Data-driven fault detection of open circuits in multi-phase inverters based on current polarity using Auto-adaptive and Dynamical Clustering

T.-H. Pham, S. Lefteriu, E. Duviella, S. Lecoeuche

Version of Record: https://www.sciencedirect.com/science/article/pii/S0019057820302561
Manuscript_5dab5c33fb4eb6c8fc5461774b1b493f

Data-driven fault detection of open circuits in multi-phase inverters based on current polarity using Auto-adaptive and Dynamical Clustering

Thanh Hung Pham[1], Sanda Lefteriu[1], Eric Duviella[1], Stéphane Lecoeuche[1]

[1] Institut Mines Telecom Lille Douai, IA, F-59508 Douai, France

Email: phamthanhhung1204@gmail.com
sanda.lefteriu@imt-lille-douai.fr
eric.duviella@imt-lille-douai.fr
stephane.lecoeuche@imt-lille-douai.fr

Corresponding Author: Thanh Hung PHAM

Data-driven fault detection of open circuits in multi-phase inverters based on current polarity using Auto-adaptive and Dynamical Clustering

Abstract

This paper proposes a data-driven method for the detection and isolation of open-circuit faults in multi-phase inverters using measurements of the motor currents. First, feature variables are formulated in terms of the averages of the phase currents and their absolute values. Next, by using an AUto-adaptive and Dynamical Clustering (AUDyC) based on Gaussian Mixture Models, feature data is clustered into different classes characterizing normal and faulty operation modes. Afterwards, these classes are used for deriving appropriate conditions for detecting and labelling faults. The proposed method requires minimal knowledge about the system operation. Furthermore, it allows us to update our knowledge of existing faults online, thus making it possible to detect unknown faults. Moreover, conditions are formulated to describe the influence of the method parameters on the detection time. Once parameters are tuned, the accuracy of the proposed method is illustrated on various experimental data sets, where single and double faults are detected with detection times in the order of the fundamental signal period.

1. Introduction

Multi-phase inverters are interesting for industrial applications because of their fault-tolerance and high power density [1]. Since power switches are the most vulnerable components in inverters, most inverter faults are due to the switches remaining closed or open longer than the defined period, leading to Short-Circuit (SC) or Open-Circuit (OC) faults, respectively [2]. With the SC fault, there are usually standard protection systems which immediately shut down the system. However, when OC faults happen, the system can keep running with degraded performance. Hence, Fault Detection and Isolation (FDI) of OC faults in multi-phase inverters associated to a multi-phase motor is an ongoing research topic [3]. Different approaches were proposed in the literature: model-based, signal-based and data-driven [4]. The model-based approach is rather involved for real-world applications (e.g., deriving the model of a multi-phase asynchronous motor is a tedious task). On the other hand, the signal-based approach is generally not robust because of the influence of varying parameters or the presence of noise in the measurements [5]. A middle ground between the two, the data-driven approach only requires some knowledge of the

Preprint submitted to Special Issue on Prognostics and Health Management of Complex Systems, June 5, 2020

system's operation, while still being robust, as it considers a large amount of measurement data. For the system including a multi-phase inverter governing a multi-phase inductive load (e.g., motor), using measurements of the phase currents in the data-driven approach is preferred to the phase voltages due to the reduced cost related to additional sensor installation [3]. This approach first extracts the features of the operation modes through feature vectors whose values are later treated in the machine learning procedure to construct feature classes corresponding to operation modes. Based on the resulting classes, conditions for the operation mode labelling and the FDI are formulated.

Both supervised and unsupervised machine learning approaches were considered for FDI problems depending on the available prior information. If input feature vectors along with their corresponding target vectors are included in the training data, the approach is called supervised learning [6], with some methods applied to FDI problem being: Support Vector Machines [7], Discriminant Analysis [8], Bayesian Networks [2], K-Nearest Neighbors [9], Neural Networks [10]. However, a supervised learning method may not be suitable for complex industrial systems in case no historical data of faulty operation modes is available, or a new fault (different from the training data) appears. To deal with these issues, the unsupervised learning approach is employed where only input feature vectors are available [6]. The most popular method of this approach is clustering. With K-means [11] or Fuzzy C-means [12], the input feature vectors are clustered such that the distance between them and the corresponding classes is minimal. This method is simple, but requires to know a priori the number of faults, which is not always available. In [13], the authors use hierarchical clustering, where two similar clusters of feature vectors are grouped to form a cluster on a higher position in a binary and hierarchical tree. This method does not need an a priori number of faults, but its results are sensitive to outliers. In [14], the density-based spatial clustering is considered, where data classes are defined by high density clusters which are separated by sparse areas. Although this method identifies well outliers as noise, it does not work well when the data density in clusters is varying. By using Gaussian Mixture Models (GMM), the authors in [15] can obtain clusters of more flexible shapes (ellipses). In real-world applications, measured data is usually non-stationary, so on-line and adaptive learning should be employed to deal with this kind of data. To this end, Artificial Neural Networks (ANN) are usually applied for unsupervised learning, a well-known example being Self-Organizing Maps [16]. Most existing methods do not possess self-organizing abilities, constructive architectures or fully auto-adaptive structures, but AUto-adaptive and Dynamical Clustering (AUDyC), which is based on GMM and ANN, does [17, 18]. Recently, this machine learning method was employed to detect and isolate converter faults in [19, 20], and is the method of choice in this paper.

Machine learning techniques are generally not used directly on measurement data, but on appropriately defined feature vectors, for effective algorithms. Methods to design feature vectors are based on time domain data, frequency domain data or their combination. Time-domain feature variables are directly derived from the measurement data such as the average current over a fundamental

signal period [3], or the Principle Component Analysis [2]. Frequency-domain feature vectors are usually computed based on the Fast Fourier Transform [2, 21] or wavelet transform [22].

The present manuscript builds on the authors' previous work in [20, 4]. Double OC faults in inverters were detected using feature variables defined in terms of the average phase currents and their average absolute values. These choices avoid expensive computations and load dependence with minimum knowledge of the system's faulty operation modes. However, there is still the need to choose an appropriate threshold making the tuning task complicated. Moreover, when dealing with experimental data obtained from a closed loop controlled system, a suitable value for the threshold is difficult to find due to the uncertainty of the motor current frequency.

Table 1: Parameter and variable notations.

Notation	Description
$\mathbb{N}$	set of natural numbers
$\mathbb{R}$	set of real numbers
$\mathbb{R}^+$	set of positive real numbers
n	number of inverter phases
f	signal frequency
t_s	sampling time
N_s	number of time samples on a fundamental signal period
k_s	sample time index
$\beta(t)$	feature variable
$\boldsymbol{\beta}(t)$	feature vector
$\mathbb{B}_k$	set of k feature vectors
$\overline{\boldsymbol{\beta}}_k$	mean vector of $\mathbb{B}_k$
$\mathbb{C}(t)$	feature class
$\overline{\boldsymbol{\beta}}(t)$	mean vector of a feature class
$\overline{\beta}_k(t)$	k^{th} element of $\overline{\boldsymbol{\beta}}(t)$
$\mathbf{I}_n$	identity matrix of dimension n
μ_{min}	class membership threshold
σ	initial covariance
$N_{min},\ N_{max}$	minimum and maximum numbers of class elements
ϵ_l	labelling threshold
N_f	sliding window width
F_f	fault threshold
t_C	faulty class creation time
t_L	faulty class labelling time
t_d	faulty class isolation time (detection time)
$\mathbf{M}(t)$	indicator matrix
$i_k(t)$	current of phase k
$\lvert i_k(t)\rvert$	absolute value of phase-k current
$\langle i_k(t)\rangle$	average value of signal $i_k(t)$ over a fundamental signal period
I_0	$\langle i_k(t)\rangle$ in the normal operation mode
$\mathbf{x}(t)$	observation vector
$g(t), g_\epsilon(t), g_\sigma(t)$	real functions of time
$g_\mu(t), g_N(t)$	real functions of time

In this paper, the FDI algorithm previously proposed in [20] is improved to avoid the tuning of a threshold value in the feature extraction procedure for a multi-phase inverter. Moreover, this manuscript presents the influence of algorithm parameters on the detection time under some specific assumptions.

The proposed method and the influence of the method parameters are validated on experimental data obtained from a closed loop system.

The remaining part of the paper is structured as follows. First, Table 1 presents the notation for the parameters and the variables used in this paper. Section 2 describes the inverter operation in the normal and faulty modes. Next, Section 3 introduces the FDI method. Section 4 studies the influence of the method parameters on the detection time. Section 5 validates the method on experimental data. Section 6 concludes the paper and presents some possible future research directions.

2. Inverter in normal and faulty operation modes

The descriptions of the 3-phase and 5-phase inverter operation modes can be found in [20] and [3], respectively. These systems contain Insulated-Gate Bipolar Transistors (IGBTs) which act as switches controlled by a Pulse Width Modulation (PWM) signal. The switches are opening and closing periodically with the PWM frequency. An OC fault occurs when a switch remains open longer than the period defined by the control.

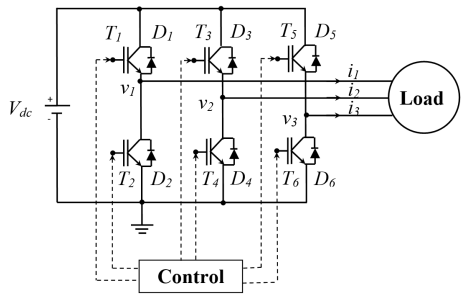

Figure 1: 3-phase inverter driving an inductive load [2].

For a 3-phase inverter, the single and double OC faults essentially lead to 4 general cases, shown in Fig. 2, even though there are 21 different possibilities. This figure presents the 3-phase current profiles in normal operation before 0.2s, and in faulty mode after that.

- In Case 1, only one IGBT, e.g., T_1 (see Fig. 1), is faulty. This results in the negative phase-1 current, i.e., $i_1(t) \leq 0$ (see Scenario 1 in Fig. 2). When T_3 or T_5 is faulty, results are similar: $i_2(t) \leq 0$ or $i_3(t) \leq 0$, respectively. When T_2, T_4 or T_6 is faulty, $i_1(t) \geq 0$, $i_2(t) \geq 0$ or $i_3(t) \geq 0$, respectively.

- In Case 2, two IGBTs of the same phase, e.g., $T_1\&T_2$ (see Fig. 1), are faulty. This results in a zero phase-1 current, i.e., $i_1(t) = 0$ (see Scenario 2 in Fig. 2). Similar behavior is observed when $T_3\&T_4$ ($i_2(t) = 0$) or $T_5\&T_6$ ($i_3(t) = 0$) are faulty.

- In Case 3, two IGBTs of different phases but on the same side, e.g., $T_1\&T_3$ (see Fig. 1), are faulty. This results in negative currents for phase 1 and 2, i.e., $i_1(t) \leq 0$, and $i_2(t) \leq 0$, while the phase-3 current is positive, i.e.,

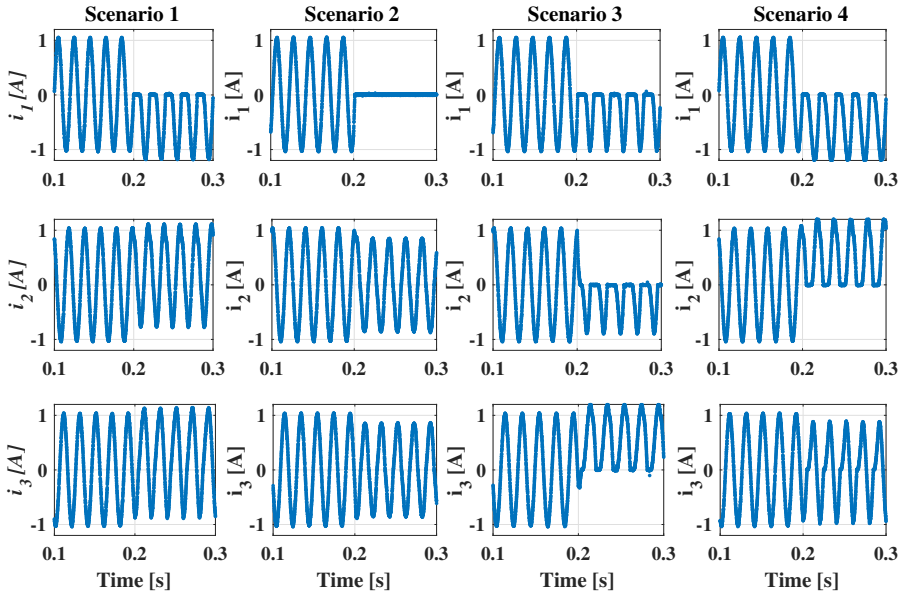

Figure 2: Current profiles in the double cases of the 3-phase inverter.

$i_3(t) \geq 0$ (see Scenario 3 in Fig. 2). The opposite behavior is noticed when $T_2\&T_4$ are faulty. When $T_1\&T_5$ are faulty, $i_1(t) \leq 0$, $i_3(t) \leq 0$ and $i_2(t) \geq 0$, which is opposite to when $T_2\&T_6$ are faulty. When $T_3\&T_5$ are faulty, $i_2(t) \leq 0$, $i_3(t) \leq 0$ and $i_1(t) \geq 0$, which is opposite to when $T_4\&T_6$ are faulty.

- In case 4, two IGBTs of different phases and on different sides, e.g., $T_1\&T_4$ (see Fig. 1), are faulty. This results in a negative phase-1 current, i.e., $i_1(t) \leq 0$, and a positive phase-2 current, i.e., $i_2(t) \geq 0$ (see Scenario 4 in Fig. 2). This is opposite to when $T_2\&T_3$ are faulty. When $T_1\&T_6$ are faulty, $i_1(t) \leq 0$, $i_3(t) \geq 0$, and this is opposite to when $T_2\&T_5$ are faulty. When $T_3\&T_6$ are faulty, $i_2(t) \leq 0$, $i_3(t) \geq 0$, which is opposite to when $T_4\&T_5$ are faulty.

In summary, if only one phase current does not change polarity, the single fault is in one of the two transistors in that branch; if one phase current is always zero, both transistors in that branch are faulty; if all three phase currents do not change polarities, two transistors on the same side in the two branches, where the currents have the same polarity, are faulty; if only two phase currents do not change the polarities, two transistors on different sides in these branches are faulty. For an inverter with more than 3 phases, it is not necessary to discriminate between Case 3 and Case 4, since a phase current does not change the polarity if and only if a transistor of that branch is faulty. Based on the polarity analysis, appropriate feature variables are derived in the next section.

3. Proposed method

This section presents the proposed FDI algorithm which includes the following procedures: feature extraction, data clustering with AUDyC, labelling and fault detection and isolation, as summarized in Fig. 3. In the feature extraction, the measurement data from the motor currents is represented in the feature space where the steady-state data corresponding to different operation modes is separated. For AUDyC learning, only representative vectors in the feature space are collected to construct feature classes. In the labelling stage, these classes are labelled using knowledge about the system operation. In the last stage, based on similarities between recent feature vectors and the labelled classes, the faulty modes are declared.

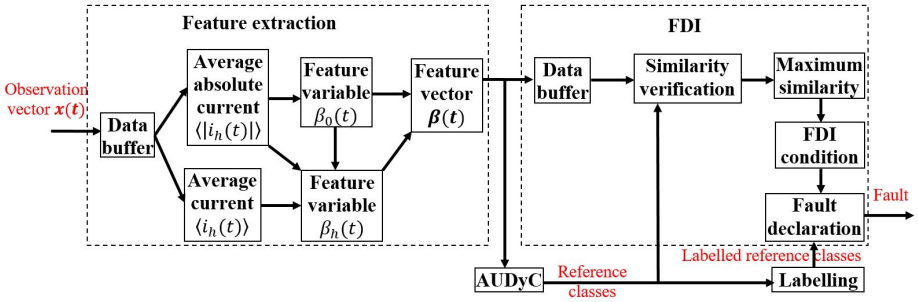

Figure 3: Diagram of the proposed fault detection method.

3.1. Feature extraction

This procedure aims at formulating an appropriate feature vector containing all feature variables, allowing us to do the clustering, labelling and FDI. In our case, the observation (or data) vector $\mathbf{x}(t)$ gathers the load currents, i.e., $\mathbf{x}(t) = \begin{bmatrix} i_1(t) & \dots & i_n(t) \end{bmatrix}^T \in \mathbb{R}^n$, where $n \in \mathbb{N}$ is the number of phases (3 or 5). In [4], the normalized average phase currents were used as feature variables. However, in Case 2 presented in Section 2, a feature variable is possibly infinite due to the zero faulty phase current. Therefore, in [20], an additional feature variable, $\beta_0(t) = \prod_{h=1}^{n} \frac{\langle|i_k(t)|\rangle}{I_0}$ with $I_0 = \langle|i_k(t)|\rangle$ in the normal operation mode, was considered to allow treating Case 2 within the whole procedure, an additional threshold was used, while the formulations for the other feature variables were modified to avoid the infinite value in Case 2.

When phase-k is faulty (Case 2), $\langle|i_k(t)|\rangle / I_0$ and, hence, $\beta_0(t)$ are zero. However, in real systems, they are usually positive due to the uncertainty. Even when $\langle|i_k(t)|\rangle / I_0$ is small, its product with corresponding terms of the other currents is still considerably large. This makes the parameter tuning even more complex. To deal with this problem, the value of $\langle|i_k(t)|\rangle / I_0$ is amplified such that the positiveness of the absolute values of the other currents is reduced. The simplest choice to this end is the application of a logarithmic function on

$\langle|i_k(t)|\rangle / I_0$, i.e., $ln\left(\langle|i_k(t)|\rangle / I_0\right)$. Moreover, to eliminate the additional threshold, zero is chosen as an invariant threshold to define the feature variables in the following way:

$$\beta_0(t) = \prod_{h=1}^{n} \sqrt[n]{1 + ln\left(\frac{\langle|i_k(t)|\rangle}{I_0}\right)}, \tag{1a}$$

$$\beta_k(t) = \begin{cases} \dfrac{\langle i_k(t)\rangle}{\langle|i_k(t)|\rangle}, & \text{if } \beta_0(t) > 0,\ h = 1, \ldots, n, \\ \dfrac{\langle|i_k(t)|\rangle}{I_0}, & \text{if } \beta_0(t) \leq 0,\ h = 1, \ldots, n, \end{cases} \tag{1b}$$

where $|i_k(t)|$ is the absolute value of phase-k current, $\langle g(t)\rangle = \sum_{k_s=0}^{N_s-1} \dfrac{g(t - k_s t_s)}{N_s}$ is the average value of function $g(t)$ over a fundamental signal period, t_s is the sampling time, $k_s \in \mathbb{N}$ is the sampling time index, and $N_s \in \mathbb{N}$ is the number of time samples on a fundamental signal period. The n^{th} root in (1a) is to simplify the tuning task for the data clustering by making the variance of $\beta_0(t)$ approximate to the other feature variables. Note that $\beta_0(t)$ is equal to 1 in the normal mode. In Case 1, 3 and 4, it is positive, while it is negative when there is at least one phase current close to zero in Case 2. Moreover, due to uncertainty, $\langle|i_k(t)|\rangle$ is not zero, and hence, $\beta_0(t)$ is not negative infinity. The feature vector is obtained from the feature variables given in (1) as $\boldsymbol{\beta}(t) = \begin{bmatrix} \beta_0(t) & \beta_1(t) & \ldots & \beta_n(t) \end{bmatrix}^T \in \mathbb{R}^{n+1}$. This vector will be used as input for the data clustering procedure.

3.2. Data clustering

In this procedure, AUDyC is used for clustering the feature vectors $\boldsymbol{\beta}(t)$ into classes, as described in [4]. A simplified formulation of AUDyC is considered, where the distribution of class data is Gaussian. A class is represented by the center, the covariance matrix and a number of feature vectors. Whenever a new feature vector $\boldsymbol{\beta}(t)$ appears, the closeness between this vector and existing classes is verified using the Mahalanobis distance with a chosen membership threshold $\mu_{min} \in (0, 1)$. If there is no class, a new class is created with a given initial covariance matrix $\sigma \mathbf{I}_{n+1}$, where $\sigma \in \mathbb{R}^+$ is the initial covariance. Moreover, the feature vector $\boldsymbol{\beta}(t)$ is used as the class center, and stored in the class memory. If there is a unique class close to $\boldsymbol{\beta}(t)$, its parameters are modified by taking into account $\boldsymbol{\beta}(t)$, and $\boldsymbol{\beta}(t)$ is added to the class memory. The number of vectors stored in a class is limited to a chosen range $[N_{min}, N_{max}]$. If there are more than one class close to $\boldsymbol{\beta}(t)$ and these classes are close enough, they will be fused to obtain a new class. Thereafter, the new class parameters are modified according to the adaptation procedure where there is a unique class close to $\boldsymbol{\beta}(t)$. Finally, if the number of feature vectors in a class is less than N_{min}, the class is eliminated. All the removed vectors will be used in the next iterative learning loop. From the algorithm description in [4], the following condition is derived for the class creation which will be used hereafter for investigating the

influence of the initial covariance σ, the membership threshold μ_{min} and the minimum class cardinality N_{min} on the detection time.

Remark 1 ([20]). *A feature class is created after a learning step if and only if there exists $N \geq N_{min}$ feature vectors which are far from existing classes but close to each other, i.e.,:*

$$exp\left(-\frac{\left\|\boldsymbol{\beta}-\overline{\boldsymbol{\beta}}_N\right\|_2}{2\sigma^2}\right) \geq \mu_{min}, \text{ for all } \boldsymbol{\beta} \in \mathbb{B}_N, \tag{2}$$

where $\mathbb{B}_N$ is the set of N feature vectors, and $\overline{\boldsymbol{\beta}}_N$ is the mean vector of $\mathbb{B}_N$.

As this is an unsupervised learning procedure, the obtained classes have no physical meaning, so they need to be named using knowledge about the system operation in the labelling procedure.

3.3. Labelling

Here, a feature class $\mathbb{C}(t)$ is labelled based on its mean vector denoted by $\overline{\boldsymbol{\beta}}(t)$. From the fault analysis in Section 2 and the feature variable formulation in Subsection 3.1, which leads to feature variable values illustrated in Fig. 4, some remarks are in place.

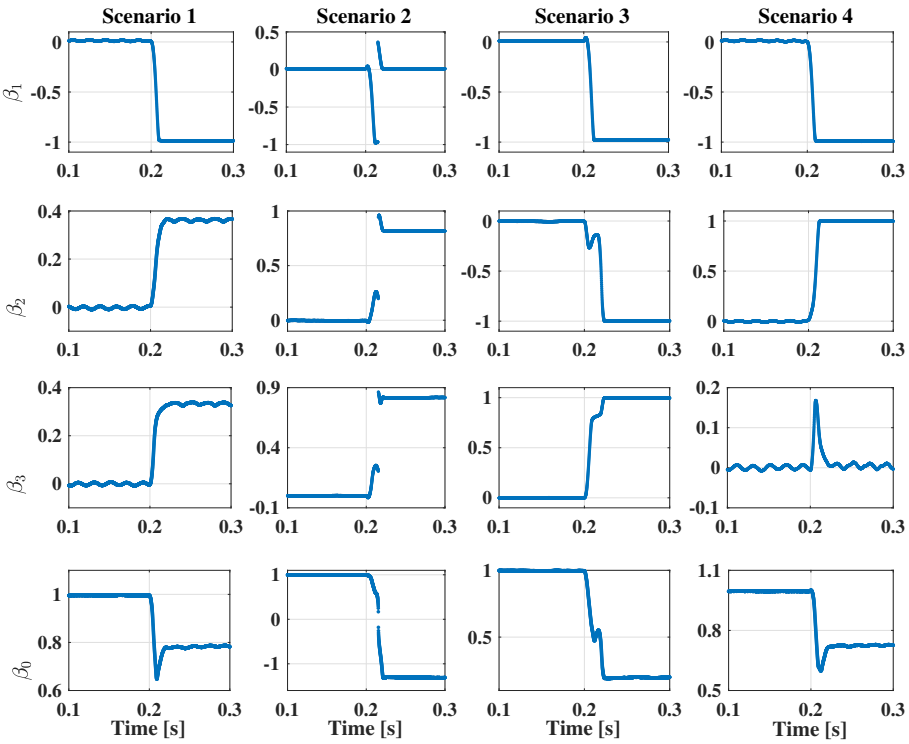

Figure 4: Feature variable profiles in Scenarios 1-4.

In the normal mode, the first feature variable $\beta_0(t)$ is positive, while the others are zero. In Case 1 and 4, $\beta_0(t)$ is positive and the feature variables corresponding to the faulty phases are equal to 1 or -1 (see Scenario 1 and 4 in Fig. 4). In Case 2, $\beta_0(t)$ is negative and the feature variable corresponding to the faulty phase is zero (see Scenario 2 in Fig. 4). In Case 3, $\beta_0(t)$ is positive and the feature variables corresponding to faulty phases are equal to 1 or -1 (see Scenario 3 in Fig. 4). The other feature variables are equal to -1 or 1 in the case of the 3-phase inverter, while they are not in the case of n-phase inverters with $n > 3$. However, when taking into account uncertainty, a positive threshold $\epsilon_l \in \mathbb{R}^+$ is used to verify the position of the mean vector $\overline{\boldsymbol{\beta}}(t)$ with respect to 0, 1 or -1. The previous remarks are summarized in Table 2.

Table 2: Labelling conditions.

Fault	Condition
None	$\overline{\beta}_0 > 0$ & $\max\limits_{m\in\{1,\dots,n\}} \left\lvert\overline{\beta}_m\right\rvert < \epsilon_l$
T_{2m-1}	$\overline{\beta}_0 > 0$ & $\left\lvert\overline{\beta}_m + 1\right\rvert \leq \epsilon_l$ & $\left\lvert\overline{\beta}_p \pm 1\right\rvert > \epsilon_l$, with $m, p \in \{1,\dots,n\}$, $m \neq p$
T_{2m}	$\overline{\beta}_0 > 0$ & $\left\lvert\overline{\beta}_m - 1\right\rvert \leq \epsilon_l$ & $\left\lvert\overline{\beta}_p \pm 1\right\rvert > \epsilon_l$, with $m, p \in \{1,\dots,n\}$, $m \neq p$
T_{2m-1} & T_{2p}	$\overline{\beta}_0 > 0$ & $\left\lvert\overline{\beta}_m + 1\right\rvert \leq \epsilon_l$ & $\left\lvert\overline{\beta}_p - 1\right\rvert \leq \epsilon_l$ & $\left\lvert\overline{\beta}_q \pm 1\right\rvert > \epsilon_l$, with $m, p, q \in \{1,\dots,n\}$, $m \neq p \neq q$
T_{2m-1} & T_{2p-1}	$\overline{\beta}_0 > 0$ & $\left\lvert\overline{\beta}_m + 1\right\rvert \leq \epsilon_l$ & $\left\lvert\overline{\beta}_p + 1\right\rvert \leq \epsilon_l$ & $\left\lvert\overline{\beta}_q - 1\right\rvert \leq \epsilon_l$, with $m, p, q \in \{1,\dots,3\}$, $m \neq p \neq q$
	$\overline{\beta}_0 > 0$ & $\left\lvert\overline{\beta}_m + 1\right\rvert \leq \epsilon_l$ & $\left\lvert\overline{\beta}_p + 1\right\rvert \leq \epsilon_l$ & $\left\lvert\overline{\beta}_q \pm 1\right\rvert > \epsilon_l$, with $m, p, q \in \{1,\dots,n\}$, $m \neq p \neq q$, $n > 3$
T_{2m} & T_{2p}	$\overline{\beta}_0 > 0$ & $\left\lvert\overline{\beta}_m - 1\right\rvert \leq \epsilon_l$ & $\left\lvert\overline{\beta}_p - 1\right\rvert \leq \epsilon_l$ & $\left\lvert\overline{\beta}_q + 1\right\rvert \leq \epsilon_l$, with $m, p, q \in \{1,\dots,n\}$, $m \neq p \neq q$
	$\overline{\beta}_0 > 0$ & $\left\lvert\overline{\beta}_m - 1\right\rvert \leq \epsilon_l$ & $\left\lvert\overline{\beta}_p - 1\right\rvert \leq \epsilon_l$ & $\left\lvert\overline{\beta}_q \pm 1\right\rvert > \epsilon_l$, with $m, p, q \in \{1,\dots,n\}$, $m \neq p \neq q$, $n > 3$
T_{2m-1} & T_{2m}	$\overline{\beta}_0 \leq 0$ & $\left\lvert\overline{\beta}_m\right\rvert \leq \epsilon_l$

3.4. Fault detection and isolation

In this subsection, the fault indicator chosen in [4] is briefly presented. To reduce the possibility of false alarms, fault declaration is investigated from the sliding window $\mathbb{B}(t) = \{\boldsymbol{\beta}(t - (N_f - 1)t_s), \dots, \boldsymbol{\beta}(t)\}$ instead of from a single feature vector $\boldsymbol{\beta}(t)$, where $N_f \in \mathbb{N}$ is the window width. This sequence is constructed online in the "Data buffer" procedure (see Fig. 3). Afterwards, it is sent to the "Similarity verification" procedure, where the closeness of its vectors and reference classes is described by an indicator matrix $\mathbf{M}(t) \in \{0,1\}^{N_f \times n_p(t)}$, with the number of reference classes being $n_p(t)$. Based on this matrix, the "Maximum similarity" procedure derives the maximum number of vectors of $\mathbb{B}(t)$ which are close to a reference class. This is equal to the maximum absolute column sum norm of the matrix $\mathbf{M}(t)$. Next, this number is used as the fault indicator $F(t)$, i.e., $F(t) = \|\mathbf{M}(t)\|_1$. Finally, the indicator and the class identity are sent to the "Fault declaration" procedure. Class $\mathbb{C}(t)$ represents the

system operation mode if $F(t) \geq F_f$, where $F_f \in \mathbb{N}$ is a chosen fault threshold. Otherwise, the system operation mode is unknown.

Some criteria to evaluate the proposed algorithm are presented in the following remarks: accuracy, tuning effort, complexity and detection time.

Remark 2 (Accuracy)**.** *The influence of white noise is drastically reduced thanks to the use of phase current averages for the feature extraction, of Gaussian Mixture Models for the feature clustering and the consideration of a number of consecutive feature vectors for the fault declaration. Moreover, the latter also helps to reduce the mode confusion possibility caused by the passage of evolving reference classes through different labelling regions in the feature space. Obviously, the algorithm accuracy depends on appropriate values of the parameters. A comprehensive investigation on the algorithm accuracy should take into account a lot of outlier cases, e.g., variations of mechanical torque and/or of system parameters, different control systems, etc.*

Remark 3 (Tuning effort)**.** *Although the proposed algorithm does not require choosing a threshold in the feature extraction procedure, there are still at least 7 parameters to tune: the labelling threshold ϵ_l, the initial covariance σ, the class membership threshold μ_{min}, the minimum and maximum number of class elements N_{min}, N_{max}, the sliding window width N_f and the fault threshold F_f. This requires an important effort for the parameter tuning. However, some ranges for choosing effective values of these parameters will be hereafter presented in Table 3, which may simplify the tuning task. Moreover, these parameters add extra freedom to the FDI algorithm to deal with the system's dynamical evolution in the real-world and the unknown impact of outliers.*

Remark 4 (Complexity)**.** *Generally, a data-driven approach for the FDI problem requires a larger computational effort than model-based and signal-based approaches. However, it usually gives high accuracy diagnosis results for the case of complex systems with different perturbations. In our work, the complexity of the machine learning technique is reduced thanks to a small number of feature variables and self learning abilities.*

Remark 5 (Detection time)**.** *The detection time is defined as the duration from the appearance of a fault to the fault isolation. Since many procedures are employed to improve the algorithm accuracy, the detection times of data-driven approaches are generally higher than of model-based and signal-based approaches. In the case where the reconfiguration of the system operation is decided by the system operator, a detection time shorter than one second may not be meaningful since the operator needs more time to perform the said action. However, with the perspective of integrating the FDI algorithm to the control system, reducing the detection time is still an important issue in the algorithm design. This motivates investigating the influence of the method parameters on the detection time in the next section.*

4. Influence of the method parameters on the detection time

Here, the influence of the method parameters on the detection time is studied when there is no mode confusion. Since a general consideration for this problem is hard to achieve, the following assumptions are made:

(A1) there are the normal mode and a faulty mode in a scenario,

(A2) for an operation mode, only one class is created,

(A3) class labels are correct,

(A4) at a class creation moment, its stored vectors are consecutive,

(A5) all feature vectors of an operation mode arriving after the creation of the corresponding class stay in this class,

(A6) at the fault isolation moment, the number of vectors stored in the faulty class is less than the sliding window width,

(A7) at the faulty class creation time, the number of feature vectors, which are not close to the normal class, is greater than the minimum class cardinality N_{min}.

Let t_C, t_L and t_d be the durations from the appearance of a fault to the following moments: i) the faulty class creation, ii) the faulty class labelling, iii) the fault isolation, respectively. According to Section 3, the presented time durations respect the inequality $t_C \leq t_L \leq t_d$. To motivate the subsequent results, the following property of a discrete-time monotonic function is recalled.

Remark 6 ([20]). *Let $t_\epsilon(\epsilon_g)$ be a function defined as $g(t_\epsilon(\epsilon_g) - t_s) < \epsilon_g \leq g(t_\epsilon(\epsilon_g))$, where $\epsilon_g \in (g(t_1), g(t_2))$, the function $g(t)$ is strictly increasing over the time interval (t_1, t_2), and t_s is the sampling time. $t_\epsilon(\epsilon_g)$ is therefore monotonically increasing. On the contrary, if $g(t)$ is strictly decreasing and $g(t_\epsilon(\epsilon_g)) \leq \epsilon_g < g(t_\epsilon(\epsilon_g) - t_s)$, $t_\epsilon(\epsilon_g)$ is monotonically decreasing.*

The influence of the window width N_f, the fault threshold F_f, the labelling threshold ϵ_l and the initial covariance σ is previously presented in [20]. $t_d(N_f)$ and $t_d(F_f)$ are increasing. The function $t_L(\epsilon_l)$ is monotonically decreasing, if

$$g_\epsilon(t) = \max_{q=1,\dots,n} \left\{ \left| \overline{\beta}_{f,q}(t) \right| \right\} \tag{3}$$

is strictly increasing, where $\bar{\boldsymbol{\beta}}_f(t)$ is the mean vector of the faulty mode class. $t_C(\sigma)$ is monotonically decreasing, if

$$g_\sigma(t) = -\frac{1}{2ln(\mu_{min})} \max_{\boldsymbol{\beta} \in \mathbb{B}_m(t)} \left\| \boldsymbol{\beta} - \bar{\boldsymbol{\beta}}_m(t) \right\|_2 \tag{4}$$

is strictly decreasing, where $\mathbb{B}_m(t) = \{\boldsymbol{\beta}(t - N_{min}t_s + t_s), \ldots, \boldsymbol{\beta}(t)\}$ is the time sequence of N_{min} consecutive feature vectors, and $\bar{\boldsymbol{\beta}}(t)$ is the mean vector of $\mathbb{B}_m(t)$.

Here, the influence of the membership threshold μ_{min} and the minimum class cardinality N_{min} on the detection time is presented in the following proposition.

Proposition 1. *Let $g_\mu(t),\ g_N(t) \in \mathbb{R}$ be given as:*

$$\begin{cases} g_\mu(t) = & exp\left(-\dfrac{1}{2\sigma^2} \underset{\boldsymbol{\beta}\in\mathbb{B}_m(t)}{max} \left\|\boldsymbol{\beta} - \bar{\boldsymbol{\beta}}_m(t)\right\|_2\right), \\ g_N(t) = & \underset{r\in\{1,\ldots,N_{max}\}}{max} r \\ & s.t.\ exp\left(-\dfrac{1}{2\sigma^2}\left\|\boldsymbol{\beta} - \bar{\boldsymbol{\beta}}_r(t)\right\|_2\right) \geq \mu_{min},\ \forall \boldsymbol{\beta} \in \mathbb{B}_r(t), \end{cases} \tag{5}$$

where $\mathbb{B}_r(t) = \{\boldsymbol{\beta}(t - rt_s + t_s), \ldots, \boldsymbol{\beta}(t)\}$ is the time sequence of $r \in \mathbb{N}$ consecutive feature vectors; $\bar{\boldsymbol{\beta}}_r(t)$ is the mean vector of $\mathbb{B}_r(t)$.

(P1) If $g_\mu(t)$ is strictly increasing, $t_C(\mu_{min})$ is monotonically increasing.

(P2) If $g_N(t)$ is strictly increasing, $t_C(N_{min})$ is monotonically increasing.

Note that this proposition is only valid for Case 1, 3 and 4 of the OC faults.

Proof.

(P1) According to Proposition 1 in [20], the faulty class is created based on the sequence $\mathbb{B}_m(t_C(\mu_{min}))$. Assume that $g_\mu(t)$ is strictly increasing, and $g_\mu(t) < \mu_{min},\ \forall t < t_C(\mu_{min})$. This implies that each vector in the sequence $\mathbb{B}_m(t)$ is far from others according to the membership function in [4] and (5). Therefore, these vectors can not be used to create a class. Until $t = t_C(\mu_{min})$, $g_\mu(t_C(\mu_{min}) - t_s) < \mu_{min} \leq g_\mu(t_C(\mu_{min}))$. Therefore, each vector of $\mathbb{B}_m(t)$ is close enough to the others. Hence, the faulty class is created, and all vectors of $\mathbb{B}_m(t_C(\mu_{min}))$ are stored in the new class. Consequently, $t_C(\mu_{min})$ is monotonically increasing according to Remark 6.

(P2) From (2) and (5), when $g_N(t) < N_{min}$, each feature vector in the sequence $\mathbb{B}_m(t)$ is not close enough to all others. Therefore, the faulty class can not be created. Assume that $g_N(t)$ is strictly increasing, $g_N(t) < N_{min}$ with $t < t_C(N_{min})$. Until $t = t_C(N_{min})$, $g_N(t_C(N_{min}) - t_s) < N_{min} \leq g_N(t_C(N_{min}))$, and thus, the feature vectors in $\mathbb{B}_m(t)$ are close enough to create the faulty class. Hence, according to Remark 6, $t_C(\mu_{min})$ is monotonically increasing. □

Note that if there is no mode confusion, $N_{min} \geq \max\{N_f/2, F_f\}$ and $\epsilon_l \geq 1 - g_\mu(t_C)$, then, $t_d = t_L = t_C$ [20]. There is still the maximum class cardinality N_{max} which is not mentioned in Proposition 1. Indeed, with Assumption (A7), this parameter has no influence on the detection time since the faulty class

creation time does not change with its different values. However, it may affect the mode confusion.

The presented algorithm and the influence of its parameters are validated on experimental data in the following section.

5. Fault detection and isolation results

In this section, the proposed FDI algorithm is applied to experimental data from 5 different scenarios. In Scenario 1-4, the data are from an open loop system containing a 3-phase inverter governing a 3-phase Resistance-Inductance series circuit with star coupling (see Fig. 5). In Scenario 5, the data provided by the L2EP laboratory in Lille, France, are of a closed loop system containing a 5-phase inverter governing a 5-phase synchronous motor (detail of the system can be found in [3]). A natural variable $fault(t) \in \mathbb{N}$ is used to simply describe the mode flag. If the operation mode is not isolated, this variable is set to -11. With Scenario 1, the influence of the method parameters on the detection time is investigated for validating Proposition 1 and the similar one in [20]. Some properties of the experimental systems used for the FDI and the values of the algorithm parameters are described in Table 3. Table 4 presents the operation modes, their labels and the detection times in the considered scenarios. The FDI algorithm is implemented in MATLAB®.

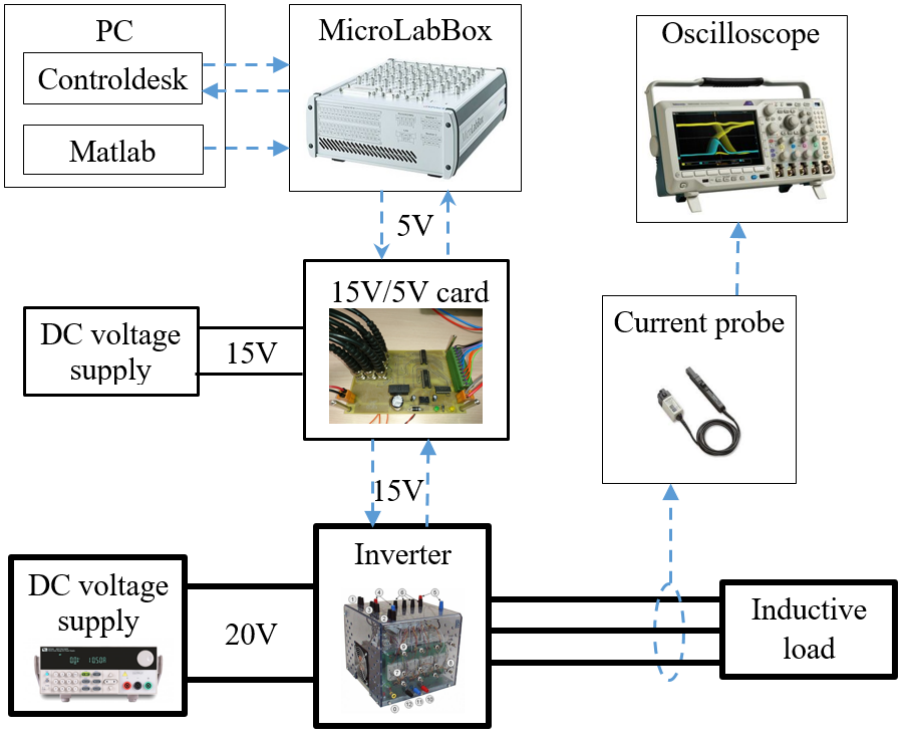

Figure 5: Test bench scheme for the studied 3-phase system.

Table 3: Parameters for the data and the algorithm.

Parameter\Scenario	All	1-5
Data		
Signal frequency [Hz]		50
Sample time t_s [μs]		80
Algorithm		
Class membership threshold μ_{min}	$0 < \mu_{min} < 1$	0.61
Initial covariance σ	$0 < \sigma < 1$	0.8
Minimum class cardinality N_{min}	$N_{min} = O(N_s)$	170
Maximum class cardinality N_{max}	$N_{max} = O(N_s)$	200
Labelling threshold ϵ_l	$0 < \epsilon_l \leq 0.5$	0.2
Sliding window width N_f	$N_f = O(N_s)$	220
Fault threshold F_f	$F_f = O(N_s)$	170

Table 4: Detection times in Scenarios 1-5.

Scenario	1		2		3		4	
Duration [s]	0.4	0.2	0.4	0.2	0.4	0.2	0.4	0.2
Fault	None	T_1	None	T_1&T_2	None	T_1&T_3	None	T_1&T_4
$fault_{ref}$	00	10	00	12	00	13	00	14
Det. time [ms]		30.5		28.2		28.0		32.0

Scenario	5							
Duration. [s]	0.25	0.25	0.25	0.25	0.25	0.25		
Fault	None	T_4	T_3	T_2	T_1	T_3&T_4		
$fault_{ref}$	00	40	30	20	10	34		
Det. time [ms]		24.8	31.9	25.0	31.8	27.9		

5.1. Scenario 1-4: Fault detection and isolation on a 3-phase inverter

Fig. 2 and Fig. 4 illustrate the profiles of the 3-phase currents and the feature variables, respectively. In these figures, the feature variables have different stable values for different modes. Moreover, the values of $\beta_0(t)$ in Scenario 2 (phase fault) are separated by zero instead of a varying threshold as in [20] (see the feature variable profile in row 4, column 2 of Fig. 4). Fig. 6 illustrates the profiles of the mean vectors $\bar{\beta}(t)$ of the normal and faulty classes, the fault indicator $F_f(t)$ and the fault flag $fault(t)$. Based on these profiles, the labelling conditions in Table 2 and the fault isolation condition in Section 3.4 can be verified. Notice that with the algorithm parameters given in Table 3, the faulty class is labelled as soon as it is created. According to this table, the detection times are in the range of [1,1.6] times the fundamental signal period ($t_s = 20$ ms).

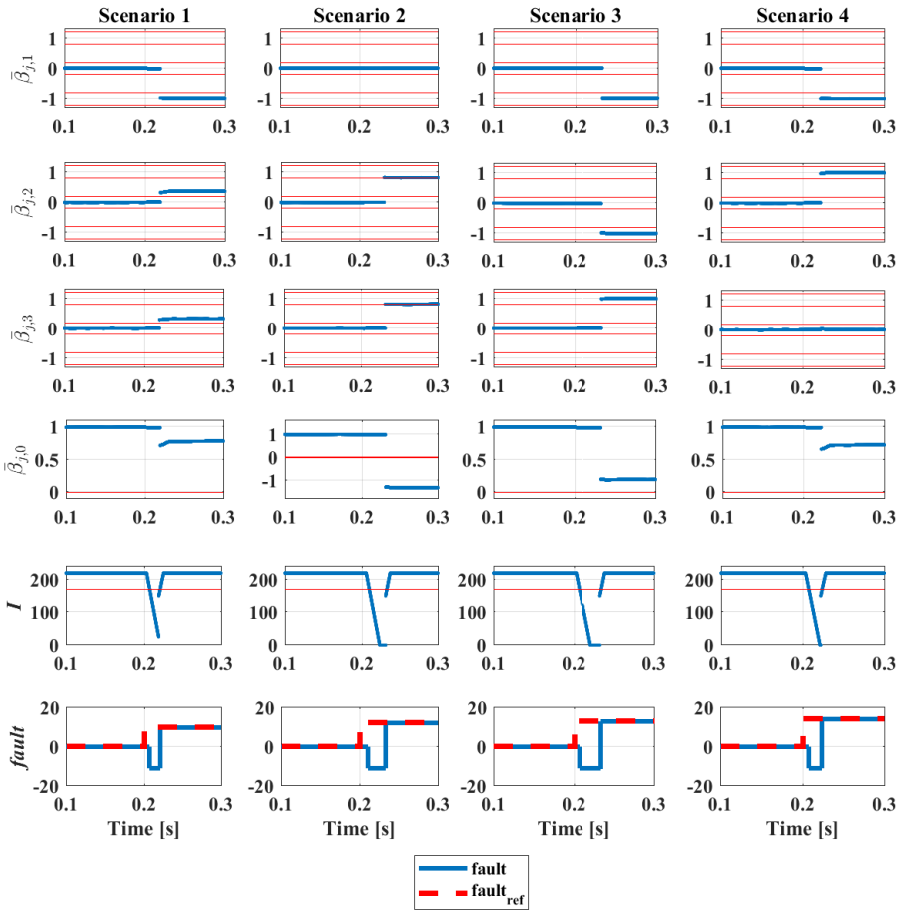

Figure 6: Profiles of the class mean vectors, the fault indicator and the fault flag in Scenarios 1-4.

5.2. Scenario 5: Fault detection and isolation on a 5-phase inverter

Figure 7 describes the profiles of the 5-phase currents and the feature variables in Scenario 5. In this figure, there are oscillations in the profiles of the feature variables during the system operation. This is due to the uncertainty in the signal frequency caused by the closed loop controller. Indeed, in (1), the number of time samples on a fundamental signal period $N_s = 1/(ft_s)$ depends on the signal frequency f. Since f is not constant, N_s and, thus, the feature variables are also varying. Fig. 8 describes the profiles of the class means of the normal and faulty modes $\bar{\beta}(t)$, the fault indicator F_f and the fault flag $fault(t)$.

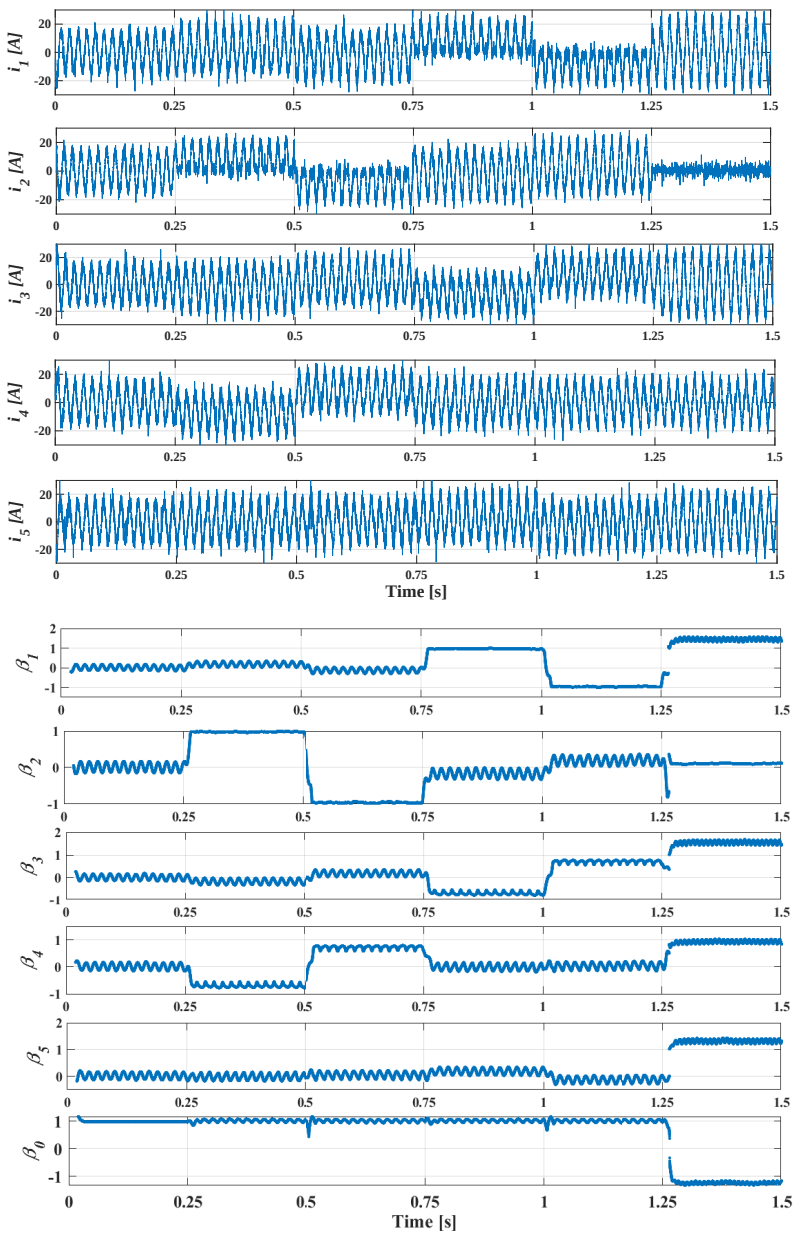

Figure 7: Profiles of the phase currents and the feature variables in Scenario 5.

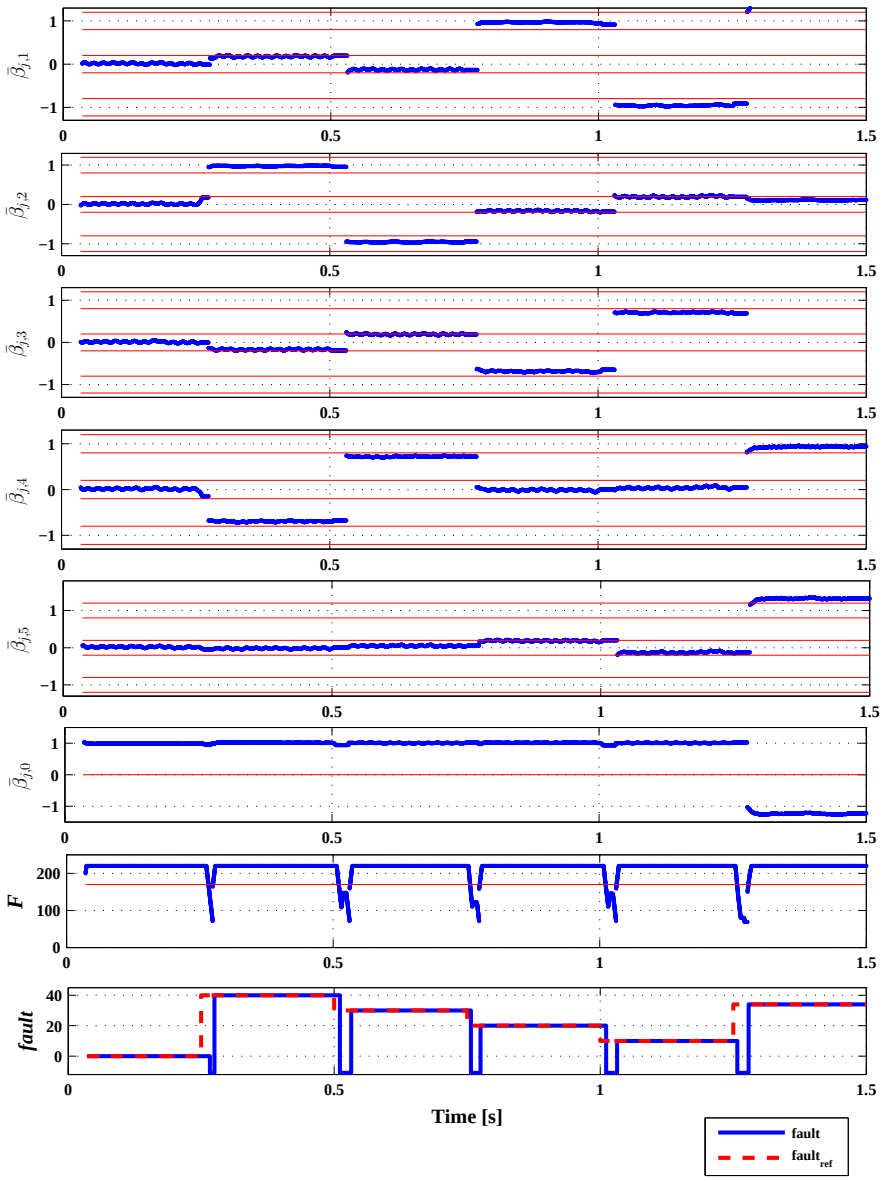

Figure 8: Profiles of the class mean vectors, fault indicator and fault flag in Scenario 5.

5.3. Scenario 1: Influence of the method parameters on the detection time

This section aims at illustrating the conclusions in Section 4 about the influence of the method parameters on the detection time (Table 3). Note that the assumptions A1-A7 hold for parameter values investigated in this subsection.

Table 5 presents the detection times in Scenario 1 with several values of the fault threshold F_f and sliding window width N_f. The detection times $t_d(F_f)$ and

$t_d(N_f)$ are monotonically increasing. Moreover, when $\max\left\{F_f, \frac{N_f}{2}\right\} \leq N_{min}$, they are constant since the fault is isolated at the faulty class labelling, i.e., $t_d(F_f, N_f) = t_L$. Note that t_L does not depend on F_f and N_f (see Subsection 3.3).

Table 5: Detection times [ms] in Scenario 1 with several values of F_f and N_f.

$F_f \backslash N_f$	220	300	340	380	420
160	30.5	30.5	30.5	32.0	33.7
170	30.5	30.5	30.5	32.0	33.7
180	31.1	31.1	31.1	32.0	33.7
190	31.9	31.9	31.9	32.0	33.7
200	32.8	32.8	32.8	32.8	33.7
220	34.5	34.5	34.5	34.5	34.5

Table 6 lists the labelling times in Scenario 1 with several values of the labelling threshold ϵ_l. The function $g_\epsilon(t)$ in (3) is strictly increasing, which is partially illustrated through its positive discrete-time derivative $\Delta g_\epsilon(t)/\Delta t$. Hence, the labelling time $t_L(\epsilon)$ is monotonically decreasing. In these tests, $N_{min} = 50$ is considered instead of 170 since the faulty class is labelled after its creation, i.e., $t_L(\epsilon_l) > t_C$. Note that t_C does not depend on ϵ_l (see Subsection 3.2).

Table 6: Labelling times [ms] in Scenario 1 with several values of ϵ_l with $N_{min} = 50$.

ϵ_l	0.1	0.15	0.20	0.25	0.3
$\Delta g_\epsilon(t)/\Delta t$ $[s^{-1}]$	5.8	16.4	32.0	53.7	53.7
$t_L(\epsilon)$ [ms]	28.2	23.2	21.1	19.9	19.9

Table 7 illustrates the class creation times in Scenario 1 for different values of the initial covariance σ, membership threshold μ_{min} and minimum class cardinality N_{min}. The function $g_\sigma(t)$ in (4) is strictly decreasing, which is partially illustrated through its negative discrete-time derivative $\Delta g_\sigma(t)/\Delta t$. Hence, the faulty class creation time $t_C(\sigma)$ is monotonically decreasing. Similarly, since the functions $g_\mu(t)$ and $g_N(t)$ in (5) are strictly increasing, the faulty class creation times $t_C(\mu)$ and $t_C(N_{min})$ are monotonically increasing according to Proposition 1.

Table 7: Class creation times [ms] in Scenario 1 for different values of σ, μ_{min} and N_{min}.

σ	0.1	0.2	0.3	0.4
$\Delta g_\sigma(t)/\Delta t$ $[s^{-1}]$	-130	-159	-151	-137
$t_C(\sigma)$ [ms]	30.7	30.1	29.4	28.7
$\sigma = 0.2$				
μ_{min}	0.15	0.30	0.45	0.61
$\Delta g_\mu/\Delta t$ $[s^{-1}]$	16	46	104	196
$t_C(\mu)$ [ms]	28.8	29.3	29.7	30.1
N_{min}	130	140	150	160
$\Delta g_N/\Delta t$ $[s^{-1}]$	12444	12444	12444	12444
$t_C(N_{min})$ [ms]	28.4	29.4	30.5	31.6

6. Conclusion

This paper presents several improvements to the data-driven fault detection and isolation algorithm previously proposed for double open-circuit faults in 3-phase inverters [20]. In this manuscript, the algorithm has been extended to deal with the multi-phase case and be applicable to experimental data. The features are firstly extracted from the measurements of the load currents. Then, the feature vectors are clustered using the Auto-adaptive and Dynamical Clustering. Using the obtained classes, the conditions for the labelling, fault detection and isolation are formulated. Briefly, the main contributions of this paper are summarized as follows: i) a new feature variable was proposed to avoid using a threshold in the feature extraction procedure; ii) labelling conditions were developed for the multi-phase inverter case; iii) rigorous conditions were formulated for describing the influence of all algorithm parameters on the detection time. Some elements of these conditions can be used in other applications of AUDyC. With the presented improvements the accuracy of the proposed method was illustrated through experimental data from real systems with several scenarios. As future work, we aim at generalizing the method to multi (more than 2) Open-Circuit fault cases and evaluating the algorithm complexity. Moreover, the proposed approach will be compared to the state of the art and will be applied to data from an industrial system.

Acknowledgement

This work has been achieved within the framework of the CE2I project (Convertisseur d'Energie Intégré Intelligent). CE2I is co-financed by European Union with the financial support of European Regional Development Fund (ERDF), the French State and the French Region of Hauts-de-France. The authors would like to thank Tiago Dos Santos Moraes, Eric Semail and Ngac Ky Nguyen from the L2EP laboratory in Lille (France) for providing the experimental data used in Scenario 5.

Research Data

The data that support the findings of this study are available on request from the corresponding author.

References

[1] E. Levi, Multiphase Electric Machines for Variable-Speed Applications, IEEE Transactions on Industrial Electronics 55 (5) (2008) 1893–1909.

[2] B. Cai, Y. Zhao, H. Liu, M. Xie, A data-driven fault diagnosis methodology in three-phase inverters for PMSM drive systems, IEEE Transactions on Power Electronics 32 (7) (2017) 5590 – 5600.

[3] M. Trabelsi, E. Semail, N. K. Nguyen, Experimental investigation of inverter open-circuit fault diagnosis for bi-harmonic five-phase permanent magnet drive, IEEE Journal of Emerging and Selected Topics in Power Electronic 6 (1) (2018) 339 – 351.

[4] T.-H. Pham, S. Lefteriu, C. Labarre, E. Duviella, S. Lecoeuche, Auto-adaptive and Dynamical Clustering for open-circuit fault diagnosis of power inverters, in: European Control Conference, IEEE, 2019, pp. 3298 – 3303.

[5] Z. Gao, C. Cecati, S. X. Ding, A survey of fault diagnosis and fault-tolerant techniques—part II: Fault diagnosis with knowledge-based and hybrid/active approaches, IEEE Transactions on Industrial Electronics 62 (6) (2015) 3768 – 3774.

[6] C. M. Bishop, Pattern Recognition and Machine Learning, Springer-Verlag, 2006.

[7] C. Delpha, D. Diallo, H. A. Samrout, N. Moubayed, Multiple incipient fault diagnosis in three-phase electrical systems using multivariate statistical signal processing, Engineering Applications of Artificial Intelligence 73 (2018) 68 – 79.

[8] Y. Zeng, Z. Jia, W. Liang, S. Gu, Fault diagnosis based on variable-weighted separability-oriented subclass discriminant analysis, Computers & Chemical Engineering 129 (2019) 106514.

[9] D. He, R. Li, J. Zhu, Plastic Bearing Fault Diagnosis Based on a Two-Step Data Mining Approach, IEEE Transactions on Industrial Electronics 60 (8) (2013) 3429–3440.

[10] S. Khomfoi, L. M. Tolbert, Fault Diagnosis and Reconfiguration for Multilevel Inverter Drive Using AI-Based Techniques, IEEE Transactions on Industrial Electronics 54 (6) (2007) 2954–2968.

[11] G. Chen, Y. Liu, Z. Ge, K-means Bayes algorithm for imbalanced fault classification and big data application, Journal of Process Control 81 (2019) 54–64.

[12] Z. Zheng, W. Jiang, Z. Wang, Y. Zhu, K. Yang, Gear fault diagnosis method based on local mean decomposition and generalized morphological fractal dimensions, Mechanism and Machine Theory 91 (2015) 151–167.

[13] Y. Liu, Z. Ge, Weighted random forests for fault classification in industrial processes with hierarchical clustering model selection, Journal of Process Control 64 (2018) 62–70.

[14] H. Li, W. Wang, P. Huang, Q. Li, Fault diagnosis of rolling bearing using symmetrized dot pattern and density-based clustering, Measurement 152 (2020) 107293.

[15] Y. Hong, M. Kim, H. Lee, J. J. Park, D. Lee, Early Fault Diagnosis and Classification of Ball Bearing Using Enhanced Kurtogram and Gaussian Mixture Model, IEEE Transactions on Instrumentation and Measurement 68 (12) (2019) 4746–4755.

[16] L. Meng, P. Wang, Z. Liu, R. Qiu, L. Wang, C. Xu, Safety assessment for electrical motor drive system based on SOM neural network, Mathematical Problems in Engineering 2016, 8 pages.

[17] S. Lecoeuche, C. Lurette, S. Lalot, New supervision architecture based on on-line modelling of non-stationary data, Neural Computing & Applications 13 (4) (2004) 323 – 338.

[18] H. A. Boubacar, S. Lecoeuche, S. Maouche, AUDyC Neural Network using a new Gaussian Densities Merge Mechanism, in: Adaptive and Natural Computing Algorithms, Springer, 2005, pp. 155 – 158.

[19] H. Toubakh, M. Sayed-Mouchaweh, Hybrid dynamic classifier for drift-like fault diagnosis in a class of hybrid dynamic systems: Application to wind turbine converters, Neurocomputing 171 (2016) 1496 – 1516.

[20] T.-H. Pham, S. Lefteriu, E. Duviella, S. Lecoeuche, Auto-adaptive and Dynamical Clustering for double open-circuit fault diagnosis of power inverters, in: 4th International Conference on Control and Fault-Tolerant Systems, IEEE, 2019, pp. 306 – 311.

[21] Y. Xia, Y. X, B. Gou, A data-driven method for IGBT open-circuit fault diagnosis based on hybrid ensemble learning and sliding-window classification, IEEE Transactions on Industrial Informatics (2019) 1–1.

[22] Q. Sun, Y. Wang, Y. Jiang, A novel fault diagnostic approach for DC-DC converters based on CSA-DBN, IEEE Access (2017) 6273 – 6285.