An Adaptive Clustering Procedure with Applications to Fault Detection

Milan R. Rapaić, Milena Petković, Zoran D. Jeličić, and Alessandro Pisano

***Abstract*—A novel adaptive clustering procedure is presented in this paper. Among the basic properties of the proposed algorithm is that the number of clusters is not known a priori, it is updated automatically based on the available data. Previous knowledge regarding data source (data generating process), if available, can be used for initialization purposes. However, the algorithm can be used even if such information is not available. The entire data set need not be known in advance, and further, the algorithm does not store previously seen data points. The computation complexity is relatively low and the entire procedure may be implemented recursively (in "real-time"). The proposed procedure is designed primarily for condition monitoring and fault detection in industrial plants. Performances of the proposed algorithm have been demonstrated by an illustrative example.**

***Index Terms*—Classification, clustering, novelty identification, fault detection and isolation (FDI), condition monitoring.**

I. Introduction

A NUMBER of classification and clustering techniques have been applied successfully in various fields of science and engineering. A general overview of different statistical procedures can be found in [1], while a gamut of soft computing techniques, including techniques based on artificial neural networks, support vector machines (SVMs) and fuzzy logic, is presented in [2].

Most of the proposed techniques are capable of distinguishing among several fixed and *a priori* known classes (cluster centers). There are however cases in which all classes are not known in advance, and also cases where the properties of the existing classes change (evolve) over time. In such cases it would be preferable to use a classification algorithm which is capable of learning from the input data stream and evolving (adapting) with it.

Fault detection and isolation (FDI) is an important filed of modern process engineering. A survey of modern approaches to fault diagnosis can be found in [3]-[5]. It has been claimed [3] that any fault diagnosis technique can be seen as (or at least reduce to) a classification procedure: a fault is either present or it is not. A more sophisticated diagnosis classifier would even be capable of distinguishing among different types of faults, and thus perform fault isolation. However, it is rather difficult to develop a diagnosis procedure which would be capable of identifying novel faulty conditions, ones not seen during design phase. Also, most FDI techniques are incapable of coping with the drift of some of the existing working conditions. The ability to identify novelties and adapt to the changing environmental conditions is identified as one of the desirable features of any fault diagnosis technique [3].

The aim of the present paper is to present an adaptive clustering procedure capable of identifying novelties and respond to changes in the input data stream. When applied in an FDI setting, the proposed procedure should be capable of detecting working conditions which occur for the first time during the exploitation of the algorithm. These newly recognized working conditions can be labeled as either nominal or faulty by an experienced operator. The proposed technique can be seen as an evolving classifier and is inspired by the on-line identification procedure proposed by Angelov and Filev in [6], as well with the *eClass* family of evolving fuzzy classifiers [7]. It is also of interest to note that a preliminary version of the proposed algorithm has been addressed also in [8] and [9], where a number of experimental case studies have been considered. The proposed technique may be seen as an alternative to the classical model-based fault detection techniques [10] and also to hybrid fault detection techniques, such as the one presented in [11].

The key underlying notion of information potential is introduced in Section II. The information potential used here is different than the information potential utilized in [7]. The general structure of the proposed adaptive clustering procedure is presented in Section III. Specialization of the proposed algorithm for FDI purposes is described in Section IV. Experimental results are presented in Section V. Concluding remarks are left for Section VI.

M.R.R, M.P. and Z.D.J. would like to acknowledge financial support of Serbian Ministry of Science and Technological Development, grant no. 32018. Some results of this paper were presented at 55th ETRAN Conference, Banja Vrućica, June 6-9, 2011.

M. R. Rapać is with Computing and Control Department, Faculty of Technical Sciences, University of Novi Sad; Trg Dositeja Obradovića 6. 21000 Novi Sad (phone: 303-555-5555; fax: 303-555-5555; e-mail: rapaja@uns.ac.rs).

M. Petković is with Computing and Control Department, Faculty of Technical Sciences, University of Novi Sad (e-mail: milena5@uns.ac.rs).

Z. D. Jeličić is with Computing and Control Department, Faculty of Technical Sciences, University of Novi Sad (e-mail: milena5@uns.ac.rs).

A. Pisano is with Deprartment of Electrical and Electronic Engineering (DIEE), University of Cagliari, Cagliari, Italy.

II. Information Potential

In [7] Angelov and Filev proposed an adaptive clustering procedure relying on the notion of the **information potential**. An information potential of a point with respect to an ordered set of points is *a measure of similarity* between the point and the set. If the information potential of a point is high, the point is representative with respect to other points within the set. Points with the highest values of the information potential are candidates for cluster centers.

Let us define the notion of the information potential more formally. Consider a set of n *features*, each of which is represented as a real number. The *feature space* $\mathcal{F}$ is therefore a subset of $\Re^n$. Let $\mathbf{z} \in \mathcal{F}$ be an arbitrary *feature vector* and let $\mathcal{Z} = \{\mathbf{z}_j \mid j = 0..N\} \subseteq \mathcal{F}$ be an ordered set of known feature vectors. The assumption is that the classification is made on-line, recursively as each new feature point becomes available. Elements of the feature set are ordered in the same sequence they become available: $\mathbf{z}_0$ is the first obtained feature vector, $\mathbf{z}_N$ is the most recent one. In order to emphasize the generality of the proposed classifier, the feature vectors will be denoted simply as the *data points*.

The **mean square distance** of a data point $\mathbf{z}$ with respect to the ordered data set $\mathcal{Z}$ is defined as [7]

$$S(\mathbf{x},\mathcal{Z}) = \frac{1}{N+1}\sum_{j=0}^{N} \|\mathbf{x}-\mathbf{x}_j\|^2 , \tag{1}$$

where $\|\cdot\|$ denotes the standard Euclidean 2-norm,

$$\|\mathbf{x}\| = \sqrt{\mathbf{x}^T\mathbf{x}} \tag{2}$$

and T in superscript denotes matrix- (or in this case, vector-) transposition. As usual, all vectors are implicitly assumed to be column-vectors. Angelov and Filev defined the information potential as a Cauchy-type function of the mean square difference

$$P(\mathbf{x},\mathcal{Z}) = \frac{1}{1+S(\mathbf{x},\mathcal{Z})} . \tag{3}$$

The important property of the information potential is that its value cannot change abruptly. The information potential is computed with respect to all previously seen data points, and thus if the underlying data shift in the feature space, it would take some time for the value of the information potential to reflect this change. Normally this is good and desirable, since it is the basic mechanism providing resilience to measurement noise. Any outliers, invalid and noisy data points, would typically be isolated and thus would not have the capability of significantly changing potential values. However, in the fault diagnosis setting, the fact that all previous data points are taken into consideration may impose a significant drawback. It is the nature of faults to appear after long periods of nominal process operations. Depending on the sampling rate, after some operation time the nominal data points in $\mathcal{Z}$ will vastly outnumber faulty ones. Consequently, the ability of the average square distance (and thus also the information potential) to detect changes in the feature data stream will be impaired.

To overcome this problem, the notion of mean square distance will be redefined in the current paper. Let us define the **exponentially windowed mean square distance** between an arbitrary data point $\mathbf{z}$ and all data points belonging to $\mathcal{Z}$ as

$$S(\mathbf{z},\mathcal{Z},\lambda) = (1-\lambda)\sum_{j=0}^{N} \lambda^{N-j} \|\mathbf{z}-\mathbf{z}_j\|^2 \tag{4}$$

where $\|\cdot\|$ may, in principle, denote an arbitrary norm in the feature space. In the sequel, however, $\|\cdot\|$ will always denote the **weighted Euclidean 2-norm**, defined as

$$\|\mathbf{x}\| = \sqrt{\mathbf{x}^T\mathbf{W}\mathbf{x}} \tag{5}$$

with $\mathbf{W}$ being a suitably chosen, positive definite, symmetric matrix. Practically, the weight matrix is usually diagonal, and serves the purpose of data scaling. Thus, the overall mean square distance (1) is replaced by the mean square distance (4) which effectively takes into consideration the recent points only. In fact, only the points appearing within an exponential time window are accounted for. The parameter λ controls the width of this window and has the same purpose as the forgetting factor in the recursive least squares algorithm. The λ parameter should be chosen in range $[0,1]$ Depending on the sample rate, the noisiness of the signal and the dynamics of the process, the usual choice is between 0.9 and 1.

The information potential can be defined as a Cauchy function of the newly introduced exponentially weighted mean square distance (3). This is the approach proposed in [7] and also used in [9]. It is however also possible to use other, more general definitions. Let us define the potential of a data point $\mathbf{z}$ with respect to the data set $\mathcal{Z}$ as a positive, real-valued mapping of the exponentially weighted mean square distance, $P(\mathbf{z},\mathcal{Z},\lambda) \equiv P(S(\mathbf{z},\mathcal{Z},\lambda))$, with the following properties: for all $S \geq 0$,

- (P1) $P(S) \in [0,1]$,
- (P2) $P(S) = 1 \Leftrightarrow S = 0$
- (P3) $P(S) = 0 \Leftrightarrow S \to \infty$
- (P4) P is a differentiable, monotonically decreasing function of S, $P'(S) < 0$.

The first property (P1) simply states that a potential takes values from a compact set $[0, 1]$. The property (P2) states that a point $\mathbf{z}$ has the potential equal to 1 if and only if all the points in $\mathcal{Z}$ are equal to $\mathbf{z}$, i.e. if and only if for all $\mathbf{z}_j \in \mathcal{Z}$, $\mathbf{z} = \mathbf{z}_j$. (P3) states that the potential becomes equal to zero if and only if the distance to at least some point in $\mathcal{Z}$ grows without an upper bound; a situation which is not to be expected in practice. (P4) implies that there is a well-defined inverse function $P^{-1}:[0,1]\to[0,+\infty]$. It is easily seen that the Cauchy-type function (3) satisfies requirements (P1)-(P4).

Any functions satisfying properties (P1)-(P4) can be considered as a mean proximity measure. In fact, the clustering procedure can be designed solely on the basis of the mean square distance, but it is more convenient to use the notion of the information potential. By using the information potential, the proposed algorithm can be tightly related to the entire family of fuzzy classifiers, as well as to the evolving eClass classifiers [2], [7].

An important property of the previously introduced average square distance and information potential is that it is possibly to compute them recursively. In other words, any classification algorithm which is based on them can be implemented in real time. This is shown by the following two theorems. The following conventions be adopted in the proof: $P(\mathbf{z},\mathcal{Z}_k,\lambda)\equiv P_k(\mathbf{z})$, $S(\mathbf{z},\mathcal{Z}_k,\lambda)\equiv S_k(\mathbf{z})$.

Theorem 1. Let $S(\mathbf{z}_k,\mathcal{Z}_k,\lambda)\equiv S_k(\mathbf{z}_k)\equiv S_k$ be defined by (4), and let the norm be defined as in (5). The following recursion holds

$$S_k=\lambda S_{k-1}+2\lambda(1-\lambda)\left(\mathbf{z}_k-\mathbf{z}_{k-1}\right)^T\mathbf{W}\mathbf{F}_k+\lambda(1-\lambda)\|\mathbf{z}_k-\mathbf{z}_{k-1}\|^2 Z_k, \quad (6)$$

$$S_0=0, \quad (7)$$

$$\mathbf{F}_k=\lambda\mathbf{F}_{k-1}+\lambda Z_{k-1}\left(\mathbf{z}_{k-1}-\mathbf{z}_{k-2}\right), \quad (8)$$

$$\mathbf{F}_0=\mathbf{0}, \quad (9)$$

$$Z_k=\lambda Z_{k-1}+1, \quad (10)$$

$$Z_0=0. \quad (11)$$

Proof. The key point is to note that

$$\begin{aligned}\|\mathbf{z}_k-\mathbf{z}_j\|^2&=(\mathbf{z}_k-\mathbf{z}_j)^T\mathbf{W}(\mathbf{z}_k-\mathbf{z}_j)\\&=(\mathbf{z}_k-\mathbf{z}_{k-1}+\mathbf{z}_{k-1}+\mathbf{z}_j)^T\mathbf{W}(\mathbf{z}_k-\mathbf{z}_{k-1}+\mathbf{z}_{k-1}+\mathbf{z}_j)\\&=\|\mathbf{z}_k-\mathbf{z}_{k-1}\|^2+2(\mathbf{z}_k-\mathbf{z}_{k-1})^T W(\mathbf{z}_k-\mathbf{z}_j)\\&\quad+\|\mathbf{z}_{k-1}-\mathbf{z}_j\|^2\end{aligned}$$

By means of the above expression, it is readily obtained that

$$\begin{aligned}S_k&=(1-\lambda)\sum_{j=0}^{k}\lambda^{k-j}\|\mathbf{z}_k-\mathbf{z}_j\|^2\\&=(1-\lambda)\sum_{j=0}^{k-1}\lambda^{k-j}\|\mathbf{z}_k-\mathbf{z}_j\|^2\\&=(1-\lambda)\|\mathbf{z}_k-\mathbf{z}_{k-1}\|^2\sum_{j=0}^{k-1}\lambda^{k-j}\\&\quad+(1-\lambda)2(\mathbf{z}_k-\mathbf{z}_{k-1})^T\mathbf{W}\sum_{j=0}^{k-1}\lambda^{k-j}(\mathbf{z}_k-\mathbf{z}_j)\\&\quad+(1-\lambda)\sum_{j=0}^{k-1}\lambda^{k-j}\|\mathbf{z}_{k-1}-\mathbf{z}_j\|^2\end{aligned}$$

By defining

$$Z_k=\sum_{j=0}^{k}\lambda^{k-j}$$

$$\mathbf{F}_k=\sum_{j=0}^{k}\lambda^{k-j}(\mathbf{z}_k-\mathbf{z}_j),$$

the recursive expressions (6), (8) and (10) are obtained using simple algebraic manipulations.

Theorem 2. Let $S(\mathbf{z}_i^*,\mathcal{Z}_k,\lambda)\equiv S_k(\mathbf{z}_i^*)$ be defined by (4), and let the norm be defined by (5). Then

$$S_k(\mathbf{z}_i^*)=(1-\lambda)\|\mathbf{z}_k-\mathbf{z}_i^*\|+\lambda S_{k-1}(\mathbf{z}_i^*). \quad (12)$$

Proof. By definition,

$$S_k(\mathbf{z}_i^*)=(1-\lambda)\sum_{j=0}^{k}\lambda^{k-j}\|\mathbf{z}_i^*-\mathbf{z}_j\|^2.$$

The recursive form is obtained by extracting the most recent term in the sum

$$\begin{aligned}S_k(\mathbf{z}_i^*)&=(1-\lambda)\sum_{j=0}^{k-1}\lambda^{k-j}\|\mathbf{z}_i^*-\mathbf{z}_j\|^2+(1-\lambda)\|\mathbf{z}_i^*-\mathbf{z}_k\|^2\\&=\lambda(1-\lambda)\sum_{j=0}^{k-1}\lambda^{k-1-j}\|\mathbf{z}_i^*-\mathbf{z}_j\|^2+(1-\lambda)\|\mathbf{z}_i^*-\mathbf{z}_k\|^2\end{aligned}$$

which is equivalent to (12).

The two theorems above testify that it is possible to implement a recursive clustering procedure based on an arbitrary potential function provided it depends solely on the exponentially windowed mean square distance (4). The Theorem 1 shows how to compute the potential of the new feature vector, while the Theorem 2 shows how to recursively update the potential of the cluster centers.

III. General Structure of the Proposed Adaptive Clustering Algorithm

The information potential defined in the previous section reflects a measure of similarity between the current and recent process behavior. If the information potential of the feature point associated with the current process behavior is high, then there has not been any sudden change in the process behavior. If, however, the potential of the current feature is low, then the abrupt change in process behavior is likely to have happened in recent past.

The algorithm itself keeps track of a certain number of data points (features) which previously had high values of the information potential. These features will be referred to as **focal points** or **foci**, and sometimes also as **nodes**. Each node is, in fact, a feature vector found to be the most representative for certain area of the feature space (and thus for certain working condition in the FDI setting). The set of all nodes defines the underlying **knowledge base** of the algorithm.

Information potential is computed recursively for each of the existing nodes using Theorem 2. The **active node (active focus)** is the one with the highest value of the information potential. The **estimated current working condition** *is the* *condition associated with the active node*. Thus, the classification is performed in a *winner-takes-all* manner.

In order to provide adaptability the information potential is also computed for the most recently observed feature point (the **current feature**). If this information potential is high, *higher than the potential of any of the existing feature points*, the current feature vector is more representative of the current working regime than any of the focal points present: *the knowledge base of the algorithm should be modified*. The modifications can be twofold. If some of the previously seen clusters changed slightly, or if a better representative of an existing cluster is found, the position of some focal points changes, but the knowledge base remain of the same size. If, however, a new cluster (potentially, a new working condition) is found, the knowledge base grows. More precisely, if the current working condition is similar to some of the existing nodes, the existing node is replaced by the current feature vector. If, on the other hand, the current feature is far from any of the existing nodes, the current feature is proclaimed as the new focal point and the underlying data structure of the algorithm grows. The pseudo-code of the algorithm is presented in Listing 1.

The algorithm may be initialized on the basis of *a priori* knowledge. In this case, the initial nodes are selected as the expected positions of typical cluster centers in the feature space. In the FDI setting, the initial nodes should be selected so to correspond to nominal working conditions and also to the typical faults. If, however, a priori knowledge is not available, the algorithm may be left uninitialized. In such a case, the first feature vector is selected as the initial node.

Listing 1. *Pseudo-code of the proposed adaptive clustering procedure.*

```
1. Choose an appropriate feature generator.
2. Choose algorithm parameters.
3. Initiate the knowledge base
4. BEGIN LOOP
   5. Obtain current feature vector
   6. Compute, on the basis of Theorem 1, the information
      potential of the current feature vector.
   7. Compute, on the basis of Theorem 2, the information
      potential of the existing nodes.
   8. IF the information potential of the current feature
      vector is higher than the information potential of all
      existing nodes THEN
      9. IF the current feature vector is close to some of the
         existing nodes THEN
         10. Modify the knowledge base: replace the closest
             existing node with the current feature vector
      11. ELSE
         12. Extend the knowledge base: create a new node
             equal to the current feature vector.
      END IF
   END IF
END LOOP
```

In any time instant, the number of working conditions the algorithm may identify is equal to the number of nodes. Each node is representative for a set of feature values, i.e. each node is representative for a specific working condition of the underlying process.

The proposed algorithm, or better say its knowledge base, may be seen as an self-adaptive artificial neural network (ANN) with structure presented in Fig. 1. The hidden layer of the network contains the nodes of the algorithm. Once a new measurement is available, the existing nodes are compared to it, similar as with the radial basis function neural networks (RBF-NN). The difference is that the information potential is used as the basis for comparison, not the algebraic distance. The advantage of using the information potential is in the considerable robustness to noise and outliers [7]. Apart from that, the neural network is constantly being trained, both in the terms of its parameters (positions of the nodes in the hidden layer) as well as in terms of its structure, since the number of nodes is growing.

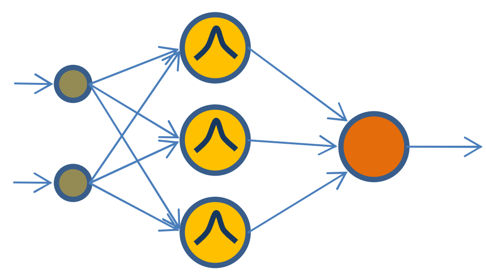

Fig. 1. The general, network-like structure of the knowledge base.

IV. SPECIALIZATION OF THE PROPOSED ALGORITHM FOR FDI

When applying the proposed adaptive clustering procedure in the field of fault detection it is important to select a **feature generator** properly. A feature generator is an algorithm, or a device, which extracts features from process measurements. In this paper, we consider faults which change the dynamic behavior of the process under consideration. Thus, it is natural to choose a *Kalman filter* [12] as a feature generator. Once the dynamic behavior of the process changes, so will the estimated values of its model parameters.

The proposed adaptive classification algorithms will search for the points in the parameter space which are most representative for current working regimes. If the underlying process is nonlinear, which is usually the case, a single working conditions may be described by multiple linear models, i.e. by several focal points in the feature space.

It is impossible for the detection algorithm to automatically detect that several focal points are in fact related to the same working condition. This should be done by an experienced operator. The operator should also be responsible for accurately describing each of the focal points, or at least as labeling them as nominal or faulty. Even more, a detailed description may be assigned to each of the recognized working conditions. In the case of faulty working regimes, the assigned description may even contain a prescription of actions needed to either fix the faulty conditions or minimize its impact. Thus, the proposed algorithm is most effective when suitably combined with expert knowledge. The overall structure of the proposed fault diagnosis system is presented in Fig. 2.

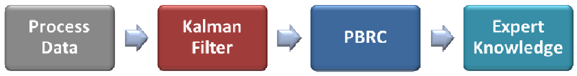

Fig. 2. The general structure of the proposed adaptive FDI scheme.

V. EXPERIMENTAL VERIFICATION

Consider the pneumatic experimental setup presented in Fig. 3. The air flow is actuated by means of a pneumatic servo-valve (**PV**), which is controlled by means of a standard current signal via current/pressure converter (**IPC**). The pressure is measured by means of a pressure meter (**PM**) and a pressure transmitter (**PT**). Behind the setup presented in Fig. 3. there is an air tank. The tank can be opened (connected to the rest of the pneumatic setup) or closed by means of a manual on/off valve (**MV**). Output valves (**OV**), simulating air consumers, are also manual.

The following experiment has been conducted. The two working conditions of the experimental setup are:

1. *the air tank is opened* (**O**);
2. *the air tank is closed* (**C**).

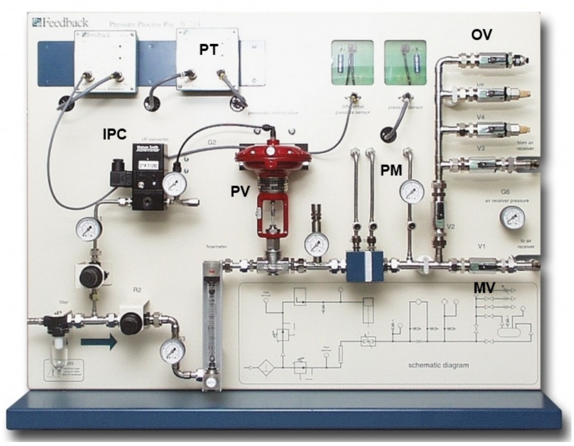

Fig. 3. The pneumatic experimental setup.

Initially, the knowledge base of the algorithm has been kept empty: the clustering procedure is unaware of both working conditions. The state of the system is changed in the manner presented in Table I. The process model is assumed in the form

$$y[k] = -a\,y[k-1] - b\,y[k-2] + c\,u[k-1] + d\,u[k-2], \quad (13)$$

where a, b, c and d are the parameters obtained by means of the Kalman filter, y is the pressure signal and u is the servo-valve command signal. The forgetting factor of the Kalman filter was selected as *0.99*. Both the system order and the forgetting factor have been selected empirically. The value of forgetting factor provided a good trade-off between detection speed and robustness to measurement noise.

TABLE I
THE SCHEDULE OF WORKING REGIMES IN THE EXPERIMENT

Start Time (sec)	Stop Time (sec)	Duration (sec)	Working Regime
0	325 (5 min 25 sec)	325 (5 min 25 sec)	TANK CLOSED
325 (5 min 25 sec)	460 (7 min 40 sec)	135 (2 min 15 sec)	TANK OPENED
460 (7 min 40 sec)	555 (9 min 15 sec)	95 (1 min 35 sec)	TANK CLOSED
555 (9 min 15 sec)	660 (11 min 00 sec)	105 (1 min 45 sec)	TANK OPENED
660 (11 min 00 sec)	875 (14 min 35 sec)	225 (3 min 45 sec)	TANK CLOSED
875 (14 min 35 sec)	1003 (16 min 43 sec)	128 (2 min 8 sec)	TANK OPENED

Even if the process is in a fixed working regime, some time is needed for the outputs of the Kalman filter to settle to their steady-state values. During this transient, the outputs of the Kalman filter are not relevant for the actual process behavior, but are primarily influenced by the internal dynamics of the Kalman filter itself. Thus it is preferable for the clustering procedure to ignore a certain amount of data initially received from the feature generator. In the current experiment the first 1000 samples coming from the Kalman filter are discarded, and the adaptive classification is effectively initiated with 1000 samples delay with respect to the feature generation process.

The λ parameter of the proposed adaptive classification scheme is chosen to be 0.99, the same as the forgetting factor of the Kalman filter. New focal points were introduced if the distance between the candidate focus (current feature vector with high value of the information potential) and the closest existing focal point is greater than 0.25.

Input and output data, i.e. valve command and pressure value, measured during the experiment are presented in Fig. 4. The algorithm has been implemented on a National Instruments cRIO (Compact RIO) real time controller. The sampling rate was *0.1* sec. It is noticeable from the figures that the dynamics of the system with opened air tank is considerably slower comparing to the dynamics of the system with air tank closed. The difference in the dynamic behavior is also clearly noticeable in the outputs of the Kalman filter, presented in Fig. 5. The final results of adaptive classification are presented in Fig. 6. The parameters of each particular focal point (node) found by the adaptive classifier are presented in Table II.

TABLE II
LINEAR MODELS OBTAINED BY THE PROPOSED ADAPTIVE CLUSTERING PROCEDURE

Model No.	a	b	C	D	Associated state
1	0	0	0	0	INITIAL
2	-1.1088	0.1203	-0.0588	0.0644	TRANSIENT
3	-1.795	0.6788	-0.0487	0.0465	**TANK CLOSED**
4	-1.3836	0.3834	-0.03973	0.0388	**TANK CLOSED**
5	-1.8992	0.9009	-0.0479	0.0479	**TANK CLOSED**
6	-3.9408	2.9310	-0.0667	0.0670	TRANSIENT
7	-2.7386	1.7312	-0.0009	0.0003	TRANSIENT
8	-0.5470	-0.4672	-0.0004	-0.0025	**TANK OPENED**
9	-0.8566	-0.1449	-0.0072	0.0036	**TANK OPENED**

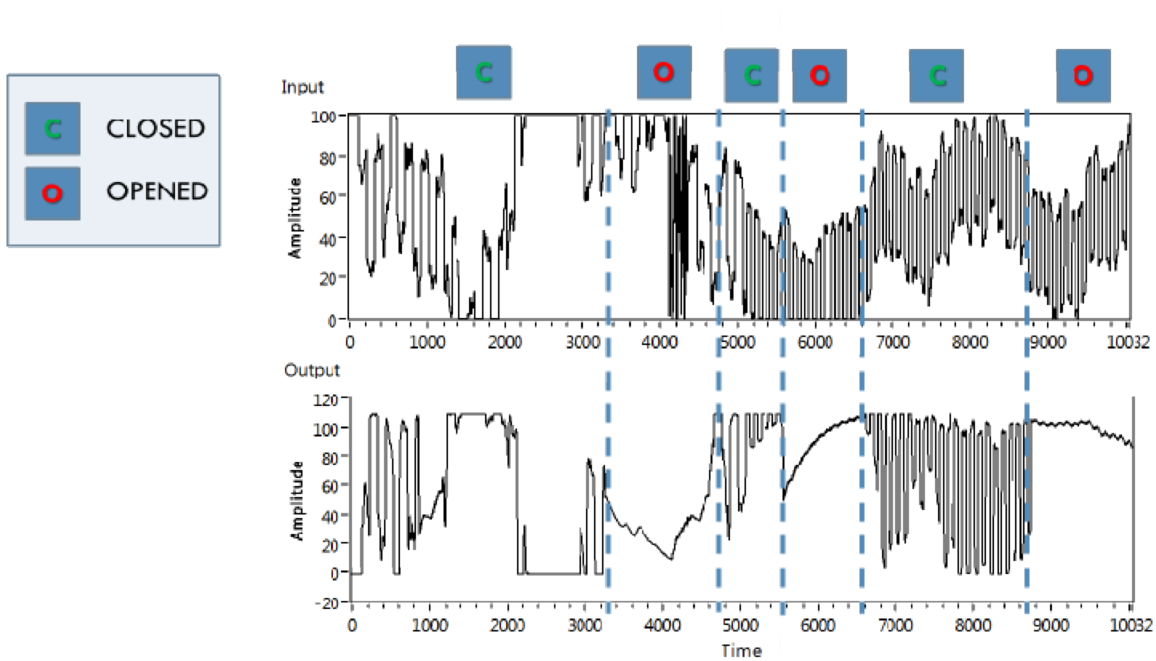

Fig. 4. Input and output data collected during the experimental run. Vertical dashed lines denote instants when the working condition has been changed. Letter above the figure denote the state of the air tank: **O** – opened, **C** – closed. The time axis shows sample indices (multiples of 100ms) from the beginning of the experiment. Vertical lines denote approximate boundaries between working conditions.

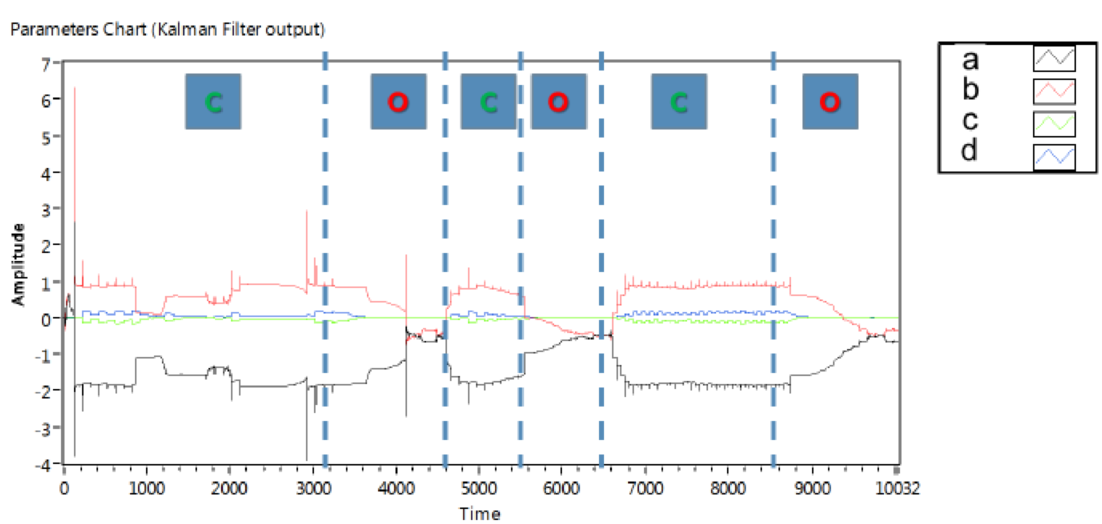

Fig. 5. Estimated values of the parameters of the second-order process model, i.e. the outputs of the Kalman filter. The time axis shows sample indices (multiples of 100ms) from the beginning of the experiment. Vertical lines denote approximate boundaries between working conditions. Upper labels denote the active working conditions: **C** denotes that the air tank is closed, **O** denotes that the air tank is opened.

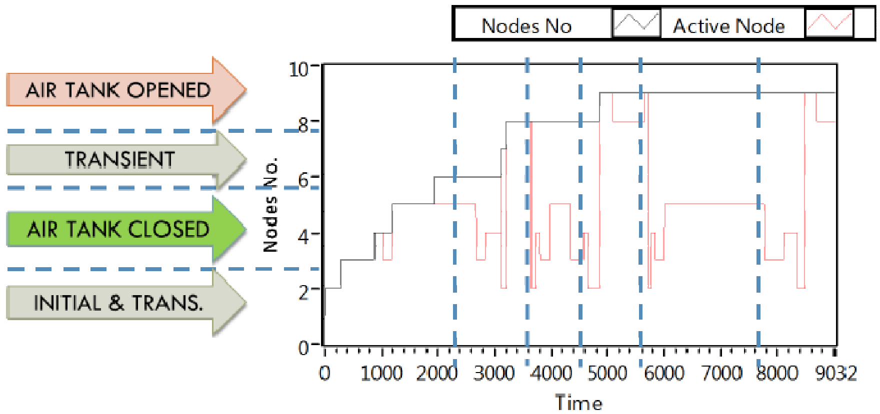

Fig. 6. The output of the proposed adaptive classification algorithm. Lower line denotes the index of the active focal point, while the upper line denotes the total number of focal points. Arrows on the left denote the focus indices corresponding to each particular working condition.The time axis shows sample indices (multiples of 100 ms).

The initial state is the one with all parameter values equal to zero. These are the initial parameters provided by the Kalman filter and it is taken as the first focus since the knowledge base is initially empty. Foci indexed 3, 4 and 5 may be associated with the working condition when the air tank is closed (disconnected from the rest of the system). Foci indexed with 8 and 9 may be associated with the working condition which is active when the air tank is opened. Multiple focal point are associated with single working condition due to the nonlinearity of the process. The association of working conditions and focal points cannot be made automatically; it should be made by an experienced operator the first time any of the focal points is detected. In this particular case, it is made after the experimental run, by observing the correlation between the actual working condition and the index of the active focus. However, once the association is created, it can be used for automatic fault detection, since during further occurrences of the same fault (or in general, of the same working condition) the regime description will remain available.

I. Conclusion

An adaptive classification procedure with emphasis to applications in fault detection and isolation has been presented in the current paper. The key underlying concept is the notion of the information potential (3) defined by means of exponentially weighted average distance of the current feature point with respect to the set of all previously seen features. The information potential defined in such way allows for recursive computations and thus, real-time, on-line implementation.

The fault detection algorithm based on the presented adaptive clustering procedure is capable of detecting novelties and to evolve in accordance to changes in the plant under consideration. However, the detection speed, especially when the measurement noise is high, may be low. For the example presented in this paper the detection delay is 30 seconds in average. For some processes and in certain applications this is acceptable. However, in a number of industrial settings such a delay is too high. In such cases, one can use standard observer based detection schemes [11] and sacrifice novelty detection for speed. A hybrid approaches, which would combine the proposed algorithm with observer based strategies, would be particularly interesting and is a subject of further line of research.

References

[1] A. J. Izenman, “Modern Multivariate Statistical Techniques: Regression, Classification and Manifold Learning”, Springer Science + Business Media, 2008

[2] V. Kecman, “Learning and soft-computing: support vector machines, neural networks and fuzzy logic models”, MIT Press, 2001.

[3] V. Venkatasubramanian, et al, „A review of process fault detecton and diagnosys, Part I: Quantitative model-based methhods“, *Computers and Chemical Engineering*, 27:293-311, 2003

[4] V. Venkatasubramanian, et al, „A review of process fault detecton and diagnosys, Part II: Qualitative models and search strategies“, *Computers and Chemical Engineering*, 27:313-326, 2003

[5] V. Venkatasubramanian, et al, „A review of process fault detecton and diagnosys, Part III: Process history based methods“, *Computers and Chemical Engineering*, 27:327-346, 2003

[6] P. P. Angelov, D. P. Filev, „An Approach to Online Identification of Takagi-Sugeno Fuzzy Models“, *IEEE Transaction on Systems, Man, and Cybernetics – Part B: Cybernetics,* 34(1), 2004

[7] P.P. Angelov, D.P. Filev, N. Kasabov (eds.), „Evolving Inteligent Systems – Theory and Applications“, John Willey and Sons, Inc. 2010

[8] FTN, Novi Sad and colaborators, „Design and implementation of real time software for patern recognition and fault detection“, *Progress Report for Task 4.2, PRODI project*.

[9] M. R. Rapaić, M. Petković, Z.D.Jeličić, „Prilagodljivi postupak za grupisanje podataka sa primena u otkrivanju grešaka u radu industrijskih procesa“, Proceedings of LV ETRAN Conference, Teslić, Bosnia and Herzegovina, 2010

[10] S.X. Ding, „Model-based fault diagnosis techniques: design schemes, algorithms, and tools“, Springer Verlag, Berlin Heidelberg, 2008

[11] M.R.Rapaić, Z.D.Jeličić, A.Pisano, E.Usai, „Second-order sliding modes and soft computing techniques for fault detection“, Proc. of the 8th European Workshop on Advanced Control and Diagnosis ACD 2010, Ferrara, Italy, pp. 271-277, November 2010.

[12] K.J.Astrom, B.Wittenmark, „Computer Controlled Systems – Theory and Design“, Prentice Hall, 1997.