Article citation info:
Benbrahim M, Hamaidi B, Haouassi H, Mahdaoui R, Mouss L-H. Explainable fault detection and diagnosis based on an IDEOA as applied to an industrial process. Diagnostyka. 2025;26(2):2025203. https://doi.org/10.29354/diag/203246.

DIAGNOSTYKA, 2025, Vol. 26, No. 2

e-ISSN 2449-5220
DOI: 10.29354/diag/203246

EXPLAINABLE FAULT DETECTION AND DIAGNOSIS BASED ON AN IDEOA AS APPLIED TO AN INDUSTRIAL PROCESS

Meriem BENBRAHIM [1, *], Brahim HAMAIDI [1], Hichem HAOUASSI [2], Rafik MAHDAOUI [2], Leila-Hayet MOUSS [3]

[1] LGEM, Electromechanical Engineering Laboratory, Electromechanics Department, Faculty of Technology, Badji Mokhtar-Annaba University, B.P. 12, Annaba, 23000, Algeria
[2] ICOSI Lab, Mathematics and Computer Science Department, Faculty of Science and Technology, Abbas Laghrour University, Khenchela, 40000, Algeria;
[3] Automation and Manufacturing Engineering Laboratory, Industrial Engineering Department, Faculty of Technology, Batna2 University, Batna, 05000, Algeria
* Corresponding author, e-mail: meriem.benbrahim@outlook.fr

Abstract

Even with all measures approved by industrial sector specialists to avoid faults leading to major accidents, this field still suffers from some issues. Therefore, the safety and reliability of these industrial systems become necessary, leading to focus more on anticipating fault occurrence by giving fault detection and diagnosis a high priority. To solve this problem, a large set of reliable methods has been developed. Machine learning-based methods have gained significant importance as they have achieved promising results. However, the black-box nature of the generated fault detection models has restricted their investigation by users. Thus, explainable models aim to show features that influence the detection model decision. In this study, an Improved Discrete Equilibrium Optimizer Algorithm (IDEOA), which aims to solve different discrete optimization problems, was proposed to generate a rule-based fault detection model easily explainable by reading its classification rules. To this end, the Opposition-Based Learning (OBL) strategy is adopted in the IDEOA to avoid being stuck in local optima. A key contribution of this study is the novel application of the methodology to the Tennessee Eastman Process. The result of this study is a fault diagnosis model that consists of 16 rules, six of them belong to normal operating conditions and the rest reveal fault occurrence (F4). Then, an accuracy value is calculated to assess the effectiveness of our approach by contrasting it with other algorithms described in the literature. The findings indicate that the proposed approach outperforms other methods.

Keywords: fault detection, rule-based classification, Improved Discrete Equilibrium Optimizer Algorithm, interpretable rules

List of Acronyms

Cr – Classification rule
DL – Deep Learning
FDD – Fault Detection and Diagnosis
IDEOA – Improved Discrete Equilibrium Optimizer Algorithm;
ML – Machine Learning
OBL – Opposite Based Learning;
TEP – Tenese Eastman Process;

1. INTRODUCTION

With industrial technical progress and development, the main challenge in fault detection and diagnosis in industrial systems is the increasing number of accidents and incidents due to system failures, despite all precautions and safety improvements. Other challenges include the complexity and nonlinearity of industrial systems, the high dimensionality and heterogeneity of data, the lack of labeled data for training and testing models, and the need for real-time and online fault detection and diagnosis. Additionally, the reliability and interpretability of the models used for FDD are also important challenges that need to be addressed.

In the last decades, fault diagnosis, which is an operation of detecting and locating faults; has become an unavoidable problem for companies. Firstly, it can help identify and locate faulty elements in the system, which can prevent further damage and reduce the risk of accidents and incidents [1].

Secondly, FDD can provide early warning of potential faults, which can allow for proactive maintenance and repair, reducing downtime and increasing availability [2]. Moreover, FDD can help analyze the consequence of a failure in the global system, which can inform decision-making and improve the overall performance of the system. As it

Received 2024-05-25; Accepted 2025-03-22; Available online 2025-03-26

can contribute to the competitiveness of production tools by improving the reliability, availability, and productivity of industrial systems [3]. This diagnosis tool has become a subject of many researches that created a growing interest among industrialists, which has led to numerous research works. The literature has mentioned several methods being developed in this field. These methods are classified into three categories, depending on the type of used information: symbolic, model-based methods, and data-based methods [4,5]- [6]. Examples include SADT, PN (Petri Nets), Parity Space, neural networks (NN), and pattern recognition methods.

Unlike other types of methods, data-driven methods or methods based on Machine Learning (ML) algorithms are widely employed in this field. The main advantage of using ML-based methods for FDD in industrial systems is that they can build a fault diagnosis model using a set of data that is being collected without using the system model [4,5]. This means that these methods can be applied to a wide range of industrial systems without requiring a detailed understanding of the system's physics or dynamics. Additionally, ML-based methods can handle high-dimensional and heterogeneous data [7], and can learn complex patterns and relationships between variables that may not be easily detectable by human experts.

However, the main disadvantage of machine learning-based methods is that they often generate black-box models, such as Neural Networks (NN) and Support Vector Machines (SVM). Which means that their internal process for making classification decisions cannot be easily interpreted by the user [6]. This can make it difficult to understand the reasons behind the model's predictions and to identify the root cause of a fault. In addition, ML-based methods require a large amount of labelled data for training and testing, which may not always be available in industrial settings. Finally, ML-based methods may be computationally expensive and may require significant computational resources to train and deploy. On the other hand, methods that produce white-box models [6], among them, we can cite Rule-Based classification methods, that can provide a clear and concise representation of the decision-making process, which can help users understand the reasons behind the model's predictions and identify the root cause of a fault. Additionally, rule-based models can be easily updated and modifies based on new data or changes in the system, which can improve their adaptability and robustness.

Another way to improve the explainability of FDD models is to use visualization techniques to represent the model's internal workings and decision-making process. Visualization techniques can help users understand the relationships between variables and the factors that influence the model's predictions. Finally, it is also important to involve domain experts in the development and validation of FDD models, as their knowledge and expertise can help ensure that the models are accurate, reliable, and interpretable. In practice, the experts of the industrial domain use the rules generated by the diagnosis system, by translating them into concrete actions, in order to enhance reliability, safety and efficiency of industrial systems. These concrete actions can be presented in the integration of new technologies such as captors and sensors…to prevent fault occurrence and to plan for predictive maintenance by monitoring equipements, For example; when a subnormal variable exceeds a certain threshold then sensors can trigger an alert to take actions. These generated rules allow also, a better planning for interventions when the diagnosed indicators indicate a real risk, which reduces unnecessary costs. In addition, to prevent fault occurrence, domain experts used to insert surveillance systems to detect anomalies as soon as they appear, also the generated rules can be inserted into a software or into an automation system to make fast decisions in case of a danger. For example, an unusual increase of temperature may signal an imminent fault, so, the domain experts act as fast as possible to reduce this temperature.

In this context, to overcome optimization problems researchers have developed several meta-heuristic methods. A novel meta-heuristic method called the Equilibrium Optimizer Algorithm (EOA) [8] which was simulated by physics, based on the principle of control volume equilibrium phenomenon, and used to estimate equilibrium states to find optimal solutions; was recently proposed by Faramarzi et al. [8] to deal with continuous optimization issues. The fundamental achievements of this study are at first a discretization of data attributes using Weka tool, to resolve the fault diagnosis issue, which is considered as a discrete problem. Then, the discrete version of the Equilibrium Optimizer Algorithm (DEOA) elucidated at (MALIK & HAOUASSI) [9], is improved by introducing an OBL technique to exploit the research space by providing promising solutions and to escape from being trapped in local optima. At second, we have developed a fault detection and diagnosis model using the IDEOA and a rule-based classification approach, to generate set of classifiers, and each classifier includes a set of reduced rules. This approach makes the model more explainable and interpretable for users, as they can easily read the classification rules and understand the features that influence the detection model decision. Additionally, the IDEOA is used to solve different discrete optimization problems and generate explainable fault detection models. Lastly, to validate the performance of the proposed approach in our manuscript, we used to apply our study for the chemical process "Tennessee Eastman plant" TEP, which is a well-known industrial process. The TEP is considered as a realistic, well defined system and an ideal benchmark process that has been widely used for testing and developing several process monitoring and fault detection algorithms in the literature. The obtained results in this paper are compared with the findings of other techniques. In the next section, we will present a brief history on diagnostic methods in industrial contexts, and how did the EOA fit into the evolution of earlier methods.

This paper is structured as follows. In literature review section, we present a survey of relevant similar studies on the TEP fault diagnosis; then a presentation of some EOA associated works is given. In background section, we give a description of the rule based classification approach and the discrete version of the EOA. The proposed methodology of fault detection and diagnosis based on RC and the IDEOA is presented in section 4, with a brief system description to be exploited in this paper. We provide an evaluation of the obtained results in experiments and results sections, then we discuss the results of the suggested approach and we compare the acquired results with respect to other works. Finally, the last section gives a conclusion to our work.

2. LITERATURE REVIEW

Over the past few years, numerous authors focused on improving the performance of industrial diagnosis of the TEP by applying a wide variety of data-based methods to identify the pertinent faults that can lead to partial or total shutdown of the system. In this section, we will give a brief history on the diagnostic methods used to detect faults in the TEP, and then we used to mention how did the EOA fit into the evolution of earlier methods.

2.1. Fault diagnosis of TEP

Some of the most known works mentioned in the literature that dealing with fault diagnosis in the chemical industries are as follows. Zou et al. [10]; proposed a new technique for recovering faults in the (TEP), using a new algorithm of neural network to build ELM network architecture, which is more compact than I-ELM, OI-ELM, and B-ELM with a fast convergence rate. The results of comparison of this new approach with other algorithms, shows it's superiority in term of accuracy. Onel et al. [11]; combined feature selection with nonlinear Kernel-dependent SVM classification algorithm to offer an accurate FDD model in continuous process, to minimise the loose of information, and enable simultaneous modeling and feature selection. The results of this study are highly promising not only in terms of detection accuracy but also detection latency. In another work; Soraya et al., successfully applied the Principal Component Analysis (PCA) method in process control to monitor large-scale plants, using numerous types of controlling statistics such as the squared prediction error (SPE), also recognized as Q statistic. This statistic was used to indicate faulty situation without giving any information about its source. For this reason, contribution charts (complete and partial decomposition contribution CDC and PDC) which are a very known PCA tool were employed to isolate faults [12]. It is worth noting that PCA method cannot deal with nonlinear or multimodal characteristics of industrial data. Therefore, Zhang et al. [13] proposed a fault detection method based on the K-nearest neighbors (Diff-PCA), to remedy the limitations of PCA that has a lower fault detection rate in industrial processes. To overcome the problem, Han et al. proposed another approach for fault diagnosis of the TEP by using an iterative method of neural network to define the number of hidden layer nodes [14]. For Deep Learning FDD techniques, they have been widely applied to a large variety of engineering fields [15]-[16]. In this context; Lomov et al. developed an advanced approach for early detection and prediction of faults on extended TEP datasets, by investigating a variety range of recurring and convolutional architectural designs to select the optimal architecture [17]. They also proposed to use neural networks to boost and improve information utilized in their research. Furthermore, to identify TEP problems, sample features are initially simplified using a ML and data-driven classification (SVM) technique in the work of Xu et al. [18] to validate the engineering application of the suggested method. Where authors used the PNN probabilistic neural networks as the principal classification tool for fault diagnostic model, based on feature selection and a bio heuristic optimizer to optimize the hidden smoothing factor (σ) of PNN to improve its classification performance. In another study, Hu et al. [19] to enhance the fault classifier's accuracy, authors proposed a deep learning-based fault diagnosis method for chemical processes. Combining feature selection and KELM classifier as an optimization strategy. The experimental results given in this study verify the applicability and effectiveness of the model.

Note that in the above-proposed strategies used to detect faults in TEP such as neural networks and SVM, and PCA, they require large amounts of data to be effective, which increase their computational complexity with high training time especially for DL architectures. At last, the most important that they are considered as black boxes methods, which makes the interpretability of their decision very difficult. Aiming to these limitations, a FDD approach using a novel meta-heuristic algorithm (IDEOA) was proposed in this study.

2.2. Meta-heuristic methods and EOA applications

With the development of computer tools and the complexity to solve optimization problems, many optimization methods have emerged. These methods may be split into two principal classes: exact and stochastic methods. In this study, we are interested in the second category and more particularly in meta-heuristic methods [20]. These methods are algorithms that aim to solve difficult optimization problems, and find the best solution that converges to a global optimum. In fact, we can find the optimization problems in many applications such as the design of new systems (dimensioning), optimization of process operating conditions, system control (stabilization, trajectory tracking), production planning, transportation, localization, also in monitoring and supervision (maintenance, diagnosis problems), and in many other fields.

Most of meta-heuristic algorithms are inspired by nature [20]: physics like Simulated Annealing (SA) method, biology such as genetic algorithms, and even animal behaviour like the ant colony. In literature, several meta-heuristic algorithms have been mentioned: starting from Genetic Algorithms (GA) inspired by the principle of natural evolution based on ideas that are derived from population genetics. Chen et al. [21] were the first to propose the use of GA in the fault diagnosis field. Many other algorithms have been modified to cope with fault diagnosis problem. We can find the Simulated Annealing method (SA) presented by Kirkpatrick et al in 1983; that was extended to the case of fault diagnosis by Wicaksana in 1992. There are also many other methods developed to solve various issues; such as; Ant Colony Optimization (ACO), Grey Wolf Optimizer (GWO), Estimation of distribution Algorithm (EDA) and Particle Swarm Optimization (PSO).

A novel optimization algorithm has been recently introduced in the field of optimization, known as "Equilibrium Optimizer" (EO), this algorithm was first suggested by Faramarzi et al. [8], where it relies on the principles of physics and draws inspiration from control volume mass balance models employed to evaluate both dynamic and steady states. The suggested method explores a dynamic mass balance approach that considers both inputs and outputs within a control volume. To determine the concentration of a non-reactive constituent in the control volume, a mass balance equation takes into account various involved sources and sinks. The EOA has demonstrated outstanding effectiveness and efficiency in achieving optimal or nearly optimal solutions in comparison to other algorithms currently available. Subsequently, Abdel-Basset et al. [22] tried to enhance the EOA by employing a method for reducing linearity diversity (LRD) and eliminating local minima (MEM). To evaluate the effectiveness of this suggested algorithm, a comparative analysis was conducted on a set of reliable algorithms, which were implemented on R.T.C France's commercial solar cells. Another study, and in the same axis of work as that of Abdel-Basset et al. was carried out. The primary goal of this study was to estimate the parameters of three distinct models of photovoltaic cell. In contrast to the initial EO, the IEO utilized by Wang et al. [23] in their investigation to solve parameter identification, utilizing a back-propagation NN to anticipate a greater amount of PV cell output data. As a result, it is capable of executing a more effective optimization process by utilizing a more logical fitness function. In their study, Too & Mirjalili, [24] introduced a novel iteration of the EO, to address the challenge of feature selection in the classification of biological data. In a further endeavour to address the feature selection issue through the latest algorithm mentioned in reference [25]; a transfer function with a V-shape was utilized to convert the continuous values generated in "EO" into a binary search area. Next, the Simulated Annealing (SA) technique was employed for improving the exploitation of the (BEO). The outcomes of the suggested methodology demonstrate a high competitive performance for solving feature selection problems.

To enhance the precision of optimization in EO, increase its exploitative abilities, and escape from getting stuck in limited optima, Fan et al. [26], proposed an adapted version of EO called (M-EO). This version incorporates Opposition-Based Learning (OBL) technique and novel update regulations. Conversely to the (EOA) proposed by Faramarzi et al. [8], that exclusively deals with continuous optimization issues, However, in [9] the authors suggested a discrete version to resolve the discrete optimization problem of rule-based classification. By combining the associative classification and intelligence-based population methodologies and introducing new discrete operators to prevent local solutions and reach global ones that improve the power of exploration and exploitation capabilities in search space.

In other works, the EOA proposed by [8] has shown its superiority over other conventional meta-heuristic algorithms. To our knowledge, the optimization algorithm (EOA) has never been adapted to the problem of detecting and diagnosing industrial failures. This motivated us to apply this novel algorithm for the detection and diagnosis of the industrial and the chemical Tenesse Eastman (TEP) faults. In this paper, we have used the discrete version (DEOA), because the original algorithm is applicable only to continuous problems. In the following section, we will give more details of the chosen method.

3. BACKGOUND

In this section, we tried to present both artificial intelligence methodologies: rule based classification and DEOA; used in this study to extract necessary informations from data system, in order to generate an explainable fault diagnosis model, and for facilitate the decision-making.

3.1. Rule based classification

With the scientific progress and technological development, the size and dimension of datasets are constantly increasing; hence, the complexity of processing this huge amount of data for decision-making, which becomes a major concern in many domains, especially in the diagnosis field. Several techniques have emerged for processing and extracting information from these data such as association, clustering, regression, and classification rules [6]. We used in this paper, the Rule-Based classification that consists in organizing big sizes of data to provide contrary to other classical techniques, highly interpretable results making easy the exploitation of the obtained classification. Rule based-classification is a very well-known technique and widely used, it was first introduced in 1989 by Liu et al. [27]. This classification technique encompasses two steps: a learning step (training) and a classification step (use). In the learning stage, a classifier which is a set of rules defined as: C = {R1, R2,...Rn}, is constructed by comprehending the training data. Every classifier rule includes an antecedent and a resultant. The resultant part portrays the category of attributes of the Cr [28]. The structure of a Cr is shown in the illustration below:

$$\text{If} = \underbrace{\overbrace{(\text{D=O})}^{\text{Term}} \text{ and If (E=P) and If (F=X)}}_{\text{Antecedent}} \text{ then } \underbrace{\text{Class = Y}}_{\text{Resultant}}$$

3.2. Overview of the Discrete Equilibrium Optimizer Algorithm

The EOA is an innovative meta-heuristic algorithm suggested by Faramarzi et al. [8]. This new algorithm draws inspiration from physics and relies on dynamic source and sink models, it is a powerful tool that has shown its superiority to solve continuous optimization problems. In other respects, FDD investigated in this study is considered as a discrete problem. Therefore, we propose to use the IDEOA for TEP diagnostics, which shows a good performance to provide adaptive, interpretable and computationally efficient solution for decision-making in diagnostics. Its ability to enhance both the precision and the explanation of diagnostic model, and handling high-dimensional data with avoiding being trapping in local optima makes it a good choice for this study. In follows, we consider to describe some outlines of the DEOA introduced in [6].

Authors of the DEOA suggested the following new discrete operators {©x%, ⊘, ⊖ and ⊗} to adapt the particle's position update equation to discrete problems. Then, the update position is defined as in Equation (1).

$$C_{i+1} = C_{eq} \copyright_{50\%} ((G \oslash \lambda) \copyright_{F\%} ((C_i \ominus C_{eq})) \quad (1)$$

with C_i and C_{i+1} represent the current and the new particle position vectors respectively, C_{eq} is the equilibrium position selected arbitrary from the equilibrium pool vector. The Equilibrium pool vector is composed of the first fourth better particles in the population. λ is a stochastic discrete vector, and the parameter G_{it} denotes the rate of generation that can be mathematically represented as shown in Equation (2) given in [6].

$$G^{it} = \begin{cases} {G_0}^{it} & \text{if } r > F/100 \\ \left|1 - {G_0}^{it}\right| & \text{else} \end{cases} \quad (2)$$

Here, the notation |.| denotes the absolute value; F indicates the exponential parameter calculated using Eq. (4). r a variable that signifies an arbitrary number between [0,1] and G0 it is calculated as in Equation (3) from [6].

$${G_0}^{it} = {X_{eq}}^{it} \ominus (\lambda \otimes {X_i}^{it}) \quad (3)$$

$$F_{it} = f(it) = \frac{(N-it)^2}{N^2} \times 100 \quad (4)$$

Here, N denotes the extreme number of iterations, and it indicates the actual iteration of the DEOA.

The symbols $\copyright_{(x\%)}$, ⊘, ⊖ and ⊗ in Equations. (1) and (3) represent discrete operators that operate on discrete vectors [6], and their definitions are as follows:

3.2.1 The ©50% operator

This operator is used to combine two discrete vectors in order to create two new additional discrete vectors. For obtaining the first one, we have to pick the first segment from the first vector and 50% last segments from the other one. Then, to get the second vector, we have to choose the second half segment from the first one and the first half segment from the other one [6].

3.2.2 Discrete subtraction (⊖) operator

$$X3 = X1 \ominus X2 = \begin{cases} X1 & \text{if } X1[1] = 1 \text{ and } X2[1] = 0 \\ X3[1] = 0 \text{ and } X3[2] = X1 \text{or } X2 \text{ randomly} & \text{if } X1[1] = X2[1] \\ X1 \text{ or } X2 \text{ randomly} & \text{if } X1[1] = 0 \text{ and } X2[1] = 1 \end{cases} \quad (5)$$

3.2.3 Discrete division (⊘) operator

$$X3 = X1 \oslash X2 = \begin{cases} X1 & \text{if } X1[1] = 0 \\ X2 & \text{if } X1[1] = 1 \text{ and } X2[1] = 1 \\ X1 \text{ or } X2 \text{ randomly} & \text{if } X1[1] = 1 \text{ and } X2[1] = 0 \end{cases} \quad (6)$$

3.2.4 Discrete multiplication (⊗) operator

$$X3 = X1 \otimes X2 = \begin{cases} X1 & \text{if } X1[1] = 0 \\ X2 & \text{if } X2[1] = 0 \\ X1 \text{ or } X2 \text{ randomly} & \text{if } X1[1] \neq 0 \text{ and } X2[1] \neq 0 \end{cases} \quad (7)$$

where, X1, X2, and X3 are discrete vectors having the structure provided in Fig. 2

The pseudo-code of Algorithm 1 depicts the particulars of the DEO algorithm. At first, all particles are initialized randomly, then the particles are displaced in the search space using the update function defined in Equation (1) and presented in [8].

4. METHODOLOGY

In this section, industrial fault detection and diagnosis is carried out through the implementation of Artificial Intelligence techniques, an IDEOA and rule-based classification are developed in this study for extracting necessary information from system Data to give a fault diagnosis model and especially for the decision-making, which is a crucial step in this field. In this work, based on TEP dataset, a classification model is constructed in order to detect a faulty situation that may threaten the system. The optimization method used here is an IDEOA dedicated for generating an explainable fault diagnosis model. The whole approach used in this work is illustrated in Algorithm 2.

Algorithm 1. Improved Discrete Equilibrium Optimizer Algorithm (IDEOA)

Input: Training DataSet (TS), Nb_ Particles, Max_iter
Output: A classification rule
{
    Arbitrary initiate all particle
    Calculate the fitness value of particles by utilizing Equation (9)
    For (t = 1 to Max_iter)
    {
    Select best particles: C_{eq1}, C_{eq2}, C_{eq3} and C_{eq4}
    $C_{eq\ avg}$ = the average of C_{eq1}, C_{eq2}, C_{eq3} and C_{eq4}
    C_{eq} = [C_{eq1}, C_{eq2}, C_{eq3}, C_{eq4}, C_{eqavg}]
        For (i = 1 to Nb_Particles)
        {   Select one vector from C_{eq}.
            Calculte the random vector λ.
            Generate the vector F using Equation (4)
            Generate the vector G_0^{it} using Equation (3)
            Generate the vector G^{it} using Equation (2)
            Generate the new particle's position using Equation (1)
        }
    Verify whether the new generated position is feasible.
    Calculate the fitness value of the new generated position utilizing Equation (9)
    }
    Returning the finest position as considered as the best rule.
}.

Algorithm 2. TEP fault diagnosis model generation methodology
Input: TEP Dataset
Output: Evaluation result of the fault diagnosis model
Begin
    Pre-treat the TEP Dataset
    Split TEP Dataset into Training Data and Testing Data
    If Dataset = Training data
    Generate the rule-based fault diagnosis model using the optimization method
    Evaluate the fault diagnosis model
    Else
        Dataset = Testing Dataset
        Evaluate the fault diagnosis model
    End if
    Give the result of the fault diagnosis model evaluation
End.

4.1. Suggested FDD approach

The rule-based methodology used in this research for FDD model generation is comprised of two principal parts as represented in Fig. 1. The first part implies the pre-treating of the TEP dataset. While, the second part consists of using of the IDEOA to generate rule discovery problem for FDD of the studied system.

4.1.1. Data-set Pretreating

In the first part, a discretization algorithm is applied to the variables of the TEP dataset, for the reason that rule-based classification method cannot be applied for numerical data and works only on categorical ones. Therefore, in this study, we used entropy-based discretization method from the Weka tool to transform continuous features to discrete intervals [29], found in [30]. This method divides continuous values in a way as to minimize the entropy in each interval, which ensures that each interval contains homogeneous classes, thus maximizing the preserved information. The key parameters of the entropy-based discretization method were configured to fine-tune the discretization process. The -B parameter, which controls the number of bins (intervals or categorical values), was set to 3. Additionally, the -M parameter, which specifies the minimum weight of instances per bin, was set to -1. We enable also the equal-frequency binning parameter –F that forces bins to contain equal number of instances. Table 2 describes the discretized values of all features of the TEP dataset using the entropy-based discretization method. Each feature is transformed to three intervals (Rank1, Rank2 and Rank3). For example, the feature ES1 was discretized to the following three intervals: interval1 =]-∞, 0.195023], interval2 = [0.195023-0.249177] and interval3 = [0.249177, + ∞]. The obtained results are represented in the last column of Table 2.

4.1.2. Rule generation

In the second part, rule generation that aims to produce a set of classification rules, resulting a classification model that enhances both the precision and the explanation of the diagnosis model.

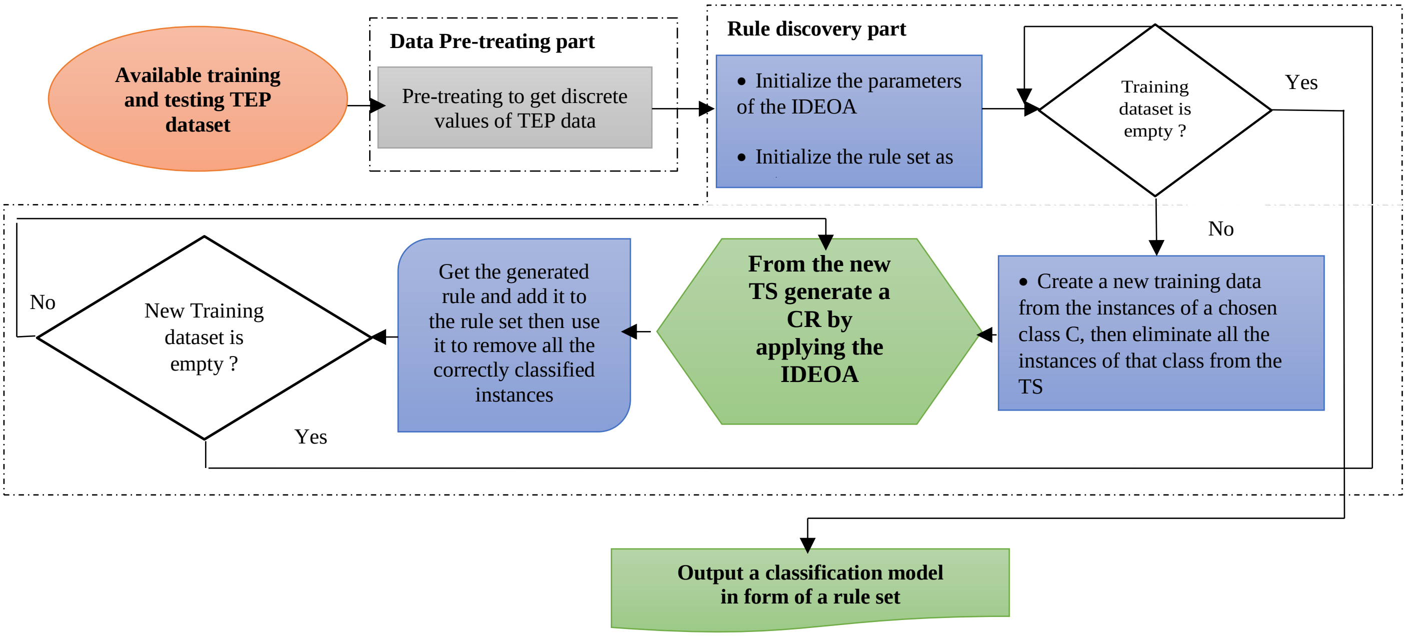


Fig. 1. Flowchart of the rule-based methodology used for fault detection and diagnosis

Therefore, the suggested IDEO is called at each iteration to create one classifier from the pre-treated dataset.

The process of the second part is unfolds over multiple iterations while the original training dataset remains not empty. It begins by an empty set of rules and initializes the parameters of the IDEOA algorithm. The IDEOA is used to generate classification rules. At each iteration, the class C with the highest number of instances is selected, and all instances of this class are transferred to a new training dataset, which consists solely of instances belonging to the chosen class. Subsequently, these instances are removed from the original training dataset. The IDEOA algorithm is then iteratively executed to generate a classification rule from the new training dataset while it remains not empty. Once a new rule is generated, it is applied to classify the instances in the new training dataset. All correctly classified instances are then removed from the new training dataset, and the rule is added to the rule set. This process stills continuous until the new training dataset is empty. The final output of this method is a rule-based classifier, which is subsequently used to classify new instances.

4.2. IDEOA for rule generation using the OBL strategy

EOA and DEOA have shown good performance on several optimization problems [25]. As all meta-heuristic based optimization techniques, exploration and exploitation are the two main strategies influencing the search ability of the algorithm [29]. In our experimental process, we found that DEOA has a good exploitation of the search space, but it falls frequently into local optima and cannot find well exploration in the search space to obtain good results.

Recently, several population-based optimization algorithms explored the use of OBL technique to improve their convergence rate [31]-[32]. Therefore, we consider use this technique (OBL) mentioned in [30] for the IDEOA algorithm, to avoid being trapped in local optima and to escape from this premature convergence. In this section, we focus on using the OBL strategy into the discrete equilibrium optimization meta-heuristic method for modifying its global search (search space exploration). Thus, DEOA with OBL can explore more search areas which directs it to converge quickly to the optimal solution. So, a new improved DEOA called IDEOA is proposed by adopting the OBL strategy in the DEOA.

The OBL technique is used in the IDEOA in two steps: the initial population and when the algorithm convergences to a local optimum.

4.2.1. Particle's position in IDEOA and OBL

The OBL strategy is called in IDEOA for the particle's diversity improvement and it is defined as follows. Let a particle p structured as a 2 N-dimensions vector [9], the initial vector is a binary one that indicates the pertinent features in the classification rule, while the second one denotes the value of every chosen vector [6]. In this work, we propose to generate only the opposite of the first vector using Equation (8), while the second vector is preserved without modification because it is a discrete vector.

$$Opposite\ (p) = op = \begin{cases} op[1,j] = 1 - p[1,j] \\ op[2,j] = p[2,j] \end{cases} \qquad (8)$$

In Fig. 2. An example of a particle and its opposite is depicted:

Original particle's position p

1	0	0	1	1	0	1
3	4	2	1	5	3	4

Opposite of p	0	1	1	0	0	1	0
	3	4	2	1	5	3	4

Fig. 2. Opposite of a particle's position

4.2.2. Fitness function

The following fitness function is employed to assess the quality of each produced solution (classification rule).

$$Obj\_Func(R) = \frac{Number\ of\ covered\ examples}{Total\ number\ of\ examples} \quad (9)$$

4.2.3. IDEOA's initialization strategy

IDEOA generates firstly the initial population P arbitrary, after that, the opposite population OP is calculated from P using the OBL strategy as defined in Equation (8) for each particle. Then, the best particles from P and OP are chosen, and considered as initial population of the algorithm.

4.2.4. IDEOA's prevention of stagnation in local optimum

When the particles of the population in the original DEOA converge to a local optimum, they cannot change the search direction. This is noticed when the obtained result did not improve during a fixed number of iterations. However, in this case the particles should displace to another search area to find a better result. Thus, an adaptive search mechanism is proposed when the particles are trapped in local optima. We check after each running of the algorithm if there is a number of iterations (predefined threshold) without improvement of the obtained result; the algorithm concludes that there is a trapping situation within the local optima and called the OBL to 50% of the population and a random re-initialization to the rest of the particles.

4.2.5. Outlines of the IDEOA

The particulars of the IDEOA are illustrated in Algorithm 3.

ALGORITHM 3. Improved Discrete Equilibrium Optimization Algorithm (IDEOA)

INPUT: DataSet (TS), N, Max_iter , MT. // MT means the maximum number of iterations without improvement
OUTPUT: Classification rule
{
Randomly generate the population P
Generate the opposite population OP by calculating the opposite of each particle in P using the Equation (8).
Evaluate the fitness of all particles of P and OP using Equation (9)
From P and OP choose N best particles as initial population.
Fit = fitness of the best particle
K=0
  For (it = 1 to it_{max})
    If K< MT then
      Generate the C_{eq1}, C_{eq2}, C_{eq3} and C_{eq4} as the four best particles
        If fitness (C_{eq1}) ≤ Fit
          K=K+1
        Else
          K=0
        End If
          Calculate C_{eqavg} as the average of C_{eq1}, C_{eq2}, C_{eq3} and C_{eq4}
          Ceq = [C_{eq1}, Ce_{q2}, C_{eq3}, C_{eq4}, C_{eqavg}]
          For (i = 1 to N)
            From C_{eq} choose one candidate.
            Arbitrary calculate the vector λ.
            Using Eq. (4) calculate the vector F
            Using Eq. (3) calculate the vector $G0^{it}$
            Using Eq. (2) calculate the vector G^{it}
            Using Eq. (1) generate the new particle's position
          End For
    Else

          For (i = 1 to N)
              Using Eq. (8) generate the new position // call the OBL strategy
          End For
    End If
          Feasibility verification of the novel generated position.
          Fitness value calculation of the novel generated position utilizing Eq. (9)
  End For
Returning the finest rule as considered as the best fitness of the particle position.
}.

4.3. Case study system description

4.3.1. Presentation of the TEP

Our case study system (TEP) is a realistic industrial complex problem and is considered as an extensively used benchmark problem in process engineering for testing a large scale of process control and monitoring technologies. The original proposal of this simulated chemical process model was made by Downs and Vogel in (1993) [33], which includes five fundamental units: a product condenser, an exothermic two-phase reactor, a separator, a stripper, and a recycle compressor as shown in Fig. 3. The process involves four gaseous reactants (A, C, D, and E) and inert B which are supplied to the reactor in order to create 2components (G and H), and the undesired byproduct F. The reaction equations are listed in (10). All of the reactions are exothermic, irreversible, and roughly first-order in terms of reactant concentrations. [34].

$$\begin{aligned} &Ag + Cg + Dg \rightarrow G(liq) \\ &Ag + Cg + Eg \rightarrow H(liq) \qquad (10) \\ &Ag + Eg \rightarrow Fl \\ &3Dg \rightarrow 2Fl \end{aligned}$$

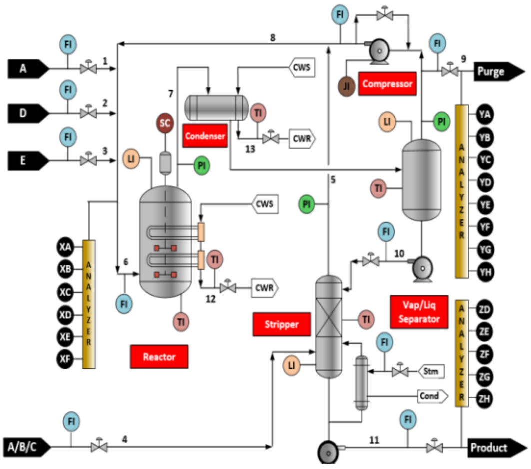


Fig. 3. Flowchart of the TEP [33]

In this study, we use the large sample TEP-data obtained from the webpage: http://web.mit.edu/braatzgroup/links.html.

Whereas, the process contains 53 variables: 41 are measured variables (ES1–ES41), among them 22 are continuous measured variables while the rest are compositions sampled measured variables. The last 12 ones are considered as manipulation variables (EV1–EV12), where, the agitation speed (EV12) is eliminated. Although, 21 pre-programmed faults (F1-F21) are generated by the TEP simulator, due to various disturbances, to mimic various situations of the process operation as indicated in Table 1. Among these faults; 16 are known faults, while the rest are unrecognized. With faults from F1 to F7 are involved with a sudden variation in the process variable. While, F8 to F12 are related to an arbitrary change of some process variables, F13 is a delayed drifts in the reaction kinetics and faults F14,F15 are related to blocking valves [35].

According to the system description, and the data issued from the system fault analysis found in the literature, the system experts concluded that the criticality of F4 is very high with a major impact, which effects critically the performance of the TEP. Hence, the neglect of this effect can be disastrous on the functioning of the system, what makes F4 very pertinent for our study.

Table 1. Process faults for the TEP simulator [35].

Faults	Description	Type
F1	A/C Feeding ratio, B composition constant	Sudden variation
F2	B Constituent, A/C ratio constant	Sudden variation
F3	D Feeding temp	Sudden variation
F4	Inlet temp of the Reactor cooling water (RCW)	Sudden variation
F5	Inlet temp of the Condenser cooling water (CCW)	Sudden variation
F6	A Feeding loss	Sudden variation
F7	C Header pressure loss–reduced validity	Sudden variation
F8	A, B, C Feeding consituent	Arbitrary change
F9	D Feeding temp	Arbitrary change
F10	C Feeding temp	Arbitrary change
F11	Inlet temp of RCW	Arbitrary change
F12	Inlet temp of CCW	Arbitrary change
F13	Reactor kinetics	Slow drift
F14	RCW valve	Stuck
F15	CCW valve	Stuck
F16	/	-
F17	/	-
F18	/	-
F19	/	-
F20	/	-
F21	The stream 4 valve was adjusted to the stable position	Constant position

4.3.2. TEP data description

In this paper, we used the benchmark TEP dataset for generating a fault detection and diagnosis model. For this purpose, two types of data were used: Training data to construct the diagnosis model and testing data for validating the model. The number of total observations generated was 500 in the training data for normal operating conditions, with a simulation time of 25 hours for each run, but only 480 observations were used after fault occurrence with the normal data matrix was defined as $X \in R^{480*52}$. While, the overall number of observations in testing data was 960 with 48 hours as a measurement duration and a sampling period of three minutes. Therefore, the testing matrix data for each fault is given as $X \in R^{960*52}$. These training and testing data matrixes are consisting of 52 observation variables, excluding the agitation speed of the reactor's stirrer, which is eliminated because it is constant-valued as mentioned in sub-section 4.2.1. In this study, the system functioning is integral, and all the variables are dependent. For this, we have selected the 52 variables because all the previous studies found that these 52 variables are relevant and they neglected a significant number of variables, which are not found in the table. Moreover, the principal objective of our study is to generate a set of classification rules, these rules are characterized by their size and their number of rules. The size include the critical variables and exclude the irrelevant ones. Therefore, the results obtained give us only the variables that have a direct impact on F4. Twelve of these variables are manipulated variables and 41 are measured ones. Among these measurable variables: 22 are continuous measured variables and the rest are composition sampled and measured variables as specified in Table 2.

5. EXPERIMENTS AND RESULTS

This section aims to study the results of our approach and compare it with other well-known approaches on the TEP dataset. The dataset has been split into two partitions: 80% of instances are used for training, while 20% of instances are used for testing the rule-based diagnosis model. The proposed approach was developed and implemented using Weka tool and Java programming languages.

The obtained results are yielded in this section; at first, we used to calculate the accuracy function to determine the percentage of the classified instances; then we compare the obtained results in this paper with results of the DEOA version. In addition, rule number and rule size were calculated to elucidate the quality of our approach. At last, the importance value of each feature was calculated in order to choose the pertinent variable that affect fault F4 occurrence, and we attempt to compare our results with other benchmark algorithms results.

5.1. IDEOA evaluation and parameter setting

For evaluating the performance of our approach we calculate the commonly accuracy function that is considered as the important metric used in literature [9], and provided in Equation (11):

$$Accuracy = \frac{TN+TP}{TN+TP+FN+FP} \tag{11}$$

Other criteria were used in this study to assess the interpretability of the fault diagnosis model generated by our approach. These criteria include the number of rules, which defines the overall size of the classifier, and the rule size. A smaller number of rules generally indicates a more compact and efficient classifier, reducing computational complexity and enhancing generalization. Meanwhile, the size of individual rules determines the level of detail and specificity in fault classification, with shorter, more concise rules contributing to better readability and interpretability.

Additionally, the trade-off between classifier size and rule complexity plays a crucial role in evaluating model performance. A large number of complex rules may improve classification accuracy but could lead to reduced interpretability. Conversely, a smaller rule set may enhance efficiency but risks losing important classification details. Thus, achieving an optimal balance between these factors is essential for developing an effective and explainable fault diagnosis system. To further validate our approach, we compared these criteria with benchmark models to assess improvements in rule conciseness and classification accuracy.

5.1.1. DEOA and IDEOA comparison

In this sub-section, the classification accuracy is calculated to evaluate the performance of the used approach of this paper. This performance measure was also used to compare the results of DEOA to those of the IDEOA, in order to find which algorithm's parameters allow us to obtain the best accuracy. To get best values, the experiment was conducted for nine times; at each time we change the number of particles (50-100-200), and for each particle increase the number of iterations (10-50-100). The evaluation results are expressed in Table 3.

From Table3, it is clear that our proposed algorithm (IDEOA) achieves best values of accuracy in most cases; it scored the highest accuracy of 95.47% using values of 100 for the number of particles and 50 for the number of iterations. In the other hand, it is denoted that at each time we run the DEOA by increasing the number of particles and the number of iterations, we are trapped in the problem of stusking in the local optimum. So, the DEOA cannot improve the obtained results. Therefore, it is quite obvious that the utilization of the suggested algorithm IDEOA helps in reaching better results by escaping from local optima.

5.1.2. Generated rules by using IDEOA

As shown in Table 4 our proposed algorithm gives a classification model, which consists of 16 rules (6 rules belong to a normal operation of the system and the rest of rules shows the fault occurrence F4), with a mean number of terms of 6.62. These results reveal that the approach used in this study is very efficient in generating accurate and short rules, which make it an explainable approach. For example, according to the rule 11 in Table 4 a fault can occur when the Reactor Temperature is less than 120.43°C; the Purge Rate is between 0.33132 and 0.35293kscmh; the Product Sep Temp is between 79.953 and 80.334°C; the Separator Cooling Water Outlet Temp is less than 77.154667°C; Component A is between 32.089 and 32.526, and Component E is less or equal to 18.755667.

Table 2. TEP dataset variables description with their discrete values [36]

Variables	Type	Description	Ranks		
			Rank 1	Rank2	Rank3
ES1	Continuous Process Measurements	A Feeding	≤0.195023	[0.195023-0.249177]	≥0.249177
ES2		D Feeding	≤3633.5	[3633.5-3682.6]	≥3682.6
ES3		E Feeding	≤4458.366667	[4458.366667-4526.833333]	≥4526.833333
ES4		A and C Feeding	≤9.3008	[9.3008-9.473]	≥9.473
ES5		Flow Recycling	≤26.731	[26.731-27.13]	≥27.13
ES6		Rate of Reactor Feeding	≤42.118333	[42.118333-42.500667]	≥42.500667
ES7		Pressure of Reactor	≤2702.633333	[2702.633333-2712.866667]	≥2712.866667
ES8		Level of Reactor	≤74.772667	[74.772667-75.761333]	≥75.761333
ES9		Temp of Reactor	≤120.43	[120.43-120.51]	≥120.51
ES10		Purge Rate in	≤0.33132	[0.33132-0.35293]	≥0.35293
ES11		Temp of Product Separator	≤79.953	[79.953-80.334]	≥80.334
ES12		Level of Product Separator	≤49.165	[49.165-50.84]	≥50.84
ES13		Pressure of Product Separator	≤2631.066667	[2631.066667-2641.533333]	≥2641.533333
ES14		Underflow of Product Separator	≤23.8283	[23.828333-25.757667]	≥25.757667
ES15		Level of the Stripper	≤48.738	[48.738-50.327]	≥50.327
ES16		Pressure of the Stripper	≤3099.733333	[3099.733333-3109.366667]	≥3109.366667
ES17		Underflow of the Stripper	≤22.180333	[22.180333-23.321667]	≥23.321667
ES18		Temp of the Stripper	≤65.531333	[65.531333-65.937667]	≥65.937667
ES19		Steam Flow of the Stripper	≤226.816667	[226.816667-234.373333]	≥234.373333
ES20		Work of the Stripper	≤339.546667	[339.546667-341.323333]	≥341.323333
ES21		Outlet Temp of the Reactor Cooling Water	≤94.473333	[94.473333-94.712667]	≥94.712667
ES22		Outlet Temp of the Separator Cooling Water	≤77.154667	[77.154667-77.634333]	≥77.634333
ES23	Sampled Process Measurements	Component A	≤32.089	[32.089-32.526]	≥32.526
ES24		Component B	≤8.842033	[8.842033-9.031467]	≥9.031467
ES25		Component C	≤26.117333	[26.117333-26.567667]	≥26.567667
ES26		Component D	≤6.8213	[6.8213-6.9733]	≥6.9733
ES27		Component E	≤18.755667	[18.755667-19.242333]	≥19.242333
ES28		Component F	≤1.643167	[1.643167-1.681933]	≥1.681933
ES29		Component A	≤32.740333	[32.740333-33.227667]	≥33.227667
ES30		Component B	≤13.767667	[13.767667-13.946333]	≥13.946333
ES31		Component C	≤23.698333	[23.698333-24.338667]	≥24.338667
ES32		Component D	≤1.150613	[1.150613-1.326107]	≥1.326107
ES33		Component E	≤18.235333	[18.235333-18.799667]	≥18.799667
ES34		Component F	≤2.250567	[2.250567-2.297233]	≥2.297233
ES35		Constituent G	≤4.7867	[4.7867-4.8936]	≥4.8936
ES36		Component H	≤2.2186	[2.2186-2.3077]	≥2.3077
ES37		Component D	≤0.010041	[0.010041-0.02523]	≥0.02523
ES38		Component E	≤0.82287	[0.82287-0.83839]	≥0.83839
ES39		Component F	≤0.091435	[0.091435-0.104042]	≥0.104042
ES40		Component G	≤53.368667	[53.368667-54.195333]	≥54.195333
ES41		Component H	≤43.715667	[43.715667-44.310333]	≥44.310333
EV1	Manipulated Variables	D Feeding Flow	≤62.563	[62.563-63.682]	≥63.682
EV2		E Feeding Flow	≤53.429667	[53.429667-54.230333]	≥54.230333
EV3		A Feeding Flow	≤21.428	[21.428-25.55]	≥25.55
EV4		A and C Flow Feeding	≤60.4	[60.4-62.39]	≥62.39
EV5		Compressor Recycle Valve	≤21.471333	[21.471333-22.177667]	≥22.177667
EV6		Purge valve	≤39.244667	[39.244667-41.898333]	≥41.898333
EV7		Flow of the Separator Pot Liquid	≤35.642667	[35.642667-40.572333]	≥40.572333
EV8		Flow of the Stripper Liquid Product	≤43.613	[43.613-47.291]	≥47.291
EV9		Stripper Steam Valve	≤46.657667	[46.657667-48.549333]	≥48.549333
EV10		Flow of the Reactor Cooling Water	≤42.146	[42.146-44.772]	≥44.772
EV11		Flow of the Condenser Cooling Water	≤17.228667	[17.228667-19.929333]	≥19.929333
EV12	M% P	Agitator Speed	/	/	/

Table 3. DEOA and IDEOA classification accuracy comparison

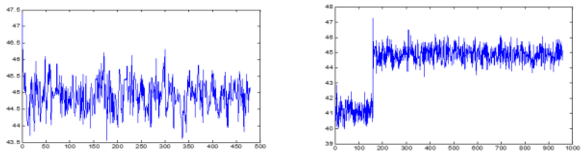
Method			**DEOA**			**IDEOA**		
No	# Particles	# Iterations	Accuracy %	# Rules	Mean size of rules	Accuracy %	# Rules	Mean size of rules
1	50	10	85.71	86	6.33	82.54	43	5.76
2	50	50	80.95	87	6.75	85.65	45	5.34
3	50	100	85.71	78	6.69	87.87	35	6.23
4	100	10	85.71	77	6.12	90.47	26	6.78
5	100	50	85.71	67	6.74	95.47	16	6.62
6	100	100	85.71	68	6.97	95.23	18	6.32
7	200	10	90.47	61	6.19	95.14	22	6.56
8	200	50	85.71	56	6.56	95.19	18	5.95
9	200	100	85.71	56	6.91	95.40	18	6.78

Table 4. Generated rules using TEP Dataset

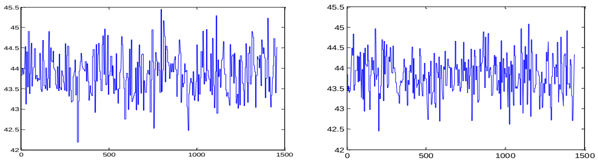
Rule No	Extracted rules	Class	Rule Accuracy %	# Terms
R1	(ES2 ≥3682.6) & (ES4≤ 9.3008) & (26.731≤ ES5 ≤ 27.13) & (ES13≤2631.066667) & (339.546667≤ ES20 ≤341.323333) & (23.698333≤ES31 ≤ 24.338667)	C0	10	6
R2	(74.772667≤ ES8 ≤ 75.761333) & (ES20 ≥ 341.323333) & (32.089≤ES23 ≤32.526) & (23.698333≤ES31≤24.338667) & (2.250567≤ES34≤ 2.297233) & (EV10 ≤42.146)	C0	10	6
R3	(ES3≥4526.833333) & (65.531333≤ES18≤65.937667) & (ES20≥341.323333) & (23.698333≤ES31≤24.338667) & (ES36≥2.3077) & (21.428 ≤ EV3 ≤ 21.428)	C0	5	6
R4	(0.195023≤ES1≤0.249177) & (79.953≤ES11≤80.334) & (26.117333≤ ES25≤26.567667) & (ES33 ≥18.799667) & (0.091435≤ES39≤ 0.104042) & (35.642667≤ EV7 ≤40.572333) & (EV8≥47.291) & (EV10 ≤42.146)	C0	10	8
R5	(2702.633333≤ ES7 ≤ 2712.866667) & (ES27≤ 18.755667) & (1.643167≤ ES28 ≤1.681933) & (4.7867≤ ES35≤4.8936) & (46.657667≤ EV9 ≤ 48.549333) & (EV10 ≤ 42.146) & (EV11≥19.929333)	C0	5	7
R6	(1.643167≤ ES28 ≤ 1.681933) & (23.698333≤ ES31 ≤ 24.338667) & (43.715667≤ ES41 ≤44.310333) & (21.428≤ EV3 ≤25.55) & (EV8≥47.291) & (EV10 ≤42.146)	C0	5	6
R7	(1.643167≤ ES28 ≤1.681933) & (ES32 ≥1.326107) & (53.368667≤ES40 ≤54.195333) & (EV1 ≤62.563) & (53.429667≤EV2≤54.230333) & (EV4 ≥62.39) & (EV5 ≥22.177667) & (EV9 ≥48.549333) & (EV10 ≤42.146) & (EV11 ≤ 17.228667)	C1	5	10
R8	(23.828333≤ ES14 ≤25.757667) & (32.089≤ ES23 ≤32.526) & (53.368667≤ ES40 ≤54.195333) & (43.715667≤ ES41 ≤44.310333) & (35.642667≤ EV7 ≤40.572333) & (EV8 ≥ 47.291)	C1	10	6
R9	(0.195023≤ ES1 ≤0.249177) & (4.7867≤ ES35 ≤4.8936) & (0.010041≤ ES37 ≤0.02523) & (0.82287≤ ES38 ≤0.83839) & (43.715667≤ ES41 ≤44.310333) & (53.429667≤ EV2 ≤54.230333)	C1	5	6
R10	(3633.5≤ ES2 ≤3682.6) & (ES16 ≥3109.366667) & (ES27 ≤ 18.755667) & (2.250567≤ ES34 ≤ 2.297233) & (0.82287≤ ES38 ≤0.83839) & (ES39 ≤ 0.091435) & (EV11 ≤ 17.228667)	C1	5	7
R11	(ES9 ≤120.43) & (0.33132≤ ES10 ≤0.35293) & (79.953≤ ES11 ≤80.334) & (ES22 ≤77.154667) & (32.089≤ ES23 ≤32.526) & (ES27≤ 18.755667)	C1	5	6
R12	(9.3008≤ ES4 ≤9.473) & (26.731≤ES5≤27.13) & (ES13 ≥ 2641.533333) & (ES29≤32.740333) & (ES41≤43.715667) & (EV11≥19.929333)	C1	5	6
R13	(ES10 ≤ 0.35293) & (ES29 ≤2.740333) & (0.82287≤ ES38 ≤0.83839) & (ES39 ≥ 0.104042) & (ES41 ≤ 43.715667) & (EV10 ≥44.772)	C1	5	6
R14	(32.089≤ ES23 ≤32.526) & (ZS27≤18.755667) & (23.698333≤ ES31≤ 24.338667) & (EV5 ≥22.177667) & (43.613≤EV8≤47.291) & (42.146≤ EV10 ≤44.772)	C1	5	6
R15	(ES18 ≤ 65.531333) & (ES19 ≤ 226.816667) & (6.8213 ≤ ES26 ≤ 6.9733) & (0.82287 ≤ ES38 ≤ 0.83839) & (EV10 ≥ 44.772) & (17.228667 ≤ EV11 ≤ 19.929333)	C1	5	6
R16	(ES28 ≥1.681933) & (18.235333≤ ES33 ≤18.799667) & (0.010041≤ ES37 ≤0.02523) & (ES41 ≤ 43.715667) & (EV1 ≤62.563) & (21.471333 ≤ EV5 ≤22.177667) & (EV8 ≥ 47.291) & (46.657667≤ EV9≤48.549333)	C1	5	8
Mean				6.625

5.1.3. Feature Importance

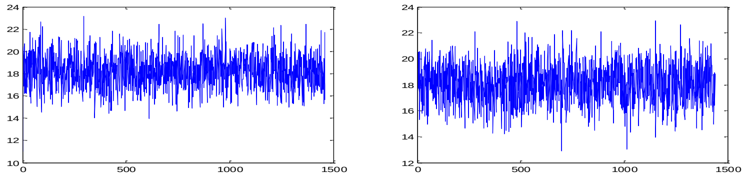
In Table 5. We present the importance value of each variable in the TEP dataset, derived from the classifier rule of the diagnosis model obtained in this study by using the IDEOA. As indicated in Table 5, the Reactor Cooling Water Flow is the crucial factor in identifying fault F4, which is associated with a sudden variation in the inlet temperature of the RCW. Thus, when fault F4 occurs we can notice a high disturbance in the variable EV10, which indicates a significant rise in the flow of the RCW. In addition, we can see that other variables are affected by the fault F4 occurrence like ES41, EV11 and others. The rest of variables remain almost steady after the fault occurrence, so, a small deviation can be found between the faulty status and the standard operating conditions. The contribution plots of Fig. 4, using Matlab12; shows the appearance of fault F4 using the training and testing TEP dataset for critical variables: ES41, EV10 and EV11:



a. Normal functioning b. abnormal functioning of EV10



a. Normal functioning b. abnormal functioning of ES41



a. Normal functioning b. abnormal functioning of EV11

Fig. 4. Variation in relevant variables contributing to F4 with normal and abnormal functioning

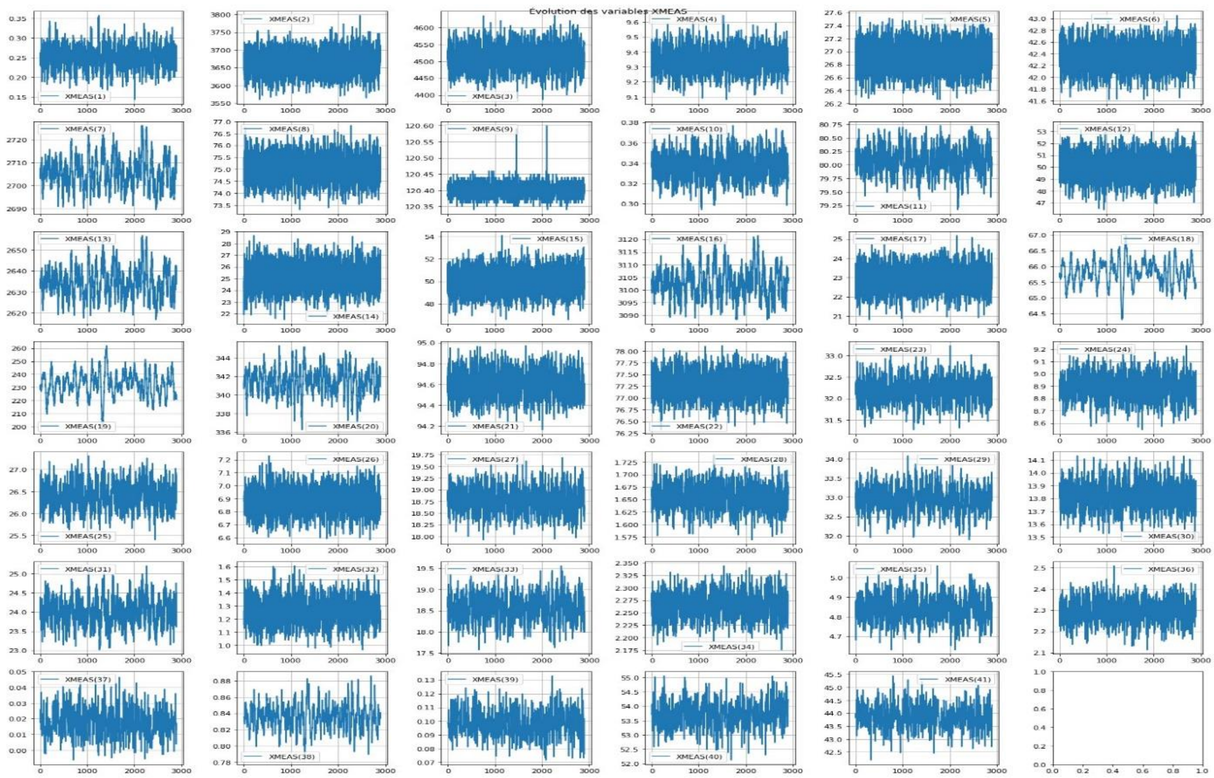


Fig. 5. Evolution of TEP variables

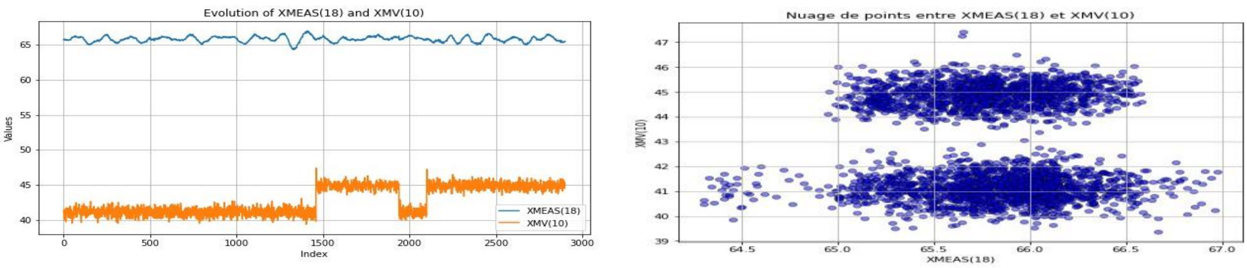


Fig. 6. Evolution of Es18 and Ev10

Table 5. Importance value of each characteristic in the IDEOA-based classifier created for TEP diagnosis

No	Characteristic	Importance value (%)	No	Characteristic	Importance value (%)	No	Characteristic	Importance value (%)
1	ES1	1.89%	19	ES19	1.89%	37	ES37	1.89%
2	ES2	1.89%	20	ES20	3.77%	38	ES38	3.77%
3	ES3	0.94%	21	ES21	0	39	ES39	2.83%
4	ES4	1.89%	22	ES22	0.94%	40	ES40	1.89%
5	ES5	1.89%	23	ES23	3.77%	41	ES41	5.66%
6	ES6	0	24	ES24	0	42	EV1	0.94%
7	ES7	0.94%	25	ES25	0.94%	43	EV2	1.89%
8	ES8	0.94%	26	ES26	0.94%	44	EV3	1.89%
9	ES9	0.94%	27	ES27	3.77%	45	EV4	0.94%
10	ES10	1.89%	28	ES28	3.77%	46	EV5	2.83%
11	ES11	1.89%	29	ES29	1.89%	47	EV6	0
12	ES12	0	30	ES30	0	48	EV7	1.89%
13	ES13	1.89%	31	ES31	4.72%	49	EV8	4.72%
14	ES14	0.94%	32	ES32	0.94%	50	EV9	2.83%
15	ES15	0	33	ES33	1.89%	51	EV10	7.55%
16	ES16	0.94%	34	ES34	1.89%	52	EV11	4.72%
17	ES17	1.89%	35	ES35	1.89%			

5.2. Comparison of the IDEOA with other approaches

To prove the efficiency of our proposed approach; we compare the results obtained by the IDEOA with other benchmark algorithms. The proposed approach was applied to the TEP dataset to generate a diagnosis rule-based classification model. For this comparison, we assess the classification accuracy of each approach for fault F4. The results of comparison are indicated in Table 6; from the obtained results, we can notice that the suggested IDEOA exhibits the highest accuracy in solving the problem of detection and diagnosis of fault 4 with a best value of 95.47%. In contrast, LAD (Logical Analysis Data), also showed a high accuracy of 95.04%. For the rest of approaches results, Random Tree achieves an accuracy of 89.55%, 76.33% was obtained by SVM, ANN had an accuracy of 69.44% while, ZeroR had the weakest value of 49.75%.

At last, as shown in this table, the result reported in the literature by [15], using Deep Learning DDSAE approach; is approximately close to our result. The difference between using DL approaches and the meta-heuristic methods (SWARM); is that our approach is particularly more efficient, ideal and faster in giving solutions for dynamic optimization problems, and it can provide better overall accuracy by avoiding local minima. So, the used approach in this manuscript outperforms the approach used in [15] in term of time of execution, because of its simplicity, robustness, and the low dependence of data. In addition, its generated rules are easy to understand by the reader contrary to the results of DL approaches, which are poorly interpretable with limited explanations for critical applications. Unlike other algorithms like DL approaches, our approach does not require modelling that is heavy and take a lot of time to be implemented.

Therefore, in this section, the results obtained by our algorithm showed its superiority in solving the problem of fault detection and diagnosis of F4 compared it to other algorithms in the literature. Fig. 5 represents the bar chart of the accuracy obtained by our method and other methods.

Table 6. Accuracy value obtained by our method compared to other methods

Method	Reference	Accuracy %
IDEOA	Our approach	95.47
LAD	[37]	95.04
SVM	Weka tool	76.33
ANN	Weka tool	69.44
ZeroR	Weka tool	49.75
RandomTree	Weka tool	89.55
DL DDSAE	[15]	99.7

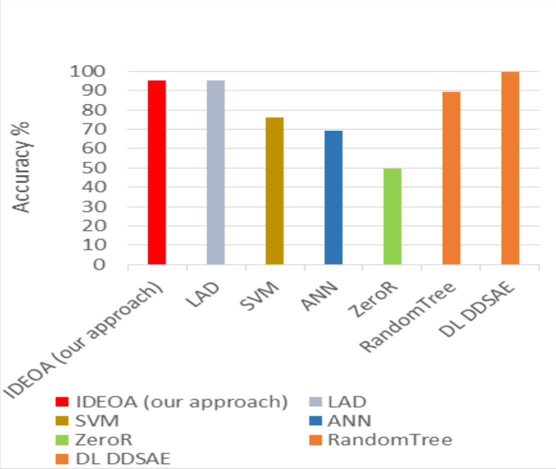


Fig. 5. Bar chart of the accuracy obtained by our method and other methods

6. CONCLUSION

In this study, we present an improved approach of the optimization method “DEOA” adapted to the field of FDD of industrial systems, which was used to generate accurate rules from the TEP dataset. The proposed approach was improved by adopting the OBL technique, allowing our algorithm to get best solutions with high accuracy, and by escaping local optima. This IDEOA is considered as a powerful method; based on the idea of the collaboration between simple agents to solve complex and optimized problems. Each agent follows a simple rule, to address the complex problem in a distributed manner. However, applying this method to problems involving complex fault scenarios with larger datasets and a high-dimensional search space, may affect the ability of search agent parameters, and the coordination between them becomes problematic. In addition, as the size of the problem and datasets increase, the iterations also increase and it could become expensive to analyze the behaviors of these large groups of agents, in terms of computing resources and computational time. Despite these limitations, the IDEOA shows a great performance for many types of problems, especially those requiring robust and distributed solutions, and becomes a key factor allowing better management of the system complexity increasing.

The obtained results of our experiments consist of 16 classification rules, six of them belong to a normal operating conditions and the rest reveals the fault F4 occurrence. These results shows that our algorithm surpasses other algorithms in terms of classification accuracy with a value of 95.47%. The power of this algorithm is in generating interpretable rules with fewest terms that made the generated diagnosis model more explainable making it a powerful tool for investigating the pattern of a fault and elucidating its causes. Ultimately, the obtained results of this study demonstrates the applicability and the explainable ability of the suggested approach in the domain of FDD of industrial systems.

Our work’s next direction will focus on two aspects: at first, as such, our approach needs additional validation on real world problems; so we consider applying it for real industrial datasets systems such as turbines and heat exchangers, to treat the problem of feature extraction in FDD. At second, we intend to link our proposed approach to other practical swarm optimization in order to enhancing its performance.

Source of funding: *This research received no external funding.*

Author contributions: *research concept and design, M.B and R.M; Collection and/or assembly of data, M.B and R.M; Data analysis and interpretation, H.H; Writing the article, M.B; Critical revision of the article, L-H.M; Final approval of the article, B.H.*

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

REFERENCES

1. Isermann R. Fault-Diagnosis Applications. Springer H. Springer Berlin Heidelberg, Berlin, Heidelberg 2011. https://doi.org/https://doi.org/10.1007/978-3-642-12767-0
2. Isermann R. Supervision FDD Methods - An Introduction. Control Eng Pract [Internet]. 1997;5(5):639–52. Available from: https://doi.org/10.1016/S0967-0661(97)00046-4
3. Venkatasubramanian V, Rengaswamy R, Yin K, Kavuri SN. A review of process fault detection and diagnosis part I: Quantitative model-based methods. Comput Chem Eng. 2003;27(3):293–311. https://doi.org/10.1016/S0098-1354(02)00160-6
4. Katipamula S, Brambley MR, Brambley MR. Methods for Fault Detection , Diagnostics , and Prognostics for Building Systems - A Review , Part I. HVAC R Res. 2005;11(1):3–25. https://doi.org/10.1080/10789669.2005.10391123
5. Venkatasubramanian V, Rengaswamy R, Kavuri SN, Yin K. A review of process fault detection and diagnosis: Part III: Process history based methods. Comput Chem Eng. 2003 Mar 15;27(3):327–46. https://doi.org/10.1016/S0098-1354(02)00162-X
6. Haouassi H, Mahdaoui R, Chouhal O, Bekhouche A. An efficient classification rule generation for coronary artery disease diagnosis using a novel discrete equilibrium optimizer algorithm. J Intell Fuzzy Syst. 2022;43(3):2315–31. https://doi.org/10.3233/JIFS-213257
7. Mahdaoui R, Mouss LH, Mouss MD, Chouhal O. A Temporal Neuro-Fuzzy Monitoring System to Manufacturing Systems. 2011;8(3):237–46. http://arxiv.org/abs/1107.3302
8. Faramarzi A, Heidarinejad M, Stephens B, Mirjalili S. Equilibrium optimizer: A novel optimization algorithm. Knowledge-Based Syst. 2020 Mar 5;191. https://doi.org/10.1016/j.knosys.2019.105190
9. Malik MM, Haouassi H. Efficient sequential covering strategy for classification rules mining using a discrete equilibrium optimization algorithm. J King Saud Univ

- Comput Inf Sci [Internet]. 2021; https://doi.org/10.1016/j.jksuci.2021.08.032
10. Zou W, Xia Y, Li H. Fault diagnosis of Tennessee-eastman process using orthogonal incremental extreme learning machine based on driving amount. IEEE Trans Cybern. 2018;48(12):3403–10. https://doi.org/10.1109/TCYB.2018.2830338
11. Onel M, Kieslich CA, Pistikopoulos EN. A nonlinear support vector machine-based feature selection approach for fault detection and diagnosis: Application to the Tennessee Eastman process. AIChE J. 2019;65(3):992–1005. https://doi.org/10.1002/aic.16497
12. Soraya B, Faouzi HM, Abderrazak L. Fault Diagnosis of Tennessee Eastman Process Based on Static PCA. Proc - 2019 1st Int Conf Sustain Renew Energy Syst Appl ICSRESA 2019. 2019;(2). https://doi.org/10.1109/ICSRESA49121.2019.9182366
13. Zhang C, Guo Q, Li Y. Fault detection in the Tennessee eastman benchmark process using principal component difference based on k-nearest neighbors. IEEE Access. 2020;8:49999–50009. https://doi.org/10.1109/ACCESS.2020.2977421
14. Han Y, Ding N, Geng Z, Wang Z, Chu C. An optimized long short-term memory network based fault diagnosis model for chemical processes. J Process Control [Internet]. 2020;92:161–8. https://doi.org/10.1016/j.jprocont.2020.06.005
15. Agarwal P, Tamer M, Budman H. Explainability: Relevance based dynamic deep learning algorithm for fault detection and diagnosis in chemical processes. Comput Chem Eng [Internet]. 2021;154:107467. https://doi.org/10.1016/j.compchemeng.2021.107467
16. Lv F, Wen C, Bao Z, Liu M. Fault diagnosis based on deep learning. In: Proceedings of the American Control Conference. 2016. p. 6851–6. https://doi.org/10.1109/ACC.2016.7526751
17. Lomov I, Lyubimov M, Makarov I, Zhukov LE. Fault detection in Tennessee Eastman process with temporal deep learning models. J Ind Inf Integr [Internet]. 2021;23(September 2020):100216. https://doi.org/10.1016/j.jii.2021.100216
18. Xu H, Ren T, Mo Z, Yang X. A Fault Diagnosis Model for Tennessee Eastman Processes Based on Feature Selection and Probabilistic Neural Network. Appl Sci. 2022;12(17). https://doi.org/10.3390/app12178868
19. Hu M, Hu X, Deng Z, Tu B. Fault Diagnosis of Tennessee Eastman Process with XGB-AVSSA-KELM Algorithm. Energies. 2022;15(9):1–25. https://doi.org/10.3390/en15093198
20. De Leon-Aldaco SE, Calleja H, Aguayo Alquicira J. Metaheuristic Optimization Methods Applied to Power Converters: A Review. IEEE Trans Power Electron. 2015;30(12):6791–803. https://doi.org/10.1109/TPEL.2015.2397311
21. Chen J, Patton RJ, Liu GP. An extension to the single bottleneck transportation problem. Int J Syst Sci. 1996;27(6):567–76. https://doi.org/10.1080/00207729608929252
22. Abdel-Basset M, Mohamed R, Mirjalili S. A Binary Equilibrium Optimization Algorithm for 0–1 Knapsack Problems. Comput Ind Eng [Internet]. 2021;151:106946. https://doi.org/10.1016/j.cie.2020.106946
23. Wang J, Yang B, Li D, Zeng C, Chen Y, Guo Z, et al. Photovoltaic cell parameter estimation based on improved equilibrium optimizer algorithm. Energy Convers Manag [Internet]. 2021;236:114051. https://doi.org/10.1016/j.enconman.2021.114051
24. Too J, Mirjalili S. General Learning Equilibrium Optimizer: A New Feature Selection Method for Biological Data Classification. Appl Artif Intell [Internet]. 2021;35(3):247–63. https://doi.org/10.1080/08839514.2020.1861407
25. Guha R, Ghosh KK, Bera SK, Sarkar R MS. Discrete Equilibrium Optimizer Combined with Simulated Annealing for Feature Selection. Res Sq [Internet]. 2022. https://doi.org/10.21203/rs.3.rs-28683/v2
26. Fan Q, Huang H, Yang K, Zhang S, Yao L, Xiong Q. A modified equilibrium optimizer using opposition-based learning and novel update rules. Expert Syst Appl [Internet]. 2021;170:114575. https://doi.org/10.1016/j.eswa.2021.114575
27. Liu B, Hsu W, Ma Y. KDD98-012.pdf. Kdd [Internet]. 1998;98:80–6. Available from: www.aaai.org
28. Hossain SKM, Ema SA, Sohn H. Rule-Based Classification Based on Ant Colony Optimization: A Comprehensive Review. Appl Comput Intell Soft Comput. 2022;2022. https://doi.org/10.1155/2022/2232000
29. Liu SH, Mernik M, Hrnčič D, Črepinšek M. A parameter control method of evolutionary algorithms using exploration and exploitation measures with a practical application for fitting Sovova's mass transfer model. Appl Soft Comput J. 2013;13(9):3792–805. https://doi.org/10.1016/j.asoc.2013.05.010
30. Tizhoosh HR. Opposition-based learning: A new scheme for machine intelligence. Proc - Int Conf Comput Intell Model Control Autom CIMCA 2005 Int Conf Intell Agents, Web Technol Internet. 2005;1:695–701. https://doi.org/10.1109/CIMCA.2005.1631345
31. Ul Hassan N, Bangyal WH, Ali Khan MS, Nisar K, Ibrahim AAA, Rawat DB. Improved opposition-based particle swarm optimization algorithm for global optimization. Symmetry (Basel). 2021;13(12):1–23. https://doi.org/10.3390/sym13122280
32. Rahab H, Haouassi H, Souidi MEH, Bakhouche A, Mahdaoui R, Bekhouche M. A Modified Binary Rat Swarm Optimization Algorithm for Feature Selection in Arabic Sentiment Analysis. Arab J Sci Eng [Internet]. 2022; https://doi.org/10.1007/s13369-022-07466-1
33. Downs JJ, Vogel EF. A plant-wide industrial process control problem. Comput Chem Eng. 1993;17(3):245–55. https://doi.org/10.1016/0098-1354(93)80018-I
34. Evan Russell LHC, Braatz and RD. Data-driven Methods for Fault Detection and Diagnosis in Chemical Processes. Springer-Verlag London Berlin Heidelberg; 2000.
35. Detroja KP, Gudi RD, Patwardhan SC. Plant-wide detection and diagnosis using correspondence analysis. Control Eng Pract. 2007;15(12):1468–83. https://doi.org/10.1016/j.conengprac.2007.02.007
36. Maran Beena A, Pani AK. Fault Detection of Complex Processes Using nonlinear Mean Function Based Gaussian Process Regression: Application to the Tennessee Eastman Process. Arab J Sci Eng [Internet]. 2021;46(7):6369–90. https://doi.org/10.1007/s13369-020-05052-x
37. Ragab A, El-koujok M, Poulin B, Amazouz M, Yacout S. Fault diagnosis in industrial chemical processes using interpretable patterns based on Logical Analysis

of Data. Expert Syst Appl [Internet]. 2018;95:368–83. https://doi.org/10.1016/j.eswa.2017.11.045

Meriem BENBRAHIM
Received her Master's degree in Environmental Engineering and Sustainable Development in 2016, and her Engineer degree in Hygiene and Industial Safety in 2014, from the Institute of Hygiene and Safety of Batna University. She is currently a PhD student in the laboratory of Electromechanical Engineering (LGEM), Badji-Mokhtar University, Annaba. Her Research interests include Industrial Systems fault diagnosis and systems safety.
E-mail : meriem.benbrahim@outlook.fr

Brahim HAMAIDI
He is currently a Professor in the Electromechanical-Engineering Department, and a Director of the LGEM (Electromechanical Engineering Laboratory) at the University of Badji Mokhtar Annaba. His research interests include electrics, maintenance and safety.
E-mail : ham5615@yahoo.fr

Hichem HAOUASSI
Received Ph.D. degree in Industrial Engineering Sciences from Batna University in 2012. He is currently a Professor in mathematics and computer science department and a team leader of software Engineering in ICOSI laboratory at the University of Abbes Laghrour Khenchela., Algeria.
His research interests include Algorithmic, computer science, data mining.
E-mail : haouassi.hichem@univ-khenchela.dz

Rafik MAHDAOUI
Received Ph.D. degree in Industrial Engineering Sciences from Batna University in 2013 for his thesis titled "Contribution to the dynamic monitoring of evolving production systems using Temporal Neuro-Fuzzy systems". He is currently a Professor in mathematics and computer science department, and a IT security team leader in ICOSI laboratory at the University of Abbes Laghrour Khenchela., Algeria.
His research interests include computer science, software Security and Industrial systems.
E-mail: mahdaoui.rafik@univ-khenchela.dz

Leial-Hayet MOUSS
Received the B.Sc. degree in Electrical Engineering from the National Polytechnic School of Algiers, Algeria, in 1979, and the M.Sc. and Ph.D. degrees in electrical and computer engineering from the University of Bordeaux—ENSERB, Bordeaux, France, in 1982 and 1985, respectively.,After graduation, she joined the University of Batna 2, Algeria, where she is currently a Full Professor of electrical and computer engineering. She is the Head of the Automatic and Computer Integrated Manufacturing Laboratory, University of Batna 2. Her current research interests include industrial diagnosis of production system using artificial intelligence techniques.,Prof. Mouss is a member of the New York Science Academy.
E-mail: h.mouss@univ-batna2.dz