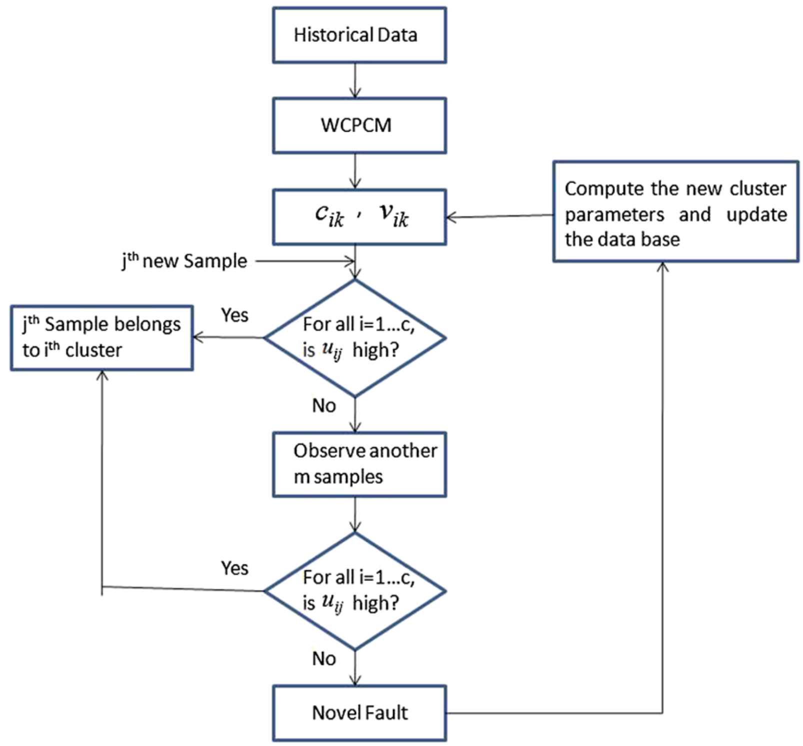


Fig. 1 The schematic diagram of the proposed online FDI algorithm

As mentioned in the initialization part of the algorithm, an initial number of clusters should be specified before running the algorithm. This is a common characteristic of partitioning clustering algorithms such as FCM and PCM [28, 32]. However, the number of the clusters may not be known beforehand and it is important to validate the number of clusters after they have been isolated in the historical data set. Nevertheless, as indicated earlier, in possibilistic clustering algorithms the number of clusters need not be specified accurately and they can be derived during the classification step [11]. In fact, even if the true value of the clusters number is unknown, the possibilistic clustering algorithms results in C 'good' clusters, i.e., dense regions. When the number of clusters is more than the actual number of clusters in the dataset, PCM algorithm results in approximately overlapping clusters, indicating that the actual number of clusters is less than the specified numbers of clusters. This could be interpreted accordingly and the clusters can be collapsed into a single cluster for another running of the partitioning algorithm. Therefore, when the number of the faults in historical data set is unknown, a roughly great number of clusters is selected as the initial number of faults, i.e. C. After running the WCPCM algorithm, if there exits two very overlapping clusters, it indicates that the actual number of faults is less than the specified number and the algorithm will run again with the new initial value of $C = C_{\text{old}} - 1$. It should be noted that this process is done in the offline mode of the algorithm, and therefore, one can run the algorithm for an arbitrary number of times to find the correct number of the clusters. In fact, this is the cost of the unsupervised algorithms compared to the supervised algorithms in which the correct number of the clusters is known as a priori.

2.3 Online FDI scheme

The proposed WCPCM algorithm is used for clustering the historical data points including normal and abnormal conditions and it is applied in an offline mode. However, for real applications, an online procedure is needed to classify the new data points to one of the identified clusters or as a novel fault. In this paper, a similar methodology proposed by Detroja et al. [11] is utilized to perform the online FDI task. A schematic diagram of the online FDI algorithm is illustrated in Fig. 1. After applying the WCPCM to historical data points, cluster centers and feature weights for different clusters are obtained. The membership values of the new incoming data point, i.e. jth data point, to different clusters are calculated by using the following equation:

$$u_{ij} = \frac{1}{\sum_{p=1}^{C} \left(\frac{\tilde{d}_{ij}^2}{\tilde{d}_{pj}^2}\right)^{\frac{1}{m-1}}} \tag{21}$$

where

$$\tilde{d}_{ij}^2 = \sum_{k=1}^{n} (v_{ik})^q d_{ijk}^2 \tag{22}$$

and d_{ijk}^2 is computed by (12). u_{ij} is the membership value of the data vector, constructed from the measurements at each instant, to each cluster and the formula is simply obtained from a simple modification at the FCM membership function in (4). In fact, instead of simple distance measure, a weighted distance measure is utilized here to measure the distances between the new data points and the previously identified cluster prototypes.

If u_{ij} exceeds a user defined threshold τ, the new data point belongs to the ith cluster which could be either a normal condition or a diagnosed fault. However, it is important to recognize that the new data point may have been created due to the short-term transitions introduced for example by a controller, measurement noise, or outlier. Thus, it is necessary to confirm the decision. For this purpose, the algorithm waits for another M samples which can be considered as the delay in detection. If u_{ij} constantly remains high, the belonging of the new data point to the ith cluster is confirmed. On the other hand, if the values of u_{ij} for different clusters do not exceed τ, it indicates that a novel fault has occurred and a new cluster is then formed. Since the objective function of the WCPCM algorithm can be seen as a set of C independent objective functions, the center and feature weights of a specific cluster are not influenced by how other clusters are placed. Therefore, it is sufficient to find only the new cluster center and feature weights from this newly collected data. An added advantage of the proposed scheme is that it reduces the emphasis on rich historical data and one can start with just the normal operating data and develop building the monitoring algorithm as new faults occur. The parameter M depends on the process dynamics and can be set based on either closed loop settling time or plant operator decision. Experimental observations recommend the threshold value τ to be set between 0.5 and 0.7. It is noted that the greater value reduces the probability of false alarms but increases the delay in fault detection. The threshold value has been set to 0.55 in all simulations studies for the online FDI purposes in this work.

3 Simulation study

In this section, the proposed FDI algorithm is tested on two industrial data sets which are TE process benchmark and DAMADICS actuator benchmark in both offline and online modes. The algorithm results are comparatively evaluated with some of the existing FDI methods.

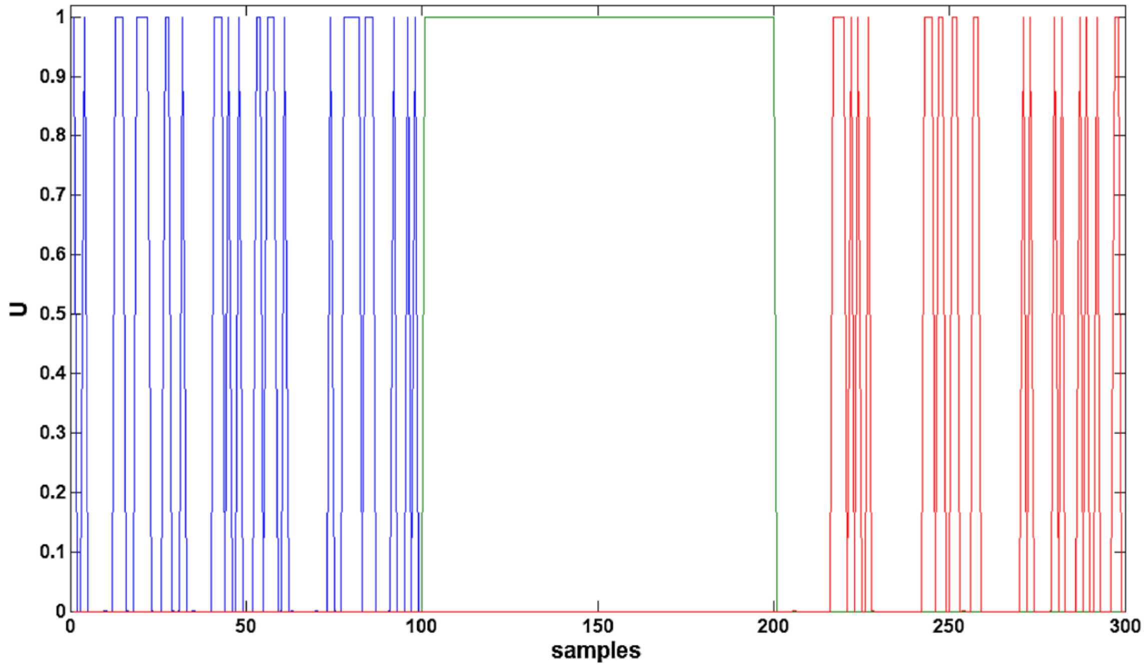
3.1 FDI on TE process benchmark

In this section, a brief description of TE challenge process is proposed followed by implementation of different FDI algorithms to investigate their comparative performances on the TE process plant in offline mode. As a first method in offline mode, some of the faults of the TE process are aimed to be isolated through the Modified Gath-Geva (MGG) algorithm, a powerful time-series segmentation method introduced in [13]. Then, DPCA as a dimension reduction technique is utilized to improve the diagnostic performance of the MGG algorithm, leading to more proper results. Finally, the performance of the proposed WCPCM algorithm is evaluated. In the online application, the WCPCM is utilized to isolate the test data which includes some of the existing faults and a novel fault.

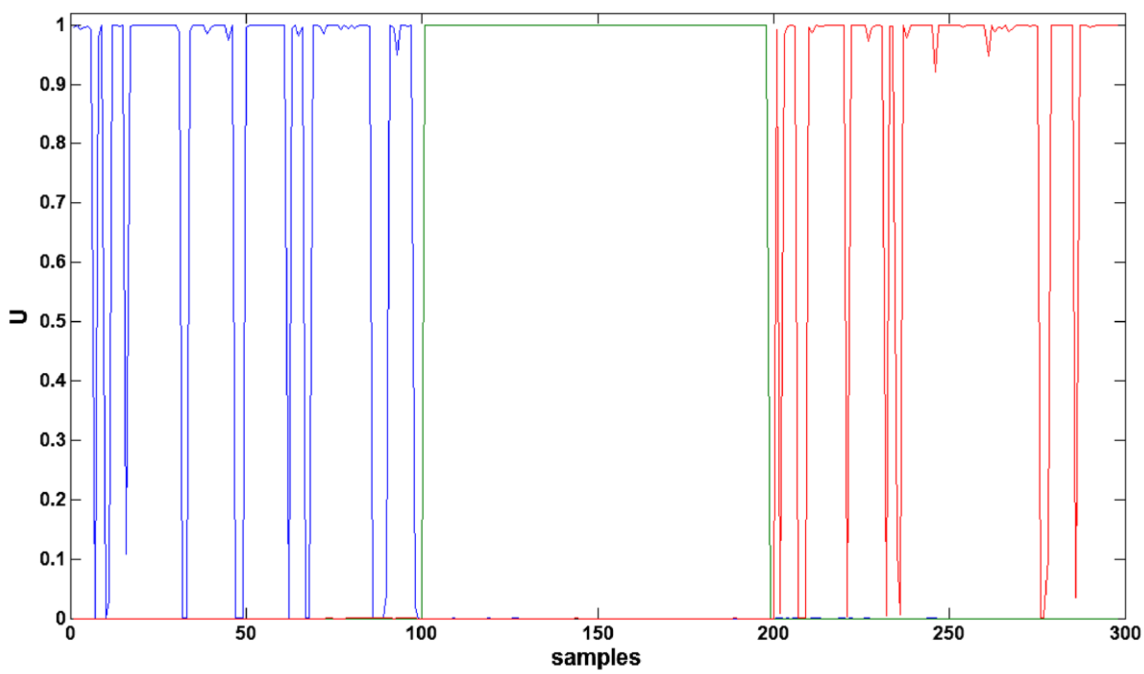
The TE process simulator has been widely used by the process monitoring community as a source of data for comparing various approaches [1, 6, 26]. The process has 22 continuous process measurements, 12 manipulated variables, and 19 composition measurements sampled less frequently. The process has also 21 fault scenarios. This process is described with more details in [1]. A sampling interval of 3 min was used to collect the simulated data for the training and testing sets. The data have been generated by Chiang et al. [1] and can be downloaded from http://brahms.scs.uiuc.edu.

3.1.1 *Clustering of historical data*

In this section, the capability of the proposed WCPCM algorithm in isolating normal operation of TE process with 2 faults of the TE, i.e. faults 4, 6 are considered and it is compared to the different FDI algorithms. A set of 100 data samples, corresponding to each normal and fault training data set, has been used in this test study. To implement the experiment, a time-series data with a length of 300 data samples was constructed by cascading a set of 100 data samples for the normal condition and each of the mentioned TE Faults. The initial parameters are set to $m = 2$, $q = 2$, $l_{\max} = 150$, $\varepsilon = 10^{-4}$. Figure 2(a) illustrates the diagnostic performance of the MGG clustering approach to isolate the TE faults and normal condition. As shown, this approach fails to perform the diagnostic task properly. One of the main reasons is due to the high dimensionality which masks the data set as discussed earlier in Sect. 1. Two of the most well-known solutions for solving high dimensionality are dimension reduction and attribute weighting. Dynamic PCA (DPCA) as a data reduction technique is utilized here to enhance the performance of the MGG algorithm and to remove auto and cross correlations among the data set [27]. DPCA is applied as a preprocessing step to reduce the number of features to the more representative ones. This method is called combined DPCA-MGG algorithm. To implement the combined DPCA-MGG approach, the number of lags h and dominant numbers of principal components (PCs) should first be determined. The method for automatically determining h is described in [27]. In this case study, h is set to 2 which has already been reported as a suitable number in [27] for the TE application. A study on the screeplot of the decomposed



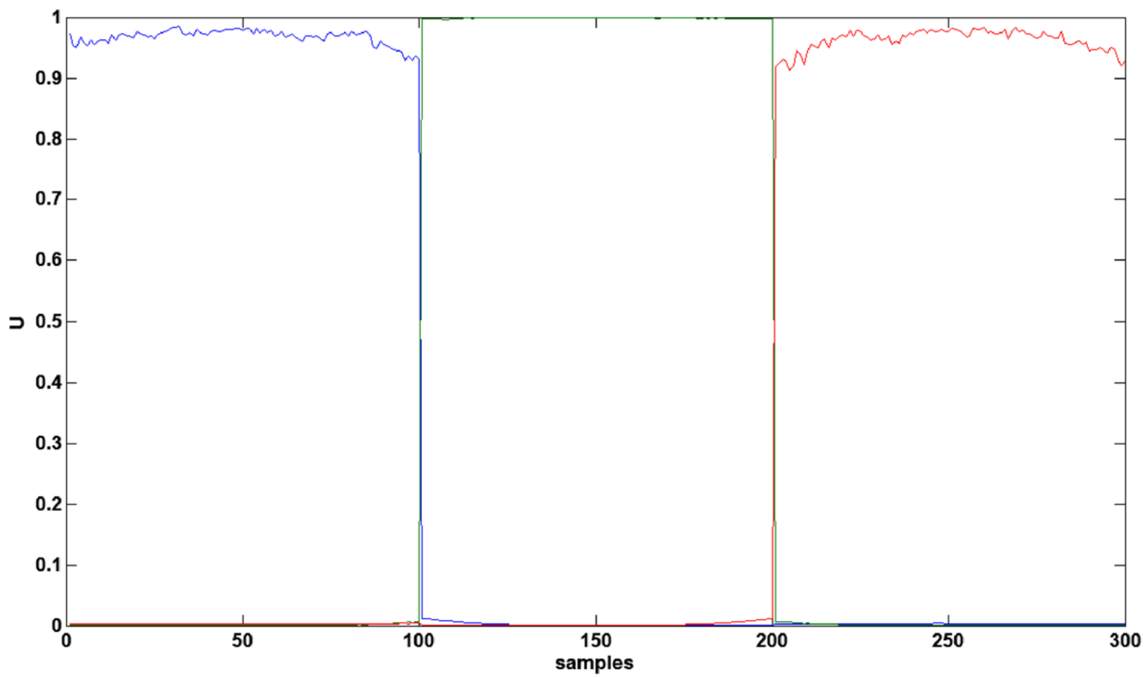
(a)



(b)

Fig. 2 FDI results for the cascaded normal operation and TE faults 4, 6: (**a**) The MGG algorithm; (**b**) The combined DPCA-MGG algorithm; (**c**) The WCPCM algorithm

eigenvalues of the data covariance matrix represents that 10 PCs are enough to capture more than 90% of the covariance of the TE fault data. Therefore, the dimension of features is reduced from 52 to 10. Figure 2(b) illustrates the isolating performance of the combined DPCA-MGG method. As shown, the DPCA-MGG approach performs better than the original MGG approach. However, it cannot discriminate all the induced faults properly. The reason is related to the fact that features are weighted equally in the PCA algorithm, but in reality different features have diverse sig-



(c)

Fig. 2 (*continued*)

nificances for representing distinctive faults. The WCPCM algorithm, in contrast to the combined DPCA-MGG algorithm, assigns variable weights for a feature while representing different clusters. Figure 2(c) illustrates the isolating performance of the WCPCM method. As shown, the faults have been properly isolated outperforming the two previous algorithms. This implies that feature weighting has better performances in clustering of high dimensional data than the data reduction techniques.

3.1.2 Online FDI on TE benchmark

In this section, two TE faults 4 and 6, which were already studied in the offline simulation study, are considered in an online manner. However, TE fault 18 is also simulated as a novel fault. For implementing the proposed online FDI scheme, the results of the WCPCM algorithm obtained in the previous section for normal operation and faults 4, and 6 are used as the training step. Delay in detection, denoted by M, is set to 20 and the threshold τ that defines the belonging of a data vector to the clusters was chosen to be 0.55.

First, a known fault scenario is evaluated using 400 samples of fault 4 which has occurred after 200 samples of normal data. As illustrated in Fig. 3, the membership value variations show that the proposed algorithm has been able to detect and isolate the known fault 4 after 120 samples. It is observed that after the fault has happened at 201th sample, the belonging of the incoming data points to the normal cluster lowers the threshold and this trend continues for another M samples. Therefore, the occurrence of fault is confirmed at 220th sample. In this period, the belongings of the data samples to the cluster of fault 4 increases as illustrated and the fault has been isolated properly.

A similar test study is conducted for fault 6 occurring after 200 samples of the normal operation and continues for the next 400 samples. The results are illustrated in Fig. 4. The fault has been detected and isolated properly after confirming M samples, i.e. the fault is identified properly at the 220th sample.

The WCPCM algorithm is also able to detect novel faults. This is an important characteristic which many existing clustering algorithms lack. To demonstrate this characteristic of the algorithm, a novel fault is tested in the online mode. Fault 18 which has not been included in the historical data clustering is supposed to occur after the 200th sample. The membership values of the faulty samples into the different existing clusters are shown in Fig. 5. The membership values of the samples in the normal cluster reduces the threshold after 200th sample and the occurrence of a fault is confirmed at the 220th sample after another M samples. However, all the other clusters have values smaller than the threshold which indicates that a new fault has occurred. This demonstrates the capabilities of the proposed algorithm to detect novel faults.

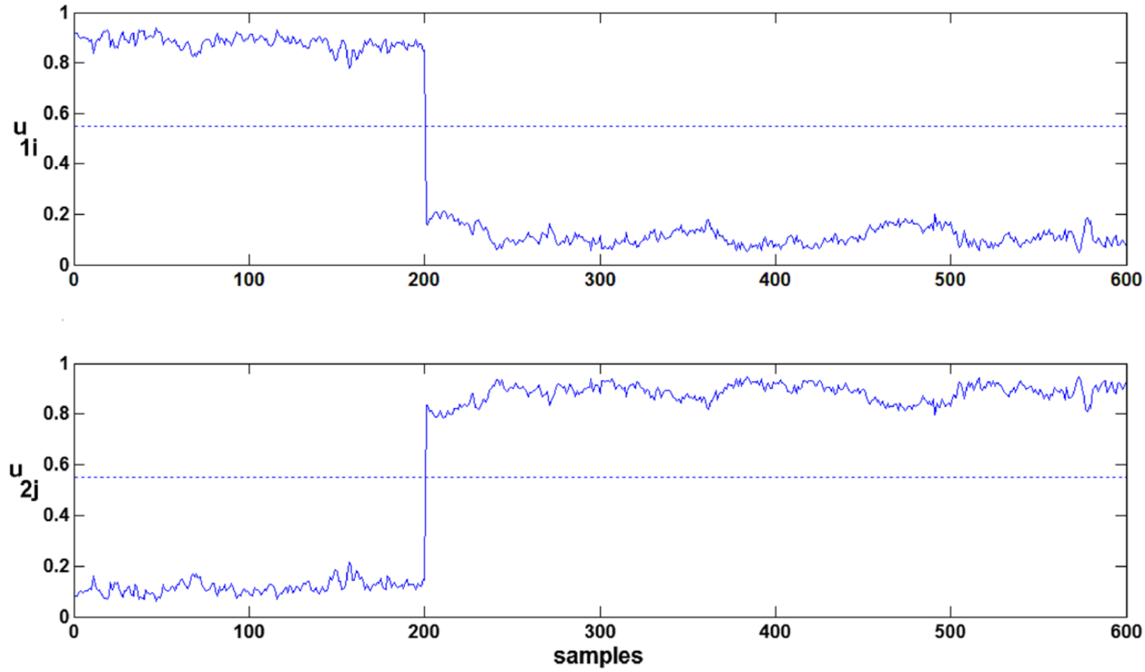


Fig. 3 FDI for a cascade of TE normal and fault 4 operations

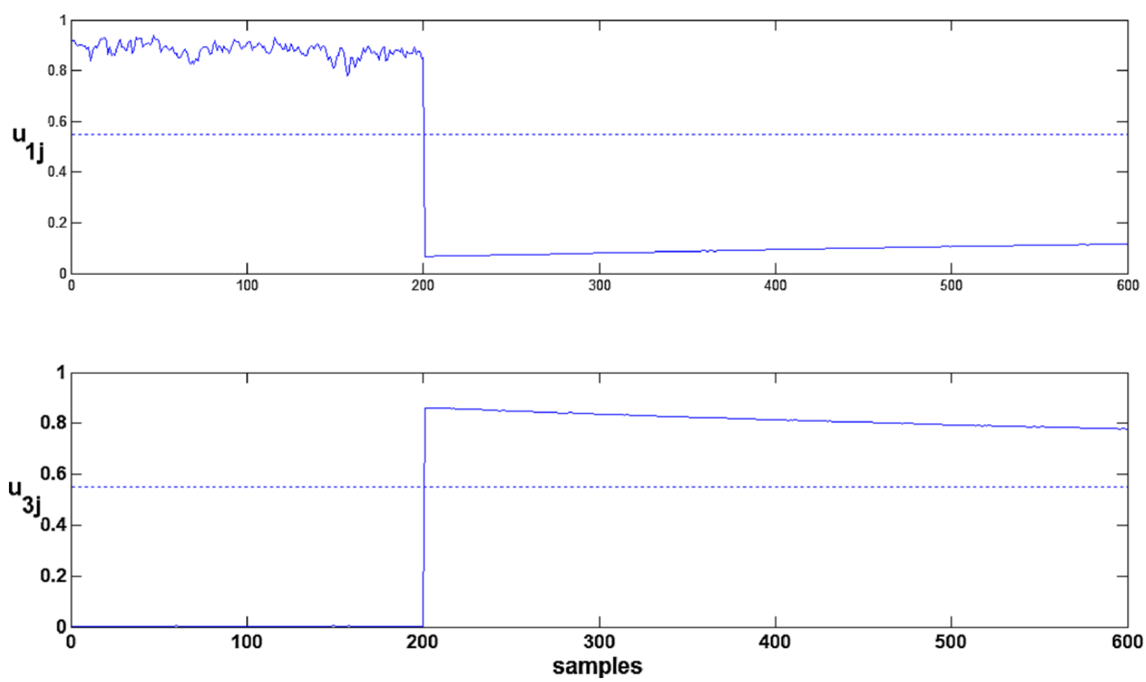


Fig. 4 FDI for a cascade of TE normal and fault 6 operations

3.2 FDI on DAMADICS actuator benchmark

The second case study corresponds to the DAMADICS benchmark (acronym for Development and Applications of Methods for Actuator Diagnosis in Industrial Control Systems). Its main purpose is to develop fault detection and isolation methods for final control elements in the industry environment. It is possible to simulate nineteen abnormal events from three actuators. The detailed description of the model can be found in [29]. A simulator of the actuator model (available for MatLab-Simulink) is employed to generate both normal and faulty data which can be downloaded from http://diag.mchtr.pw.edu.pl/damadics/.

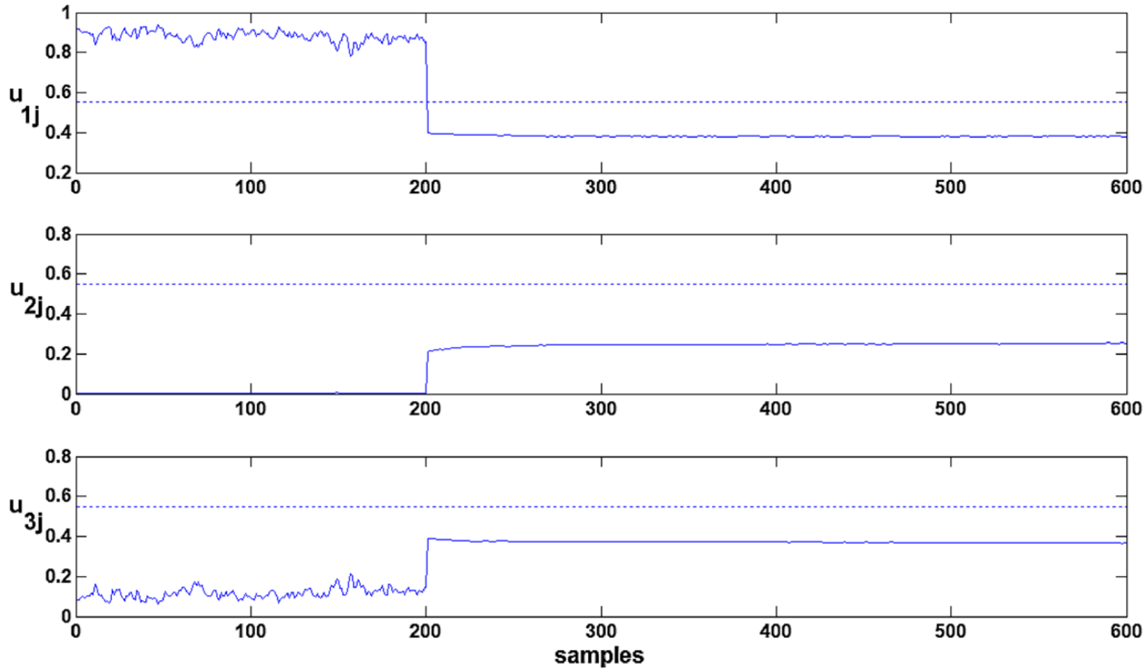


Fig. 5 FDI for a cascade of TE normal and novel fault 18 operations

Table 1 Simulated DAMADICS faults

Fault	Description
f16	Positioner supply pressure drop
f17	Unexpected pressure change across the valve
f18	Fully opened by-pass valve
f19	Flow rate sensor fault

The data set contains 33 variables including inlet and outlet juice pressure, inlet and outlet juice temperature, density, flow and etc. Some of the DAMADICS faults which are used in this section for FDI purpose have been listed in Table 1.

3.2.1 *Clustering of historical data*

The cascade of 5 faults of the DAMADICS along with the normal operation, as listed in Table 2, is considered here. The number of samples in each condition is also listed in Table 2. To implement the experiment, a time-series data with a length of 600 data samples was constructed by cascading the corresponding data samples for each of the mentioned DAMADICS faults and normal operation. The initial parameters were set to the same values considered in the previous test studies. Figure 6 illustrates the isolating performance of the WCPCM algorithm on the simulated cascaded time-series. As it is shown, the algorithm is able to isolate the DAMADICS faults properly. The results of the MGG and the combined DPCA-MGG algorithms are not satisfactory and hence have not been reported for the sake of brevity.

Table 2 Induced time-series for DAMADICS normal and faulty operations

Samples	Cluster
0–100	Normal Operation
101–200	Fault 16 of Valve 1
201–250	Fault 17 of Valve 1
251–350	Fault 18 of Valve 1
351–500	Fault 17 of Valve 2
501–600	Fault 18 of Valve 3

To evaluate the realistic situations in which the measured data may be corrupted by measurement noises, the simulation study is repeated by imposing all the used data set with random Gaussian noises having 0.1 standard deviations and zero mean values. The simulation was run under the same settings. As it is shown in Fig. 7, the WCPCM algorithm is able to successfully isolate the six different clusters irrespective of the corrupted nature of the measured data set. This demonstrates the important robust characteristic of the proposed algorithm in the face of the corrupted data set.

3.2.2 *Determining the number of faults in historical data set*

Although it is known here that we have six classes or regions of operation in this 32 dimensional space, in general this information may not be known beforehand. Therefore, it is important to validate the number of faults after they have been isolated in the historical data. For this purpose,

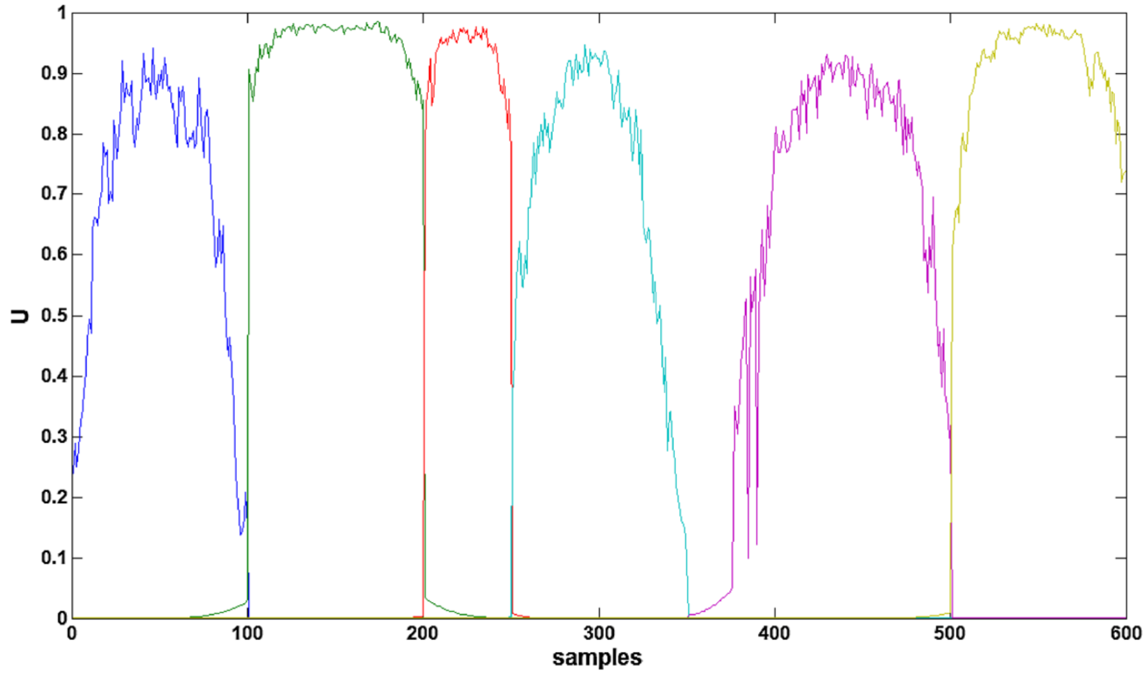


Fig. 6 FDI results for normal operation and 5 faults of the DAMADICS benchmark listed in Table 2

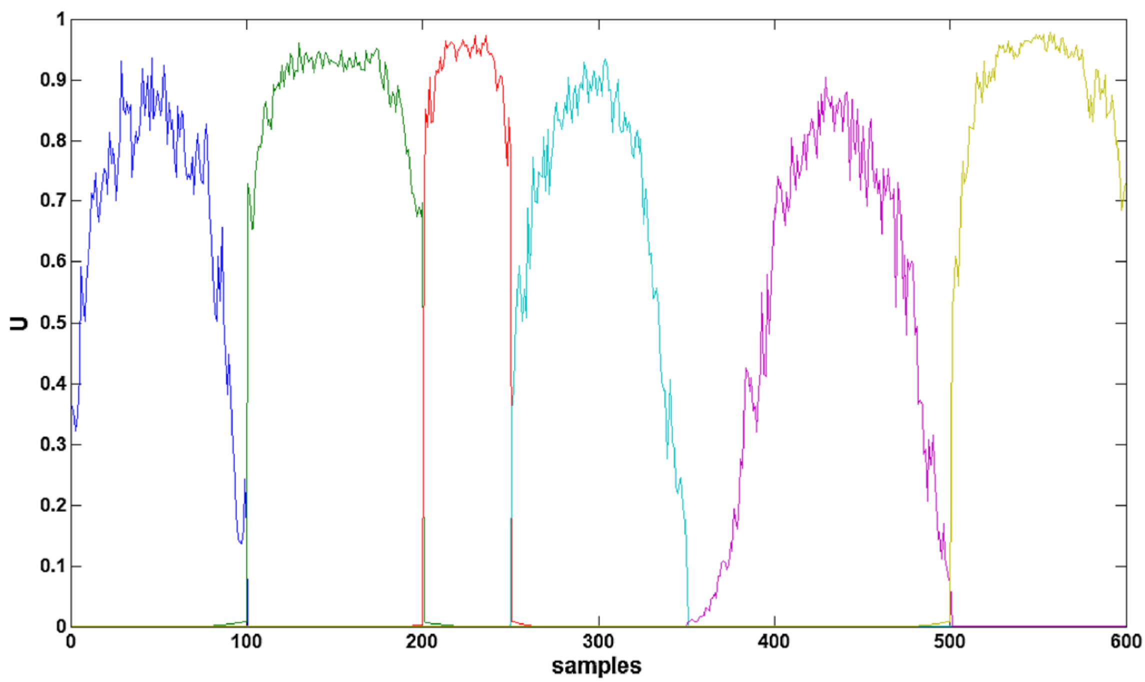


Fig. 7 FDI results for normal operation and 5 faults of the DAMADICS benchmark listed in Table 2 with corrupted data set

it is assumed that the number of clusters is unknown here and should be determined. When applying the WCPCM algorithm, however, a specified number of clusters should be first selected. The number of clusters is set to 7, i.e. more than the real number of the operation conditions. Figure 8 illustrates the results of the WCPCM algorithm in clustering the data sets into seven clusters. As shown, two clusters overlap, indicating that there are indeed only six clusters present in the data set. In fact, the two overlapping clusters can be merged into a single cluster. To study this capability of the algorithm further, the values of the prototypes for the first five feature variable of the DAMADICS benchmark,

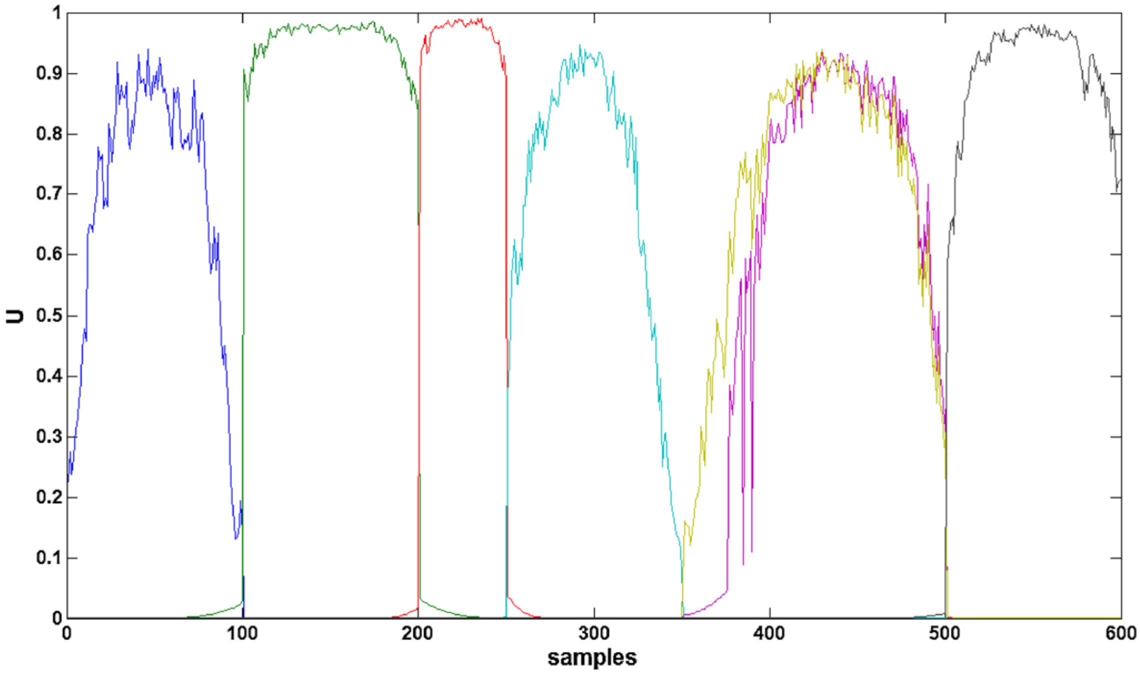


Fig. 8 The WCPCM results for isolating 6 operation conditions with wrong assumed No. of clusters i.e. 7 initial clusters

Table 3 Values of the cluster prototypes for the first five variables of the DAMADICS benchmark obtained by FCM and WCPCM algorithms

Cluster	FCM	WCPCM
1	$(-1.4352, 0.4719, 0.3540, 0.6125, -0.8832)$	$(-1.513, 0.4839, 0.3408, 0.6502, -0.9109)$
2	$(-0.0406, 0.4334, -0.7414, 1.0329, 1.7525)$	$(-0.0846, 0.4425, -0.7422, 1.0802, 1.7418)$
3	$(1.5587, -3.1778, -1.4466, -1.2703, -0.0552)$	$(1.553, -3.1826, -1.446, -1.2635, -0.0565)$
4	$(0.3865, 0.1260, -1.0656, 0.1307, -0.01285)$	$(0.3687, 0.1181, -1.0866, 0.1317, -1024)$
5	$(0.0831, 0.2770, 1.3703, -0.9407, -0.3170)$	$\mathbf{(0.0597, 0.3149, 1.4535, -0.8244, -0.3355)}$
6	$(0.0382, 0.3209, 1.4604, -0.8100, -0.3434)$	$\mathbf{(0.0725, 0.3047, 1.4420, -0.8502, -0.3312)}$
7	$(0.7790, 0.0138, 0.0259, -0.3942, -0.1866)$	$(0.7858, 0.01138, 0.0259, -0.3992, -0.1840)$

obtained by FCM and WCPCM algorithms, are listed in Table 3. The FCM algorithm gives seven distinct cluster centers as shown in Table 3. Cluster validity measures such as the one introduced by Xie–Beni index [31] may be necessary to obtain the correct number of clusters when the FCM algorithm is used. On the other hand, the WCPCM algorithm suggests two overlapping clusters as well as 5 distinct clusters indicating that there are indeed only six clusters present in the data set, i.e. the cluster centers are close to each other for the two overlapping clusters (see Table 3). Thus, The WCPCM clustering algorithm has the capability to lead to the proper number of clusters.

3.2.3 Online FDI

In this section, the simulated DAMADICS faults are considered in order to evaluate the proposed FDI algorithm in an online manner. For examining the evaluation studies with a more challenging situation, the DAMADICS fault 19 of valve 3 is also simulated as a novel fault. To implement the proposed online FDI scheme, the results of the WCPCM algorithm obtained in the previous section for the normal operation and known faults are used as the training step. Delay in detection, denoted by M, is set to 20 and the threshold τ that defines the belonging of a data vector to the clusters was chosen to be 0.55.

A known fault scenario is evaluated using 100 samples of fault 16 of valve 1 which has occurred after 100 samples of normal data samples. As illustrated in Fig. 9, the membership value variations show that the proposed algorithm has been able to detect and isolate the known fault. It is observed that after the fault has happened at the 10th sample, the membership values of the incoming data points to the normal cluster reduces the predefined threshold and this reduction trend continues for another M samples. Therefore, the occurrence of fault is confirmed at 120th sample. Similar test studies have been conducted for the other known faults listed in Table 2 in which the faults occur after 100 samples

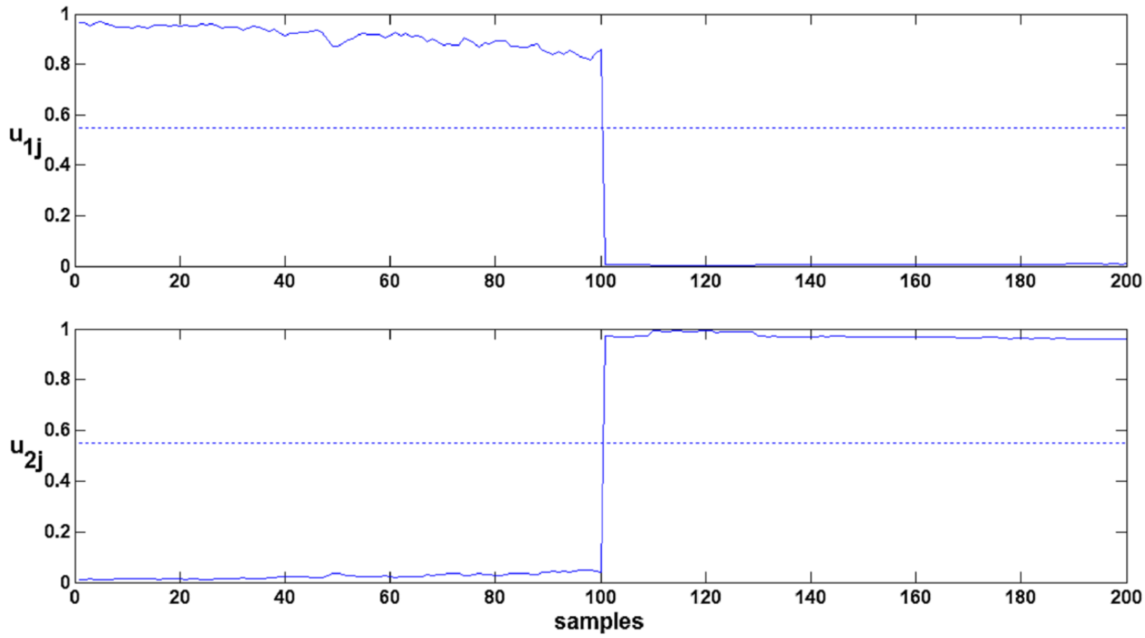


Fig. 9 FDI for a cascade of DAMADICS normal operation and fault 16 corresponding to valve 1 operation

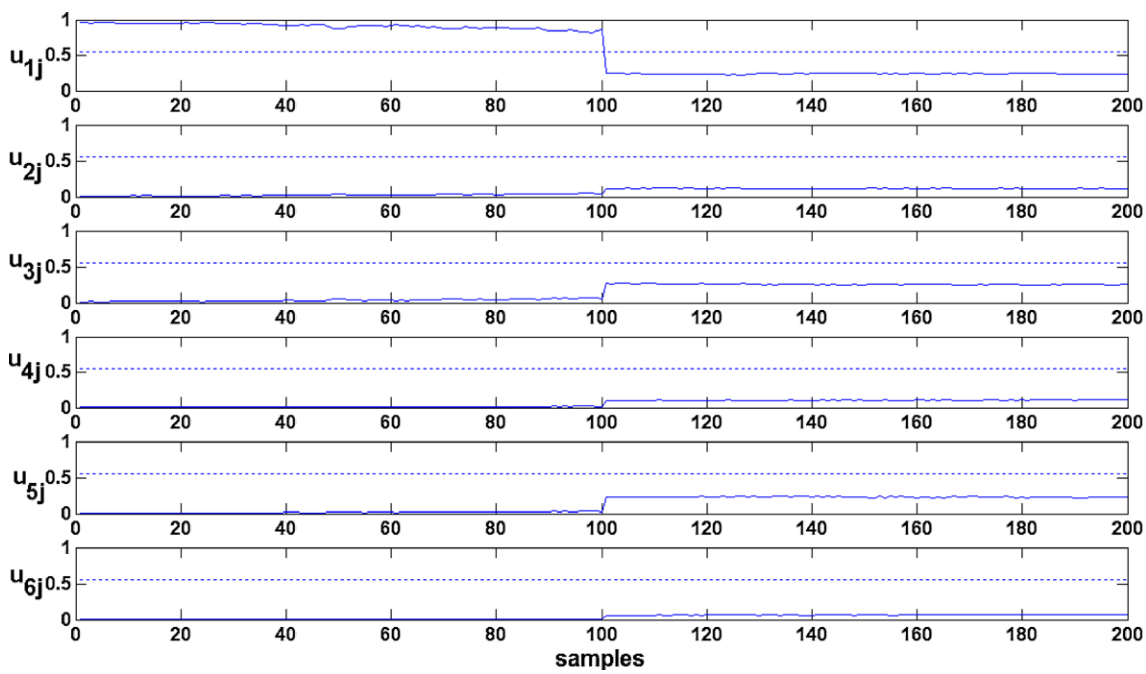


Fig. 10 FDI for a cascade of DAMADICS normal operation and novel fault 19 corresponding to valve 3 operation

of normal operation. The number of false positives in the simulations is 10 out of 500.

To demonstrate the capability of the WCPCM algorithm to detect novel faults, a novel fault is simulated in the online mode. Fault 19 of valve 3 which has not been included in the historical data clustering is supposed to occur after the 100th sample. The membership values of the faulty samples in the different existing clusters are shown in Fig. 10. The membership values of the samples in the normal cluster reduces the threshold after 100th sample and the occurrence of a fault is hence confirmed at the 120th sample after another M samples. However, all the other clusters have membership values smaller than the threshold which indicates that a new fault has been occurred.

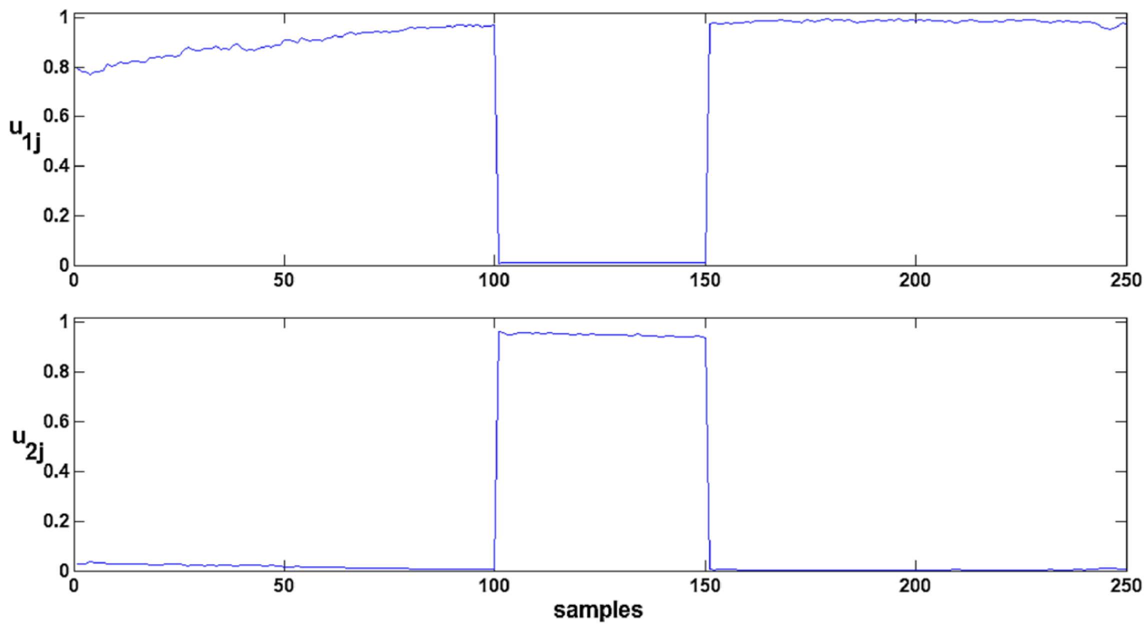


Fig. 11 FDI for a non-stationary fault 16 corresponding to valve 1 operation occurring at samples between 150 to 200

3.3 FDI for non-stationary fault situations

The proposed approach can also detect and isolate faults that can occur at a certain point and then disappear later on. A simulation was carried out to demonstrate this capability of the proposed approach. For this purpose, fault 16 of valve 1 has been induced after 100 samples of normal operation. The fault is then removed suddenly at 150th sample. Figure 11 shows the FDI result of this faulty scenario. As can be seen, the fault is properly detected and isolated by the proposed approach. Later, when the fault is canceled, the monitoring scheme shows that the fault is removed and process operation is back to normal function. Thus, the proposed approach is capable of detecting and isolating different types of fault scenarios.

4 Conclusions

WCPCM has been proposed in this paper as a new clustering algorithm for process FDI applications. The algorithm is based on a weighted and constrained possibilistic C-means approach to effectively accomplish simultaneous possibilistic clustering and feature weighing. This allocates a localized feature weighting set for each fault by optimizing a proposed weighted cost function. This is in contrast to most of the available methods in which a single set of weights is allocated to the entire data set. The resulting scheme facilitates the FDI tasks by finding more dense clusters. The output of the WCPCM algorithm is informative in which the number of the faults can be estimated by some preliminary simulation exploration test runs on the basis of historical data sets and hence it is not necessary to be precisely known as a priori. The performance of the proposed WCPCM algorithm has been evaluated on the TE process plant and the DAMADICS actuator benchmark as two high-dimensional benchmark problems. The resulting comparative studies carried out with the MGG algorithm and the improved MGG algorithm, i.e. the combined DPCA-MGG algorithm, demonstrate the superiority of the proposed algorithm to accomplish exact FDI operations. Simulation results show the capability of the proposed FDI method to detect novel faults, as well.

References

1. Chiang LH, Russell EL, Braatz RD (2001) Fault detection and diagnosis in industrial systems. Springer, London
2. Dunia R, Qin SJ, Edgar TF, McAvoy TJ (1996) Identification of faulty sensors using principal component analysis. AIChE J 42:2797–2812
3. Lee JM, Qin SJ, Lee IB (2006) Fault detection and diagnosis based on modified independent component analysis. AIChE J 52(10):3501–3514
4. Chiang LH, Russell EL, Braatz RD (2000) Fault diagnosis in chemical processes using Fisher discriminant analysis, partial least squares, and principal component analysis. Chemom Intell Lab Syst 50:243–492
5. Wang X, Kruger U, Lennox B (2003) Recursive partial least squares algorithms for monitoring complex industrial processes. Control Eng Pract 11(6):613–632
6. Raich AC, Cinar A (1996) Statistical process monitoring and disturbance diagnosis in multivariable continuous processes. AIChE J 42:995–1009
7. Lieftucht D, Kruger U, Irwin GW (2006) Improved reliability in diagnosing faults using multivariate statistics. Comput Chem Eng 30(5):901–912

8. Krishnapuram R, Keller JM (1993) A possibilistic approach to clustering. IEEE Trans Fuzzy Syst 1(2):98–110
9. Widodo A, Yang BS (2007) Support vector machine in machine condition monitoring and fault diagnosis. Mech Syst Fault Diagn 21:2560–2574
10. Palade V, Patton RJ, Uppal FJ, Quevedo J, Daley S (2002) Fault diagnosis of an industrial gas turbine using neuro-fuzzy methods. In: IFAC, 2002
11. Detroja KP, Gudi RD, Patwardhan SC (2006) A possibilistic clustering approach to novel fault detection and isolation. J Process Control 16:1055–1073
12. Vasko KT, Toivonen HTTT (2002) Estimating the number of segments in time series data using permutation tests. IEEE Int Conf Data Mining 466–473
13. Abonyi J, Feil B, Nemeth S, Arva P (2005) Modified Gath–Geva clustering for fuzzy segmentation of multivariate timeseries. Fuzzy Sets Syst 140:39–56
14. Berkhin P (2002) Survey of clustering data mining techniques. Accrue Software, Inc., Fremont
15. Berry M, Dumais S, Landaure T, Obrien G (1995) Using linear algebra for intelligent information retrieval. SIAM Rev 37:573–595
16. Mardina K, Kent J, Bibby J (1980) Multivariate analysis. Academic Press, San Diego
17. Law MHC, Figueiredo MAT, Jain AK (2004) Simultaneous feature selection and clustering using mixture models. IEEE Trans Pattern Anal Mach Intell 26:1154–1166
18. Wettschereck D, Aha DW, Mohri T (1997) A review and empirical evaluation of feature weighting methods for a class of lazy learning algorithms. Artif Intell 11:273–314
19. Domingos P (1997) Context sensitive feature selection for lazy learners. Artif Intell 11:227–253
20. Srinivasan R, Qian MS (2007) State-specific key variables for monitoring multi-state processes. Chem Eng Res Des 85:1630–1644
21. He QP, Wang J, Qin SJ (2005) A new fault diagnosis method using fault directions in Fisher discriminant analysis. AIChE J 51(2):555–571
22. Teppola P, Minkkinen P (1999) Possibilistic and fuzzy C-means clustering for process monitoring in an activated sludge wastewater treatment plant. J Chemom 13:445–459
23. Frigui H, Nasraoui O (2004) Unsupervised learning of prototypes and attribute weights. Pattern Recogn 37:567–581
24. Bezdek JC (1981) Pattern recognition with fuzzy objective function algorithms. Plenum Press, New York
25. Downs JJ, Vogel EF (1993) A plant-wide industrial-process control problem. Comput Chem Eng 17:245–255
26. Lin W, Qian Y, Li X (2000) Nonlinear dynamic principal component analysis for on-line process monitoring and diagnosis. Comput Chem Eng 24:423–429
27. Ku W, Storer RH, Georgakis Ch (1995) Disturbance detection and isolation by dynamic principal component analysis. Chemom Intell Lab Syst 30:179–196
28. Kelly PM (1994) An algorithm for merging hyperellipsoidal clusters. Technical Report LA-UR-94-3306, Los Alamos National Laboratory, Los Alamos, NM
29. Syfert M, Patton R, Bartys M, Quevedo J (2003) Development and application of methods for actuator diagnosis in industrial control systems (Damadics): a benchmark study. In: Proceedings of the IFAC symposium safe process, pp 939–950
30. Krishnapuram R, Keller JM (1996) A possibilistic C-means algorithm: insights and recommendations. IEEE Trans Fuzzy Syst 4(3):385–393
31. Xie XL, Beni GA (1991) Validity measure for fuzzy clustering. IEEE Trans Pattern Anal Mach Intell 3(8):841–846
32. Jain AK, Murty MN, Flynn PJ (1999) Data clustering: a review. ACM Comput Surv 31(3):264–323
33. Chen J, Howell J (2002) Towards distributed diagnosis of the Tennessee Eastman process benchmark. Control Eng Pract 10:971–987