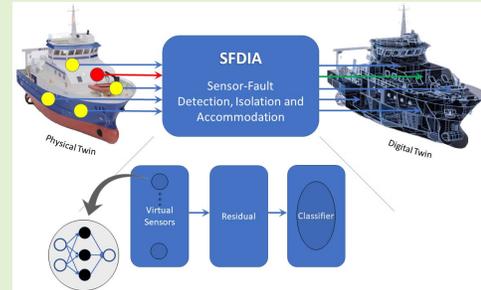


Sensor-Fault Detection, Isolation and Accommodation for Digital Twins via Modular Data-Driven Architecture

Hossein Darvishi^{ID}, Graduate Student Member, IEEE, Domenico Ciunzo^{ID}, Senior Member, IEEE, Eivind Rosón Eide, and Pierluigi Salvo Rossi^{ID}, Senior Member, IEEE

Abstract—Sensor technologies empower Industry 4.0 by enabling integration of in-field and real-time raw data into digital twins. However, sensors might be unreliable due to inherent issues and/or environmental conditions. This article aims at detecting anomalies in measurements from sensors, identifying the faulty ones and accommodating them with appropriate estimated data, thus paving the way to reliable digital twins. More specifically, we propose a general machine-learning-based architecture for sensor validation built upon a series of neural-network estimators and a classifier. Estimators correspond to virtual sensors of all unreliable sensors (to reconstruct normal behaviour and replace the isolated faulty sensor within the system), whereas the classifier is used for detection and isolation tasks. A comprehensive statistical analysis on three different real-world data-sets is conducted and the performance of the proposed architecture validated under hard and soft synthetically-generated faults.

Index Terms—Digital twin, fault tolerance, Industry 4.0, Internet of Things, machine learning, sensor validation.



I. INTRODUCTION

INDUSTRY 4.0 identifies the current fourth industrial revolution, whose aim is an increased level of automation through the effective combination of the Internet of Things (IoT), cyber-physical systems and cloud computing technologies [2]. Within this concept, sensors play a crucial role by measuring different physical parameters, thus enabling monitoring, controlling and decision-support capabilities [3]. While systems are highly dependent on data collected by sensors, the latter are unfortunately prone to errors. These errors can occur because of several reasons such as a harsh working environment, low battery level, limited life span (aging),

Manuscript received September 10, 2020; accepted October 3, 2020. Date of publication October 7, 2020; date of current version January 15, 2021. This work was supported in part by the Research Council of Norway within the framework IKTPLUSS under Project 311902 (SIGNIFY). This article was presented in part at the IEEE SENSORS 2020 Conference, Rotterdam, The Netherlands, October 2020. The associate editor coordinating the review of this article and approving it for publication was Prof. Okyay Kaynak. (Corresponding author: Hossein Darvishi.)

Hossein Darvishi and Pierluigi Salvo Rossi are with the Department of Electronic Systems, Norwegian University of Science and Technology, 7491 Trondheim, Norway (e-mail: hossein.darvishi@ntnu.no; salvorossi@ieee.org).

Domenico Ciunzo is with the Department of Electrical Engineering and Information Technologies, University of Naples Federico II, 80125 Naples, Italy (e-mail: domenico.ciunzo@unina2.it).

Eivind Rosón Eide is with the Department of Advanced Analytics and Machine Learning, Kongsberg Digital AS, 1383 Oslo, Norway (e-mail: eivind.roson.eide@kdi.kongsberg.com).

Digital Object Identifier 10.1109/JSEN.2020.3029459

improper calibration and hardware failures [4]. Corrupted data from sensors with failures may negatively affect both simple and more advanced functionalities of the system and result in overall system performance degradation and increased risk level. This would lead to consequences ranging from financial losses to serious safety issues (including life losses).

Reliable sensor measurements are vital for effective control and action-taking chain, and early reaction to faulty scenarios plays a critical role in risk management strategies while increasing safety and reliability. More specifically, a properly-working system should be able to perform: (i) *detection* (promptly detecting a fault condition within the system); (ii) *isolation* (identifying the faulty sensor) and (iii) *accommodation* (replacing the faulty data with some other trusted data). Accordingly, in this article we propose a machine-learning-based framework for sensor validation. This framework allows developing a general sensor-fault detection, isolation, and accommodation (SFDIA) scheme to be easily adapted to different application domains, e.g. renewables in maritime scenarios [5]. In detail, the *contributions* of this article are:

- 1) A novel machine-learning-based architecture for SFDIA is proposed. The proposed architecture jointly takes advantage of the temporal correlation of the measurements and of both reliable and unreliable sensors within the system to achieve a higher sensor validation performance.

- 2) The focus of generated faults is on *weak faults*, which are very hard to detect and usually ignored in the literature [6]–[10].
- 3) The performance of the proposed approach (in terms of probabilities of detection, false alarm, correct classification, misclassification, etc.) is evaluated on *three* different real-world data-sets [11]–[13] corrupted with synthetically-generated sensor faults (bias and drifts) and compared with two state-of-the-art techniques [14], [15]. The data-sets considered are *publicly-available*: this fosters reproducibility and further advances on the topic. Synthetically-generated sensor faults have been considered to perform a systematic performance assessment of the proposed architecture.
- 4) The impact of different hyperparameters, such as the number of layers and the number of nodes per layer, is assessed for the considered scenarios.

The rest of this article is organized as follows. Sec. II provides a literature review regarding the related work. In Sec. III we introduce the proposed general SFDIA architecture and describe the different blocks for fault detection, isolation and accommodation. Then, in Sec. IV, we present the data description, contamination and pre-processing related to three independent data-sets with different applications. Accordingly, Sec. V highlights and compares the numerical performance for all the data-sets with different setups.

Finally, in Sec. VI we provide some concluding remarks and highlight future directions of research.

Notation: Lower-case bold letters denote vectors, $(\cdot)^T$ is the transpose operator, and $\mathcal{O}(\cdot)$ indicates the Landau notation.

II. RELATED WORKS

First practices for sensor validation were based on *hardware redundancy* [16]. These approaches used multiple sensors to measure the same parameter at the same point as well as a voting scheme to compensate sensors faults [16], [17]. However, hardware redundancy is unable to handle system noise and has some other serious drawbacks in terms of cost, weight, power consumption and size. Even more importantly, it is sensitive to simultaneous failure of all redundant sensors subject to the same harsh environmental conditions. Due to these reasons, alternative approaches based on *analytical redundancy* have gained more attention. Analytical-redundancy approaches attempt to develop reliable virtual sensors based on system model(s). More specifically, measurements collected by real sensors are compared with the values from the virtual ones to detect presence of faults and provide reliable measurements for replacement [9], [15], [18]. Various model-based and model-free (viz. data-driven) algorithms such as Kalman filter (KF) [19], [20], hidden Markov model [21], artificial neural networks (NN) [7], [22], and support vector machine (SVM) [14] have focused on detection and isolation tasks with application on aircraft sensor technologies, cyber-physical systems and wireless sensor networks (WSNs).

Early KF-based algorithms for detection and isolation were developed with an inherited drawback of being unable to deal with non-linearities [19]. Extended KF and multiple hybrid

KFs were shown to overcome this issue through linearization around the state estimate and piece-wise linear models, respectively [20], [23]. Nevertheless, such solutions were heavily dependent on domain knowledge about the system which is not necessarily available.

As for data-driven approaches, multi-layer perceptron (MLP) architectures were considered for reducing probabilities of false alarm and miss detection through time-variant thresholds-based tests [22]. A method based on the SVM classifier was also proposed to detect faults through abnormal behaviors in the last three data measurements [14]. However, this method makes decision using redirected data to the server which results in delayed fault detection. Since the SVM classifier was only able to classify the faulty data, a deep belief network [7] coupled with a maximum squared error method for fault detection and isolation purposes was investigated. To address large data requirement of data-driven approaches, fault detection and isolation filters were derived in the state-space representation form by estimating system impulse response coefficients in the frequency domain via fast Fourier transform of input/output signals [24].

In the context of industrial WSNs, a threshold-free error detection (TED) method was developed [25]. TED relies on both temporal and spatial correlation between sensor readings. Recently, a method named TPE-FTED [10] based on an adjustable step window was proposed for online learning the changes of sensor readings in a dynamic environment. TPE-FTED deals with fault detection and isolation problem as a trajectory pattern extraction problem extracted from different sensing states. Then, TPE-FTED starts pattern matching as well as spatial-temporal constraint violation checking to detect the faulty sensor.

In summary, model-based algorithms require good knowledge of system model/ parameters and are difficult to implement in presence of nonlinearities. Conversely, data-driven algorithms may represent a valid alternative to analytical model-based algorithms: ease of implementation and capabilities to capture non-linear behavior by learning from historical data have increased attention toward data-driven algorithms for SFDIA schemes [8], [9], [15], [26]–[28].

An SFDIA scheme based on MLPs by consociating one main NN and a set of decentralized NNs has been proposed to create a system for detecting failures of gyro sensors of an aircraft [26]. Previous-time measurements of sensors under estimation were also used as the input of MLP NNs. A minimal radial basis function (MRAN) NN presented in [27] was able to reduce NN complexity by ignoring hidden neurons with less effect on the NN output. This algorithm was relatively slow in detecting faults after the occurrence of the faults. The performance of MLP and Extended MRAN NNs on sensor failure accommodation scheme were evaluated and compared through a study for failure on air data system [28]. This study showed similar performance of both NNs as online estimators, with slightly better performance of MLP NN in the training phase. SFDIA scheme presented in [15] employed a fully connected cascade (FCC) NN with only one neuron per layer connected to all previous layers. The proposed FCC NN was able to perform efficiently with a limited number of neurons

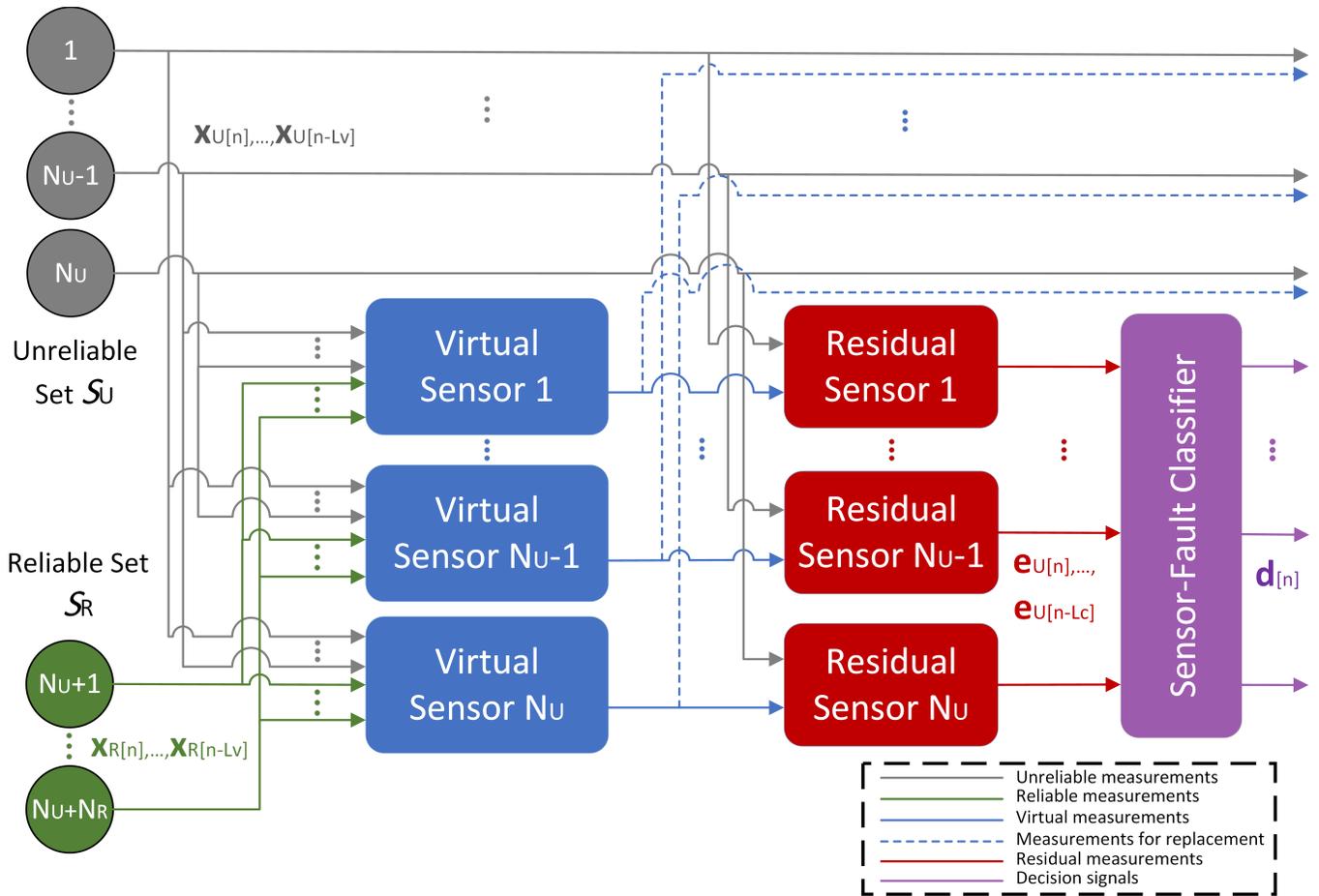


Fig. 1. Block diagram of the SFDIA system.

and reduced computational complexity in comparison to MLP NN.

A NN-based sensor validation scheme for heavy-duty diesel engines was proposed using two banks of NN approximators to generate a residual signal for isolating faults and to produce an approximation of faulty sensor measurements [9]. A hybrid structure constructed of adaptive linear (ADALINE) NN for linear dominant operating conditions as well as MRAN NN for non-linear dominant operating conditions were considered to decrease complexity and computational load. However, the proposed scheme is still slow in detecting faults and requires a high number of neurons to approximate sensor output. In [8], the SFDIA approach based on artificial hydrocarbon networks (AHN) over WSN was presented. AHN is exploited to predict the temperature and detect the faulty sensor using in-field sensors and comparing it with information from a web service.

A distributed spike fault detection method was presented for linear time-invariant systems based on online learned pair-wise relationships of sensors using auto-regressive with exogenous input time-series model [29]. Another method utilized a seasonal auto-regressive integrated moving average models for forecasting surface temperature variation of concrete sewer pipes [6]. Predicted values were used as a reference measure for fault detection and replacement for faulty data. However,

the presence of faults and anomalies reduces the forecasting performance of this method as it relies on previous measurements of the faulty sensor.

III. SYSTEM ARCHITECTURE FOR SFDIA SCHEME

In the proposed SFDIA scheme, sensors are split into two groups: the unreliable set \mathcal{S}_U with N_U sensors that are prone to failures, and the reliable set \mathcal{S}_R with N_R reliable sensors. Indeed, in some applications some sensors could be more reliable because of sensor quality, hardware redundancy, proper design and working environment, being at middle of life time [30], or some other forms of protection in higher architectural layers. The proposed SFDIA algorithm can also handle the case of \mathcal{S}_R being the empty set ($N_R = 0$). The objective is to detect, identify and accommodate failure of faulty sensors among the unreliable set whenever they happen. In the following, $x_s[n]$ denotes the measurement from the generic s th sensor at time instant n . Without loss of generality, we number sensors 1 to N_U those belonging to the unreliable set, and $N_U + 1$ to $N_U + N_R$ those belonging to the reliable set, then we denote $\mathbf{x}_{U,s}[n]$ and $\mathbf{x}_R[n]$ the vectors collecting the measurements from the unreliable sensors with s th sensor excluded and from the reliable sensors, respectively, at time instant n .

The block diagram of the proposed SFDIA scheme is shown in Fig. 1, where similar blocks and similar data are reported in the same color. The input to the system is the set of measurements from all sensors. The system is based on *three* stages: (i) the first stage is made of N_U virtual sensors (representing estimation of unreliable sensors); (ii) the second stage is made of N_U analogous residual-computation units; and (iii) the third stage is made of a (multi-task) classifier. The classifier at the third stage is performing detection and isolation, while accommodation is done by exploiting the estimators' output.

More specifically, at the *first stage*, the virtual sensor $s \in \mathcal{S}_U$ receives as input the measurements from all sensors excluding sensor s (i.e. $(\mathcal{S}_U \cup \mathcal{S}_R - \{s\})$) for time instant n and L_v previous time instants (i.e. a sliding window), and produces as output an estimate of the measurement of sensor $s \in \mathcal{S}_U$, whose n th sample is denoted $y_s[n]$.

Then, at the *second stage*, the residual-computation unit $s \in \mathcal{S}_U$ receives as input the measurement $x_s[n]$ of sensor $s \in \mathcal{S}_U$ and the corresponding estimate $y_s[n]$ from the virtual sensor $s \in \mathcal{S}_U$ and produces as output a measure of dissimilarity of the pair, whose n th sample is denoted $e_s[n]$. Residual measurements are reflecting inconsistencies between the normal and faulty sensor operating status of unreliable sensors.

At the *third stage*, the classifier receives as input the dissimilarity measures from all the sensor pairs in the unreliable set \mathcal{S}_U for time instant n and L_c previous time instants, and produces as output a decision vector about if and which sensor has undergone a failure. According to Fig. 1, the n th (soft-) decision vector is denoted $\mathbf{d}[n] = (d_1[n], d_2[n], \dots, d_{N_U}[n])^T$ where $d_i[n] \in [0, 1]$, $i = 1, \dots, N_U$ denotes the probability of the i th sensor (corresponds to a specified unreliable sensor) being faulty. Ideally, a vector $\mathbf{d}[n]$ with all elements set to 0 denotes the event that no sensor has been declared in failure, while the set of unreliable sensors \mathcal{S}_U is mapped bijectively into the first N_U positive integers with an arbitrary labeling function. The final decision is made based on whether the maximum element of vector $\mathbf{d}[n]$ exceeds a given threshold γ . Nonetheless, the proposed SFDIA architecture (cf. Fig. 1), can detect, isolate and accommodate more than one sensor simultaneously. In this case, SFDIA scheme would present better performance for large scale systems. However this issue falls beyond the scope of this article and will be explored in future works.

It is implicitly assumed that in the case that sensor $s \in \mathcal{S}_U$ is declared in failure, its measurement $x_s[n]$ is replaced with the estimate $y_s[n]$ from the corresponding virtual sensor. It is apparent how the considered architecture implements all the tasks of a SFDIA system: i.e. decision vector $\mathbf{d}[n]$ with an over threshold element represents the *detection task*; after a fault is detected, the specific sensor index i corresponding to the maximum element $d_i[n]$ of the decision vector performs the *isolation task* and replacing $x_s[n]$ with $y_s[n]$ employs the *accommodation task*, with the sensor s identified through the inverse labeling function. In what follows, we detail each of three aforementioned stages.

1) Virtual Sensor: An MLP NN, with $(L_v + 1)(N_U + N_R - 1)$ inputs, 1 output, and H_v hidden layers, each with N_v hidden nodes, has been considered for the implementation of the generic virtual sensor, i.e.

$$y_s[n] = f_s^{(H_v, N_v)}(\mathbf{x}_{U, s}[n], \dots, \mathbf{x}_{U, s}[n - L_v], \mathbf{x}_R[n], \dots, \mathbf{x}_R[n - L_v]), \quad (1)$$

where f_s represents the MLP-based function model of the s th sensor. Each MLP has been trained using the Nesterov-accelerated adaptive moment estimation (Nadam) optimization algorithm using real-world data-sets [31], [32]. The Nadam algorithm takes advantage of properties of adaptive moment estimation (Adam) algorithm and incorporates Nesterov Accelerated Gradients to Adam. Hyperbolic tangent (tanh) and identity activation functions are employed in hidden layers and the output layer, respectively. Mean square error (MSE) loss function is used for loss calculation in training phase.

The MLP is a simple architecture with proved performance of estimating nonlinear behavior [26], [28]. Numerical results show the excellent performance of MLP architecture. However, in the case of further requirement of extrapolating long-term impact of the temporal dimension for time series data-sets, more complicated architectures (e.g. convolutional neural networks, recurrent neural networks (RNNs) and long short term memory networks (LSTMs) [33], [34]) are expected to present more appropriate results for the implementation of each virtual sensor. Data description, data pre-processing (in order to make it suitable for model training) and data contamination procedure (via synthetically-generated faults) are described in the next section.

2) Residual Computation: For dissimilarity measure, we simply considered the error between the estimated value and the actual value, i.e.

$$e_s[n] = y_s[n] - x_s[n]. \quad (2)$$

In fault-free condition, it is expected that the residual measurements $e_s[n]$ be equal to zero, but in practice, it always contains non-zero value due to noise and imperfect estimation of sensor output. Hence, the classifier is introduced to discriminate faulty measurements from non-faulty measurements via pattern analysis of residual signals.

3) Classifier: An MLP NN, with N_U inputs, N_U discrete output, and H_c hidden layer with N_c hidden nodes, has been considered for the implementation of the classifier, i.e.

$$\mathbf{d}[n] = g^{(H_c, N_c)}(\mathbf{e}_U[n], \dots, \mathbf{e}_U[n - L_c]). \quad (3)$$

where $\mathbf{e}_U[n]$ is a vector of the dissimilarity measurements of the unreliable set at time instant n . Since there is a certain level of correlation between temporal samples of residual signals, L_c previous time instants are also fed to the classifier to exploit the temporal correlation among measurements.

The binary cross-entropy loss function along with the same optimization algorithm (Nadam) and activation function (tanh) for hidden layers as in the virtual sensors are employed in the classifier. Moreover, N_U sigmoid activation function is used at the output layer of the classifier. The fault-signal generation is described in the next section.

TABLE I
COMPUTATIONAL COMPLEXITY OF THE MLPs CONSTITUTING
THE PROPOSED SFDIA ARCHITECTURE

Layers	MLP	Complexity
first hidden layer	virtual sensor	$\mathcal{O}(L_v N_U N_v + L_v N_R N_v)$
	classifier	$\mathcal{O}(L_c N_U N_c)$
other hidden layers	virtual sensor	$\mathcal{O}(N_v^2)$
	classifier	$\mathcal{O}(N_c^2)$
output layer	virtual sensor	$\mathcal{O}(N_v)$
	classifier	$\mathcal{O}(N_U N_c)$
in total	virtual sensor	$\mathcal{O}(L_v N_U N_v + L_v N_R N_v + H_v N_v^2)$
	classifier	$\mathcal{O}(L_c N_U N_c + H_c N_c^2)$

Computational Complexity: The computational complexity of the proposed SFDIA structure is calculated hereunder in terms of the big- \mathcal{O} notation for one input sample. The computational complexity for each layer of the virtual sensor and classifier is specified in Tab. I.

It is worth noticing that in Tab. I, the impact of tanh and sigmoid operations for virtual sensors and the classifier has been neglected. Finally, with respect to the computational complexity of both MLPs and assuming equal number of hidden layers ($H_v = H_c = H$), nodes per hidden layer ($N_v = N_c = N$) and time delays ($L_v = L_c = L$), the computational complexity involved with the proposed architecture is approximately $\mathcal{O}(LN_U^2 N + LN_R N_U N + HN_U N^2)$. Thus, the proposed architecture has *polynomial complexity*, and the complexity grows *quadratically* as a function of the number of nodes per layer (N) and number of unreliable sensors (N_U).

IV. DATA DESCRIPTION, PRE-PROCESSING, AND CONTAMINATION

A. Data Description

Three real-world data-sets are applied to the proposed SFDIA system to evaluate the qualification of the system in different scenarios.

1) *Air Quality (AQ) Data-Set:* The first data-set contains hourly-averaged measurements of an array of 5 metal oxide chemical sensors embedded in a gas multi-sensor device deployed on the field in an Italian city along with gas concentrations references from a certified analyzer [11]. The device was located in a polluted area, at road level of the city. AQ data-set was recorded during Mar. 2004-Feb. 2005.

Measurements contain carbon monoxide (CO), non-metanic hydrocarbons (NMH), nitrogen oxides (NO_x), nitrogen dioxide (NO₂) and ozone (O₃) gas concentrations, as well as measurements of temperature and humidity. For our analysis, the ground truth hourly-averaged concentrations provided by a co-located reference certified analyzer along with absolute humidity are ignored. Accordingly, in our numerical analysis, the five gas sensors are considered as the unreliable set ($N_U = 5$), whereas temperature and relative humidity are considered as the reliable set ($N_R = 2$).

2) *Wireless Sensor Network (WSN) Data-Set:* The second data-set used in our evaluation has been collected at the University of North Carolina at Greensboro [12]. A labeled

data-set collected from a single-hop and a multi-hop WSN using TelosB motes. The data-set consists of 4 sensors located indoor and outdoor measuring humidity and temperature. Measurements were collected during 6 hours at 5 seconds interval. Anomalies indicated with label "1" in the original data-set were introduced to two sensors by using a water kettle which increased the temperature and humidity.

In what follows, only the multi-hop data-set with 4 temperature (T1 to T4) measurements is used as unreliable set ($N_U = 4$), and data with the indicated label "1" were ignored from this data-set. No reliable set is considered for this data-set ($N_R = 0$).

3) *Permanent Magnet Synchronous Motor (PMSM) Data-Set:* The third data-set comprises several sensor data measurements from a permanent magnet synchronous motor collected by the LEA department at Paderborn University [13], [35]. Data-set measurements include ambient temperature, coolant temperature (CT), voltage q and d components, current q and d components, motor speed (MS), torque (TRQ), rotor temperature, stator yoke temperature, stator tooth temperature, and stator winding temperature. Original measurements contain 52 sessions, with each session being 1 ~ 6h long and sampled at intervals of 0.5 seconds.

We have considered a sample interval of 15 seconds (by down-sampling) and ignored the ambient and rotor measurements. Summation of q and d components of voltage and current are treated as final voltage (V) and current (C) measurements. The reliable set consists of 3 stator temperatures ($N_R = 3$), and other remaining measurements form the unreliable set ($N_U = 5$).

B. Pre-Processing

As commonly done in machine-learning applications, in order to avoid polarization in the training due to different ranges of different variables, measurements of each sensor have been normalized such to span the range [0, 1] via min-max scaling

$$x'_s[n] = \frac{x_s[n] - x_{\min}}{x_{\max} - x_{\min}}, \quad (4)$$

where $x'_s[n]$ represents the normalized measurements of the s th sensor, whereas x_{\max} and x_{\min} are the minimum and maximum of the training set for given sensor measurements. It is worth mentioning, in the normalization process, x_{\max} and x_{\min} are derived based on the training set of each data-set to present the real-world condition. Besides normalization, entire rows with missed data in data-sets are omitted. No other pre-processing has been considered, such as feature extraction, to help the learning procedure of the virtual sensors. Although, for noisy data-sets, smoothing techniques (e.g. moving average, Savitzky-Golay filter or quadratic regression) or low-pass filtering can be performed allowing the important patterns of data to stand out.

In proposed architecture, instead of using all sensor except the one under estimation as input of each virtual sensor, only the most correlated sensors could be considered as input. This would help containing complexity, specially for large-scale systems, while ensuring acceptable performance. Correlation matrix of all sensors could be obtained from the training set.

However, this issue is beyond the scope of this article and will not be here investigated. Architectures with different number of hidden layers has been compared in order to verify if a deep architecture can overcome the need for feature extraction for the specific problem.

C. Data Contamination

In order to build data-sets including sensor failures for training the SFDIA classifier and testing its performance, synthetic fault signals have been generated and injected to all three data-sets. Failure of a sensor could manifest in several ways [36]–[38]. The most common fault models are bias, drift, freezing and random fault. In this article, without loss of generality, we considered *bias* and *drift* faults to represent hard and soft failures, respectively. The mathematical model for each of them is described in what follows.

1) **Bias Fault:** In this type of failure (also known as step fault), a constant bias b for M consecutive samples was added to the sensor measurements, namely

$$x_s[n] = \begin{cases} a_s[n] + v_s[n] + b, & 0 \leq n - m < M \\ a_s[n] + v_s[n], & \text{else} \end{cases} \quad (5)$$

where $a_s[n]$ is the ideal (without fault) measurement of the s th sensor and m is the starting time instant of the fault, while $v_s[n]$ denotes the measurement noise. Sensor measurements in all three data-sets are (naturally) including measurement noise (i.e. they provide $a_s[n] + v_s[n]$).

2) **Drift Fault:** This additive fault happens in $M + N$ consecutive samples when sensor output drifts up to the bias level b with M time instants

$$x_s[n] = \begin{cases} a_s[n] + v_s[n] + \frac{b(n-m+1)}{M}, & 0 \leq n - m < M \\ a_s[n] + v_s[n] + b, & M \leq n - m < M + N \\ a_s[n] + v_s[n], & \text{else} \end{cases} \quad (6)$$

where N is the number of consecutive samples that the drift fault remains at the saturated bias level b . Also, we considered $M > N$ to stress the effect of the drift.

V. NUMERICAL RESULTS

In this section, performance of the proposed SFDIA architecture is examined and compared with recent research works by using the aforementioned real-world data-sets. Each data-set is divided into three parts. On each data-set, we used 70% of data for training MLPs (training set), 15% for validating (validation set) and last 15% block of data for testing purposes (test set). *Early stopping* method is used to avoid over-fitting during the training phase [39]. In this method, error on the validation set is monitored and if after 20 consecutive epochs validation set error did not improve, the training process is stopped.

We denote *variation domain* the size of the range spanned by a sensor with reference to the training set. Maximum level b of generated faults is assumed uniformly distributed between 0.2 and 0.4 (i.e. accounting for 20 to 40 percent of the corresponding variation domain) to represent weak

TABLE II
VARIATION DOMAIN FOR EACH SENSOR

Data-set	Sensors Variation Domain				
	CO	NMH	NO _x	NO ₂	O ₃
AQ	1392	1830	2360	2118	2270
WSN	T1 (°C)	T2 (°C)	T3 (°C)	T4 (°C)	-
	3.57	3.72	2.23	1.99	-
PMSM	CT	V	C	MS	TRQ
	3.50	6.00	7.24	3.25	6.33

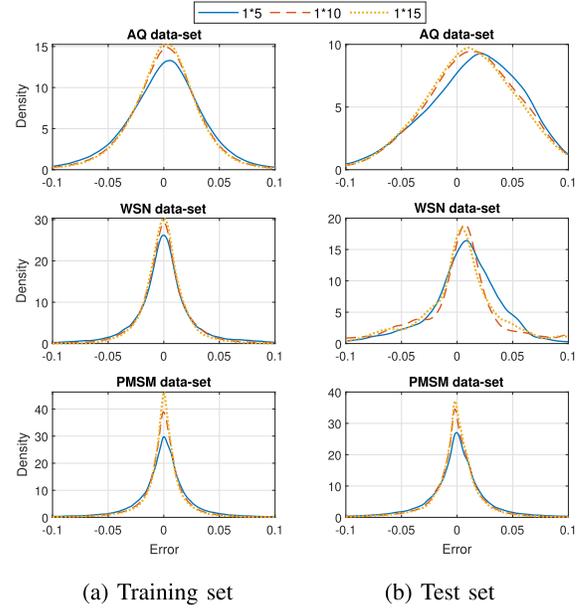


Fig. 2. Averaged performance of the virtual sensors for different number of nodes N_v in terms of PDF of the error signals on each data-set. Different configurations are denoted with $H_v \times N_v$.

fault signals. Positive and negative faults are generated randomly. Uniform distribution of maximum level b assures that the classifier will not learn on a specific level. Table II reports the variation domain for each sensor. The variation domain, which is always less or equal to the true range of each sensor (e.g. on WSN data-set in Tab. II, maximum variation domain is 3.72°C while usually temperature sensors range are around 150°C or even higher), is used as criterion since the true ranges were unknown. In addition, to better understand the effect of fault strength on detection accuracy, strong fault signals with maximum level b uniformly distributed between 0.6 and 0.9 are considered for comparison with weak fault signals.

A. Virtual Sensors Performance

Virtual sensors with $N_v \in \{5, 10, 15\}$ nodes per hidden layer and $H_v \in \{1, 2, 3\}$ hidden layers have been trained and compared. In detail, virtual sensors' overall performance on both training and test sets are shown in Figs. 2 and 3 in terms of PDF of all sensors error signals ($e_{ij}[n]$) in each data-set.

The improvement of the performance with increasing the number of nodes (N_v) and hidden layers (H_v) is apparent, but variable for different data-sets. Fig. 2 seems to suggest

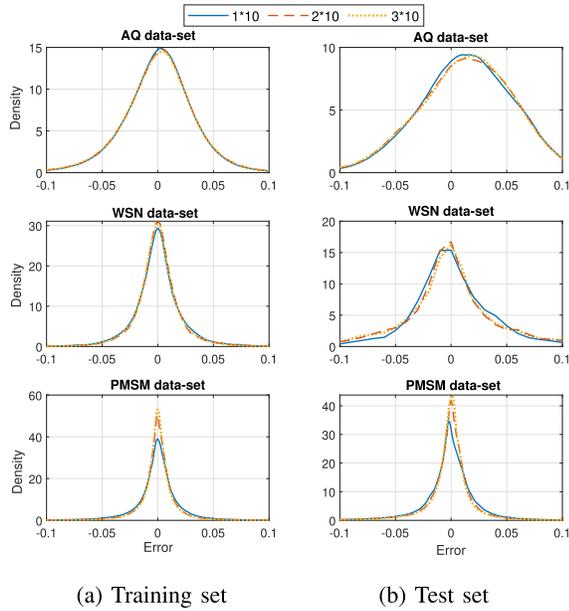


Fig. 3. Averaged performance of the virtual sensors for different number of hidden layers H_v in terms of PDF of the error signals on each data-set. Different configurations are denoted with $H_v \times N_v$.

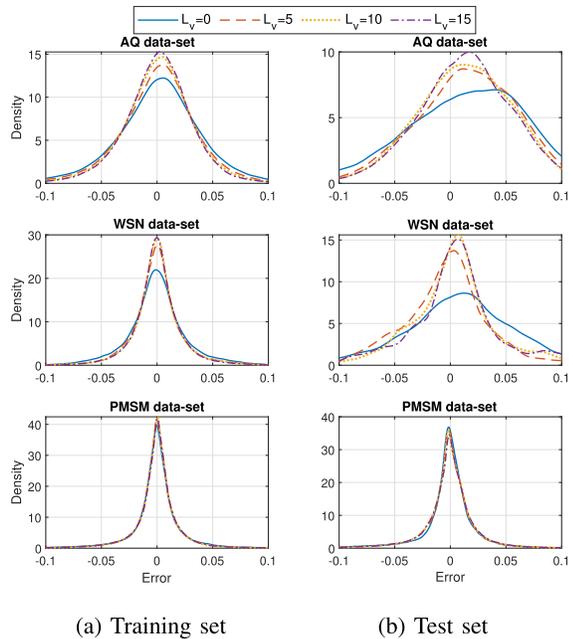


Fig. 4. Averaged performance of the virtual sensors in configuration 1×10 for different number of previous time instants L_v in terms of PDF of the error signals on each data-set.

the improvements with respect to the number of nodes per layer saturate approximately with N_v , while, as it can be seen in Fig. 3, adding more layers has only a relevant effect on the largest data-set (PMSM data-set). It must be said that deeper network structures require larger data-sets to update their weights and biases, thus the saturation effect might be due to the limited amount of available data. Fig. 4 illustrates the impact of input window size L_v on the virtual sensors performance. By employing delayed samples, the virtual sensors can exploit the temporal correlation between data samples to enhance estimation performance. However, the PMSM data-set has a very limited temporal correlation.

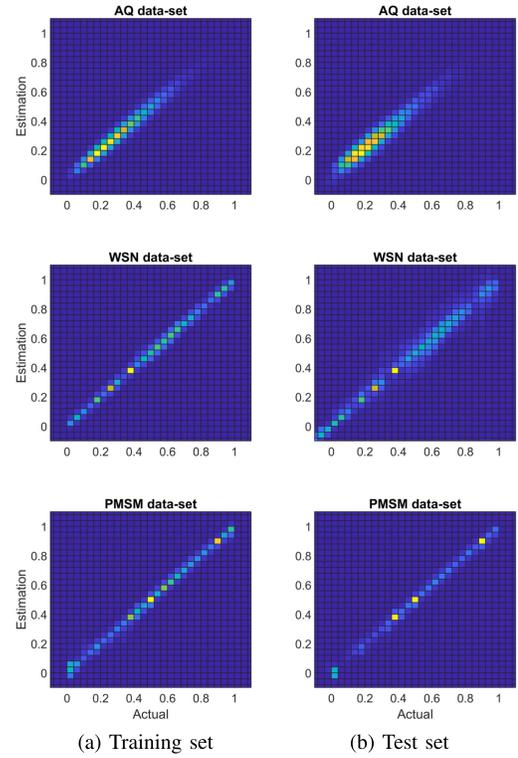


Fig. 5. Averaged performance of the virtual sensors in configuration 1×10 and $L_v = 10$ in terms of 2D PDFs of the estimated and actual values.

TABLE III
NUMBER OF TRAINABLE PARAMETERS (WEIGHTS AND BIASES)

		Virtual Sensor ^a				Classifier ^a			
N_c ,	H_c ,	L_v				L_c			
N_v	H_v	0	5	10	15	0	5	10	15
5	1	31	131	231	331	60	185	310	435
	2	61	161	261	361	90	215	340	465
10	1	61	261	461	661	115	365	615	865
	2	171	371	571	771	225	475	725	975
15	1	91	391	691	991	170	545	920	1295
	2	331	631	931	1231	410	785	1160	1535

^a No reliable sensor ($N_R = 0$) and $N_U = 5$ unreliable sensors considered for calculations.

Performance of the configuration with $H_v = 1$ hidden layer, $N_v = 10$ nodes per hidden layer and $L_v = 10$ is considered acceptable, thus in the following, we will refer to this specific configuration. The 2D-PDF plots of the estimated and actual values for virtual sensors in configuration 1×10 are shown in Fig. 5, both for the training and the test sets. It is worth noticing that the test set of the WSN data-set exceeds the defined normalization lower-bound which is the result of normalization on the training set.

B. Classifier Fault Detection and Classification Performance

Synthetically-generated faults have been added to unreliable set of sensors to emulate faulty sensors. Different configurations for the classifier are compared in the following. Table III lists the number of parameters (weights and biases) to be

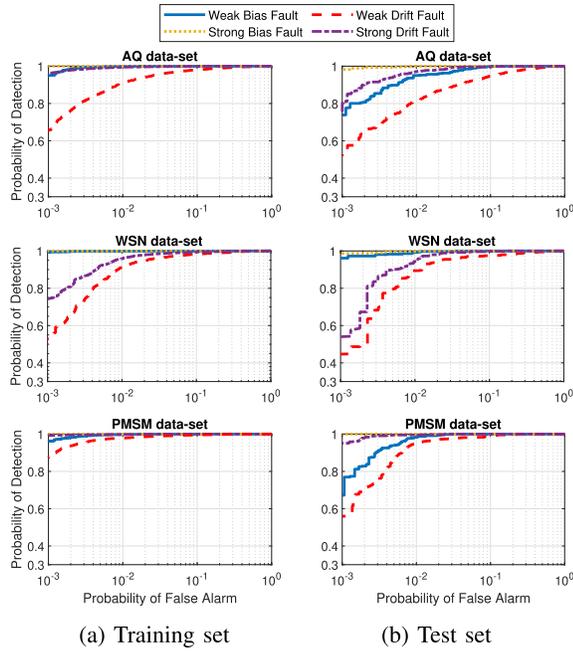


Fig. 6. ROC curves of proposed SFDIA structure for all data-sets under bias and drift faults.

trained during training phase in the classifier and each virtual sensor for different configurations.

A classifier with $H_c = 2$ hidden layers, $N_c = 15$ nodes per hidden layer and a memory of $L_c = 10$ has been trained. In this configuration, according to Tab. III, a total number of 725 trainable parameters of the classifier are required to be updated through training phase over AQ and PMSM data-set.¹

The probabilities of detection and false-alarm are two important metrics for evaluating the performance of a detector. Accordingly, in Fig. 6, fault detection performance is investigated in terms of both metrics by using the well-known receiver operating characteristic (ROC) curves (i.e. by varying the threshold γ). Results highlight that, although the classifier is facing weak fault signals, it is still capable to detect them with a very high probability for negligible false-alarm probability. Detection probability of bias faults is noticeably higher than drift faults over different false alarm rates. This is originally due to the ramp up phase of drift faults which takes classifier more samples to detect faults. As illustrated in Fig. 6, WSN data-set has somewhat lower performance in comparison with the other two data-sets (in case of drift faults). It is mainly because of very weak fault levels on this data-set according to its sensors' variation domains (see Tab. II). Conversely, detection performance of proposed architecture under strong faults are significantly higher than the detection performance under weak faults as shown in Fig. 6, which highlights the importance of detection and isolation of weak faults.

The detection rate of the classifier with 5 and 15 nodes per hidden layer is assessed in Fig. 7 in case of drift faults. It is apparent from both train and test sets that 5 nodes per hidden layer are not enough for distilling relevant features

¹The number of trainable parameters of the classifier is different for WSN data-set due to different Number of unreliable sensors ($N_U = 4$).

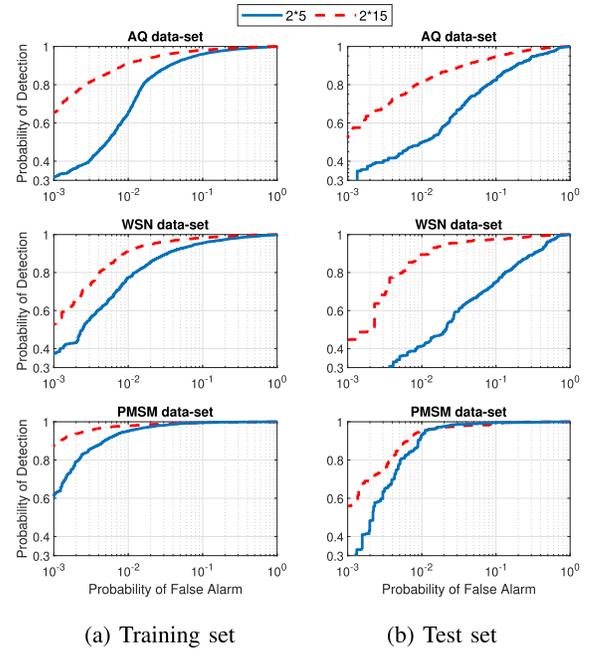


Fig. 7. Detection performance of the classifier for different number of nodes per hidden layer N_c in terms of ROC on each data-set.

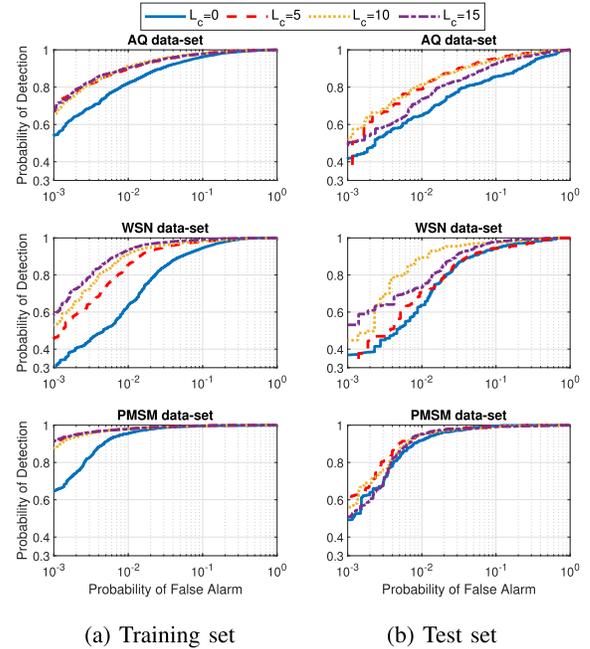


Fig. 8. Detection performance of the classifier in configuration 2×15 for different number of previous time instants L_c in terms of ROC on each data-set.

from the data sequences. In general, the accuracy on test set is lower than the accuracy on train set since the classifier is optimized for the latter. Figs. 8 and 9 demonstrate the effect of using time-delayed samples on the classifier in the case of drift fault. There are certain improvements in detection performance and averaged classification (isolation) performance² when temporal correlation exists in sensor measurements.

²Averaged classification performance is the average of correct classification probability on all sensors in data-set. Non-fault occurrence is considered as a separate class.

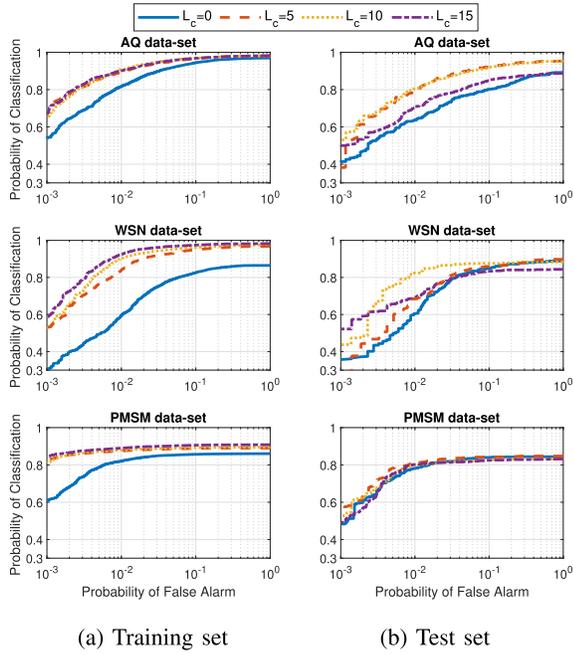


Fig. 9. Averaged classification (isolation) performance of the classifier in configuration 2×15 for different number of previous time instants L_c in terms of ROC on each data-set.

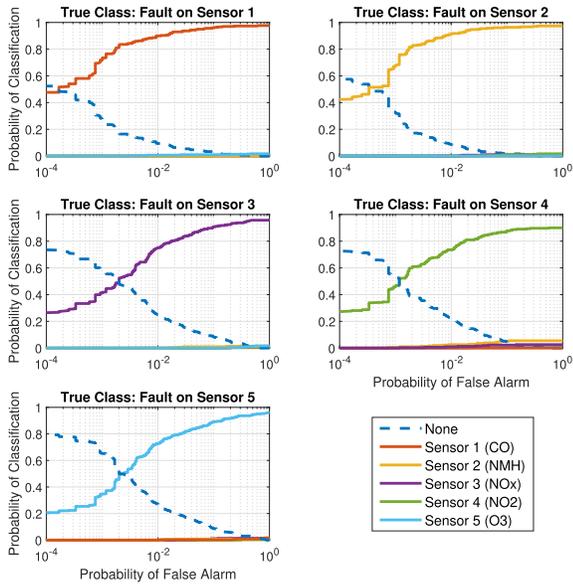


Fig. 10. Classification ROC curves for AQ data-sets under drift faults.

However, as it can be seen on both Fig. 8.(b) and 9.(b), the performance slightly reduces with increasing number of time delays ($L_c = 15$) due to the negligible temporal correlation between older samples and current sample in the measurements. Besides, in this scenario, increasing the window size should potentially lead to a performance improvement, however a larger number of nodes in the hidden layers might be required to handle properly the increased number of input nodes. Differently, with a fixed network structure, increasing the window size might in practice saturate the learning capability.

Figure 10 shows the performance in terms of “multi-class ROC” for each detected class for AQ data-set under drift faults,

TABLE IV
DETECTION AND CLASSIFICATION ACCURACY
BASED ON YODEN’S INDEX

Data-set	Fault Type	γ	P_d (%)	P_f (%)	P_{di} ^a (%)	Sensors Classification Performance (%)				
						CO	NMH	NO _x	NO ₂	O ₃
AQ	Bias	0.109	95.2	1.1	91.3	94.8	99.8	93.0	68.5	99.9
	Drift	0.1345	93.1	7.0	90.6	95.3	96.0	89.1	86.5	86.2
WSN	Bias	0.151	97.3	0.2	89.8	94.1	99.9	94.7	70.1	-
	Drift	0.213	95.0	2.1	86.0	92.7	95.4	83.7	72.4	-
PMSM	Bias	0.001	99.9	7.4	89.0	99.0	94.6	91.7	68.1	91.4
	Drift	0.107	97.0	2.0	81.1	90.2	76.4	83.3	65.7	90.0

^a P_{di} = Averaged probability of correct classification on all sensors.

i.e. no failure and sensor-1 to sensor-5 failures. More specifically, each subfigure refers to a specific true sensor fault (including the no-fault scenario represented with a dashed line) obtained through varying the selected threshold.³ The probability of correct classification for all 5 sensors reaches $\approx 95\%$. Also, it is apparent how good detection and identification results are obtained at the expenses of reduced misclassification rates. Apart from misclassification with the none case, the case with NO₂ sensor failure being misclassified as a NMH sensor failure is the most difficult misclassification case to avoid in AQ data-set. In all data-sets, the results with bias faults are notably improved in comparison to those with drift fault.⁴

There exists several criteria for setting the optimal threshold value to maximize the probability of detection. In this study we selected Youden index J , i.e. maximization the vertical distance between the 45-degree line (equality line) and the point on the ROC curve [40]

$$J = \max_{\gamma} (P_d - P_f). \quad (7)$$

where P_d is the probability of detection and P_f is the probability of false alarm.

Sensors classification performance on test sets of all data-sets with Youden index criteria are summarized in Tab. IV. Thresholds in Tab. IV are set by applying Youden index criteria to ROC curves from training sets. Next, all recorded probabilities are derived from test sets for obtained thresholds. On the whole, the achieved accuracy with bias faults is comparatively higher than drift faults. The best detection accuracy of 99.9% as well as very good detection

³Plots are not depicted with respect to the selected threshold, but with respect to the corresponding probability of false alarm. It is worth noticing that well-known confusion matrices may be obtained from these plots by selecting a desired point of operation (corresponding to a specific value of the numerical threshold γ providing the classifier output).

⁴Classification performance on different sensors of other two data-sets as well as bias faults are not shown for brevity.

TABLE V
DETECTION ACCURACY OF THE PROPOSED ARCHITECTURE
COMPARED TO THE SVM CLASSIFIER AND THE FCC
TECHNIQUE ON THE TEST SET

Data-set	Architecture	Metrics	Bias (%)		Drift (%)	
			Weak	Strong	Weak	Strong
AQ	SVM FCC Proposed	P_f	2.82	0.01	2.32	0.17
		P_d	79.2	98.0	70.4	88.8
		P_d	98.5	-	85.2	87.9
		P_d	97.5	98.9	84.1	95.9
		P_f	22.7	0.15	21.7	1.0
WSN	SVM FCC Proposed	P_d	95.9	98.5	88.2	90.3
		P_d	100	-	94.4	96.3
		P_d	100	98.9	98.2	94.2
		P_f	0.05	0.06	0.11	0.15
PMSM	SVM FCC Proposed	P_d	34.9	92.3	31.8	77.7
		P_d	15.9	99.7	25.0	50.8
		P_d	58.1	99.8	56.0	96.2
		P_d	58.1	99.8	56.0	96.2

accuracy of 97.3% with the lowest false alarm rate of 0.2% respectively obtained on PMSM and WSN data-sets under bias fault condition which shows excellent detection performance of the proposed SFDIA scheme. Moreover, good classification performance on most sensors is evident with highest average correct classification of 91.3%, with MS sensor on PMSM data-set as the hardest classification case.

C. Performance Comparison

Table V compares the proposed architecture with two state-of-the-art techniques previously outlined in Sec. II: (i) the SVM classifier [14] and (ii) the FCC NN [15] with 6 nodes. The SVM classifier has no control over the probability of false alarm since it does not have any threshold mechanism. Hence, to provide a fair comparison, we tuned the threshold on the proposed architecture and on the FCC technique to achieve the same probability of false alarm as the SVM classifier, and compared the probability of detection for all techniques in Tab. V. Apparently, the detection performance of the proposed architecture outperforms the SVM technique for all fault types. The performance gap between these two techniques in terms of detection accuracy becomes more evident under weak faults. More specifically, under weak drift fault for the PMSM data-set, the performance improvement in fault detection of the proposed architecture over the SVM technique is approximately 24.2%. The main reason lies in the fact that the SVM classifier takes raw-sensor data as input while the proposed architecture exploits the estimations of each sensor and feeds the residual data as input to the classifier which contains easy-to-interpret information about faults. The FCC technique exhibits similar detection performance as the proposed architecture over AQ and WSN data-sets, while on the PMSM data-set the proposed architecture turns to be better performing. In Tab. V, the detection accuracy of the FCC technique with respect to the corresponding probability of false alarm was not available for the WSN and AQ data-sets under

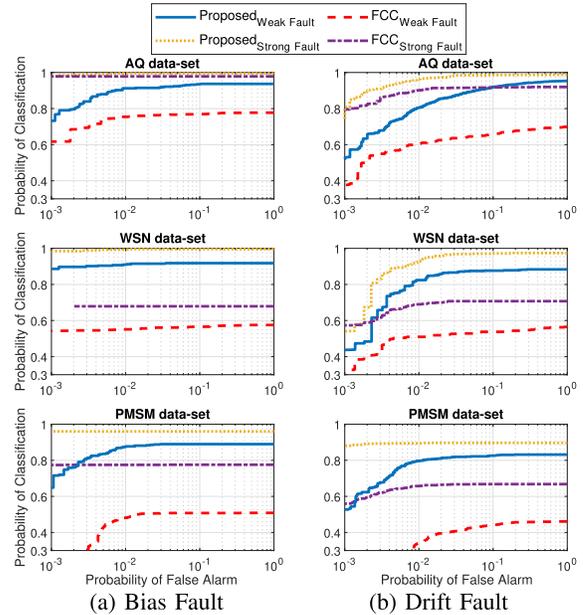


Fig. 11. Averaged classification (isolation) performance comparison in terms of ROC for the test set on each data-set.

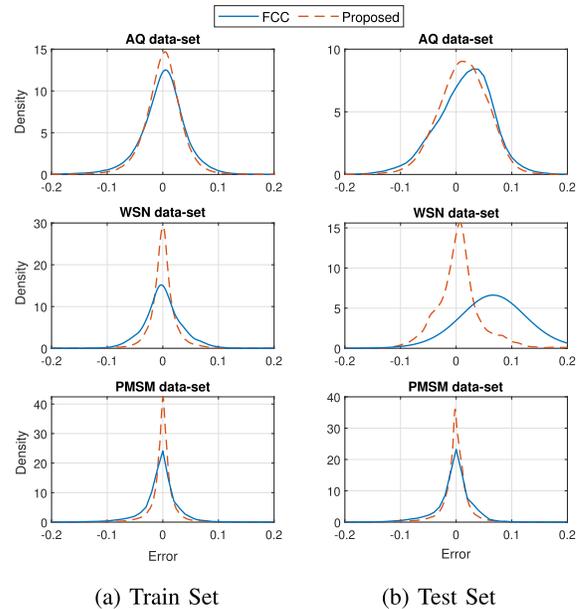


Fig. 12. Accommodation performance comparison in terms of PDF of the error signals on each data-set.

strong bias faults (as can be seen also in Fig. 11(a)). It is worth mentioning that the detection performance on the training set resembles those shown for the test set in Tab. V.

As for the isolation task, the proposed architecture achieves significant gains over the FCC technique as observed in terms of classification performance shown in Fig. 11. More specifically, the proposed architecture takes advantage of MLP classifier while the FCC technique merely uses a sliding window mechanism. The relevance of the proposed architecture as an effective SFDIA scheme is apparent.

Finally, as for the accommodation task, Fig. 12 compares the accuracy of the virtual sensors which reveals better estimation capability of the MLPs from the proposed architecture

against the FCC NNs. The improvement is mainly due to the capability of the proposed technique to exploit temporal correlation. Finally, it is worth noticing that isolation and accommodation performances of the SVM technique cannot be compared due to its incapability to classify and estimate faulty sensors.

VI. CONCLUSION

In this article, we presented a three-stage SFDIA architecture with capability to adapt with different applications. The classifier at the third stage detects and isolates the faulty sensor from patterns within the input residual signals. The bank of estimators at the first stage allows to accommodate unreliable sensors by replacing the measurements from the identified faulty sensors. Estimators are also used at the second stage to derive the residual signals for the classifier. An extensive evaluation on three real-world data-sets from different applications indicated that the proposed SFDIA architecture attains high probability of detection and correct classification with low probability of false alarm in presence of weak bias and drift faults.

The same architecture allows large flexibility with the components in each layer (e.g. replacing the considered MLPs with RNNs), thus might achieve further performance improvements under specific circumstances. In addition, although not investigated in this work, the proposed architecture is potentially capable of handling multiple simultaneous faults, a feature to be considered in future works.

REFERENCES

- [1] H. Darvishi, D. Ciunzo, E. R. Eide, and P. Salvo Rossi, "A data-driven architecture for sensor validation based on neural networks," in *Proc. IEEE SENSORS*, Oct. 2020.
- [2] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The future of industrial communication: Automation networks in the era of the Internet of Things and industry 4.0," *IEEE Ind. Electron. Mag.*, vol. 11, no. 1, pp. 17–27, Mar. 2017.
- [3] S. Yin, S. X. Ding, and D. Zhou, "Diagnosis and prognosis for complicated industrial systems—Part I," *IEEE Trans. Ind. Electron.*, vol. 63, no. 4, pp. 2501–2505, Jan. 2016.
- [4] A. Mahapatro and P. M. Khilar, "Fault diagnosis in wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 4, pp. 2000–2026, Mar. 2013.
- [5] H.-T. Pham, J.-M. Bourgeot, and M. E. H. Benbouzid, "Comparative investigations of sensor fault-tolerant control strategies performance for marine current turbine applications," *IEEE J. Ocean. Eng.*, vol. 43, no. 4, pp. 1024–1036, Oct. 2018.
- [6] K. Thiagarajan, S. Kodagoda, L. Van Nguyen, and R. Ranasinghe, "Sensor failure detection and faulty data accommodation approach for instrumented wastewater infrastructures," *IEEE Access*, vol. 6, pp. 56562–56574, 2018.
- [7] S. Mandal, B. Santhi, S. Sridhar, K. Vinolia, and P. Swaminathan, "Nuclear power plant thermocouple sensor-fault detection and classification using deep learning and generalized likelihood ratio test," *IEEE Trans. Nucl. Sci.*, vol. 64, no. 6, pp. 1526–1534, Jun. 2017.
- [8] S. Gutiérrez and H. Ponce, "An intelligent failure detection on a wireless sensor network for indoor climate conditions," *Sensors*, vol. 19, no. 4, p. 854, Feb. 2019.
- [9] G. Campa, M. Thiagarajan, M. Krishnamurthy, M. R. Napolitano, and M. Gautam, "A neural network based sensor validation scheme for heavy-duty diesel engines," *J. Dyn. Syst., Meas., Control*, vol. 130, no. 2, Mar. 2008.
- [10] L. Liu, G. Han, Y. He, and J. Jiang, "Fault-tolerant event region detection on trajectory pattern extraction for industrial wireless sensor networks," *IEEE Trans. Ind. Informat.*, vol. 16, no. 3, pp. 2072–2080, Mar. 2020.
- [11] S. De Vito, E. Massera, M. Piga, L. Martinotto, and G. Di Francia, "On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario," *Sens. Actuators B, Chem.*, vol. 129, no. 2, pp. 750–757, Feb. 2008.
- [12] S. Suthaharan, M. Alzaharani, S. Rajasegarar, C. Leckie, and M. Palaniswami, "Labelled data collection for anomaly detection in wireless sensor networks," in *Proc. 6th Int. Conf. Intell. Sensors, Sensor Netw. Inf. Process. (ISSNIP)*, Dec. 2010, pp. 269–274.
- [13] W. Kirchgässner, O. Wallscheid, and J. Böcker, "Empirical evaluation of exponentially weighted moving averages for simple linear thermal modeling of permanent magnet synchronous machines," in *Proc. IEEE 28th Int. Symp. Ind. Electron. (ISIE)*, Jun. 2019, pp. 318–323.
- [14] S. Zidi, T. Moulahi, and B. Alaya, "Fault detection in wireless sensor networks through SVM classifier," *IEEE Sensors J.*, vol. 18, no. 1, pp. 340–347, Jan. 2018.
- [15] S. Hussain, M. Mokhtar, and J. M. Howe, "Sensor failure detection, identification, and accommodation using fully connected cascade neural network," *IEEE Trans. Ind. Electron.*, vol. 62, no. 3, pp. 1683–1692, Mar. 2015.
- [16] P. Goupil, "AIRBUS state of the art and practices on FDI and FTC in flight control system," *Control Eng. Pract.*, vol. 19, no. 6, pp. 524–539, Jun. 2011.
- [17] S. Yin, B. Xiao, S. X. Ding, and D. Zhou, "A review on recent development of spacecraft attitude fault tolerant control system," *IEEE Trans. Ind. Electron.*, vol. 63, no. 5, pp. 3311–3320, May 2016.
- [18] S. Gururajan, M. L. Fravolini, M. Rhudy, A. Moschitta, and M. Napolitano, "Evaluation of sensor failure detection, identification and accommodation (SFDIA) performance following common-mode failures of Pitot tubes," SAE Tech. Paper 2014-01-2164, Sep. 2014.
- [19] G. Heredia and A. Ollero, "Detection of sensor faults in small helicopter UAVs using observer/Kalman filter identification," *Math. Problems Eng.*, vol. 2011, pp. 1–20, Jan. 2011.
- [20] B. Pourbabaee, N. Meskin, and K. Khorasani, "Sensor fault detection, isolation, and identification using multiple-model-based hybrid Kalman filter for gas turbine engines," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 4, pp. 1184–1200, Jul. 2016.
- [21] C. Alippi, S. Ntalampiras, and M. Roveri, "Model-free fault detection and isolation in large-scale cyber-physical systems," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 1, no. 1, pp. 61–71, Feb. 2017.
- [22] F. Balzano, M. L. Fravolini, M. R. Napolitano, S. d'Urso, M. Crispoltoni, and G. del Core, "Air data sensor fault detection with an augmented floating limiter," *Int. J. Aerosp. Eng.*, vol. 2018, pp. 1–16, Nov. 2018.
- [23] M. Carminati, G. Ferrari, R. Grassetti, and M. Sampietro, "Real-time data fusion and MEMS sensors fault detection in an aircraft emergency attitude unit based on Kalman filtering," *IEEE Sensors J.*, vol. 12, no. 10, pp. 2984–2992, Oct. 2012.
- [24] E. Naderi and K. Khorasani, "Data-driven fault detection, isolation and estimation of aircraft gas turbine engine actuator and sensors," *Mech. Syst. Signal Process.*, vol. 100, pp. 415–438, Feb. 2018.
- [25] J. Gao, J. Wang, P. Zhong, and H. Wang, "On threshold-free error detection for industrial wireless sensor networks," *IEEE Trans. Ind. Informat.*, vol. 14, no. 5, pp. 2199–2209, May 2018.
- [26] M. R. Napolitano, Y. An, and B. A. Seanor, "A fault tolerant flight control system for sensor and actuator failures using neural networks," *Aircr. Design*, vol. 3, no. 2, pp. 103–128, Jun. 2000.
- [27] M. L. Fravolini, G. Campa, K. Napolitano, and Y. Song, "Minimal resource allocating networks for aircraft SFDIA," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics*, vol. 2, Jul. 2001, pp. 1251–1256.
- [28] S. Gururajan, M. L. Fravolini, H. Chao, M. Rhudy, and M. R. Napolitano, "Performance evaluation of neural network based approaches for airspeed sensor failure accommodation on a small UAV," in *Proc. 21st Medit. Conf. Control Autom.*, Jun. 2013, pp. 603–608.
- [29] C. Lo, J. P. Lynch, and M. Liu, "Distributed reference-free fault detection method for autonomous wireless sensor networks," *IEEE Sensors J.*, vol. 13, no. 5, pp. 2009–2019, May 2013.
- [30] P. Tchakoua, R. Wamkeue, M. Ouhrouche, F. Slaoui-Hasnaoui, T. A. Tameghe, and G. Ekemb, "Wind turbine condition monitoring: State-of-the-art review, new trends, and future challenges," *Energies*, vol. 7, no. 4, pp. 1–36, Apr. 2014.
- [31] T. Dozat, "Incorporating nesterov momentum into adam," in *Proc. Int. Conf. Learn. Representations (ICLR)*, 2016, pp. 1–4.
- [32] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, Banff, AB, Canada, Dec. 2014, pp. 1–15.

- [33] B. Zhao, H. Lu, S. Chen, J. Liu, and D. Wu, "Convolutional neural networks for time series classification," *J. Syst. Eng. Electron.*, vol. 28, no. 1, pp. 162–169, Feb. 2017.
- [34] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [35] W. Kirchgässner, O. Wallscheid, and J. Böcker, "Deep residual convolutional and recurrent neural networks for temperature estimation in permanent magnet synchronous motors," in *Proc. IEEE Int. Electric Mach. Drives Conf. (IEMDC)*, May 2019, pp. 1439–1446.
- [36] H. Alwi, C. Edwards, and C. P. Tan, "Fault tolerant control and fault detection and isolation," in *Fault Detection and Fault-Tolerant Control Using Sliding Modes*. London, U.K.: Springer-Verlag, 2011, pp. 7–27.
- [37] T. Muhammed and R. A. Shaikh, "An analysis of fault detection strategies in wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 78, pp. 267–287, Jan. 2017.
- [38] G. Campa, M. L. Fravolini, M. Napolitano, and B. Seanor, "Neural networks-based sensor validation for the flight control system of a B777 research model," in *Proc. Amer. Control Conf.*, vol. 1, 2002, pp. 412–417.
- [39] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [40] W. J. Youden, "Index for rating diagnostic tests," *Cancer*, vol. 3, no. 1, pp. 32–35, 1950.



Hossein Darvishi (Graduate Student Member, IEEE) received the B.Sc. degree from the Kermanshah University of Technology, Iran, in 2016, and the M.Sc. (Hons.) degree in telecommunications engineering from the K. N. Toosi University of Technology, Iran, in 2018. He is currently pursuing the Ph.D. degree in electronics and telecommunications with the Department of Electronic Systems, Norwegian University of Science and Technology, Trondheim, Norway.

His research interests include statistical signal processing, machine learning, the Internet of Things, and wireless sensor networks.



Domenico Ciunzo (Senior Member, IEEE) received the Ph.D. degree in electronic engineering from the University of Campania "Luigi Vanvitelli," Italy. Since 2011, he has been holding several visiting researcher appointments. He is an Assistant Professor with the University of Naples Federico II. His research interests include data fusion, statistical signal processing, wireless sensor networks, the Internet of Things, and machine learning. He was a recipient of the Best Paper Award at IEEE ICCS 2019,

the 2019 Exceptional Service Award from the IEEE Aerospace and Electronic Systems Society, and the 2020 Early-Career Technical Achievement Award from the IEEE Sensors Council for Sensor Networks/Systems. Since 2014, he has been an (Area) Editor of several IEEE, IET, and ELSEVIER journals.



Eivind Rosón Eide received the M.Eng. degree in information and computer engineering from the University of Cambridge in 2018. He is currently working at Kongsberg Digital AS. He has been working on the use of machine learning for anomaly detection in the IoT systems and combining machine learning with first principle simulators for real time optimization in digital twins.



Pierluigi Salvo Rossi (Senior Member, IEEE) was born in Naples, Italy, in 1977. He received the Dr. Eng. (*summa cum laude*) degree in telecommunications engineering and the Ph.D. degree in computer engineering from the University of Naples Federico II, Italy, in 2002 and 2005, respectively. He held visiting appointments at the Department of Electrical and Computer Engineering, Drexel University, USA; the Department of Electrical and Information Technology, Lund University, Sweden; the Department of Electronics and Telecommunications, Norwegian University of Science and Technology (NTNU), Norway; and the Excellence Center for Wireless Sensor Networks, Uppsala University, Sweden. From 2005 to 2008, he held postdoctoral positions at the Department of Computer Science and Systems, University of Naples Federico II; the Department of Information Engineering, Second University of Naples, Italy; and the Department of Electronics and Telecommunications, NTNU. From 2008 to 2014, he was an Assistant Professor (tenured in 2011) of Telecommunications with the Department of Industrial and Information Engineering, Second University of Naples. From 2014 to 2016, he was an Associate Professor of Signal Processing with the Department of Electronics and Telecommunications, NTNU. From 2016 to 2017, he was a Full Professor of Signal Processing with the Department of Electronic Systems, NTNU. From 2017 to 2019, he was a Principal Engineer with the Department of Advanced Analytics and Machine Learning, Kongsberg Digital AS, Norway. Since 2019, he has been a Full Professor of Statistical Machine Learning with the Department of Electronic Systems, NTNU, and the Director of the IoT@NTNU. His research interests include communication theory, data fusion, machine learning, and signal processing. He was awarded as an Exemplary Senior Editor of the IEEE COMMUNICATIONS LETTERS in 2018. He has been serving as an Executive Editor for the IEEE COMMUNICATIONS LETTERS since 2019, an Area Editor for the IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY since 2019, and an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL AND INFORMATION PROCESSING OVER NETWORKS since 2019 and the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS since 2015. He was a Senior Editor (from 2016 to 2019) and an Associate Editor (from 2012 to 2016) of the IEEE COMMUNICATIONS LETTERS.

From 2016 to 2017, he was a Full Professor of Signal Processing with the Department of Electronic Systems, NTNU. From 2017 to 2019, he was a Principal Engineer with the Department of Advanced Analytics and Machine Learning, Kongsberg Digital AS, Norway. Since 2019, he has been a Full Professor of Statistical Machine Learning with the Department of Electronic Systems, NTNU, and the Director of the IoT@NTNU. His research interests include communication theory, data fusion, machine learning, and signal processing. He was awarded as an Exemplary Senior Editor of the IEEE COMMUNICATIONS LETTERS in 2018. He has been serving as an Executive Editor for the IEEE COMMUNICATIONS LETTERS since 2019, an Area Editor for the IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY since 2019, and an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL AND INFORMATION PROCESSING OVER NETWORKS since 2019 and the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS since 2015. He was a Senior Editor (from 2016 to 2019) and an Associate Editor (from 2012 to 2016) of the IEEE COMMUNICATIONS LETTERS.