

Linguistic OWA and two time-windows based Fault Identification in wide plants

A. Sánchez Fernández^a, G.I. Sainz-Palmero^a, J.M. Benítez^b, M.J. Fuente^{a,*}

^a*Department of System Engineering and Automatic Control, EII, University of Valladolid, Valladolid, Spain*

^b*Department of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain*

Abstract

Fault detection and diagnosis in industrial processes are challenging tasks that demand effective and timely decision making procedures. The multivariate statistical approaches for fault detection based on data have been very useful. However, they are known to be less powerful for fault diagnosis because they normally require prior knowledge of the problem involved. In this context, this proposal is based on an on-line, distributed fault isolation approach to provide a scored rank of variables considered as responsible for the faults in a more robust and earlier way than usual approaches. The fault isolation is carried out considering some top Fault Isolation (FI) methods, without prior knowledge regarding faults, in a distributed and collaborative way by a linguistic based decision making. The isolation of faulty variables provided by each FI approach is aggregated to provide a fault identification based on a scored ranking at two time points: after the fault detection and when the plant has recovered. In both cases, the final fault isolation is provided as a scored ranking obtained by Ordered Weighted Average operators (OWA) and Regular Increasing Monotone (RIM) aggregation functions, which permit the implementation of linguistic aggregation functions. The risk aversion during this multicriteria isolation is tuned by the user and can provide several strategies or policies. The fault isolation at two key times searches for the origin of faults and evaluates the evolution of the system after the fault's occurrence in the new working position of the plant. This is because faults in an industrial plant are propagated to different variables due to the actions of the process controllers. This method has been applied to two complex benchmark plants obtaining an earlier and more robust isolation.

*Corresponding author, M.J. Fuente. Department of Systems Engineering and Automatic Control, EII, C/ Paseo del Cauce N.59, Post code: 47011, University of Valladolid, Valladolid, Spain, Tlf: +34 983 42 39 84

Email addresses: alvar.sanchez@uva.es (A. Sánchez Fernández), gresai@eii.uva.es (G.I. Sainz-Palmero), j.m.benitez@decsai.ugr.es (J.M. Benítez), mjfuente@eii.uva.es (M.J. Fuente)

Keywords: Fault identification, Multicriteria decision making, OWA operator, contribution plots

1. Introduction

Fault detection and diagnosis (FDD) has been an active research area over the last few decades. The purpose of fault detection is to determine the occurrence of an abnormal event in a process, i.e., a fault, and the diagnosis task aims to identify the root cause or the source of the faults. Investigating the root cause of abnormal events in industrial processes is crucial for maintaining efficiency and optimal operation. As industrial processes become large scale and complex systems due to efforts to reduce energy costs, environmental releases, increasing profit, etc., the potential for faults to dynamically propagate and produce significant damage in equipment, life or environment has increased. This has encouraged the development of methods to detect faults quickly, and also to isolate the variables associated with faults, preferably before their effects become a major safety issue.

Several techniques have been developed to detect and diagnose faults, which can be categorized into model-based and data driven methods [1–3]. The former require a strong background to the plant and a long process of developing and testing the model. The latter only requires historical data from the plant, and the model developing process is easier and faster. So, recently, methods based on signals, such as Principal Component Analysis (PCA) or Partial Least Squares (PLS), have been widely used [4–9]. These methods work with data supplied by all the sensors in the plant and return only a few statistics that are very easy to work with. This is because it is only necessary to check whether the current value of one of these statistics is over a limit value to know if the system is working under fault conditions.

When the method detects a fault, it is important to know where the problem is, i.e., the faulty variables need to be identified in order to diagnose the root cause of the fault. This is helpful in repairing the installation quickly and safely. Traditionally, contribution plots [10] was the most popular tool for identifying the faulty variables, where previous knowledge is not necessary, i.e., data from all possible faults occurring in the system, which might not exist for some industrial processes. However, it is a tool that, as the authors note, does not reveal the cause of the fault. It can, nevertheless, find a group of variables with abnormal behaviour, i.e., it isolates which variables are pushing statistics beyond their control limits.

Other authors proposed different solutions to the fault identification problem: Dunia and Qin (1998) [11] worked on the reconstruction of variables in order to isolate faulty variables. Yue and Qin (2001) [12] combined the statistics T^2 and Q into a new statistic, φ . Their method, based on the cited reconstruction method [11], aims to minimize φ after carrying out reconstruction over different fault directions. These directions constitute the fault subspace. One drawback of this technique is that this subspace changes when the fault is

propagated to other variables, as the fault situation evolves due to the action of controllers. Alcala and Qin (2009) [13] developed the reconstruction-based contribution (RBC), which is based on the missing variable approach. RBC has problems related to smearing between faulty and non-faulty variables. So, RBC is practical for identifying faults which involve only one sensor. Another solution, proposed by Detroja et al. (2007) [14], uses Correspondence Analysis (CA) to isolate faulty variables, in a similar way to the contribution plots of PCA.

Liu and Chen (2014) [15] introduced the Reduction of Combined Index (RCI) and Modified Contribution Plots [16]. This methodology avoids the smearing of non-faulty variables, calculating a contribution plot by maximizing the reduction of a combined index through a missing data method. However, it does not give a correct fault identification when several abnormal events happen, because it supposes that all faulty data have the same defective variables. More recently, Jiang et al. (2015) [17] used Canonical Variate Analysis (CVA) to perform fault identification; here, two types of contributions are developed based on variations in the canonical state space and the residual space. Another method was proposed by Kariwala et al. (2010) [18], which applies the Branch and Bound algorithm (B&B) to search for faulty variables by minimizing the monitoring statistics of Probabilistic Principal Component Analysis (PPCA). This method is limited by the size of the variable set. Finally, Mahadevan et al. (2009) [19] used Support Vector Machine techniques (SVM) to identify faults. The results of some of these methods applied to the same plant are different [10, 14, 15, 17–19], showing that there is no unique, effective technique to isolate faulty variables.

On the other hand, there are techniques based on the fusion of alternative results for fault detection and identification, as in Zhang et al. (2015) [20]. Here, the authors apply various methods to detect and identify faults, and take decisions that fuse the results of each one. However, it is focused on fault classification, which requires available previous data regarding the different faults, which is not realistic for industrial processes, since known event lists might not exist and there is usually no faulty data available for most industrial processes. In addition, an incorrect fault diagnosis result occurs when a new type of fault appears. Similar ideas are in Gosh et al. (2011) [21], which uses multiple and heterogeneous FI methods, such as Kalman Filters, PCA and artificial neural networks, fusing their results with several decision fusion strategies. Nevertheless, as before, this has to be understood as a classification problem, with the same prior challenges.

Additionally, there exists the challenge regarding the time when the diagnosis must be carried out: diagnosis is sometimes developed at the time when the fault detection is made [10–14, 18, 19]; while, in other cases, it is developed over a long period of time, as long as the available data samples, as in [15–17]. The first approach has the problem of the existence of disturbances, noise, etc., which distort the fault identification, while only taking into account the variables affected by the fault first. The second approach analyses both variables affected in the first moments after the fault and variables affected by the propagation

of the fault. If the plant has a control system, this control aims to correct the controlled variables, modifying others. This means that, after a certain period of time, the controlled variables return to their normal values, while the manipulated ones go to unusual values. This leads to an untrue or untrusted diagnosis.

The proposal of this work is focused on overcoming these problems: a lack of an effective method for all situations and the requirement of prior knowledge of fault origins and their influence on the plant. In order to face both challenges, two ideas are considered: the aggregation of different FI methods through a decision making to provide a final identification of the faulty variables, and giving a fault isolation at two times: when the fault is detected and when the plant works under faulty conditions.

First, the combination of different fault identification methods is based on a multicriteria decision making approach, the Ordered Weighted Average (OWA) linguistic operator, based on the Regular Increasing Monotone (RIM) function, to find the variables that are candidates to be considered responsible for the fault. This RIM OWA operator provides a scored ranking for the candidate variables involved in the fault identification.

The second idea presented in this paper is to apply the fault identification procedure in two different times: just after the fault is detected (Early fault identification), and after the occurrence of the fault, when the system has reached a new steady point (Steady fault identification). Our proposal aggregates various fault identification methods in a linguistic way in order to carry out a final decision making. This aggregation is carried out throughout a time-window, not only in just one sampling time. So the fault identification is made through two time-windows at the two times previously described. Between both time instants, a slide time-window is used in order to check if the system has reached a new steady state, when the second fault identification must be carried out. So the contributed idea is to use two time-windows for two fault isolations, i.e., a fault identification in two different instants of time, using two time-windows, each corresponding to a different state of the system. This double time-window based fault identification provides more reliable results, taking into account the state of the system at each time.

Other authors, such as [22], use slide time-windows to detect faults, computing on-line the principal components along a moving time-window, as the method is termed Moving Principal Component Analysis (MPCA). However, in [22], the time-window is used to detect faults: a reference principal component is calculated in normal operation conditions, and the differences between the reference principal components (PCs) and the PCs representing the current operation condition, i.e., the PCs calculated in the time-window, updated step by step, are used as indexes for monitoring. The fault detection index is calculated on-line for each sampling time by the slide time-window, and compared to a threshold to know if the system is in fault. In addition, the update of the PCs for each sampling time can be done using a recursive PCA algorithm (RPCA) [23, 24] instead a time-window. In this case, a process data block in normal operation conditions is used to build an initial PCA model, and this PCA model

is updated when a new data block becomes available in a recursive way. So, in these two methods, the windows are used to update the PCA algorithm in each sampling time in order to detect faults when the system is time varying. In this paper, the time-windows are used for fault identification not for fault detection, and in this paper the FI methods used are not adapted recursively based on the new data.

Otherwise, the time-windows are also used to identify faulty variables, i.e., once the fault is detected, the time-window is used to discover which variables are responsible for the fault, as in [15–17]. In the traditional contribution plot [10], the variable contribution is plotted for the observation in which the fault is detected, but in [17], a contribution plot based on the CVA statistics is carried out; not only during the detection time, the procedure is repeated to generate one contribution plot at each observation, from the observation in which the fault is detected until the final data, and a 2-D contribution map is plotted. So, in this case, a large time-window is used and the final decision, i.e., the fault identification, is taken at the end of the observation data. This is a long time to identify the faulty variables in the on-line operation in industrial plants. In addition, [15, 16] use the same idea, calculating the contribution plot based on RCI in a long time-window, from the fault detection time to the final observation data set, and at this final time a 2-D map is plotted. However, as before, this decision-making time is too long in an industrial plant.

In our approach, the idea is to take advantage of both situations and to give an initial fault identification when a fault is detected, as many authors have done [10, 13, 14, 18, 19], in order to provide fast fault identification. What is more, this first time-window permits the faulty variables to be identified at the fault detection time, revealing the candidate variables for causing the fault, or the variables that are firstly affected by the fault. This is done in order to locate the fault origin in the system. Simultaneously, the proposed method gives a new fault identification after the system has evolved to a new steady post-fault state, providing a more consistent fault identification, as the authors in [15–17] have done, but faster than them, because instead of waiting for the final observation to take the decision, this decision is taken at the time when the plant has reached the new stationary point. This second time-window shows how the system works under faulty conditions, giving information about the new working condition of the plant and giving support to decide whether the plant can continue working under the new situation or whether it needs some type of modification.

Considering all this, the main contributions of this proposal can be summarized as follows:

- A more robust identification provided by the aggregation of several top FI approaches and an OWA operator. This aggregation is tuned by the user, according to his/her own admissible risk in multicriteria decision making, providing a scored ranking for the candidate variables that may be responsible for the fault.
- A more trusted and earlier fault identification in two times: ahead of

fault detection and when the plant is again steady. This improves the fault identification performance, avoiding initial disturbance and plant transition to a new working point, all of which could blur the diagnosis.

- This fault identification in two times permits a more robust diagnosis in the short term, and a more robust and earlier diagnosis is supplied in the long term in comparison with the more usual approaches.

The remainder of this paper is organized as follows: Section 2 explains the PCA method and the different fault identification methods used in this proposal. Also, in the same section, there is the description of the linguistic operator OWA. The proposed method is explained in Section 3. Section 4 explains the workbenches used to test the proposed method, a wastewater treatment plant and the Tennessee Eastman Process (TEP) plant, as well as presenting the results of these tests. Finally, the conclusions of this work are explained in section 5.

2. Related works

In this proposal, the faults are detected using Principal Component Analysis (PCA) and fault identification is based on the results of some of the most used fault isolation data-driven methods without any prior fault knowledge. The various alternative results are aggregated by a linguistic operator in a similar way to a human technician.

2.1. Principal component analysis for Fault Detection

Let the data matrix $X \in R^{n \times m}$ with n observations and m measured variables. This matrix is normalized to zero mean and unit variance; then, its covariance matrix is decomposed:

$$S = P\Lambda P^T + \tilde{P}\tilde{\Lambda}\tilde{P}^T \quad (1)$$

The diagonal of Λ contains the most significant eigenvalues of S , sorted in decreasing order, and P are their associated eigenvectors. The residual eigenvectors and eigenvalues are included in \tilde{P} and $\tilde{\Lambda}$, respectively.

The fault detection task is developed with *Hotelling's statistic* (T^2) [10, 13, 25], which is a measure for the variation of the systematic part of the data, the *Square Prediction Error (SPE)* statistic [10, 13, 25], which calculates the variation of residual parts of the data, and a combined index of both statistics called φ [17, 26, 27].

The plant is under normal conditions, for an α significance level, if T^2 is under its threshold T_α^2 :

$$T^2 = x^T D x < T_\alpha^2 \quad (2)$$

where $D = P\Lambda^{-1}P^T$.

For the *SPE* statistic (also called Q) and a new observation x , the system is under control if the value of Q is lower than Q_α :

$$Q = x^T \tilde{C} x < Q_\alpha \quad (3)$$

where $\tilde{C} = \tilde{P} \tilde{P}^T$. Jackson (1991) [28] gives the way in which the thresholds T_α^2 and Q_α can be calculated.

Finally, the combined statistic [12, 17, 27], φ , combines T^2 and Q , and is under control if $\varphi < \varphi_\alpha$:

$$\varphi = \frac{Q}{Q_\alpha} + \frac{T^2}{T_\alpha^2} = x^t \Phi x < \varphi_\alpha \quad (4)$$

where $\Phi = \frac{\tilde{C}}{Q_\alpha} + \frac{D}{T_\alpha^2}$ and φ_α can be found in [13].

The thresholds of these statistics, (T_α^2 , Q_α and φ_α) are calculated theoretically as described in [28] and [13], respectively, for a significance level (ISL or α) of 1%, i.e., only a certain percentage of anomalous observations are over the tuned limit in the faultless training data. Then, these values are tested in the faultless test data, and retuned experimentally if necessary in order to assure this false alarm rate. However, since false alarms are unavoidable, an out-of-control value of a statistic can be the result of a fault or a false alarm. In order to drastically decrease the rate of false alarms, it is necessary to establish a number of consecutive anomalous observations to indicate a fault.

2.2. Fault identification

After a fault alarm, it is necessary to identify the variable or variables responsible for the fault. There are several methodologies to do this, but only the methods used in this paper are explained here.

2.2.1. Overall average variable contribution: $Cont^{T^2}$

In the observation x , the contribution of the variables to T^2 for each of the higher normalized scores are calculated [10, 16, 29]. The contribution of variable x_j in the a normalized score is:

$$cont_{a,j} = \frac{t_a}{(\sigma_a)^2} (x_j - \mu_j) \quad (5)$$

if this value is negative, $cont_{a,j}$ is set to zero. The total contribution of each variable j is:

$$CONT_j = \sum_{a=1}^n cont_{a,j} \quad (6)$$

The variables with the highest contribution value are selected as candidates for the fault.

2.2.2. *Normalized errors of the variables: Cont^Q*

If a new observation x falls outside the limit of Q , the normalized error of each variable x_j is [10, 16, 30, 31]:

$$cont_{x_j} = (x_j - \hat{x}_j)^2 = e_j^2 \quad (7)$$

where \hat{x}_j is the value of the variable predicted by the model, and e_j is the residual.

2.2.3. *φ contribution: Cont ^{φ}*

For the observation x , the contribution of each variable i to the index φ is [12, 13]:

$$c_i^\varphi = \left(\xi_i^T \Phi^{\frac{1}{2}} x \right)^2 \quad (8)$$

where ξ_i is the i -th column of the identity matrix.

2.2.4. *Reconstruction based index: RBC^{T²}, RBC^Q and RBC ^{φ}*

This method, used by various authors [32–34], minimizes the influence of each variable on the detection index (T^2 , SPE or φ) [11]. Based on the work of Alcalá and Qin (2009) [13], the amount of reconstruction index of a variable is equivalent to the contribution of this variable to the out-of-control index. The reconstruction based index (RBC) for a variable i is obtained as:

$$RBC_i^{index} = x^T M \xi_i (\xi_i^T M \xi_i)^{-1} \xi_i^T M x \quad (9)$$

where x is the current measurement and M is the matrix D for the T^2 index, the matrix \tilde{C} for the Q index and the matrix Φ for the φ index.

2.2.5. *Modified Contribution Plots: RCI*

This method first finds the variable that maximizes the reduction of the combined index (RCI) after the reconstruction of this variable [15, 16]. Then, this variable is put in the group of faulty variables. Following that, the index φ is recalculated using the reconstruction along the selected variable in the previous steps and, if the reconstructed combined index is under its control limit, the process stops. However, if the index is still over its limit, the process is repeated with the variables in the non-faulty variables group. Finally, the variables inside the faulty variables group are analysed [35].

The reduction of the combined index after reconstructing the faulty data is:

$$\varphi - \varphi_{nf}^* = (\mathbf{x}_{nf} - \mathbf{x}_{nf}^*)^T (\xi^T \Phi \xi) (\mathbf{x}_{nf} - \mathbf{x}_{nf}^*) \quad (10)$$

where \mathbf{x}_{nf} is the set of faulty variables, and \mathbf{x}_{nf}^* the same set of variables after reconstruction. Also, the reconstruction of \mathbf{x}_{nf} is:

$$\mathbf{x}_{nf}^* = -(\xi^T \Phi \xi)^{-1} \xi^T \Phi (1 - \Gamma) \mathbf{x} \quad (11)$$

where Γ is a diagonal matrix with ones in the faulty variables indexes and zeros in the remaining positions. x is the data vector with all variables. The contribution of the variable i to the RCI is:

$$c_i^{RCI} = [(\mathbf{x}_{nf} - \mathbf{x}_{nf}^*)^T (\xi^T \Phi \xi)^{0.5} \xi_i]^2 \quad (12)$$

2.3. Multicriteria Decision Making based on OWA operators

Multicriteria decision making (MCDM) is a well-known approach in different domains, such as financial investments [36, 37], maintenance [38], energy management [39], fault detection and identification [20], etc., to give some global solution when there are several options evaluated by many criteria [40, 41]. One popular approach to MCDM is the aggregation functions with RIM quantifiers. One of these are the Ordered Weighted Averaging (OWA) operators introduced by Ronald R. Yager (1988) [42]. Liu and Han (2008) [43] gave a summary of this family of operators.

RIM quantifier based MCDM aims to obtain final and global decisions (Eq. 13) regarding the scores, $\langle \alpha_1, \alpha_2, \alpha_3, \dots, \alpha_n \rangle$, obtained by every option in each criterion, ordered from the highest to the lowest, resulting in $\langle \beta_1, \beta_2, \beta_3, \dots, \beta_n \rangle$, where β_1 is the largest and β_n is the smallest.

$$F(\alpha_1, \alpha_2, \alpha_3, \dots, \alpha_n) = \sum_{j=1}^n \omega_j \beta_j \quad (13)$$

where ω_j is the weight for the j th β . OWA operators are an option to implement RIM quantifiers, $Q(r)$, (Eq.13) in aggregation functions [44].

$$\begin{aligned} Q_\alpha(r) &= r^\alpha \\ \omega_j &= Q\left(\frac{j}{n}\right) - Q\left(\frac{j-1}{n}\right) \\ \omega_j &= \left(\frac{j}{n}\right)^\alpha - \left(\frac{j-1}{n}\right)^\alpha \end{aligned} \quad (14)$$

The OWA can be characterized by their *orness* and/or their *andness* [45–47], taking into account eq. (14):

$$\begin{aligned} orness(\omega) &= \frac{1}{n-1} \sum_{j=1}^n (n-j)\omega_j \\ andness(\omega) &= 1 - orness(\omega) \end{aligned} \quad (15)$$

$$Q_\alpha(r) = r^\alpha \Rightarrow orness = \frac{1}{1+\alpha}$$

Orness is included within the range $[0, \dots, 1]$. Higher *Orness* values mean taking into account options with good scores in only a few criteria (Low risk aversion), while lower values are more conservative, because they only consider those options with good scores in the majority of the criteria (High risk aversion).

3. OWA-RIM aggregation based Fault Identification (ORAFI)

This proposal develops a fault identification approach based on the aggregation of seven top FI methods, which do not use prior knowledge of the faults: Overall average variable contribution [10], Normalized errors of the variables [10], φ contribution [13], Reconstruction based index (for T^2 , Q and φ) [13] and Modified Contribution Plot [16], through a linguistic operator (OWA-RIM) as a multicriteria decision-making method tuned by the user according to his/her own tolerable risk. This fault identification is performed with the information collected during two time-windows, supplied at the end of each time-window: the first is applied immediately after the fault detection, while the second is applied after the system has evolved to a new steady condition.

A general overall view of the proposal is shown in Algorithm 1. The fault detection is based on a standard PCA using the statistics T^2 , Q and φ , the first of these exceeding its threshold during a consecutive number of samples establishes the fault detection time. Then, during a short time-window after the fault is detected, all the fault identification methods considered for this proposal (see Section 2) provide their own results. These outcomes are aggregated using a linguistic approach based on a RIM-OWA function, resulting in a global scored ranking of candidate variables responsible for the faults. This is the *Early fault identification*. After that, the system is checked until a new steady working condition after the fault is reached. Then a new fault isolation is carried out in the new stationary point, based on the same techniques and RIM-OWA decision-making, considering the data collected along this second time period; this is the *Steady fault identification*.

A key issue is to tune the different critical parameters of the method, such as: the risk considered during fault identification, the length of the time-windows and the detection of a new stationary point. The criteria used to adjust these issues are as follows:

- Risk assumed during fault identification: an OWA-RIM operator is used to aggregate the candidates of the FI methods. The RIM aggregation function: $Q_\alpha(r) = r^\alpha$ allows the weights for the OWA operator to be calculated. The α is adjusted for the user to establish the desired risk aversion set-point. Higher values of α give lower *orness* values (see Eq. 15), which means high risk aversion or less risky decisions.
- Time-window lengths used to collect data for fault identification: It is necessary to adjust the length of the first (after the fault) time-window: Δt_{Early} and the second (after the plant stabilization) time-window: Δt_{Steady} .
 - Δt_{Early} : this time is tuned looking for the smallest value that provides the best *Separability* between the scores obtained by the variables in the ranking as a result of the aggregation. The objective is to delay the fault identification as little as possible, thus using only a few samples for the identification, permitting an easier discrimination between the candidate variables (see Section 4).

Algorithm 1 OWA-RIM aggregation based Fault Identification

```

1: Tuning method:
2: for Faultless data do
3:   Set a Risk Aversion value for MCDM.
4:   FI methods:  $\{Cont^{T^2}, Cont^Q, Cont^\varphi, RBC^{T^2}, RBC^Q, RBC^\varphi$  and
    $RCI\}$ 
5:   Adjust  $\Delta t_{Early}$  for  $min(\text{delay})$  with  $max(\text{separability})$ 
6:   Adjust  $\Delta t_{Steady}$ ,  $Th_{Steady}$  and  $Var_{Steady}$  for  $min(\text{delay})$  with
    $max(Detect_{Stabilization})$ 
7: end for
8:
9: Data under analysis
10: for  $t=1$  to  $n$  do ▷ For each instant
11:   Early fault identification:
12:   if Fault Detection = TRUE then
13:     while  $t < (t_{FaultDetection} + \Delta t_{Early})$  do ▷ Early time window
14:        $D_{FI_j} \leftarrow DiagnosisBasedOn\{FI_j\}$  ▷ Isolation from each FI
   method
15:        $F_t^{Variable_k}(D_{FI_1}, \dots, D_{FI_n}) = \sum_{j=1}^n \omega_j \beta_j$  ▷ Scores ( $\omega_j$ : OWA
   weights)
16:       ( $\beta_j = \text{Ordered value of } Variable_k \text{ in } \langle D_{FI_1}, D_{FI_2}, \dots, D_{FI_n} \rangle$ )
17:     end while
18:     if  $t = (t_{FaultDetection} + \Delta t_{Early} - 1)$  then ▷ End of Early window
19:        $F^{Variable_k} \leftarrow average(F_t^{Variable_k}), t \in \{Early\ time\ window\}$ 
20:        $Diagnosis(Early) \leftarrow Ranking(F^{Variable_1}, \dots, F^{Variable_m})$ 
21:     end if
22:   end if
23:
24:
25:   Steady fault identification:
26:   Check SystemStabilization
27:    $D_{FI_j} = DiagnosisBasedOn\{FI_j\}$  ▷ isolation from each FI method
28:    $F_t^{Variable_k}(D_{FI_1}, D_{FI_2}, \dots, D_{FI_n}) = \sum_{j=1}^n \omega_j \beta_j$  ▷ Scores in  $t$  ( $\omega_j$ : OWA weights)
29:   ( $\beta_j = \text{Ordered value of } Variable_k \text{ in } \langle D_{FI_1}, D_{FI_2}, \dots, D_{FI_n} \rangle$ )
30:   if  $F_t^{Variable_k}$  not change in last  $\Delta t_{Steady}$  then
31:     SystemStabilization = TRUE
32:      $F^{Variable_k} = average(F_{t_s}^{Variable_k})$  with  $t_s \in \text{last } \Delta t_{Steady}$ 
33:      $Diagnosis(Steady) = Ranking(F^{Variable_1}, \dots, F^{Variable_m})$ 
34:   end if
35: end for
36:
37: Stabilization Criteria:
38: First positions of Ranking  $F_t^{Variable_k}$  don't change in  $\Delta t_{Steady}$ 
39: Scores of first positions vary less than  $Var_{Steady}$  in  $\Delta t_{Steady}$ 

```

- Δt_{Steady} : the process to adjust Δt_{Steady} consists of attempting to find the value from which the identified variables do not change, using data from different faulty situations. This value is adjusted looking for the shortest delay for the fault identification, while at the same time maximizing the number of faulty cases in which the system once more reaches a steady set point, at t_{Steady} (t_{Steady} is different for each fault).
- Stabilization time, t_{Steady} : this is obtained by applying the RIM-OWA diagnosis proposed for Δt_{Steady} . This provides a ranking of candidate variables for each instant. The plant is considered stabilized when:
 - the top ranking positions for the fault identification, with scores overpassing a threshold: Th_{Steady} , do not change during a time-window of length Δt_{Steady} .
 - these variable scores remain below some thresholds (selected by the user), such as variance (Var_{Steady}), along the same time-window.

The average score of each variable in this time-window is obtained and sorted to obtain a ranking which is used for the *Steady fault identification*.

Regarding the usual fault isolation methodologies, this proposal introduces some advantages concerning fault isolation time and robustness: regarding the approaches, such as [10, 13, 18], that carry out the fault identification when the fault is detected, ORAFI provides a more robust fault isolation due to the use of many aggregated top FI approaches. ORAFI is also open to incorporating any other FI method. So the time-window helps to avoid initial disturbances at the plant, and the proposed methodology gives information about the evolution of the plant after the fault, not only about its origin.

Regarding other approaches that consider 'long time periods' to make the fault identification, such as [15–17], ORAFI supplies information about the origin of the fault, not only about the faulty stabilized working condition. On the other hand, in the steady fault identification, a more robust identification is given earlier than in the other approaches, i.e., a shorter time-window collecting data is enough, so both issues are key for any FDI system.

On the other hand, the fault identification is supplied as a scored ranking of candidate variables to be responsible of the faults: this can be tuned according to the user's own risk criteria, permitting different fault identification policies to be defined.

This approach can be developed and implemented in a decentralized or distributed, and parallel way, using many agents each apply one of the different fault identification methods. After that, a central processor joins the results using the MCDM method to give a global fault identification.

4. Illustrative examples: Simulation case studies

This section presents the results of applying the proposed methodology to two benchmarks: a Waste Water Treatment Plant (WWTP) and the Tennessee

Eastman Process. The first was used to test the proposal with a plant where the origin of the fault was known, so it would be possible to check its effectiveness. The second was a well-known simulated process used by many authors [15, 17, 48–50] to test their approaches. This was used to compare the performance of this proposal regarding other solutions.

4.1. Case Study 1: Waste Water Treatment Plant

The approach presented in this paper has been tested in a simulated WWTP (Figure 1). This model is the BSM2 (Benchmark Simulation Model No. 2) developed by the Working Groups of COST Action 682 and 624 and the IWA Task Group [51, 52].

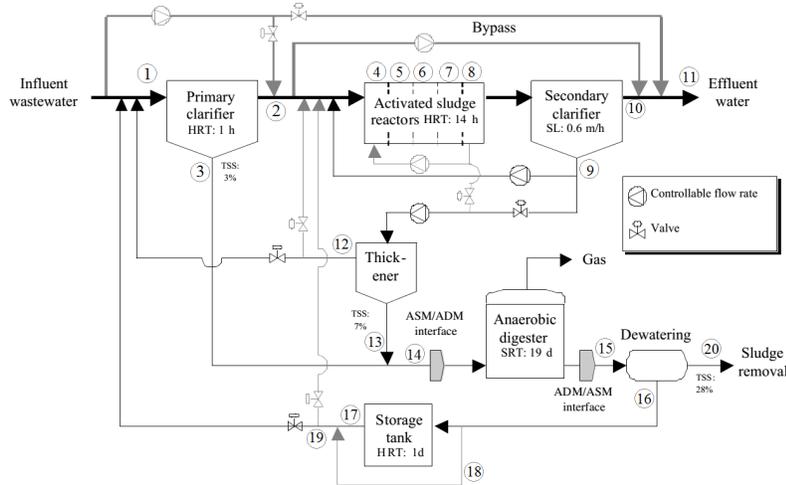


Figure 1: BSM2 plant

WWTP is an installation designed to remove biological and chemical products from water, making it possible to use the treated water for other purposes. The objective is to reduce toxicity and increase the water quality.

The model of the WWTP is available for Simulink (Matlab), and some modifications were made to produce anomalies or faults in this installation. The built-in control system was not modified and was running throughout the simulation.

The mathematical model has 16 state variables, such as inert soluble material, slowly biodegradable substrate, nitrate and nitrite, etc. [52], in each measurement point (there are 20 measurement points, see Fig. 1, a total of 320 measurements are available). However, it is not possible to measure this set of variables instantly in a real WWTP, so in this paper, a much more realistic set of 7 variables are taken into account to monitor the process. They are obtained

by combining some of the model variables (see Table 1). The model simulation runs for 609 days, giving measured values every 15 minutes.

Table 1: Variables in BSM2 plant

Used variables	Variables in the model
COD (Chemical Oxygen Demand)	S_i, S_s, X_i, X_s
O_2	S_O
Alkalinity	S_{alk}
Nitrogen	S_{NO}, S_{NH}, S_{ND}
Solids suspended	TSS (Total suspended solids)
Flow	Flow rate
Temperature	Temperature

Three kinds of fault have been considered: O_2 sensor fault, alkalinity variation and problems with flows (leaks and pipe jams), with different fault sizes. So a set of 13 tests have been carried out (see Table 2). Table 2 shows the variables that are responsible for each fault.

Table 2: Faults in BSM2 plant

Fault #	Description	Responsible variables
1	O_2 sensor failure, + 70 %	42 125
2	O_2 sensor failure, + 50 %	42 125
3	O_2 sensor failure, + 20 %	42 125
4	O_2 sensor failure, - 20 %	42 125
5	O_2 sensor failure, - 50 %	42 125
6	Change in influent alkalinity, - 50 %	8 15
7	Change in influent alkalinity, + 20 %	8 15
8	Change in influent alkalinity, + 40 %	8 15
9	Change in alkalinity in reactor 1, + 10 %	22
10	Change in alkalinity in reactor 1, + 20 %	22
11	Change in alkalinity in reactor 1, - 30 %	22
12	Change in flows Q_r and Q_w , Q_r - 50 %	80 86
13	Change in flows Q_r and Q_w , Q_r - 25 %	80 86

4.1.1. Experimental methodology

The parameters and features used for this experimental work are detailed in Algorithm 2.

First of all, a PCA model was created using the faultless data with 70% of variance captured by the selected principal components. This value was

Algorithm 2 *Experimental methodology*

```
1: Faulty train data preprocessing: Z-score normalization
2: PCA model: 70% of variance captured, thresholds adjusted to percentile
   99.5.
3:
4: Tuning  $\Delta t_{Early}$  (After fault detection):
5: for each fault do
6:   for RiskAversion=HIGH to LOW do
7:     for  $\Delta t_{Early} = 2$  to 10 do
8:       Calculate:  $Separability = \text{Variance}(\text{Scores}(\Delta t_{Early}))$     ▷ Scores
       separation
9:     end for
10:    return ( $\Delta t_{Early}$ ) which  $\min(\Delta t_{Early})$  and  $\max(Separability)$ 
11:  end for
12: end for
13:
14: Tuning  $\Delta t_{Steady}$  (After fault stabilization):
15: for each fault do
16:   for RiskAversion=HIGH to LOW do
17:     for  $\Delta t_{Steady} = 3$  to 20 do
18:       for  $Var_{Steady} = 0.00001$  to 0.005 do
19:         for  $Th_{Steady} = 0.3$  to 0.7 do
20:           Calculate:  $Delay_{Stabilization}$  and  $Detect_{Stabilization}$ 
21:         end for
22:       end for
23:     end for
24:     return ( $\Delta t_{Steady}$ ) and ( $Var_{Steady}$ ) which  $\min(Delay_{Stabilization})$ 
       and  $\max(Detect_{Stabilization})$ .
25:   end for
26: end for
27:
28: Fault detection and identification:
29: Faulty test data preprocessing: Z-score normalization
30: if FaultDetection = TRUE then
31:   Develop Early fault identification in  $\Delta t_{Early}$ 
32:   for each instant do
33:     Check stabilization
34:     if SystemStabilization = TRUE then
35:       Develop Steady fault identification in  $\Delta t_{Steady}$ 
36:     end if
37:   end for
38: end if
```

selected after cross-validation tests looking for the best trade-off between the number of principal components and the variance retained by these principal components. Lower values lost too much information, while higher values did not significantly increase the variance retained. The thresholds for the T^2 and Q statistics were adjusted so that they only had 0.5% of observations over them in normal conditions, avoiding false alarms before the fault happened.

The OWA-RIM based aggregation process: the weights were calculated for High risk aversion (orness= 0.2), Moderate risk aversion (orness = 0.5) and Low risk aversion (orness= 0.9). In this way, three alternative fault identification policies could be checked. The (*Early* fault identification) time-window was set to length 2 after doing many tests with different values (see Algorithm 2). The objective was the minimum window length, maximizing the *Separability* of the scores in the OWA-RIM ranking. This allowed for an easier discrimination between candidate variables. In Table 3, the results of the tests for window lengths of 2, 3 and 5 samples are shown.

Table 3: Tuning of window length for *Early* fault identification

Risk aversion in fault identification	High (orness=0.2)			Moderate (orness=0.5)			Low (orness=0.9)		
Window length (samples)	2	3	5	2	3	5	2	3	5
Separability	0.0012	0.0011	0.00079	0.009	0.008	0.006	0.055	0.049	0.038

The parameters for *Steady* fault identification must be adjusted to detect the stabilization as early as possible, but always when the system has reached a truly steady situation:

- Δt_{Steady} was adjusted to length 10. Higher values delayed the stabilization detection and smaller values detected the untrusted stabilization.
- The score over $Th_{Steady} = 0.5$ for ranking variables has to be maintained over 10 consecutive observations. Many other values were tested (see Algorithm 2), but their influence was not appreciable.
- The variation of the scores (Var_{Steady}) of these variables should be under 0.001. Lower values delayed the stabilization detection too much, and higher values gave false stabilization times.

Tables 4, 5 and 6 show some tests developed to tune these parameters: for different window lengths, the delay (in samples) obtained to detect the stabilization of the plant, and the number of faulty datasets where the algorithm correctly detects the stabilization of the plant (stabilization detections). For example, in Table 4, with orness = 0.2 and a window length of 5 samples, the stabilization time is 9554 samples, and the algorithm is able to correctly detect the stabilization of the plant in 10 out of the 13 faults.

4.1.2. Results and discussion

The following tables show the results of the fault identification; in order to make these results more readable, only the first six candidate variables and their

Table 4: Tuning of window length for *Steady* fault identification, Variance=0.0005

Risk aversion in Fault identification	High (orness=0.2)			Moderate (orness=0.5)			Low (orness=0.9)		
Window length (samples)	5	10	15	5	10	15	5	10	15
Average delay (samples)	9554	10501	12542	3125	3406	4998	3457	12549	15988
Stabilization detections	10	12	11	12	12	10	12	13	13

Table 5: Tuning of window length for *Steady* fault identification, Variance=0.001

Risk aversion in Fault identification	High (orness=0.2)			Moderate (orness=0.5)			Low (orness=0.9)		
Window length (samples)	5	10	15	5	10	15	5	10	15
Average delay (samples)	1580	3501	6189	510	2406	4965	2385	11744	15885
Stabilization detections	10	13	13	12	13	13	12	13	13

scores are shown. This introduces the most relevant variables.

4.1.2.1. *Early fault identification.*

The *Early* identification, the first fault isolation according to this methodology, were provided shortly after fault detection. Comparing this fault identification time to a standard PCA method (based on 70% of variance in its principal components), using T^2 and Q statistics to detect the faults and the contribution to Q to identify the fault at detection time, our proposal gave the fault isolation one sample later than PCA, due to the use of a time-window of length 2. In this way, the fault identification provided is more robust, avoiding the initial perturbations which are a consequence of the fault, as can be seen in Tables 7, 8 and 9.

In this fault identification, the difference between the risk aversion values is clear: lower values (more strict criteria) imply a higher capability of discrimination between candidate variables, as only candidates with high scores in the top positions in most contribution methods considered appear in the global ranking in the top position with high scores. So there is less risk of taking irrelevant candidate variables into account for the fault identification. Higher risk aversion values (more permissive criteria) give more similar final scores to the candidates, so this allows variables identified by only a few methods (because the rest are unable to detect them) to be taken into account. This can be another fault identification option or policy.

For the first 5 faults, the variables where the fault was inserted are 42 and 125, so these are the variables that must be in the first positions of the ranking. This is clear with high risk aversion (orness = 0.2 for fault identification), but as the risk aversion increases, more variables are taken into account, though they do not have as much influence as variables 42 and 125. For example, fault 5 with high risk aversion (orness = 0.2) is a difficult situation, but the *Early* fault identification window is able to distinguish the right candidates from the others, whereas PCA does not identify the correct variables. The same happens

Table 6: Tuning of window length for *Steady* fault identification, Variance=0.002

Risk aversion in Fault identification	High (orness=0.2)			Moderate (orness=0.5)			Low (orness=0.9)		
Window length (samples)	5	10	15	5	10	15	5	10	15
Average delay (samples)	1352	3412	3987	487	2105	4875	2133	10144	13578
Stabilization detections	10	12	12	11	12	12	11	12	12

Table 7: Early fault identification comparison. High risk aversion (orness = 0.2)

Fault	OWA-RIM						PCA	
	Delay (samples)	Early ranking			Early score			Delay (samples)
1	2	42 125 49 83 76 56	0.1229 0.0606 0.0382 0.0242 0.0242 0.0242			1	125 42 62	
2	2	42 125 83 76 56 82	0.1353 0.0873 0.0363 0.03634 0.0363 0.0270			1	125 21 42	
3	2	125 42 56 51 76 82	0.1490 0.0838 0.0103 0.0095 0.0089 0.0086			1	125 49 21	
4	2	125 49 42 76 83 56	0.3705 0.1142 0.0714 0.0539 0.0539 0.0539			1	21 125 42 28 49 35	
5	2	42 49 125 28 41 124	0.2395 0.1307 0.1174 0.0382 0.0136 0.0124			1	21 35 28 62 69 56	
6	2	8 15 90 121 103 109	0.5315 0.5315 0.0155 0.00096 0.00096 0.00096			1	8 15	
7	2	15 8 90 21 83 56	0.4999 0.4915 0.0078 0.0042 0.0040 0.0040			1	8 15	
8	2	8 15 90 102 120 96	0.5115 0.5115 0.0122 0.0017 0.0017 0.0017			1	8 15	
9	2	22 55 75 21 7 14	0.32895 0.1676 0.1649 0.1551 0.1248 0.1248			1	22	
10	2	22 29 21 55 75 7	0.33559 0.1016 0.0594 0.0561 0.0539 0.0443			1	22	
11	2	22 29 36 21 75 43	0.81715 0.3294 0.1037 0.0206 0.0192 0.0174			1	22	
12	2	86 80 8 15 14 7	0.7665 0.4341 0.0295 0.0295 0.0282 0.0282			1	86 80	
13	2	86 80 8 15 14 7	0.6439 0.4271 0.0366 0.0366 0.0290 0.0290			1	86 80	

with medium and low risk aversion (orness values of 0.5 and 0.9), where the time-window can include these two variables in the head group of the ranking, mixed with other variables.

Otherwise, comparing with the standard PCA for these first 5 faults, in the high risk aversion mode (Table 7), our proposal revealed a 100% effectiveness in 3 out of 5 faults, and for the other two, the responsible variables were in the first three positions, clearly discriminated by their scores regarding the rest of the candidate variables. The PCA, meanwhile, was not 100% effective in any of these five faults, i.e., it did not clearly distinguish the variables which originate the fault.

The results were less clear for the moderate risk aversion mode, because our proposal did not give such a good discrimination between candidates as in the previous case; however, this methodology has a higher performance than the standard PCA. For the low risk aversion situation, however, the OWA-RIM fault identification methodology took in account too many variables to give a reliable diagnosis.

As for the remaining faults, the results of the OWA-RIM were very similar to the PCA ones, giving valuable information to discriminate between the responsible variables.

Summarizing, using an aggregation of fault identification methods for a time-window supplies, in the shown results, a more robust fault identification, detecting the variables responsible for the faults, avoiding instantaneous disturbances and discriminating these variables more clearly when the methodology is tuned to have high or moderate risk aversion behaviour. On the other hand, as the

Table 8: Early fault identification comparison. Medium risk aversion (orness = 0.5)

Fault	OWA-RIM				PCA	
	Delay (samples)	Early ranking	Early score		Delay (samples)	Ranking
1	2	42 125 83 76 56 55	0.3170 0.2638 0.2603 0.2603 0.2603 0.2525		1	125 42 62
2	2	42 125 83 76 56 62	0.3589 0.3106 0.2784 0.2784 0.2784 0.2615		1	125 21 42
3	2	125 42 76 56 44 62	0.4061 0.3387 0.2739 0.2704 0.2484 0.2454		1	125 49 21
4	2	42 125 83 56 49 62	0.5973 0.3016 0.3016 0.3016 0.2879 0.2724		1	21 125 42 28 49 35
5	2	21 42 125 76 83 56	0.43822 0.3405 0.3057 0.1978 0.1978 0.1978		1	21 35 28 62 69 56
6	2	8 15 97 103 115 121	0.5547 0.5547 0.0824 0.0824 0.0824 0.0824		1	8 15
7	2	15 8 83 76 56 7 14	0.5514 0.5481 0.1271 0.1271 0.1271 0.1179		1	8 15
8	2	8 15 102 96 120 108	0.5412 0.5412 0.0805 0.0805 0.0805 0.0804		1	8 15
9	2	22 55 75 7 14 82	0.71789 0.5036 0.4621 0.4392 0.4393 0.4269		1	22
10	2	22 29 55 7 14 75	0.7379 0.4041 0.3239 0.3021 0.3021 0.2853		1	22
11	2	22 29 36 89 83 76	0.9350 0.5960 0.2729 0.1778 0.1702 0.1702		1	22
12	2	86 80 95 110 116 104	0.8795 0.6652 0.1302 0.1301 0.1301 0.1301		1	86 80
13	2	86 80 8 15 95 104 110	0.8319 0.6585 0.1446 0.1446 0.1310 0.1304		1	86 80

Table 9: Early fault identification comparison. Low risk aversion (orness = 0.9)

Fault	OWA-RIM				PCA	
	Delay (samples)	Early ranking	Early score		Delay (samples)	Ranking
1	2	55 59 62 69 42 125	0.825 0.8204 0.8141 0.8115 0.8089 0.7819		1	125 42 62
2	2	55 59 62 125 79 42	0.8264 0.8211 0.8174 0.8149 0.8096 0.7877		1	125 21 42
3	2	62 55 69 59 83 42	0.8417 0.8408 0.8392 0.830 0.8264 0.8263		1	125 49 21
4	2	55 59 62 42 79 125	0.8223 0.8162 0.8162 0.8138 0.8049 0.7941		1	21 125 42 28 49 35
5	2	62 69 42 76 125 79	0.7812 0.7791 0.7588 0.7588 0.6158 0.6158		1	21 35 28 62 69 56
6	2	8 15 115 97 103 109	0.6443 0.6443 0.4078 0.4078 0.4078 0.4078		1	8 15
7	2	15 8 76 83 56 95	0.6388 0.6383 0.4788 0.4787 0.4787 0.4774		1	8 15
8	2	8 15 95 98 104 110	0.6358 0.6358 0.3900 0.3898 0.3897 0.3897		1	8 15
9	2	22 55 75 59 79 7	0.94692 0.9019 0.8302 0.8220 0.8121 0.8045		1	22
10	2	22 55 59 79 75 7	0.9558 0.8471 0.7870 0.7783 0.7735 0.7649		1	22
11	2	22 29 89 56 83 76	0.98784 0.7877 0.7303 0.7205 0.7204 0.7205		1	22
13	2	86 80 95 116 104 110	0.9461 0.8462 0.6884 0.6877 0.6877 0.6877		1	86 80
13	2	86 80 95 104 110 116	0.9395 0.8397 0.6792 0.6762 0.6762 0.6762		1	86 80

PCA with the Q contribution is one of the seven FI methods used by ORAFI, it is clear that when this method fails, the fault identification of the ORAFI method can compensate for it with the outcome of the remaining FI methods, making it more robust.

In order to quantify the performance of the fault identification method for the 13 faults, Table 10 gives the success rate for each fault. This rate depends on the matching of the true variables responsible for the faults with the group of candidate variables supplied by our methodology, and its capability of score based discrimination of these variables regarding the remaining candidates.

Table 11 shows that the proposed method gives a better result than the standard PCA, considering the high or moderate risk cases. The discrimination of the candidate variables for the low case is poorer and, for the first five faults, the faulty variables are not even in the first ranking positions, providing a worse fault identification.

4.1.2.2. Steady fault identification.

Table 10: Rates for each fault identification

Qualification	Score	Conditions
Excellent	100	Only responsible variables in the candidates group
Good	67	Responsible variables in the candidates group
Poor	33	Not all the responsible variables in the candidates group or no candidates group, but responsible variables in first positions
Very poor	0	None of the responsible variables in the candidates group, or no candidates group

Steady fault identification is the second part of this methodology, the time delay for this identification after the fault detection is shown in Table 12. The proposed approach gives its *Steady* fault identification when stabilization is detected. Comparing this fault identification time with the approach in [15, 16], which uses all the available data, and which is used in this paper to compare the results, the identification time is lower for our methodology. Here, there are 58464 samples available for this plant, the fault appearing at time 18800, so the fault identification would be delayed 39664 samples for the [15] approach.

In the moderate value of risk aversion case, $\text{orness} = 0.5$, it gave the lowest average and standard deviation values for the time delay. However, the lowest risk aversion value ($\text{orness} = 0.9$) returned the worst value; this is because more, different variables appeared as top candidates in the rankings during the stabilization detection of the system; so, consequently, the delays were greater. In any case, the diagnosis times were considerably smaller than the delay obtained with methods that need all the data.

The results for this fault identification, based on each selected risk aversion value, are shown in Tables 13, 14 and 15, as well as a comparison with the RCI method [16] (which is one of the seven FI methods aggregated by ORAFI) with the entire data set after fault detection.

Once the system has stabilized its operation, the faults can affect other variables apart from those responsible for the faults. In this case study, faults 1 to 5 and 9 to 11 suffered from fault expansion, while for faults 6 to 8 and 12 and 13, the fault remained concentrated in the original faulty variables.

High risk aversion based fault identification: some other variables not related with the fault appeared as candidates for the first 5 faults (the origin is in reactor 4, O_2 sensor, variables 42 and 125). Specifically, variables measuring the O_2 level downstream in the plant (21,49,56,62,69,76,83,89) or the DQO level (101,107,113), related with O_2 . This result was similar for the RCI method. So the fault identification was less clear than for *Early* fault identification: the scores were very similar for some variables, not only for the variables that originate the fault. In any case, it is possible to identify the variables that caused these faults, except the fifth one, where all the candidate variables have the same score.

Table 11: Diagnosis rates for each *Early* fault identification

Fault	OWA-RIM			PCA
	Risk aversion			
	High	Moderate	Low	
1	100	33	33	67
2	100	100	33	67
3	100	33	33	33
4	67	33	33	33
5	67	67	67	0
6	100	100	100	100
7	100	100	100	100
8	100	100	100	100
9	100	100	100	100
10	100	100	100	100
11	100	100	100	100
12	100	100	100	100
13	100	100	100	100
Average rate	94.5	80.5	74.92	76.92

The fault smeared other parts of the plant for faults 9 to 11, and the fault identification reflected this change (the discrimination between scores was very small, so a lot of variables were fault candidates); thus the isolation of the variables which really originate the fault was not possible for these faults as for the RCI method. However, they give us an idea of the propagation of the fault for the plant.

The other faults did not suffer a notable change in the diagnosis, since the fault did not expand its effect over the plant; in fact, the differences between the scores of the top candidate variables and the others had increased in the majority of them. The variable candidates for Faults 12 and 13, considering the low risk case, were clearly identified. This confirms that the fault mainly affects its origin. ORAFI based isolation coincides here with what really happened in the plant. The RCI method for these faults did not put both responsible variables (80 and 86) in the first positions, and it also identified other variables not affected by the fault expansion. This confirms the robustness of the ORAFI method.

Summarizing, fault identification using data from a 'long time interval' (all available data) does not imply a reliable fault isolation. This is because the fault, after it happens, often expands its effect over the plant. So the origin of the fault is only found if a fault identification is also performed in the first few instants, as in the *Early* fault identification.

In this example, the *Steady* identification of faults 1 to 5 and 9 to 11 did not reveal the causes of the faults, as there were many variables as candidates, and not all of them were the truly responsible ones. In faults 12 and 13, as the

Table 12: *Steady* fault identification delay (in samples). BSM2

Fault	Steady fault identification delay		
	Risk Aversion (orness)		
	High (0.2)	Moderate (0.5)	Low (0.9)
1	759	211	13630
2	113	4782	8282
3	396	6042	8281
4	15001	210	27743
5	1236	788	788
6	82	62	72
7	76	64	286
8	64	64	86
9	670	210	2476
10	278	122	288
11	1348	110	1344
12	104	100	146
13	150	130	286
Average	1559.77	991.92	4900.65
Std. Dev.	4062.73	1987.5	8109.9

fault did not expand, the diagnosis of ORAFI was sufficient to know the fault’s origin, unlike with RCI, which did not give a good diagnosis. In real situations, it is not possible to know if the next fault will expand over the plant or not, so a fast and early identification is also necessary to make the diagnosis. Thus, based on these results, in a real plant, it may be necessary to perform an *Early* fault identification to see the original root of the fault and a *Steady* identification, which either confirms the fault identification of the early one, if the fault has not expanded throughout the plant, or gives another fault identification, i.e., explains how the fault is propagated over the plant.

4.2. Case study 2: TEP plant

The proposed method was also applied to the Tennessee Eastman Process (TEP) data [53]. This is a case study widely used to test new control and monitoring strategies. In the last field monitoring, it was used to check the behaviour of fault detection and diagnosis techniques [15, 17, 48–50].

The plant processes five reactives: A, B, C, D and E, obtaining two products: G and H. A schematic diagram of the plant is represented in Figure 2

The available data for this plant consists of 22 continuous process measurements, 12 manipulated variables (but the last one, *Agitator speed*, is constant, so it was not taken into account) and 19 sampled process measurements. So, 52 variables are available. Two datasets are available: training (480 observations) and test (960 observations) datasets. Each of them consists of 22 data groups: the first one corresponds to faultless operation and the other 21 correspond to

Table 13: Steady fault identification. High risk aversion (orness = 0.2)

Fault	OWA-RIM						RCI	
	Steady ranking	Steady scores					Candidate variables	
1	42 125 62 69 56 76	0.6730	0.6730	0.6190	0.3389	0.3089	0.3089	42 125 21
2	125 42 62 69 56 76	0.9425	0.8653	0.4239	0.3859	0.3585	0.3585	125 42
3	125 42 113 107 101 49	0.7182	0.4253	0.3779	0.3363	0.3103	0.1716	125 42
4	125 28 42 21 89 56	0.2187	0.1679	0.1182	0.1043	0.0465	0.0244	42 125
5	56 76 83 62 69 49	0.7760	0.7760	0.7760	0.6982	0.6932	0.4796	21 28 56 76 83 62
6	8 15 90 36 22 43 63	0.9698	0.9698	0.0340	0.0073	0.0067	0.0054	8 15
7	8 15 90 121 109 103	0.9025	0.9025	0.02107	0.0026	0.0026	0.0026	8 15
8	8 15 90 29 102 120	0.9319	0.9319	0.02678	0.0023	0.0021	0.0021	8 15
9	50 70 63 43 36 84	0.7271	0.7259	0.7242	0.7190	0.718	0.6917	63 70 84 50 70 36
10	36 43 50 63 70 77	0.8274	0.8211	0.8003	0.7882	0.7876	0.7506	50 77 43 63 70 36
11	84 77 57 63 70 36	0.90692	0.9069	0.9069	0.8969	0.8915	0.8359	84 77 57 63 70 36
12	80 86 113 101 107 64	0.6917	0.0982	0.0417	0.0417	0.0417	0.0087	28 80
13	80 86 107 113 101 90	0.6528	0.2401	0.0347	0.0347	0.0347	0.0254	80 28

Table 14: Steady fault identification. Moderate risk aversion (orness = 0.5)

Fault	OWA-RIM						RCI	
	Steady ranking	Steady scores					Candidate variables	
1	62 69 56 76 83 21	0.4508	0.4028	0.3549	0.3549	0.3549	0.3482	42 125 21
2	62 69 56 76 83 21	0.5367	0.4583	0.4026	0.4026	0.4026	0.3022	125 42
3	109 115 97 103 121 96	0.2816	0.2816	0.2816	0.2816	0.2816	0.2812	125 42
4	56 83 76 62 69 7	0.41484	0.3950	0.3950	0.3649	0.3643	0.2739	42 125
5	21 56 76 62 69 83	0.81401	0.6709	0.6709	0.6245	0.6234	0.6047	21 28 56 76 83 62
6	8 15 115 103 109 97	0.98134	0.9813	0.1339	0.1339	0.1339	0.1339	8 15
7	8 15 102 96 108 120	0.93853	0.9385	0.1352	0.1352	0.1352	0.1352	8 15
8	8 15 97 103 109 115	0.97341	0.9734	0.1439	0.1439	0.1439	0.1439	8 15
9	36 43 50 84 77 57	0.80177	0.80004	0.7775	0.7609	0.7609	0.7609	63 70 84 50 70 36
10	36 43 22 50 29 84	0.78794	0.7706	0.736	0.7262	0.7136	0.562	50 77 43 63 70 36
11	84 77 57 36 63 70	0.9028	0.9028	0.9028	0.9005	0.8905	0.8859	84 77 57 63 70 36
12	80 86 113 101 107 95	0.8940	0.2863	0.1985	0.1985	0.1985	0.1465	28 80
13	80 107 101 113 86 95	0.9013	0.2395	0.2395	0.2395	0.2393	0.1645	80 28

different faults, including step faults, random variation, etc. The faults start at the first observation in the training dataset, and at the 160th observation in the test dataset. The data were generated by Chiang et al. (2001) [54], and it can be downloaded from <http://web.mit.edu/braatzgroup/links.html>. A brief description of the process faults can be seen in Table 16.

Some of these faults are easy to detect, while others are more difficult; in fact, some of them are usually not detected: faults 3, 9 and 15 [17].

4.2.1. Experimental methodology

The methodology for this benchmark is as for the previous case study. Process parameters and features are detailed in Algorithm 2.

A PCA model was created using the faultless data with 70% of variance captured by the selected principal components: as in the previous case study, this value was selected after cross-validation tests, getting a reduced number of

Table 15: Steady fault identification. Low risk aversion (orness=0.9)

Fault	OWA-RIM							RCI
	Steady ranking	Steady scores						Candidate variables
1	62 69 76 56 83 80	0.8552	0.8542	0.8020	0.8018	0.8016	0.6534	42 125 21
2	101 107 113 50 97 103	0.8182	0.8182	0.8182	0.8005	0.7607	0.7607	125 42
3	101 107 113 97 103 109	0.8207	0.8207	0.8207	0.8052	0.8052	0.8052	125 42
4	69 62 56 76 83 49	0.7556	0.7554	0.7288	0.728	0.7257	0.7033	42 125
5	21 56 76 62 69 83	0.8140	0.6709	0.6709	0.6245	0.6233	0.6046	21 28 56 76 83 62
6	8 15 115 97 103 109	0.9911	0.9911	0.6565	0.6565	0.6565	0.6565	8 15
7	8 15 98 104 110 116	0.9905	0.9905	0.7534	0.7534	0.7533	0.7533	8 15
8	8 15 103 109 115 121	0.9906	0.9906	0.7788	0.7788	0.7788	0.7788	8 15
9	84 77 57 63 36 70	0.9229	0.9229	0.9222	0.905	0.9029	0.9013	63 70 84 50 70 36
10	84 77 57 63 70 36	0.9721	0.9721	0.9721	0.9622	0.9573	0.9495	50 77 43 63 70 36
11	70 63 84 77 57 50	0.9642	0.964	0.9625	0.9625	0.9625	0.944	84 77 57 63 70 36
12	80 95 110 116 104 98	0.9897	0.7542	0.7511	0.7511	0.7511	0.7506	28 80
13	80 95 110 104 116 98	0.9918	0.7482	0.7443	0.7443	0.7443	0.7436	80 28

principal components and a high value of variance retained by principal components. The thresholds for the T^2 and Q statistics were adjusted to have only 0.5% of observations over them in normal conditions, avoiding the appearance of false alarms before the fault happens.

The OWA-RIM based fault identification: the weights were calculated for High risk aversion (orness= 0.2), Moderate risk aversion (orness = 0.5) and Low risk aversion (orness= 0.9). The length of the first time-window was set to 3 observations for (*Early* fault identification), getting the earliest identification with the highest value of *Separability* (discrimination) between the scores (see Table 17).

A length of 10 observations was selected for the *Steady* fault identification time-window in order to achieve the fastest identification, but avoiding false stabilization times. Also, these selected variables must have a score variance along the time-window under 0.002 (the stabilization detection time increased for lower values, but higher values gave false stabilization times). Tables 18, 19 and 20 show some of the experiments developed to select these values. As in the previous benchmark, the stabilization criteria require the variables of the ranking with a score over 0.5 to be the same during these 10 consecutive observations (see Algorithm 2 for the rest of the values tested.)

4.2.2. Results and discussion

As before, the tables with the results of the fault identification only show the first six candidate variables and their scores. This is enough to see the most relevant variables.

The performance of the method was compared with other proposals. The *Early* fault identification was compared to PCA with the contribution of Q [10], Branch and Bound (B&B) [18], Correspondence Analysis (CA) [14], and the One class Support Vector Machine (SVM) [19]. All these methods returned a fault identification at the time of fault detection. The *Steady* fault identification was

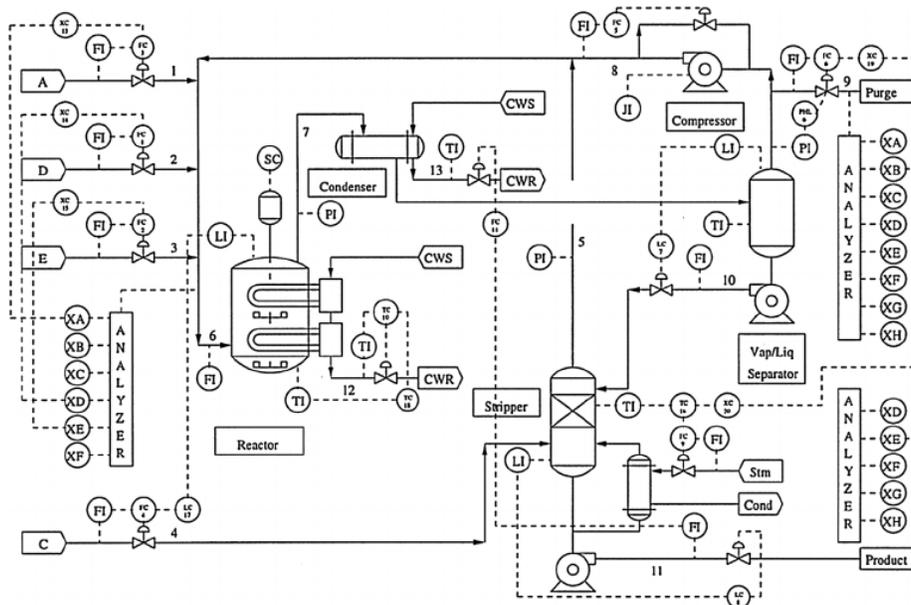


Figure 2: Tennessee Eastman Process diagram

compared with usual well-known methods using whole dataset, such as Canonical Variate Analysis [17] and Modified Contribution Plots (or Reconstruction Combined Index: RCI) [15].

4.2.2.1. Early fault identification.

The results for this *Early* fault identification are shown in Tables 21, 22 and 23, taking into account many risk aversion values for the MCDM diagnosis. The fault identification shows that other methods select candidate variables that are not always coincident between them. However, in many faults, our methodology gathers the variables revealed, in an individual way as candidates, by other proposals. Also, our proposal gives an ordered and quantified list of all these candidate variables.

Nevertheless, the fault isolation task only has to provide clues for diagnosing the root causes of a process fault. In fact, the right diagnosis depends on the results of isolating faulty variables and on the judgements made from the knowledge of the process. The definition of the right diagnosis for this plant is vague and not known, so the comparison with a right diagnosis is not possible. As mentioned above, however, an ordered and quantified ranking of faulty candidate variables is supplied in this approach. This ranking includes most of the variables for each method of the comparison considered as faulty, that is, the variables with a bigger change in the first instants of the fault's occurrence.

Table 16: TEP faults

Fault #	Description	Type
1	A/C feed ratio, B composition constant (Stream 4)	Step
2	B composition, A/C ratio constant (Stream 4)	Step
3	D feed (Stream 2)	Step
4	Reactor cooling water inlet temperature	Step
5	Condenser cooling water inlet temperature	Step
6	A feed loss (Stream 1)	Step
7	C header pressure loss-reduced availability (Stream 4)	Step
8	A, B and C compositions (Stream 4)	Random variation
9	D feed temperature (Stream 2)	Random variation
10	C feed temperature (Stream 4)	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	Unknown	-
17	Unknown	-
18	Unknown	-
19	Unknown	-
20	Unknown	-
21	Stream 4 valve	Sticking

Table 17: Tuning of time window length for *Early* fault identification

Risk aversion in MCDM	High (orness=0.2)				Moderate (orness=0.5)				Low (orness=0.9)			
Window length (samples)	2	3	5	7	2	3	5	7	2	3	5	7
Separability	0.006	0.0062	0.0049	0.0039	0.009	0.012	0.01	0.008	0.024	0.027	0.022	0.019

There were no remarkable differences between the three Risk Aversion values. With the High value, the scores were lower, but with a better separability between scores. Nevertheless, all Risk aversion values gave a well-discriminated group of candidates for most faults, allowing a clear identification.

4.2.2.2. Steady fault identification.

The time for *Steady* fault identification is shown in Table 24, compared with the CVA and RCI methods. CVA and RCI used the whole available dataset, so they took 800 observations to get a diagnosis. When the proposed methodology was not able to find the system stabilization, then the time delay was the same as the other methods: 800 samples, but in the other faults, the diagnosis time was much lower. The Moderate risk aversion value (orness=0.5) gave the lowest average value of delay, but there were no significant differences between the three cases.

Table 18: Tuning of time window length for *Steady* fault identification. Variance=0.001

Risk aversion in MCDM	High (orness=0.2)					Moderate (orness=0.5)					Low (orness=0.9)				
	5	7	10	12	15	5	7	10	12	15	5	7	10	12	15
Window length (samples)	5	7	10	12	15	5	7	10	12	15	5	7	10	12	15
Average delay (samples)	351.21	357.45	458.51	465.89	461.52	326.84	354.21	399.54	416.27	429.95	388.14	409.53	419.74	457.15	468.29
Stabilization detections	12	15	16	17	17	15	16	17	17	18	15	17	17	17	18

Table 19: Tuning of time window length for *Steady* fault identification. Variance=0.002

Risk aversion in MCDM	High (orness=0.2)					Moderate (orness=0.5)					Low (orness=0.9)				
	5	7	10	12	15	5	7	10	12	15	5	7	10	12	15
Window length (samples)	5	7	10	12	15	5	7	10	12	15	5	7	10	12	15
Average delay (samples)	320.46	347.56	453.05	458.88	459.55	316.75	348.19	398.63	405.33	424.49	382.18	403.88	414.47	449.11	459.5
Stabilization detections	13	16	18	18	18	16	16	18	18	18	16	17	18	18	18

It is worth nothing that the proposed methodology reduced this fault identification time for the stabilized system by up to 50% in comparison with other approaches. In some cases, this was reduced more than 15 times. The variance of the delays obtained was high because the faults where the proposed method cannot reach a stabilization point were included. At the end of Table 24, there are values of mean and standard deviation delay, taking into account only the stabilized faults, which reduced the delay by nearly a quarter of the time for the CVA [17] and RCI [15] proposals. So, in most faults, the OWA-RIM aggregation based Fault Identification is a lot faster.

Tables 25, 26 and 27 show the results of the fault identification for each selected Risk Aversion (orness) value.

When the Risk Aversion value was reduced, the scores were higher, as expected, and the fault identification was similar to the CVA and RCI methods, but it could be seen earlier. In addition, the user can adjust the risk of the fault identification.

For example, with a low Risk Aversion value (orness = 0.2), the first candidate variables obtained a notably higher score than the next variables (better separability), allowing an easier decision-making.

Also, the proposal gave information about the most relevant variables, while the fault identification obtained by CVA, in many cases (faults 2, 6, 8, etc.), gave too long a list of candidate variables, making it difficult to analyze the condition of the plant. Compared with RCI, ORAFI gave a scored ranking of all the variables, that is, the level of influence of the fault for each variable.

These results lead us to believe that the two-step fault identification is necessary. In fact, the results obtained by the methods that analyze the first instants are very different from the ones that analyze the whole available dataset. The system evolves after the fault and, in many cases, the identification of faulty variables is different between the first instants after the fault and when the system has reached a new steady condition. In the first case, the information obtained refers to the origin of the fault and, in the second, to the new working conditions of the plant.

Table 20: Tuning of time window length for *Steady* fault identification. Variance=0.003

Risk aversion in MCDM	High (orness=0.2)					Moderate (orness=0.5)					Low (orness=0.9)				
	5	7	10	12	15	5	7	10	12	15	5	7	10	12	15
Window length (samples)	310.55	331.55	442.15	449.84	451.57	306.53	344.84	401.36	402.81	421.41	373.22	399.51	409.74	431.01	439.57
Average delay (samples)	10	14	15	17	17	13	14	16	16	16	14	16	17	17	17
Stabilization detections															

Table 21: Early fault identification comparison. High risk aversion (orness=0.2)

Fault	Early identification-ranking	Early identification-score	PCA	B & B	CA	SVM
1	16 20 7 25 13 23	0.5483 0.3303 0.2298 0.2212 0.1836 0.1127	16 20 46	16	16 13 7 20	16 18
2	30 24 10 47 16 7	0.46213 0.2364 0.1476 0.1089 0.0602 0.0318	30 24 16 40	21 24 30	40 4 16 13 20 7	47 10 16
3	37 32 8 45 14 18	0.1333 0.0931 0.0866 0.0760 0.0744 0.0709	nd	nd	nd	nd
4	51 9 24 30 28 35	0.5870 0.0158 0.0152 0.0109 0.0102 0.0085	51 21 9	9 51	51	51
5	11 35 8 14 24 30	0.5098 0.4717 0.1348 0.0908 0.0844 0.0816	11 9 35 22 18	11 22	4 21 22 11	52 17
6	44 1 20 21 13 7	0.4531 0.1856 0.0367 0.0180 0.0122 0.0119	44 1	1 44	44 1	1
7	16 7 13 4 45 21	0.5968 0.4071 0.3660 0.2771 0.2078 0.1355	4 16 36 38 31 25	4 6 9 16 22 45 51	16 7 13 21 4	45
8	20 16 46 31 25 23	0.5221 0.397 0.3239 0.2416 0.2330 0.1594	3 46 29 16 20	37	46 13 7 16 20	16 23 20 30
9	nd	nd	nd	nd	nd	nd
10	18 8 23 20 3 31	0.2911 0.1154 0.0918 0.0871 0.0782 0.0704	19 34 35		19 20 18	18 38 50
11	51 9 39 42 21 14	0.3858 0.0279 0.0086 0.0065 0.0061 0.006	51 9 21	51	51	51
12	11 22 35 9 36 34	0.366 0.2214 0.0685 0.0344 0.0264 0.0204	11 37 22 4	22	21 4 11 22	11 22 4
13	16 7 32 33 13 6	0.3655 0.2484 0.2407 0.2181 0.2023 0.0933	37 32 26 42	37	51 16 20 13 7	34 16
14	51 9 21 2 42 28	0.2783 0.2737 0.1914 0.0246 0.0137 0.0061	21 51 9 2 42	9 51	21 51	9 51
15	nd	nd	nd	nd	nd	nd
16	19 18 32 29 1 8	0.5599 0.384 0.0787 0.0413 0.0310 0.0268	50 3 27		50 19	50 19
17	21 2 42 27 24 35	0.5977 0.0125 0.0091 0.0042 0.0038 0.0033	21 2 42		51 21	29 9
18	22 11 35 33 51 24	0.8027 0.0711 0.0108 0.0096 0.0083 0.0082	22 11 40		4 11 40 22	51 17
19	5 32 39 43 3 27	0.3553 0.0555 0.0353 0.0331 0.0313 0.0274	5 27 37 46 38		40 51 20 4	5 20 46
20	46 13 20 34 5 40	0.6478 0.1466 0.0932 0.0446 0.0265 0.0235	46 39 13 23		8 42 11 40	13 52 17 7
21	19 21 50 40 23 3	0.167 0.1220 0.0979 0.0754 0.0715 0.0641	39 26 5		13 7 42 20	19 37 16

nd = not detected

4.2.3. Fault 7

Here, a brief description is given of the behaviour of the proposed method for an example fault: fault 7, which is a pressure loss in the C header, i.e., the reduced availability of stream 4. In the *Early* fault identification, the most important candidate variables, according to the scores, were 16,7,13,4, 21 and 45 (for High and Moderate Risk Aversion, orness = 0.2 and 0.5). For the *Steady* fault identification, it was the variable 45. Looking at Figure 3, it is clear that, at the start of the fault, the variables 4, 7, 13, 16, 21 and 45 suffered a strong change in their value. This fault was a pressure drop in stream 4, and the ORAFI indicated that it particularly affected variables 4, 16, 7 and 13, which measure the flowrate of stream 4 (variable 4) that has suddenly dropped, as well as the pressures in the stripper (variable 16), the reactor (variable 7) and the product separator (variable 13), respectively. Since these pressures were not regulated by controllers, they varied proportionally with the gas flow rate, because these elements are located downstream from Stream 4. The valve (variable 45) was also opened to compensate for the reduced availability of stream 4. So the fault identification is correct.

Some instants later, these variables started to return to their original values, due to the effect of the control system; except for variable 45, which remained at an abnormal value. This was reflected by the *Steady* fault identification and is the expected behaviour of the plant, which modifies the stream 4 valve set-point to compensate the pressure loss in the above mentioned stream. This suggests that the proposed approach was capable of locating the major faulty variables with a high score in the ranking and filtering out the minor faulty ones with a

Table 22: Early fault identification comparison. Moderate risk aversion (orness=0.5)

Fault	Early identification-ranking	Early identification-score	PCA	B & B	CA	SVM
1	16 20 25 7 23 46	0.6688 0.5514 0.3217 0.2976 0.2807 0.2722	16 20 46	16	16 13 7 20	16 18
2	30 24 10 47 31 25	0.7136 0.4399 0.3570 0.3453 0.202 0.1871	30 24 16 40	21 24 30	40 4 16 13 20 7	47 10 16
3	8 37 45 32 24 14	0.3073 0.2953 0.2849 0.2744 0.2301 0.2151	nd	nd	nd	nd
4	51 9 34 24 6 14	0.7876 0.1627 0.0735 0.0669 0.0647 0.0610	51 21 9	9 51	51	51
5	35 11 4 14 22 24	0.6524 0.6151 0.3231 0.2916 0.2795 0.246	11 9 35 22 18	11 22	4 21 22 11	52 17
6	44 1 20 46 13 7	0.77 0.5422 0.2089 0.1537 0.1118 0.1051	44 1	1 44	44 1	1
7	16 4 7 13 45 31	0.8118 0.5931 0.5528 0.5126 0.3457 0.2974	4 16 36 38 31 25	4 6 9 16 22 45 51	16 7 13 21 4	45
8	20 16 46 31 25 29	0.7388 0.5849 0.5830 0.3528 0.3222 0.2801	3 46 29 16 20	37	46 13 7 16 20	16 23 20 30
9	nd	nd	nd	nd	nd	nd
10	18 20 4 23 31 3	0.5900 0.2833 0.2601 0.2385 0.1984 0.1983	19 34 35	nd	19 20 18	18 38 50
11	51 9 6 21 27 39	0.4677 0.1938 0.0575 0.0482 0.0467 0.0416	51 9 21	51	51	51
12	11 22 35 41 36 6	0.5313 0.4558 0.1965 0.1279 0.1122 0.0981	11 37 22 4	22	21 4 11 22	11 22 4
13	16 32 33 7 13 21	0.5989 0.4806 0.4682 0.4142 0.3614 0.2817	37 32 26 42	37	51 16 20 13 7	34 16
14	51 9 21 2 42 6	0.4287 0.4249 0.3707 0.1625 0.1137 0.0608	21 51 9 2 42	9 51	21 51	9 51
15	nd	nd	nd	nd	nd	nd
16	19 18 29 32 31 50	0.7436 0.5232 0.2450 0.1937 0.1509 0.1487	50 3 27	nd	50 19	50 19
17	21 2 42 51 9 41	0.7028 0.1031 0.0862 0.0427 0.0387 0.0347	21 2 42	nd	51 21	29 9
18	22 11 13 7 2 40	0.8921 0.3307 0.0818 0.0707 0.0554 0.0528	22 11 40	nd	4 11 40 22	51 17
19	5 43 24 46 32 34	0.5538 0.1825 0.1793 0.1517 0.1517 0.1487	5 27 37 46 38	nd	40 51 20 4	5 20 46
20	46 13 20 7 5 23	0.7359 0.2849 0.2179 0.1628 0.1570 0.1295	46 39 13 23	nd	8 42 11 40	13 52 17 7
21	19 50 21 40 18 29	0.5139 0.3886 0.3439 0.2643 0.2599 0.25064	39 26 5	nd	13 7 42 20	19 37 16

nd = not detected

Table 23: Early fault identification comparison. Low risk aversion (orness=0.9)

Fault	Early identification-ranking	Early identification-score	PCA	B & B	CA	SVM
1	16 20 46 23 18 25	0.8361 0.7934 0.5562 0.5550 0.5312 0.5088	16 20 46	16	16 13 7 20	16 18
2	30 24 31 47 25 10	0.9153 0.7108 0.598 0.5673 0.5642 0.5578	30 24 16 40	21 24 30	40 4 16 13 20 7	47 10 16
3	8 37 45 24 32 29	0.5919 0.5365 0.5276 0.48566 0.4832 0.4781	nd	nd	nd	nd
4	51 9 34 20 14 6	0.9210 0.4219 0.2461 0.2151 0.2024 0.19943	51 21 9	9 51	51	51
5	35 11 4 22 26 14	0.8426 0.7423 0.6653 0.6149 0.5854 0.5771	11 9 35 22 18	11 22	4 21 22 11	52 17
6	44 1 20 46 13 7	0.9345 0.8263 0.5801 0.4979 0.3622 0.3403	44 1	1 44	44 1	1
7	16 4 7 13 31 8	0.9437 0.7974 0.7398 0.7215 0.7190 0.6742	4 16 36 38 31 25	4 6 9 16 22 45 51	16 7 13 21 4	45
8	20 46 16 18 29 31	0.8985 0.7994 0.7876 0.5995 0.5526 0.5492	3 46 29 16 20	37	46 13 7 16 20	16 23 20 30
9	nd	nd	nd	nd	nd	nd
10	18 20 4 29 23 3	0.8054 0.5397 0.5149 0.4374 0.4225 0.3912	19 34 35	nd	19 20 18	18 38 50
11	51 9 27 21 6 29	0.6107 0.5558 0.1684 0.1621 0.1589 0.1525	51 9 21	51	51	51
12	11 22 35 41 6 24	0.8320 0.7427 0.4922 0.4377 0.3396 0.3111	11 37 22 4	22	21 4 11 22	11 22 4
13	16 32 33 21 7 40	0.7887 0.7641 0.7417 0.6040 0.6027 0.5611	37 32 26 42	37	51 16 20 13 7	34 16
14	51 9 21 2 42 22	0.5604 0.5545 0.5355 0.4569 0.3639 0.2035	21 51 9 2 42	9 51	21 51	9 51
15	nd	nd	nd	nd	nd	nd
16	19 18 29 50 31 32	0.9176 0.6899 0.6144 0.5488 0.4319 0.413	50 3 27	nd	50 19	50 19
17	21 2 42 51 9 22	0.8322 0.2951 0.2511 0.1296 0.1139 0.108	21 2 42	nd	51 21	29 9
18	22 11 13 7 16 40	0.9628 0.7411 0.3181 0.2880 0.202 0.1879	22 11 40	nd	4 11 40 22	51 17
19	5 24 43 29 46 34	0.7335 0.5093 0.4483 0.4450 0.4240 0.4197	5 27 37 46 38	nd	40 51 20 4	5 20 46
20	46 13 7 5 23 20	0.8671 0.5103 0.4676 0.3955 0.3828 0.3807	46 39 13 23	nd	8 42 11 40	13 52 17 7
21	19 50 21 18 3 29	0.9080 0.7351 0.5726 0.5140 0.5078 0.49079	39 26 5	nd	13 7 42 20	19 37 16

nd = not detected

low score in the ranking, which would be practical for engineers in diagnosing a process fault in a limited time frame.

Once more, the idea of using both time-windows for fault identification in two different times, early and steady fault identification, is supported by the results because some variables were affected at the first instant of the faults, and others as the fault was propagated over time. If only one of the windows (the early or the steady time-window) was used, a lot of information about the faults would be lost. Also, the aggregation of different FI methods can give a more robust diagnosis.

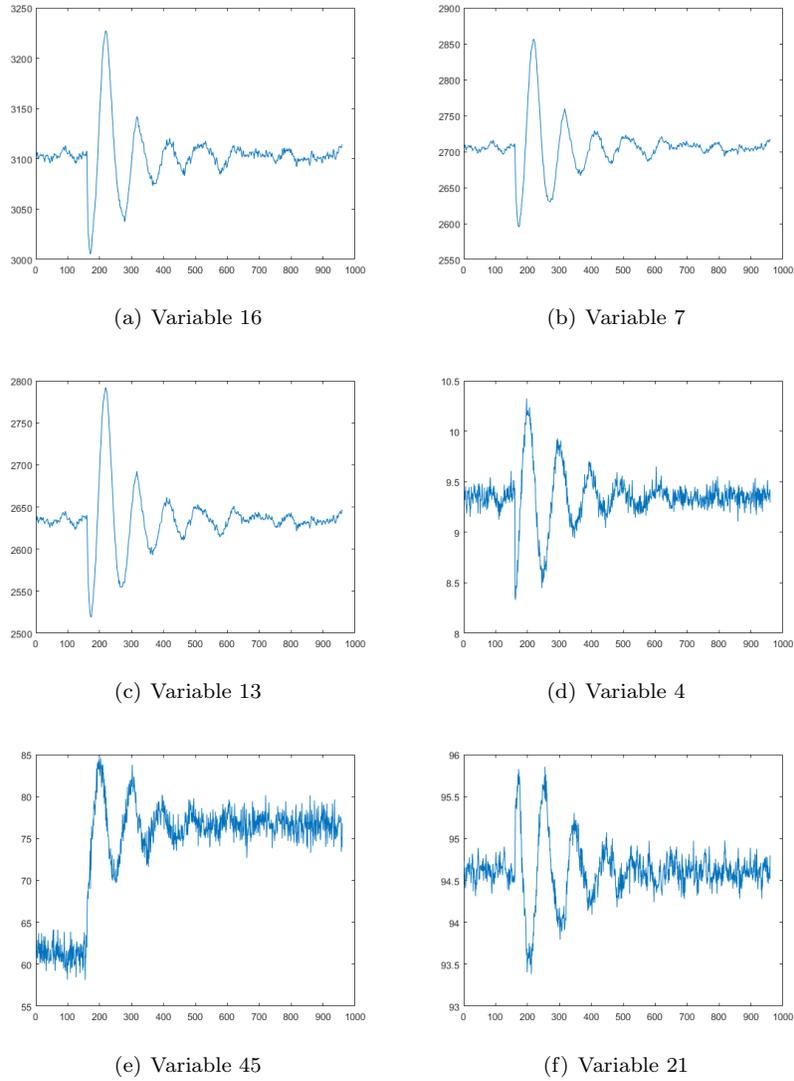


Figure 3: Fault 7. Evolution of the candidate variables.

5. Conclusions

A method to carry out the fault identification of a plant is introduced here. The main innovations of the OWA-RIM Aggregation based Fault identification are: the aggregation of different well known fault identification methods carried out with a multicriterion decision-making approach and the fault identification is carried out in two steps using time-windows.

The aggregation of different methods allows the results of a method to be used when it is really effective, and to discard it when it is not valid (because it does not give a similar result to other methods). This makes the Fault Identification more robust. The operator also has the option to decide on the risk level of the fault identification, taking into account only the variables that are selected as candidates by nearly all the methods, or including in the diagnosis the variables with a good score in only a few methods.

The fault identification is carried out in two steps: the first one, developed just after the fault detection, helps to identify the origin of the fault. The second identification, performed when the system reaches a stable operation point after the fault happens, gives information to the user about the new working condition after the fault and whether it is advisable to continue operating the plant or carry out repairs. Both fault identifications are developed in a time-window which prevents the effect of noise, disturbances, etc., making the method more robust.

The method has been tested in two plants: a waste water treatment plant and the Tennessee Eastman Plant, showing its effectiveness in performing the fault identification and showing that both identification (*Early* and *Steady*) are necessary to correctly diagnose the state of the plant.

In this proposal, seven FI methods have been considered, but it is open to extending the ORAFI aggregation with more methods. This can improve the effectiveness and robustness of the method.

Acknowledgement

This work has been partially supported by the Spanish Government and the European Regional Development Fund (FEDER) through the Project DPI2015-67341- C2-2-R.

6. References

- [1] V. Venkatasubramanian, R. Rengaswamy, S. Kavuri, A review of process fault detection and diagnosis. Part I: quantitative model-based methods, *Computers & Chemical Engineering* 27 (2003) 291–311.
- [2] V. Venkatasubramanian, R. Rengaswamy, S. Kavuri, A review of process fault detection and diagnosis. Part II: qualitative models and search strategies, *Computers & Chemical Engineering* 27 (2003) 313–326.
- [3] V. Venkatasubramanian, R. Rengaswamy, S. Kavuri, K. Yin, A review of process fault detection and diagnosis. Part III: process history based methods, *Computers & Chemical Engineering* 27 (2003) 327–346.
- [4] B. Wise, N. Gallagher, The process chemometrics approach to process monitoring and fault detection, *Journal of Process Control* 6 (1996) 329–348.

- [5] D. X. Tien, K. W. Lim, L. Jun, Comparative study of PCA approaches in process monitoring and fault detection, in: 30th Annual Conference of IEEE Industrial Electronics Society, IECON 2004, 2594-2599, Vol. 3, 2004.
- [6] S. Ding, P. Zhang, E. Ding, S. Yin, A. Naik, P. Deng, W. Gui, On the application of PCA technique to fault diagnosis, *Tsinghua Science and Technology* 2 (15) (2010) 138–144.
- [7] S. Qin, Data-driven fault detection and diagnosis for complex industrial processes, in: *Proceedings of the 7th IFAC Symposium on Fault Detection and Supervision and Safety of Technical Processes*, 1115-1125, 2009.
- [8] M. Livani, M. Abadi, Distributed PCA-based anomaly detection in wireless sensor networks, in: *2010 IEEE International Conference for Internet Technology and Secure Transactions*, 1-8, 2010.
- [9] C. Lau, K. Ghosh, M. Hussain, C. C. Hassan, Fault diagnosis of Tennessee Eastman process with multi-scale PCA and ANFIS, *Chemometrics and Intelligent Laboratory Systems* 120 (2013) 1–14.
- [10] T. Kourti, J. MacGregor, Multivariate SPC methods for process and product monitoring, *Journal of Quality Technology* 28 (1996) 409–428.
- [11] R. Dunia, S. J. Qin, Subspace approach to multidimensional fault identification and reconstruction., *AIChE Journal* 44 (1998) 1813–1831.
- [12] H. Yue, S. J. Qin, Reconstruction based fault identification, using a combined index, *Industrial & Engineering Chemistry Research* 40(20) (2001) 4403–4414.
- [13] C. F. Alcalá, S. J. Qin, Reconstruction-based contribution for process monitoring, *Automatica* 45 (2009) 1593–1600.
- [14] K. Detroja, R. Gudi, S. Patwardhan, Plant-wide detection and diagnosis using correspondence analysis, *Control Engineering Practice* 15 (2007) 1468–1483.
- [15] J. Liu, Fault diagnosis using contribution plots without smearing effect on non-faulty variables, *Journal of Process Control* 22 (2012) 1609–1623.
- [16] J. Liu, D. S. Chen, Fault isolation using modified contribution plots., *Computers & Chemical Engineering* 61 (2014) 9–19.
- [17] B. Jiang, D. Huang, X. Zhu, F. Yang, R. D. Braatz, Canonical variate analysis-based contributions for fault identification, *Journal of Process Control* 26 (2015) 17–25.
- [18] V. Kariwala, P. E. Odiwei, Y. Cao, T. Chen, A branch and bound method for isolation of faulty variables through missing variable analysis, *Journal of Process Control* 20 (2010) 1198–1206.

- [19] S. Mahadevan, S. L. Shah, Fault detection and diagnosis in process data using one-class support vector machines, *Journal of Process Control* 19 (2009) 1627–1639.
- [20] F. Zhang, Z. Ge, Decision fusion systems for fault detection and identification in industrial processes, *Journal of Process Control* 31 (Supplement C) (2015) 45–54.
- [21] K. Ghosh, Y. S. Ng, R. Srinivasan, Evaluation of decision fusion strategies for effective collaboration among heterogeneous fault diagnostic methods, *Computers and Chemical Engineering* 35 (2011) 342–355.
- [22] M. Kano, S. Hasebe, I. Hashimoto, H. Ohno, A new multivariate statistical process monitoring method using principal component analysis, *Computers & Chemical Engineering* 25 (2001) 1103–1113.
- [23] S. Qin, W. Li, H. Yue, Recursive PCA for adaptive process monitoring, in: *Proceedings of IFAC World Congress*, 8590. N, Beijing, China, 1999.
- [24] W. Li, H. Yue, S. Valle-Cervantes, S. Qin, Recursive PCA for adaptive process monitoring, *Journal of Process Control* 10(5) (2000) 471–486.
- [25] I. Jolliffe, *Principal component analysis*, Springer, 2002.
- [26] S. Qin, *Statistical process monitoring: basics and beyond*, *Journal of Chemometrics* 17 (2003) 480–502.
- [27] J. Liu, D. Wong, D. S. Chen, Bayesian filtering of the smearing effect: fault isolation in chemical process monitoring, *Journal of Process Control* 24 (2014) 1–21.
- [28] J. E. Jackson, *A user’s guide to principal components*, Wiley, 1991.
- [29] R. W. Hopkins, P. Miller, R. E. Swanson, J. J. Scheible, Method of controlling a manufacturing process using multivariate analysis. US Patent 5442562 (August 1995).
- [30] J. Westerhuis, S. Gurden, A. Smilde, Generalized contribution in multivariable statistical process monitoring, *Chemometrics and Intelligent Laboratory Systems* 51 (2000) 95–114.
- [31] S. Qin, S. Valle, J. Piovoso, On unifying multiblock analysis with application to decentralized process monitoring, *Journal of Chemometrics* 15 (2001) 715–742.
- [32] G. Li, C. F. Alcalá, S. J. Qin, D. Zhou, Generalized reconstruction-based contributions for output-relevant fault diagnosis with application to the Tennessee Eastman process, *IEEE Transactions on Control Systems Technology* 19 (5) (2011) 1114–1127.

- [33] G. Li, S. J. Qin, T. Yuan, Data-driven root cause diagnosis of faults in process industries, *Chemometrics and Intelligent Laboratory Systems* 159 (Supplement C) (2016) 1–11.
- [34] J. Yang, Y. Chen, Z. Sun, A real-time fault detection and isolation strategy for gas sensor arrays, in: *2017 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 1-6, 2017.
- [35] X. Guo, M. Yang, Y. Li, Modified reconstruction-based contribution plots for fault isolation, *Chinese Journal of Scientific Instrument* 36 (5) (2015) 1193–1200.
- [36] P. J. Gudiel-Pineda, J. Liou, C. C. Hsu, Y. C. Chuang, An integrated MCDM model for improving airline operational and financial performance, *Journal of Air Transport Management*. (In Press, Corrected Proof, Available online 22 February 2017).
- [37] G. Kou, Y. Peng, G. Wang, Evaluation of clustering algorithms for financial risk analysis using MCDM methods, *Information Sciences* 275 (Supplement C) (2014) 1–12.
- [38] V. V. Dhanisetty, W. Verhagen, R. Curran, Multi-criteria weighted decision making for operational maintenance processes, *Journal of Air Transport Management* (In Press, Corrected Proof, Available online 15 September 2017).
- [39] G. Villacreses, G. Gaona, J. Martinez-Gomez, D. J. Jijn, Wind farms suitability location using geographical information system (GIS), based on multi-criteria decision making (MCDM) methods: The case of continental Ecuador, *Renewable Energy* 109 (C) (2017) 275 –286.
- [40] A. Mardani, A. Jusoh, E. Zavadskas, Fuzzy multiple criteria decision-making techniques and applications Two decades review from 1994 to 2014, *Expert Systems with Applications* 42 (2015) 4126–4148.
- [41] M. Behzadian, S. Otaghsara, M. Yazdani, J. Ignatius, A state-of-the-art survey of TOPSIS applications, *Expert Systems with Applications* 200 (2012) 198–215.
- [42] R.R. Yager, On ordered weighted averaging aggregation operators in multi-criteria decision making, *IEEE Transactions on Systems, Man and Cybernetics* 18 (1988) 183–190.
- [43] X. Liu, S. Han, Orness and parameterized RIM quantifier aggregation with OWA operators: a summary, *International Journal of Approximate Reasoning* 48 (2008) 77–97.
- [44] R. Fuller, Exploring the limits of support systems, *Turku Centre for Computer Science*. Abo, 1996, Ch. OWA Operators in Decision Making, pp. 90–103.

- [45] R. Yager, Families of OWA operators, *Fuzzy Sets and Systems* 59 (1993) 125–148.
- [46] R. Yager, Quantifier guided aggregation using OWA operators, *International Journal of Intelligent Systems* 11 (1996) 49–73.
- [47] R. Fuller, P. Majlender, An analytic approach for obtaining maximal entropy OWA operator weights, *Fuzzy Sets and Systems* 124 (2001) 53–57.
- [48] M. Grbovic, W. Li, P. Xu, A. Usadi, L. Song, S. Vucetic, Decentralized fault detection and diagnosis via sparse PCA based decomposition and maximum entropy decision fusion, *Journal of Process Control* 22 (2012) 738–750.
- [49] P. Odiowei, Y. Cao, State-space independent component analysis for non-linear dynamic process monitoring, *Chemometrics and Intelligent Laboratory Systems* 103 (2010) 59–65.
- [50] Z. Ge, Z. Song, Distributed PCA model for plant-wide process monitoring, *Industrial and Engineering Chemistry Research* 52 (2013) 1947–1957.
- [51] The COST Simulation Benchmark: Description and Simulator Manual, <http://www.benchmarkwwtp.org>.
- [52] I. Nopens, L. Benedetti, U. Jeppsson, M. N. Pons, J. Alex, J. B. Copp, K. V. Gernaey, C. Rosen, J. P. Steyer, P. A. Vanrolleghem, Benchmark simulation model No 2: finalisation of plant layout and default control strategy, *Water Science and Technology* 62 (2010) 1967–1974.
- [53] J. J. Downs, E. F. Vogel, A plant-wide industrial process control problem, *Computers & Chemical Engineering* 17 (1993) 245–255.
- [54] L. H. Chiang, E. L. Russell, R. D. Braatz, *Fault detection and diagnosis in industrial systems*, Springer London, 2001.

Table 24: *Steady* fault identification delay.

Fault	Steady identification			CVA	RCI
	High risk aversion (orness=0.2)	Moderate risk aversion (orness=0.5)	Low risk aversion (orness=0.9)		
1	417	464	494	800	800
2	43	74	122	800	800
3	800	800	800	nd	nd
4	69	34	33	800	800
5	168	168	168	800	800
6	41	42	189	800	800
7	59	52	158	800	800
8	800	89	773	800	800
9	nd	nd	nd	nd	nd
10	800	800	800	800	800
11	800	800	800	800	800
12	800	800	800	800	800
13	113	101	114	800	800
14	184	83	74	800	800
15	nd	nd	nd	nd	nd
16	800	800	265	800	800
17	93	56	55	800	800
18	342	276	116	800	800
19	800	800	800	800	800
20	800	800	800	800	800
21	679	535	514	800	800
Average	453	399	414	800	800
Std. Dev.	339	342	324	0	0
Average (only stabilized)	201	165	237	800	800
Std. Dev. (only stabilized)	201	171	221	0	0

nd = not detected

Table 25: Steady fault identification comparison. High risk aversion (orness=0.2)

Fault	Steady ranking	Steady scores	CVA-SSFV	CVA-RSFV	RCI
1	50 1 44 18 19 20	0.49177 0.29929 0.29469 0.14841 0.095174 0.059851	18 19 50	1 4 34 44 45	1 4 18 44 45 50
2	47 30 10 24 31 16	0.24747 0.24404 0.2428 0.10482 0.025131 0.021431	18 19 28 34 47 50	3 9 13 25 28 29 34 43 46 47	34 28 10 47
3	36 40 30 4 24 41	0.11424 0.061272 0.054644 0.052433 0.051343 0.048675	nd	nd	nd
4	51 14 9 5 8 42	0.51602 0.015004 0.012828 0.012176 0.010698 0.0098518	nd	51	51
5	50 19 18 38 7 13	0.54234 0.42538 0.25552 0.14817 0.11315 0.10701	50	52	38 19
6	44 1 16 29 20 7	0.58565 0.22914 0.17015 0.1086 0.10025 0.068958	2 3 7 11 13 16 21 22 23 25 26 30 31 33 35 36 38 43 44 47	1 2 3 7 8 9 11 13 16 17 21 25 28 29 31 34 35 36 37 38 39 42 43 45 46 51 52	51
7	45 4 8 21 20 36	0.7031 0.13786 0.12558 0.023644 0.021061 0.017697	nd	45	45
8	16 7 13 10 47 34	0.70973 0.46838 0.40685 0.18112 0.15369 0.14811	7 10 11 13 16 18 19 20 23 30 47 50	1 4 7 10 13 20 24 25 28 29 31 34 35 43 44 46 47	16 10 47 7 30 13
9	nd	nd	nd	nd	nd
10	32 6 19 27 14 5	0.069529 0.059034 0.055383 0.048206 0.044145 0.039671	7 13 19 50	18	18
11	51 9 32 6 5 40	0.3401 0.041143 0.0094807 0.0076341 0.0075326 0.0073716	nd	9 51	51 9
12	38 11 50 22 21 8	0.28026 0.15735 0.086265 0.065918 0.056208 0.053948	7 9 11 13 16 18 19 20 22 23 27 29 30 31 33 36 38 47 50	1 3 4 8 20 21 25 35 43 52	11 38 4 22
13	19 50 18 16 7 13	0.75033 0.44267 0.32921 0.10664 0.077435 0.07053	7 11 13 16 18 19 20 21 22 23 26 27 30 31 33 36 38 41 44 47	3 4 7 8 11 13 16 18 19 20 25 27 28 29 30 32 34 35 36 37 43 50 51	19 38 50 16 7 13
14	51 9 21 2 42 6	0.29523 0.24123 0.18186 0.010421 0.0094964 0.0067851	50	9 21 51	51 9 21
15	nd	nd	nd	nd	nd
16	19 18 38 35 40 32	0.28998 0.14713 0.13261 0.067832 0.058264 0.055488	7 13 18 19 20 50	16	19 80 18
17	21 9 2 42 51 6	0.53238 0.020847 0.0085033 0.0058579 0.0028783 0.0014499	7 9 11 13 16 18 19 21 30 50	9 21 51	21
18	43 51 46 42 45 4	0.8269 0.56166 0.20131 0.10881 0.071067 0.054741	4 7 9 11 13 14 16 20 22 23 24 26 27 28 29 30 33 35 36 39 41 46 47 50 51	6 7 12 13 16 17 19 21 23 25 27 28 30 32 33 35 36 37 38 41 42 43 45 46 48 51 52	43 51 46
19	5 46 35 41 39 34	0.07696 0.060251 0.05253 0.049232 0.048419 0.026763	nd	5	5
20	46 20 13 11 7 29	0.51899 0.14688 0.064884 0.025334 0.01418 0.012639	7 11 13 16 18 20 13 50	20 46	46
21	8 19 50 7 16 13	0.46479 0.21241 0.14141 0.12369 0.12307 0.11559	nd	nd	8

nd = not detected

Table 26: Steady fault identification comparison. Moderate risk aversion ($\alpha=0.5$)

Fault	Steady ranking	Steady scores	CVA-SSFV	CVA-RSFV	RCI
1	44 1 4 50 19 18	0.6462 0.6412 0.6061 0.4114 0.3218 0.2524	18 19 50	1 4 34 44 45	1 4 18 44 45 50
2	30 10 47 24 25 37	0.5662 0.5495 0.5473 0.181 0.1134 0.1120	18 19 28 34 47 50	3 9 13 25 28 29 34 43 46 47	34 28 10 47
3	36 40 30 4 24 41	0.1142 0.0613 0.0546 0.0524 0.0513 0.0486	nd	nd	nd
4	51 9 28 34 6 43	0.6504 0.1527 0.0811 0.0634 0.0628 0.0431	nd	51	51
5	38 1 44 50 19 52	0.6612 0.2032 0.201 0.1073 0.1046 0.1041	50	52	38 19
6	44 1 16 29 20 7	0.7842 0.5434 0.301 0.2545 0.1719 0.1567	2 3 7 11 13 16 21 22 23 25 26 30 31 33 35 36 38 43 44 47	1 2 3 7 8 9 11 13 16 17 21 25 28 29 31 34 35 36 37 38 39 42 43 45 46 51 52	51
7	45 4 8 46 20 30	0.8464 0.3166 0.2049 0.1609 0.1538 0.1155	nd	45	45
8	16 7 13 10 47 34	0.8154 0.5558 0.4963 0.3885 0.344 0.338	7 10 11 13 16 18 19 20 23 30 47 50	1 4 7 10 13 20 24 25 28 29 31 34 35 43 44 46 47	16 10 47 7 30 13
9	nd	nd	nd	nd	nd
10	38 19 50 36 24 26	0.4743 0.27479 0.2723 0.1899 0.1761 0.1675	7 13 19 50	18	18
11	51 9 32 6 5 40	0.3401 0.0411 0.0095 0.0076 0.0075 0.0073	nd	9 51	51 9
12	11 16 35 7 13 36	0.512 0.3317 0.2744 0.2595 0.2348 0.1909	7 9 11 13 16 18 19 20 22 23 27 29 30 31 33 36 38 47 50	1 3 4 8 20 21 25 35 43 52	11 38 4 22
13	16 7 19 13 38 50	0.6855 0.498 0.4719 0.4536 0.441 0.3798	7 11 13 16 18 19 20 21 22 23 26 27 30 31 33 36 38 41 44 47	3 4 7 8 11 13 16 18 19 20 25 27 28 29 30 32 34 35 36 37 43 50 51	19 38 50 16 7 13
14	51 21 9 2 6 42	0.3991 0.3748 0.3484 0.0874 0.0821 0.0794	50	9 21 51	51 9 21
15	nd	nd	nd	nd	nd
16	50 34 38 29 20 28	0.5429 0.1863 0.16 0.1542 0.1423 0.1373	7 13 18 19 20 50	16	19 80 18
17	21 2 42 9 51 6	0.5630 0.0861 0.0735 0.0612 0.0370 0.0259	7 9 11 13 16 18 19 21 30 50	9 21 51	21
18	22 11 13 7 16 35	0.5848 0.2474 0.1233 0.1166 0.1067 0.0521	4 7 9 11 13 14 16 20 22 23 24 26 27 28 29 30 33 35 36 39 41 46 47 50 51	6 7 12 13 16 17 19 21 23 25 27 28 30 32 33 35 36 37 38 41 42 43 45 46 48 51 52	43 51 46
19	5 46 35 41 39 34	0.0769 0.0602 0.0525 0.0492 0.0484 0.0267	nd	5	5
20	46 11 13 20 7 22	0.4836 0.1423 0.1319 0.1212 0.1196 0.1145	7 11 13 16 18 20 13 50	20 46	46
21	8 19 50 16 7 44	0.5760 0.2629 0.2252 0.1845 0.1500 0.1451	nd	nd	8

nd = not detected

Table 27: Steady fault identification comparison. Low risk aversion (orness=0.9)

Fault	Steady ranking	Steady scores	CVA-SSFV	CVA-RSFV	RCI
1	1 44 4 45 24 21	0.9133 0.9103 0.7810 0.4619 0.395 0.3745	18 19 50	1 4 34 44 45	1 4 18 44 45 50
2	10 47 30 37 34 25	0.9013 0.8562 0.7942 0.5145 0.4254 0.4139	18 19 28 34 47 50	3 9 13 25 28 29 34 43 46 47	34 28 10 47
3	36 40 30 4 24 41	0.1142 0.0612 0.0546 0.0524 0.0513 0.0487	nd	nd	nd
4	51 9 28 6 34 21	0.8689 0.4666 0.2311 0.1998 0.1653 0.1568	nd	51	51
5	52 17 40 33 38 23	0.5027 0.4126 0.3544 0.3238 0.3115 0.3029	50	52	38 19
6	51 9 16 1 7 44	0.9955 0.7129 0.7108 0.5244 0.5158 0.4922	2 3 7 11 13 16 21 22 23 25 26 30 31 33 35 36 38 43 44 47	1 2 3 7 8 9 11 13 16 17 21 25 28 29 31 34 35 36 37 38 39 42 43 45 46 51 52	51
7	45 20 29 4 50 46	0.8831 0.7119 0.7102 0.652 0.5044 0.442	nd	45	45
8	10 47 30 37 34 25	0.8914 0.8741 0.7907 0.6181 0.4568 0.3878	7 10 11 13 16 18 19 20 23 30 47 50	1 4 7 10 13 20 24 25 28 29 31 34 35 43 44 46 47	16 10 47 7 30 13
9	nd	nd	nd	nd	nd
10	38 19 50 36 24 26	0.4743 0.2748 0.2723 0.1899 0.1762 0.1675	7 13 19 50	18	18
11	51 9 27 6 21 34	0.6658 0.5394 0.2029 0.1888 0.1883 0.1789	nd	9 51	51 9
12	22 11 35 40 16 21	0.5881 0.5441 0.3418 0.2689 0.2347 0.2059	7 9 11 13 16 18 19 20 22 23 27 29 30 31 33 36 38 47 50	1 3 4 8 20 21 25 35 43 52	11 38 4 22
13	19 16 50 7 18 13	0.9651 0.8872 0.7976 0.6509 0.6312 0.5948	7 11 13 16 18 19 20 21 22 23 26 27 30 31 33 36 38 41 44 47	3 4 7 8 11 13 16 18 19 20 25 27 28 29 30 32 34 35 36 37 43 50 51	19 38 50 16 7 13
14	21 51 9 2 42 6	0.6249 0.4823 0.4453 0.2879 0.2671 0.2402	50	9 21 51	51 9 21
15	nd	nd	nd	nd	nd
16	50 19 27 29 5 3	0.8526 0.4840 0.3763 0.3377 0.3281 0.3278	7 13 18 19 20 50	16	19 80 18
17	21 2 42 9 51 6	0.6541 0.2516 0.2097 0.1422 0.1156 0.0904	7 9 11 13 16 18 19 21 30 50	9 21 51	21
18	22 11 13 7 16 38	0.7692 0.5792 0.4750 0.4519 0.3908 0.1634	4 7 9 11 13 14 16 20 22 23 24 26 27 28 29 30 33 35 36 39 41 46 47 50 51	6 7 12 13 16 17 19 21 23 25 27 28 30 32 33 35 36 37 38 41 42 43 45 46 48 51 52	43 51 46
19	5 46 34 41 6 30	0.5828 0.4188 0.3129 0.3032 0.2987 0.2787	nd	5	5
20	46 7 13 22 42 11	0.6063 0.2564 0.2514 0.249 0.2288 0.2249	7 11 13 16 18 20 13 50	20 46	46
21	8 16 19 25 50 33	0.7612 0.3623 0.3516 0.3276 0.3139 0.3003	nd	nd	8

nd = not detected