

Received October 29, 2021, accepted November 7, 2021, date of publication November 9, 2021, date of current version November 19, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3127084

Auto-NAHL: A Neural Network Approach for Condition-Based Maintenance of Complex Industrial Systems

TAREK BERGHOUT¹, MOHAMED BENBOUZID^{1,2,3}, (Fellow, IEEE),
S. M. MUYEEN⁴, (Senior Member, IEEE), TOUFIK BENTRACIA¹,
AND LEÏLA-HAYET MOUSS¹

¹Laboratory of Automation and Manufacturing Engineering, University of Batna 2, Batna 05000, Algeria

²UMR CNRS 6027, University of Brest, 29238 Brest, France

³Logistics Engineering College, Shanghai Maritime University, Shanghai 201306, China

⁴Department of Electrical Engineering, Qatar University, Doha, Qatar

Corresponding author: S. M. Muyeen (sm.muyeen@qu.edu.qa)

This work was supported by the Qatar National Library.

ABSTRACT Nowadays, machine learning has emerged as a promising alternative for condition monitoring of industrial processes, making it indispensable for maintenance planning. Such a learning model is able to assess health states in real time provided that both training and testing samples are complete and have the same probability distribution. However, it is rare and difficult in practical applications to meet these requirements due to the continuous change in working conditions. Besides, conventional hyperparameters tuning via grid search or manual tuning requires a lot of human intervention and becomes inflexible for users. Two objectives are targeted in this work. In an attempt to remedy the data distribution mismatch issue, we firstly introduce a feature extraction and selection approach built upon correlation analysis and dimensionality reduction. Secondly, to diminish human intervention burdens, we propose an Automatic artificial Neural network with an Augmented Hidden Layer (Auto-NAHL) for the classification of health states. Within the designed network, it is worthy to mention that the novelty of the implemented neural architecture is attributed to the new multiple feature mappings of the inputs, where such configuration allows the hidden layer to learn multiple representations from several random linear mappings and produce a single final efficient representation. Hyperparameters tuning including the network architecture, is fully automated by incorporating Particle Swarm Optimization (PSO) technique. The designed learning process is evaluated on a complex industrial plant as well as various classification problems. Based on the obtained results, it can be claimed that our proposal yields better response to new hidden representations by obtaining a higher approximation compared to some previous works.

INDEX TERMS Compressed sensing, condition monitoring, fault detection, hydraulic systems, industrial system, machine learning, neural networks, particle swarm optimization, predictive maintenance.

I. INTRODUCTION

Real time automated Condition Monitoring (CM) tools are the primary effective means for ensuring the continuity of systems operations and maximizing their revenue [1]. By fulfilling a well structured Condition-based Maintenance (CBM) policy, systems are repaired just at the right necessary time without affecting the entire workflow by unsuitable downtimes. Generally speaking, maintenance decisions are based

The associate editor coordinating the review of this manuscript and approving it for publication was Zheng Chen¹.

on the failure detection process, where in most cases Time-To-Repair (TTR) is known or can be estimated by qualified persons. As a result, accelerating the diagnostic process (failures/faults detection and identification) may constitute the predominant function in the design of a reliable forecasting model [2]. In fact, there are three fundamental ways to develop a coherent model capable of emulating the real operating behavior, namely: physical modeling, data-driven modeling and hybrid modeling. Deriving mathematical formulas from physical laws is an efficient way to provide an authentic simulation [3], [4]. However, the increasing

complexity of systems in terms of subcomponents and interaction dynamics like in the recent era of “Industry 4.0” makes the modeling procedure based on physical interpretations an intractable task to achieve. This situation therefore shifted the modeling tendencies toward hybrid and data-driven modeling processes [5]. It has been noticed that among data-driven analytical approaches (e.g. statistical analysis), Machine Learning (ML) methods have gained a central position as reflected by the large amount of recently published works. The cornerstone of the model construction depends on solving one of the three problems related to the nature of the timely driven-samples: classification, regression and clustering. The Volume, Velocity and Variety (i.e. the 3V of Big data) of the training samples reflect the depth of the encountered problem in the sense that deep 3V problems are tightly correlated with more sophisticated solutions [6] (The problem is also referred as ‘credit assignment path’, see Schmidhuber [7], §3). This massive growth in data due to advanced sensor technology results in a tremendous evolution of ML modeling procedures. Thus, this leads in turn to a shift from traditional ML modeling with both hybrid and ensemble learning to deep learning (DL). Furthermore, the extension to recent knowledge-driven learning paradigms such as Generative Adversarial Networks (GANs) and Transfer Learning (TL) cannot be neglected.

In the context of traditional ML tools, Roosefert Mohan *et al.* [8] proposed a hybrid algorithm that integrates both Adaptive Auto-Regressive Integrated Moving Average (ARIMA) and Support Vector Machine (SVM) to reduce catastrophic breakdowns of industrial machinery. Results of their application on a hydraulic sand-molding machine, by involving the oil contamination level as the primary health indicator, revealed that zero downtime could be achieved by increasing the Mean Time Between Failures (MTBF) up to 800%. Steurtewagen and Van den Poel [9] suggested to add a new aspect to ML prediction by considering statistical analysis using Shapley values to an XGBoost classifier when predicting different health stages of an oil refiner. Two gas compressor units, which are a very crucial part of an Atmospheric Residue DeSulpherizer (ARDS) unit, have been carefully studied through vibration analysis. Chen *et al.* [10] integrated a dynamic adaptation mechanism to a Radial Basis Function (RBF) neural network. This is done in order to permit adaptive resilient control under highly changing data with time delayed responses. The continuity of this work is provided in [11], where further improvement of the adaptive learning is proposed using a sliding mode control design.

On the other hand and under deep architectures, numerous interesting works are available such as the one conducted by Souza *et al.* [12]. Authors employed a deep Convolutional Neural Network (CNN) for classification of rotating machines faults by measuring the vibrations of motors and bearings. She and Jia [13] used an adaptive DL algorithm, named Bidirectional Gated Recurrent Unit (BiGRU), to solve the regression problem related to the Remaining Useful Life (RUL) prediction of the ball bearing. The algorithm

using vibration signals has been improved using the bootstrap method. A two hidden layers Recurrent Neural Network (RNN) is used by Chu *et al.* [14], to perform a double adaptive control process of dynamic systems.

In order to take advantage of the new generation of knowledge-guided learning paradigms such as GANs and TL in CBM applications, the works of Ding *et al.* [15] and Zhai *et al.* [16] are mentioned. Both GANs and TL can be integrated in a single framework when filling the gaps produced by incomplete list of learning patterns. For example, GANs are able to generate new instances so that the learning set can be reasonably augmented. In the meantime, TL is able to collect the necessary information from different learning domains to import some additional assumptions and extend the generalization process over both training/testing samples.

We conclude from the aforementioned works that almost all considered problems concern multi-class classifications treating a single target. Data is acquired via a few types of sensors, which are in general vibration-recording accelerometers with similar sampling rates. Multiple tools with different architectures and paradigms have been applied to study rotating machines. However, in real world applications, CM systems are usually characterized by multiple types of measurements with multi-rate recording sensors, multiple targets, and multi-classes for each target. In view of these constraints, we are motivated by gaining additional benefits through the support of such aspects at the level of CM within complex industrial plants. To the best of our knowledge, the case study of Helwig *et al.* [17] can be considered as the most appropriate choice as it offers a wide range of general investigations. By inspecting the available literature, a brief overview of works conducted so far on this scope is presented.

Helwig *et al.* [17] proposed a study involving a simulation model that makes it possible to mimic a reversible degradation of components conditions associated to a complex hydraulic system. After validating the physical modeling process and collecting necessary learning patterns from different scenarios including several types of sensors, the authors gave indications on how to solve such a complex prediction problem. The recorded data is massive with a higher level of 3V similar to real industrial applications. In their work, they refer to the problem as a feature selection problem rather than a pure prediction problem. Hence, a features extraction and selection process is initially opted to reduce dimensionality (i.e. reduce algorithmic complexity), which results in a more simplified version of the problem that can be solved by exploiting ML techniques. They used Spearman’s correlation analysis for features selection before dimensionality reduction with a Linear Discriminant Analysis (LDA) to perform a classification task. The same problem was treated by Schneider *et al.* [18], where a more efficient selection and extraction approach was developed with full automation for similar industrial applications. The authors introduced correlation analysis to select important learning features from different temporal and frequency statistical characteristics. After that, they fed the learning parameters into the LDA

algorithm for further processing (i.e. reducing dimensionality) before training the Mahalanobis classifier. To enhance the previous findings, Wu *et al.* [19] used a stacking based on SVM classifier in the form of an ensemble learning algorithm to solve the multi-class classification problem. The Random Forest (RF) algorithm has been adopted as a meta-model that brings together the results of several SVM models. In contrast to the previous original work, they used Pearson's correlation coefficient for selection after extraction of statistical characteristics. The last published paper on this topic was introduced by Ma *et al.* [20], where they proposed a Multi-Rate Sensor Information Fusion Strategy (MRSIFS) with parallel convolutional mappings for the extraction of the necessary learning patterns. In addition, time-frequency analysis has been investigated for the detection of defect signatures before the fine-tuning process of a CNN model.

We deduce from these works that very complex hybrid, ensemble or deep architectures have been elaborated to obtain higher classification results in the multi-task faults detection. Such type of faults is computationally expensive and lacks flexibility due to the high number of needed interventions. In addition, most of these algorithms rely on manual adjustment of hyperparameters or grid search via manually assigned parameters. Besides all these drawbacks, the prediction is a complex process since it is defined basically as a feature extraction and selection under a multi-class and multi-output prediction problem.

Therefore, and due to the efficiency of the DL representations paradigm, our main objective in this work is to reduce the generated data-driven modeling complexity in terms of computational cost and human intervention without disturbing learning performances. For this purpose, we used new efficient feature extraction and selection tools in order to reach satisfactory performances. In addition, we develop an Automatic artificial Neural network with an Augmented Hidden Layer (Auto-NAHL) to support addressing a non-complex automatic DL process.

By consequence, our contributions dealing with faults classification of complex industrial systems can be summarized as follows: i) Using multiple time-frequency features to collect necessary statistical information via a sliding window for an efficient training process; ii) Involving a two consecutive features selection approach, namely Spearman's correlation and Compressed Sensing (CS), to guarantee more meaningful and robust compressive representations; iii) Developing a new deep architecture for artificial neural network by extending hidden layer representation with multiple abstractions and non-linear mapping; iv) Employing a simple and non-complex learning scheme for the deep neural network based on Least Squares (LS) methods to avoid expensive iterative tuning of standard backpropagation algorithm when searching for the optimal solutions; v) Fully automating the new network thanks to Particle Swarm Optimization (PSO) for avoiding human intervention and exhaustive grid search during hyperparameters tuning. To validate this new learning path, it is firstly tested with the help of a set of classification

problems. After obtaining sufficient results, the next step consists of assessing its performance using a complex industrial plant.

The rest of this paper is organized as follows: Section II describes the exploited methods and the designed learning process. Section III depicts the studied system and associated dataset. Section IV is dedicated to the application and results discussion. Finally, Section V concludes with some remarks and perspectives.

II. PROPOSED METHODOLOGY

The flow diagram of Fig.1 elucidates different followed steps during health states classification. It can be noticed that three main steps are necessary to accomplish the learning process namely extraction, selection and automatic training of the prediction network. This section provides a separate brief description for each step.

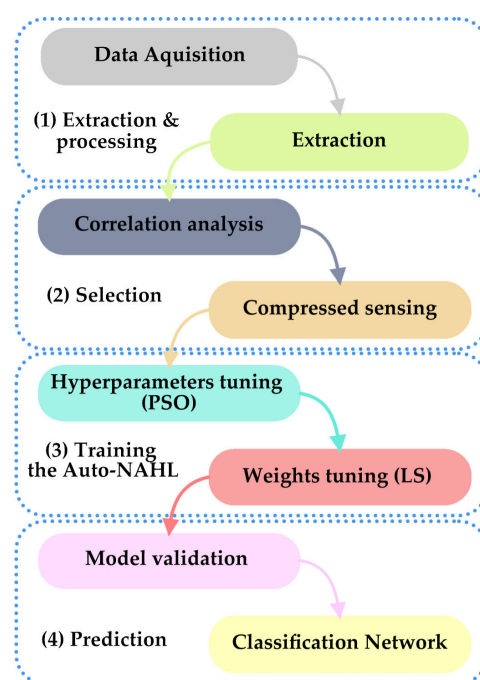


FIGURE 1. Design methodology of the prediction model.

A. EXTRACTION

We have 15 different features obtained from both time and frequency domains namely: mean, standard deviation (Std), skewness, kurtosis, peak to peak indices (Peak2Peak), root mean square (RMS), crest factor, shape factor, impulse factor, margin factor energy, spectral kurtosis mean (SKMean), standard deviation of spectral kurtosis (SKStd), skewness of spectral kurtosis (SKSkewness), kurtosis of spectral kurtosis (SKKurtosis). These features are collected from time and frequency domains by sliding a window over each one of the recorded cycles in the training data. More information about mathematical formulas of such statistical metrics can be found in [21].

The next step is to normalize these measurements in the interval [0, 1] using the common min-max normalization as expressed in (1) to equally distribute contribution chances in the learning process. The set $\{x_i, \tilde{x}_i\}$ denotes the original and normalized inputs, respectively at time instant i .

$$\tilde{x}_i = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \quad (1)$$

B. SELECTION

Although previous works [17]–[19] entirely depend on a single feature selection process based on correlation analysis, we propose in this work two consecutive steps relying on both correlation analysis and sparse coding using Compressed Sensing (CS).

1) CORRELATION ANALYSIS

As proved in Helwig et al. [17], the best selection based on correlation analysis has been achieved via involving Spearman’s coefficient. We have adopted the same strategy for features selection, however in our case the selection parameters have been chosen by involving expert’s knowledge. Therefore, the elements associated to the correlation heatmap in which their values belong to the interval [0.80, 1] have been selected as the default criterion for the first step of each extracted complete set of features.

2) COMPRESSED SENSING

Compressed Sampling (CS) recognized by different appellations such as “compressed sensing”, “magic reconstruction”, “compressive sampling”, is one of the most powerful sparse coding algorithms developed so far. Its role is to shrink any full rank representation to a sparse domain with higher level of zero coefficients. By doing so, a set of important descriptive patterns is pushed to a single zone within some sort of compression. Sparse domain transforms such as wavelets, Fourier and discrete cosine transform can be involved for this purpose. To measure the quality of results, CS theories entail the exact and unique restoration of such non-zero coefficient signals from their versions of higher level of incomplete data.

Accordingly, and based on the reference paper published by Donoho [22], the l_1 norm optimization-based solutions provide more adequate reconstruction leading to higher compression.

For a given non-zero coefficients signal vector x representing learning inputs of the machine learning model in our case, the full rank mapping Φ of such a vector at higher dimensions can be established via the rectangular matrix A with random linearly independent parameters as illustrated by (2).

$$\Phi = Ax \quad (2)$$

Obtaining the sparse representation Θ as given in (3) can be achieved using several frequency domain transforms tf including the previously mentioned

time frequency transforms.

$$\Theta = \Psi \Phi \quad (3)$$

where Ψ stands for the sparse matrix of the identity matrix I corresponding to the full rank feature mapping as explained by (4).

$$\Psi = tf(I) \quad (4)$$

The reconstruction vector \tilde{A} subject to solving the linear problem in (5) can be obtained by considering the l_1 norm minimization problem f_{\min} provided by (6).

$$\Theta = \tilde{A}x \quad (5)$$

$$f_{\min} = \min(\|x\|_1) \quad (6)$$

C. TRAINING OF THE AUTO-NAHL

Auto-NAHL is an artificial neural network designed with the aim of providing a more robust representation under simple non-iterative learning behavior. Training of the Auto-NAHL involves two main algorithms for two different processes. The naturally inspired random search algorithm PSO is used to tune the training hyperparameters. In the meantime, the LS method is used for setting the learning weights. This section presents the most important basics of the proposed network, passing through its architecture and learning rules.

1) NETWORK ARCHITECTURE

Fig. 2 showcases the architecture of the proposed Auto-NAHL. As observed, the structure consists of three main layers; the input layer, the augmented hidden layer and the output layer. In the learning paradigms of conventional Artificial Neural Networks (ANNs) and deep ANNs, the network is typically initialized by generating random learning weights and biases. After that, the weights and biases are updated with iterative backpropagation by moving around the training samples. If the training process achieves the desired results, the learning must be completed; otherwise the learning will start over with other initial weights. This reflects that learning is a very expensive tuning process [23]. Alternative solutions use a set of ANNs trained in parallel with different sets of weights and biases to be able to analyze most of the possible cases and judge which network is the best approximator [24]. However, and regardless of retraining, ensemble learning with ANNs is a long and tedious approach. As a simple solution to the weight selection problem, we have come up with an augmented representation, which is a simple and more effective way to train once the hidden layer without even considering retraining.

Firstly, the driven samples must be mapped to multiple representations in a form of several temporary hidden layers without activation \tilde{h} , as explained by (7). The elements $\{w, b\}$ refer to the input weight matrices and biases vectors of each feature mapping, respectively. Secondly, each temporary hidden layer is multiplied with a discount factor $\gamma \in [0, 1]$ to determine the importance of each feature map in the learning process. Thirdly, these feature mappings are added together

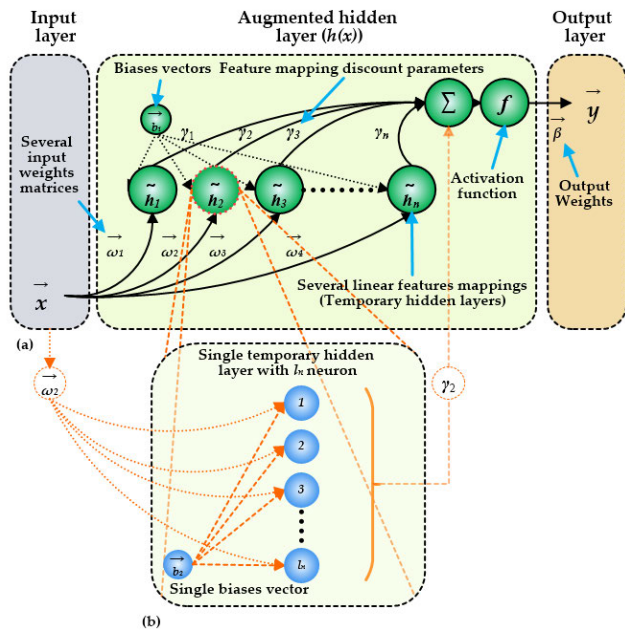


FIGURE 2. Architecture of the proposed Auto-NAHL (a) General architecture with several temporary hidden layers; (b) “Zoom in” on a single temporary hidden layer.

element by element and the result will be activated with an activation function f as explained by (8). The parameter n is the number of temporary hidden layers. The formula in (8) is applicable only in case of temporary hidden layers having the same number of neurons l_n .

$$\tilde{h}_i = w_i x + b_i \tag{7}$$

$$h = f\left(\sum_{i=1}^n \gamma_i (w_i x + b_i)\right) = f\left(\sum_{i=1}^n \gamma_i (\tilde{h}_i)\right) \tag{8}$$

Once the hidden layer is analytically determined, the outputs y will be simply settled according to (9). The output weights β are responsible for the process of universal approximation and generalization of the entire neural network.

$$y = h\beta \tag{9}$$

2) WEIGHTS TUNING

To avoid the complexity of expensive iterative learning of the backpropagation algorithm, we propose to train the neural network by considering the output weights as the main element of the approximation. As a result, the input weights for each temporary hidden layer are generated randomly and independently of the training data as well as of each other. Therefore, the output weights are determined analytically according to LS methods as explained in so-called Extreme Learning Machine (ELM) theories [25], [26]. Formula (10) illustrates the method for weights tuning which is primarily focused on using the hidden layer pseudo-inverse matrix $pinv(h)$. The simplest mathematical definition of the $pinv$ function is given by the inverse inv of the dot product of

matrix h and its transpose h' , with h a linearly independent matrix (full rank). The reason for the term “pseudo” lies in its similarity to the generalized known inverse of matrix variant.

$$\beta = pinv(h)y \mid pinv(h) = inv(h \cdot h') \tag{10}$$

In this context, the main learning parameters (output weight β) are calculated once through a single iteration. Hence, it does not require any updating, which is different from traditional iterative learning such as backpropagation or contrastive divergence [27]. For this reason, the NAHL by itself becomes an offline learning algorithm (cannot be dynamically updated). Such situation motivates its adoption in real time monitoring systems only if the monitored system presents a static process.

3) HYPERPARAMETERS TUNING

In the current study, the learning hyperparameters are listed as follows: number of neurons in each layer, number of temporary hidden layers, activation function type, and discount factors of each temporary hidden layer, which can be represented by $P = \{l_n, n, f, \gamma_1, \dots, \gamma_n\}$. The notation P is considered as a single particle (individual) in the PSO population ρ .

The PSO algorithm is one of the algorithms inspired by Swarm Intelligence (SI) and it was introduced in 1995 [28]. It is widely used in the optimization of ML algorithms as well as in ensemble learning and parameter selection due to its simplicity, precision and speed of convergence. PSO is an iterative algorithm that studies the vectors coordinates movements of a complete set of randomly generated population (initial solutions). It begins with a randomly generated initial population and updates their positions and movement velocities based on variations in the fitness function. Once the search conditions are reached, the algorithm takes the best particle as the best solution of the optimization problem. PSO algorithm can be divided into the following steps [29]:

- 1) Initialize random search parameters including inertia weights, lower bound LB and upper bound UB constraints for each element of the particle P ;
- 2) Define the objective function and its constraints. In this case, the approximation process of the Auto-NAHL is our main objective;
- 3) Generate randomly an initial population $\vec{\rho}$, which represents a set of particles \vec{P} ;
- 4) Evaluate the fitness function f_{it} for each particle and determine whether the stopping criterion satisfies the tolerance error. In this case of the Auto-NAHL training, the fitness function is defined as in (11). The notation C is the classification rate of the testing set.

$$f_{it} = \|1 - C\|_2 \tag{11}$$

- 5) Determine the best particle so far;
- 6) Update velocity \vec{v} and current population using (12) and (13). Also, check the boundary constraints.

$$\vec{v}_{new} = \vec{v}_{old} + \vec{U}(0, \Omega_1) \otimes (\vec{p} - \vec{\rho}) + \vec{U}(0, \Omega_2) \otimes (\vec{p} - \vec{\rho}) \tag{12}$$

$$\vec{\rho}_{new} = \vec{\rho}_{old} + \vec{v}_{old} \tag{13}$$

\vec{U} is randomly generated for each particle for different iterations. Variables $\{\Omega_1, \Omega_2\}$ are the inertia weights for population directional purposes. The operator \otimes is component-wise multiplication;

- 7) If the expected tolerance error is achieved or the maximum number of iterations is not reached, go to step 8. Otherwise, the population update must be continued;
- 8) Generate a new population and repeat steps 4, 5 and 6 with rechecking the condition in step 7.

Since the main focus of this paper is to prove the NAHL capability in universal approximation, automation of the learning process was made upon the use of PSO in its standard form. The pseudo-code presented in Algorithm 1 provides further simplified representation of PSO technique.

Algorithm 1: Standard Algorithm of PSO Technique

- Inputs: $LB, UB, \Omega_1, \Omega_2$, desired tolerance, iterations number.
 Outputs: best particle P
- 1 Generate randomly initial population ρ of particles P ;
 - 2 Evaluate fitness function $f_{it} = \|1 - C\|_2$;
 - 3 **While** tolerance or iterations conditions are not met, then
 - 4 Update velocity
 $\vec{v}_{new} = \vec{v}_{old} + \vec{U}(0, \Omega_1) \otimes (\vec{p} - \vec{\rho}) + \vec{U}(0, \Omega_2) \otimes (\vec{p} - \vec{\rho})$;
 - 5 Update population $\vec{\rho}_{new} = \vec{\rho}_{old} + \vec{v}_{old}$;
 - 6 **Otherwise**;
 - 7 Break the random search mechanism;
 - 8 **End While**.

III. PROBLEM AND DATASET DESCRIPTION

Our faults prediction case study is built on a complex hydraulic system dataset. This dataset is the simulation result of reversible degradation of several sub-components of the system. Eventually, it was made publically available by the Center for Mechatronics and Automation Technology in Saarbrücken, Germany [17], [19], [30]. As shown in Fig. 3, this system is composed of two main subsystems: the cooling-filtration system and the working system. These two main parts are connected to an oil tank.

In this system, the element MP1 appearing in Fig.3a is a 3.3 kW pumping motor of the working system and is mainly controlled by the pressure relief valve V11. To simulate the operating states of this moving machine, several cycles are randomly generated with load variations according to previously well assigned boundary constraints. A total of 18 sensors with different acquisition rates was placed at different locations of the test rig to collect enough information for the design of the condition monitoring system.

By saying “enough information” we are referring to the fact that the training data should contain learning patterns from the entire life cycle of the system. These learning patterns may describe several health states under different scenarios (working conditions). The amount of collected samples for each health state in a particular scenario should

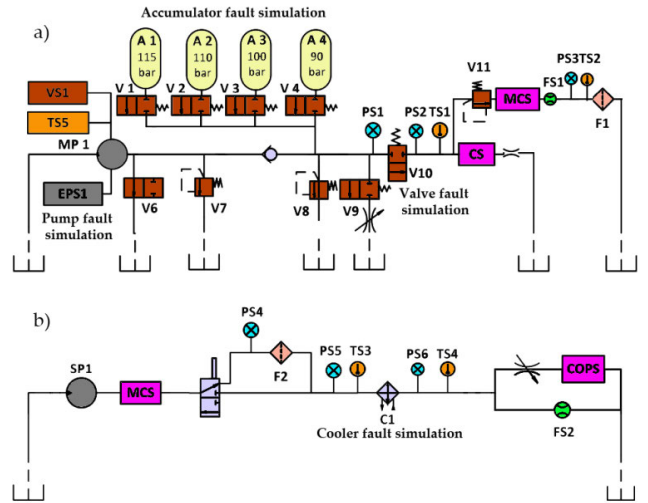


FIGURE 3. Components of the simulated hydraulic system [17] a) Working system; b) Cooling and filtration system.

be quite balanced and contains no missing information or labels. Besides, a primarily data-driven evaluation process on acceptable amount of new unseen samples will further judge if the training data is “complete” or not. Table 1 is elaborated to recapitulate these sensors types, their names and sampling rates.

TABLE 1. Components installed sensors on test rig.

Name	Rate	Type
PS1→PS6	100 Hz	Pressure
EPS1	100 Hz	Motor power
FS1 and FS2	10 Hz	Volume flow
TS1→TS5	1 Hz	Temperature
VS1	1 Hz	Vibration
CE	1 Hz	Cooling efficiency
MCS	1 Hz	Cooling power
COPS	1 Hz	System efficiency

The collected samples are continuously stored in a personal computer using the “Beckhoff CX5020 PLC” data acquisition system. 2205 work cycles are recorded simultaneously every 60s. Each cycle begins with the closing of V10 for the first 10 seconds, followed by 50s of random variation of the proportional pressure of V11. Depending on the recording rate of each sensor, each cycle (observation) can contain from 60 to 6000 samples. The fault injection process was carried out on four main components with each component being subject to different types of faults. Table 2 summarizes in more detail the important information related to the categories of health conditions. From Table 2, it can be observed that the classes are distributed quietly in a balance regarding abnormal function. The only subset which could provide more prediction problems is the last one related to the accumulator. This is the reason why even advanced methods (see Wu et al. [19], Fig.7d) are not sufficient to achieve 100% correct classification.

TABLE 2. Studied health states of targeted complex system.

Component	Working condition	Health state	Labels ¹	Cycles
Cooler	Cooling power decrease	Max efficiency	1	741
		Efficiency decrease	2	732
		Nearly total failure	3	732
Valve	Switching characteristic degradation	Optimal condition	1	1125
		Small lag	2	360
		Severe lag	3	360
		Nearly total failure	4	360
Pump	Internal leakage	No leakage	1	1221
		Small leakage	2	492
		Severe leakage	3	492
Accumulator	Gas leakage	Optimal pressure	1	808
		Small pressure loss	2	599
		Severe pressure loss	3	399
		Nearly total failure	4	399

¹User-given label during the current study

IV. APPLICATION AND RESULTS DISCUSSION

This section is devoted to the study of the designed algorithm performance during the classification of different health states of the targeted system. Therefore, in order to provide a more rigorous investigation, we have divided the evaluation process into two main phases; “*phase I*” where a set of well-known classification benchmarks are adopted to primarily test the designed network and “*phase II*” where we apply our methodology on the industrial complex system.

A. PHASE I

In this phase, the experiments were carried out on the list of datasets presented in Table 3 and previously used in the study of Lu *et al.* [31]. Datasets are already prepared, min-max normalized, and split into training and testing sets using *k*-fold cross validation (*k* = 5).

TABLE 3. Primarily evaluation data.

Dataset	Attributes	Classes	Samples
Wine	13	3	178
Vehicle	18	4	846
Segment	19	7	2310
Page	10	5	5473
Satimage	36	6	6435
USPS	256	10	9497
Letter	16	26	15000

It is known in ELM theories of neural networks with a single layer that incrementing the number of neurons will automatically increase the precision of learning. As a result, it leads us to introduce a comparison experiment as a first attempt to provide conclusions on the newly designed Auto-NAHL architecture. Consequently, the constraints of the upper bound *UB* related to the number of neurons I_n will be gradually increased during the learning process. The accuracy of the tests will be captured and recorded at each iteration. Fig.4 depicts some examples of the incremental learning. The convergence behavior of classification performance confirms that Auto-NAHL divulges higher accuracy resulting in faster stable responses. Consequently, augmented representations

have shown to be helpful in providing a more meaningful representation than ordinary hidden layers, which directly yields a more accurate estimation.

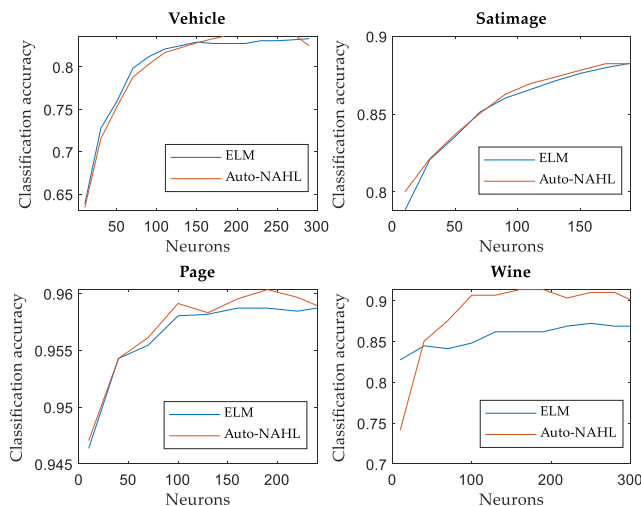


FIGURE 4. Variation of classification accuracy during incremental learning.

Table 4 summarizes obtained classification rates compared to ELM and the original source of these datasets. It is indicated that Auto-NAHL is able to accurately classify unseen samples better than both ELM and original works providing the primarily evaluation datasets. This proves that the adopted deep architecture exhibits better learning representation than ordinary kernel or full rank mapping.

TABLE 4. Evaluation results of phase I.

Dataset	Auto-NAHL	ELM	Lu <i>et al.</i> [31]
Wine	0.9917	0.9917	0.9832
Vehicle	0.8699	0.8699	0.8214
Segment	0.9732	0.9042	0.9686
Page	0.9885	0.9504	0.9600
Satimage	0.9295	0.8050	0.9108
USPS	0.9601	0.9392	0.9586
Letter	0.9250	0.8820	0.9011
Mean	0.9483	0.9061	0.9291

B. PHASE II

The basic evaluation of the proposed architecture shows significant improvement in the neural network learning performance. Now, we are in position to tackle its application on the complex industrial system. This part introduces feature selection mechanism and classification results versus previous published works dealing with the same data.

1) FEATURES SELECTION

After extracting statistical features with the help of a sliding window, each sensor is separately exposed to correlation analysis with Spearman’s heatmap. Only important parameters that satisfy the selection conditions (i.e. [0.8, 1]) are passed to the next step of compression with CS. The example

of Fig.5 illustrates selected features in single sensor measurements (i.e. Vibration sensor (VS1)). The labels tickets are named after the type of extracted features. Green and blue colors in Fig5a explains the weakness of correlation between the learning features (they do not behave similarly). However, in Fig.5b such non-correlation is reduced which further improves quality of learning samples.

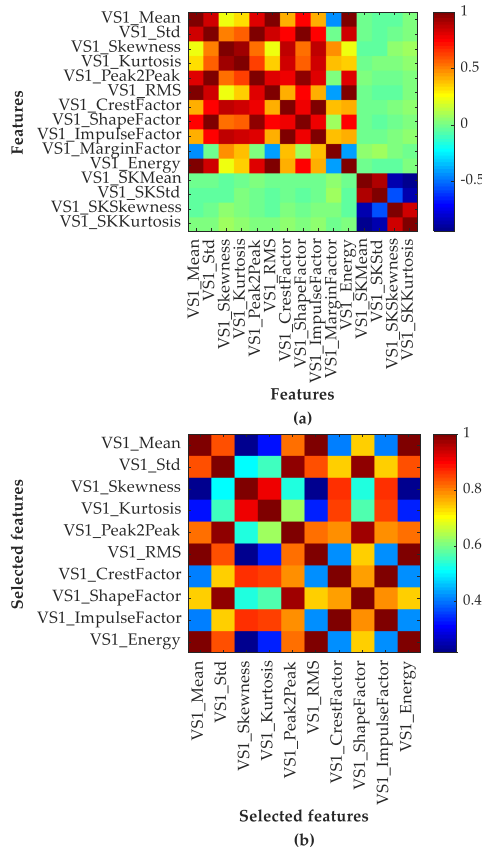


FIGURE 5. Selected features based on Spearman’s heat map (sensor CE) a) Original features; b) Chosen features.

After selection based on correlation analysis, a feature compression with CS is adopted. As indicated in Fig.6 and with the help of frequency transform “Discrete Cosine Transform (DCT)” algorithm, the more expressive features are automatically pushed into a single zone with less number of expressive elements, which makes it easier to be selected according to predefined coefficient with respect to each sensor.

In this case, the used software to train the CS algorithm is the l_1 magic reconstruction toolbox provided by Candes et al. [32] at Stanford statistics web page [33]. The selection mechanism is straightforward by choosing the first 10 maximum consecutive elements from the sparse version of the dataset in which it is specified using the red circle (Fig.6d).

2) PREDICTING WITH AUTO-NAHL

The training process is immediately followed the feature selection with compressed sensing in order to avoid algorithmic complexity by repeating the experiments using cross validation as in Helwig et al. [17]. Also, in order to prove that the algorithm can improve data distribution by involving the NAHL technique, we divide the simulated data randomly into training and test sets with a training ratio equal to 80% then; there is no need to tune the learning hyperparameters.

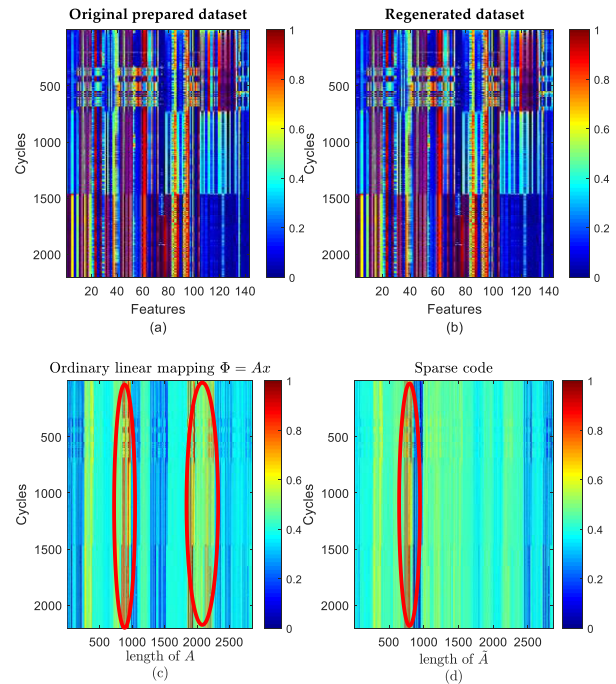


FIGURE 6. Features selection with compressed sensing a) Prepared data after feature extraction; b) Reconstructed data from the sparse domain (RMSE = 1.30×10^{-16}); c) Ordinary full rank mapping; d) Sparse version of the features space.

In fact, the tuning process is fully automated with the help of PSO algorithm. Fig.7 depicts discount parameters γ updates when moving from the initial population to the final best particles that can be considered for selection. Population movement is recorded when the Auto-NAHL was trained on the Cooler health state classification. It can be seen that the population coordinates are entirely changed toward certain specific points. As a result, such generated points are elected as discount parameters among initial ones to design the best universal approximation function.

In the meantime, Fig.8 emphasizes random search mechanism on the hidden layers neurons l_n and best activation function f for the Auto-NAHL.

Initial generated and finally updated populations are clearly shown. After a well spread solutions, final judgment with PSO entails the push of the learning parameters toward a more effective architecture. This is a very useful approach compared to traditional incremental construction with additive hidden nodes such as in Huang et al. [34].

In each prediction problem, the best solution changes from an application to another. However, all solutions agree on the use of three temporary feature maps in case of the Auto-NAHL. Accordingly, Table 5 is introduced to address results of automatic hyperparameters tuning for each prediction problem. The algorithm of random search PSO makes it easy to study the adequate architecture for the learning process, as it is able to determine the appropriate feature mapping and necessary activation for the neural network.

Table 6 indicates the classification performances of the designed algorithm with respect to some existing methods in the literature.

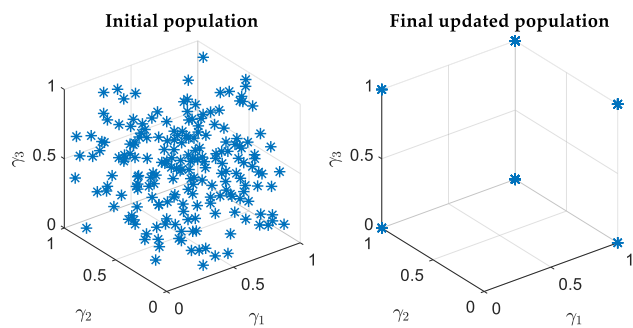


FIGURE 7. Discount factor updates (for temporary hidden layers) during random search with PSO algorithm.

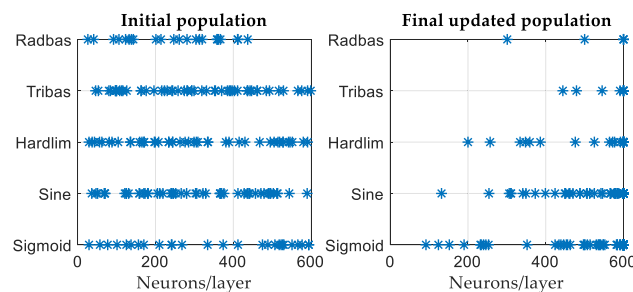


FIGURE 8. Population updates when searching for the appropriate number of neurons and activation functions.

TABLE 5. Results of automatic hyperparameters tuning.

Component	l_n	f	γ_1	γ_2	γ_3
Cooler	566	sigmoid	0.99	0.87	0.54
Valve	528	sine	0.56	0.95	0.57
Pump	274	sine	0.25	0.19	0.75
Accumulator	417	sigmoid	0.27	0.07	0.02

It is noticed that the Auto-NAHL algorithm achieves promising performances by providing the best classification accuracy. This is due to the effective extraction and selection approach that follows two main steps namely the Spearman’s correlation analysis and CS mapping. Moreover, the advantage of the time window sliding through multiple characteristics should not be neglected.

TABLE 6. Evaluation results on the complex system.

Component	Auto-NAHL	ELM	Helwig et al. [17]	Wu et al.[19]
Cooler	1	1	1	1
Valve	1	0.93	1	1
Pump	1	0.9977	0.98	1
Accumulator	0.9637	0.9637	0.904	0.765
Mean	0.9909	0.9728	0.9710	0.9413

The confusion matrix plotted in Fig.9 is an additional metric that confirms the results of Table 6 by providing more explanation regarding real and predicted classes. Moreover, it is quite clear that the false negatives of confusion matrix related to failure modes are less abundant to alter the prediction process.

Numerical experiments of this study are carried out on a personal computer having the following characteristics: Intel (R) Core (TM) i5-3427 CPU @ 1.80 GHz 2.30 GHz, and RAM memory of 8.00 GB. Table 7 addresses the time consumed during the training process. It is shown that even with additive search algorithm PSO, the entire learning process needs less time compared to some similar works. Hence, it becomes a very suitable CBM tool when diagnosing health condition.

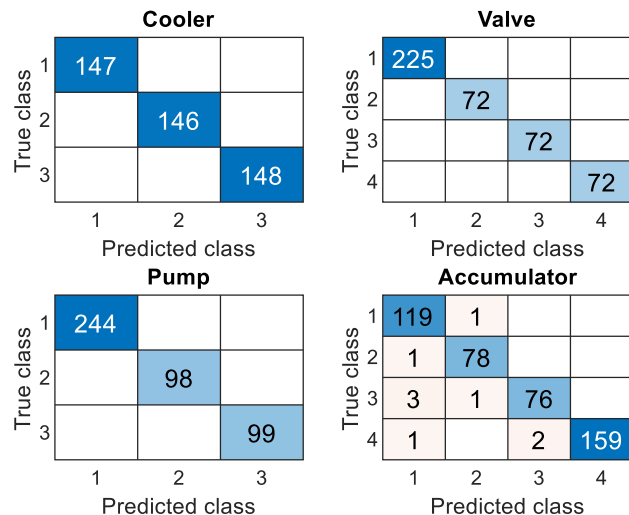


FIGURE 9. Confusion matrix of classification accuracy results.

TABLE 7. Learning time (s).

Component	Auto-NAHL	Wu et al.[19]
Cooler	0.37	0.68
Valve	0.63	1.07
Pump	0.24	0.78
Accumulator	0.54	2.57
Mean	0.4450	1.2750

For a predictive maintenance algorithm, or ML prediction algorithm in general, to be qualified as a good tool, three elementary properties should be satisfied including accuracy,

reduced complexity (i.e. be user-friendly) and low computation costs. The later criterion has been already explored by Huang [35] (see [35], Fig.4), when investigating certain comparisons between ELM and familiar iterative machine learning methods. In our work, accuracy of the proposed learning scheme is demonstrated by conducting comparisons with respect to a set of previous published results as indicated in subsections IV-A and IV-B. To consolidate this aspect, numerical evaluation and visual metrics have been provided as convincing arguments (Fig.4, Table 4 and Table 6). Furthermore, the simplicity of the learning rules of the proposed algorithm reflects its low complexity. Finally, the computation time depicted in Table 7 constitutes another prominent measure that cannot be ignored for the evaluation of the learning process simplicity.

At the current stage, we would like to point out that our contributions are not only limited to enhancing the observability of the studied hydraulic system based on ML techniques but also to showcasing the Auto-NAHL superior capacity for this task. Hence, numerous comparative analyses of our findings against similar works available in literature dedicated to hydraulic systems. However, we did not focus much more on recent deep complex networks due to their well-known computational expensiveness even they may offer good results [36].

To shed more light on the Auto-NAHL performance in terms of classification rate, Table 8 highlights the comparison with basic deep learning algorithms (i.e. LSTM, 1 dimensional CNN (1D-CNN) and deep belief network (DBN)). To make sure that the algorithms are studied under the same criteria, we use only a single layer of feature maps for each deep network (as in the Auto-NAHL where only a single augmented layer is used). Hyperparameters are set to be updated within a grid search mechanism. The generated classification accuracy results also form in this experiment strong evidence in favor of the Auto-NAHL. Both LSTM and DBN illustrate universal approximation capacity better than 1D-CNN, which justifies the compatibility of CNN with more complex prediction procedures such as image processing.

The Auto-NAHL has an advantage of turning deep complex and ensemble learning (i.e. multiple representations for a single layer in the same classifier) into a very simple tuning scheme using the proposed tuning theories. With regard to classification, the deep layered representations boost the NAHL response for universal approximation tasks. In our Auto-NAHL framework, there is no need of initial

hyperparameters tuning since the whole process is automatically preformed when user unleashes the training process.

V. CONCLUSION

Our goal in this work has been twofold. On the one hand, a new artificial neural network approach named Auto-NAHL specifically has been designed to evaluate health condition of complex industrial systems. The main idea is to allow the hidden layer to scan multiple effective representations when dealing with the approximation problem. The tuning mechanism has been simplified by following least squares method as explained in ELM theories. Hyperparameters search has been fully automated with the help of PSO algorithm. On the other hand, this work has also introduced a new feature extraction and selection scheme by involving time frequency characteristics, correlation analysis and CS. This learning methodology was firstly investigated on several known classification problems to assess at a first glance the algorithm performance, where satisfactory measures were obtained. The next step concerns the algorithm application on a complex hydraulic system. The obtained results support the preprocessing scheme by showing promising performances of reconstruction. In addition, classification metrics including classification accuracy and confusion matrix allowed demonstrating the efficacy of the Auto-NAHL. It is worth mentioning that the computational time associated to weights and biases tuning was significantly improved. As the current Auto-NAHL was limited to deep architecture with a single augmented hidden layer, one of the possible trends could be the use of a deeper architecture with multiple augmented hidden layers. Also, it is worthy to involve more representation learning paradigms of deep learning. Regarding training frameworks, it is recommended to consider reprogramming the Auto-NAHL by exploiting the standard backpropagation method in order to gain further insights about the computation time behavior of our proposed network.

REFERENCES

- [1] T. Berghout, L.-H. Mouss, T. Bentrucia, E. Elbouchikhi, and M. Benbouzid, "A deep supervised learning approach for condition-based maintenance of naval propulsion systems," *Ocean Eng.*, vol. 221, Feb. 2021, Art. no. 108525, doi: [10.1016/j.oceaneng.2020.108525](https://doi.org/10.1016/j.oceaneng.2020.108525).
- [2] G. A. Susto, A. Schirru, S. Pampuri, S. McLoone, and A. Beghi, "Machine learning for predictive maintenance: A multiple classifier approach," *IEEE Trans. Ind. Informat.*, vol. 11, no. 3, pp. 812–820, Jun. 2015, doi: [10.1109/TII.2014.2349359](https://doi.org/10.1109/TII.2014.2349359).
- [3] T. Wang, M. Lu, X. Zhu, and R. Patton, "Aggressive maneuver oriented robust actuator fault estimation of a 3-DOF helicopter prototype considering measurement noises," *IEEE/ASME Trans. Mechatronics*, early access, Jun. 8, 2021, doi: [10.1109/TMECH.2021.3087193](https://doi.org/10.1109/TMECH.2021.3087193).
- [4] K. Liu, X. Hu, H. Zhou, L. Tong, D. Widanalage, and J. Marco, "Feature analyses and modelling of lithium-ion batteries manufacturing based on random forest classification," *IEEE/ASME Trans. Mechatronics*, early access, Jan. 5, 2021, doi: [10.1109/TMECH.2020.3049046](https://doi.org/10.1109/TMECH.2020.3049046).
- [5] Y. Liu, W. Yu, T. Dillon, W. Rahayu, and M. Li, "Empowering IoT predictive maintenance solutions with AI: A distributed system for manufacturing plant-wide monitoring," *IEEE Trans. Ind. Informat.*, vol. 18, no. 2, pp. 1345–1354, Feb. 2022, doi: [10.1109/tii.2021.3091774](https://doi.org/10.1109/tii.2021.3091774).
- [6] T. Ivanov, N. Korfiatis, and R. V. Zicari, "On the inequality of the 3 V's of big data architectural paradigms: A case for heterogeneity," 2013, *arXiv:1311.0805*.

TABLE 8. Comparison results against basic deep networks.

Component	Auto-NAHL	LSTM	1D-CNN	DBN
Cooler	1	0.998	0.997	0.997
Valve	1	0.791	0.510	0.725
Pump	1	0.980	0.554	0.951
Accumulator	0.9637	0.812	0.509	0.680
Mean	0.9909	0.895	0.642	0.838

- [7] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015, doi: [10.1016/j.neunet.2014.09.003](https://doi.org/10.1016/j.neunet.2014.09.003).
- [8] T. R. Mohan, J. P. Roselyn, R. A. Uthra, D. Devaraj, and K. Umachandran, "Intelligent machine learning based total productive maintenance approach for achieving zero downtime in industrial machinery," *Comput. Ind. Eng.*, vol. 157, Jul. 2021, Art. no. 107267, doi: [10.1016/j.cie.2021.107267](https://doi.org/10.1016/j.cie.2021.107267).
- [9] B. Sturtewagen and D. Van den Poel, "Adding interpretability to predictive maintenance by machine learning on sensor data," *Comput. Chem. Eng.*, vol. 152, Sep. 2021, Art. no. 107381, doi: [10.1016/j.compchemeng.2021.107381](https://doi.org/10.1016/j.compchemeng.2021.107381).
- [10] Z. Chen, F. Huang, W. Sun, J. Gu, and B. Yao, "RBF-neural-network-based adaptive robust control for nonlinear bilateral teleoperation manipulators with uncertainty and time delay," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 2, pp. 906–918, Apr. 2020, doi: [10.1109/TMECH.2019.2962081](https://doi.org/10.1109/TMECH.2019.2962081).
- [11] Z. Chen, F. Huang, W. Chen, J. Zhang, W. Sun, J. Chen, J. Gu, and S. Zhu, "RBFNN-based adaptive sliding mode control design for delayed nonlinear multilateral telerobotic system with cooperative manipulation," *IEEE Trans. Ind. Informat.*, vol. 16, no. 2, pp. 1236–1247, Feb. 2020, doi: [10.1109/TII.2019.2927806](https://doi.org/10.1109/TII.2019.2927806).
- [12] R. M. Souza, E. G. S. Nascimento, U. A. Miranda, W. J. D. Silva, and H. A. Lepikson, "Deep learning for diagnosis and classification of faults in industrial rotating machinery," *Comput. Ind. Eng.*, vol. 153, Mar. 2021, Art. no. 107060, doi: [10.1016/j.cie.2020.107060](https://doi.org/10.1016/j.cie.2020.107060).
- [13] D. She and M. Jia, "A BiGRU method for remaining useful life prediction of machinery," *Meas. J. Int. Meas. Confed.*, vol. 167, no. Aug. 2019, Jan. 2021, Art. no. 108277, doi: [10.1016/j.measurement.2020.108277](https://doi.org/10.1016/j.measurement.2020.108277).
- [14] Y. Chu, J. Fei, and S. Hou, "Adaptive global sliding-mode control for dynamic systems using double hidden layer recurrent neural network structure," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 4, pp. 1297–1309, Apr. 2020, doi: [10.1109/TNNLS.2019.2919676](https://doi.org/10.1109/TNNLS.2019.2919676).
- [15] Y. Ding, M. Jia, Q. Miao, and P. Huang, "Remaining useful life estimation using deep metric transfer learning for kernel regression," *Rel. Eng. Syst. Saf.*, vol. 212, Aug. 2021, Art. no. 107583, doi: [10.1016/j.ress.2021.107583](https://doi.org/10.1016/j.ress.2021.107583).
- [16] S. Zhai, B. Gehring, and G. Reinhart, "Enabling predictive maintenance integrated production scheduling by operation-specific health prognostics with generative deep learning," *J. Manuf. Syst.*, Mar. 2021, doi: [10.1016/j.jmsy.2021.02.006](https://doi.org/10.1016/j.jmsy.2021.02.006).
- [17] N. Helwig, E. Pignatelli, and A. Schutze, "Condition monitoring of a complex hydraulic system using multivariate statistics," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (IMTC) Proc.*, May 2015, pp. 210–215, doi: [10.1109/I2MTC.2015.7151267](https://doi.org/10.1109/I2MTC.2015.7151267).
- [18] T. Schneider, N. Helwig, and A. Schutze, "Automatic feature extraction and selection for condition monitoring and related datasets," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (IMTC)*, May 2018, pp. 1–6, doi: [10.1109/I2MTC.2018.8409763](https://doi.org/10.1109/I2MTC.2018.8409763).
- [19] J. Wu, P. Guo, Y. Cheng, H. Zhu, X.-B. Wang, and X. Shao, "Ensemble generalized multiclass support-vector-machine-based health evaluation of complex degradation systems," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 5, pp. 2230–2240, Oct. 2020, doi: [10.1109/TMECH.2020.3009449](https://doi.org/10.1109/TMECH.2020.3009449).
- [20] X. Ma, P. Wang, B. Zhang, and M. Sun, "A multirate sensor information fusion strategy for multitask fault diagnosis based on convolutional neural network," *J. Sensors*, vol. 2021, pp. 1–17, Jun. 2021, doi: [10.1155/2021/9952450](https://doi.org/10.1155/2021/9952450).
- [21] J. B. Ali, L. Saidi, S. Harrath, E. Bechhoefer, and M. Benbouzid, "Online automatic diagnosis of wind turbine bearings progressive degradations under real experimental conditions based on unsupervised machine learning," *Appl. Acoust.*, vol. 132, pp. 167–181, Mar. 2018, doi: [10.1016/j.apacoust.2017.11.021](https://doi.org/10.1016/j.apacoust.2017.11.021).
- [22] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006, doi: [10.1109/TIT.2006.871582](https://doi.org/10.1109/TIT.2006.871582).
- [23] X.-F. Liu, Z.-H. Zhan, and J. Zhang, "Resource-aware distributed differential evolution for training expensive neural-network-based controller in power electronic circuit," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, May 7, 2021, doi: [10.1109/TNNLS.2021.3075205](https://doi.org/10.1109/TNNLS.2021.3075205).
- [24] J.-Y. Kim and S.-B. Cho, "A deep neural network ensemble of multimodal signals for classifying excavator operations," *Neurocomputing*, Jan. 2021, doi: [10.1016/j.neucom.2020.01.127](https://doi.org/10.1016/j.neucom.2020.01.127).
- [25] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, nos. 1–3, pp. 489–501, 2006, doi: [10.1016/j.neucom.2005.12.126](https://doi.org/10.1016/j.neucom.2005.12.126).
- [26] X. Shi, Q. Kang, J. An, and M. Zhou, "Novel L1 regularized extreme learning machine for soft-sensing of an industrial process," *IEEE Trans. Ind. Informat.*, vol. 18, no. 2, pp. 1009–1017, Feb. 2022, doi: [10.1109/TII.2021.3065377](https://doi.org/10.1109/TII.2021.3065377).
- [27] J. Zhang, Y. Li, W. Xiao, and Z. Zhang, "Non-iterative and fast deep learning: Multilayer extreme learning machines," *J. Franklin Inst.*, vol. 357, no. 13, pp. 8925–8955, 2020, doi: [10.1016/j.jfranklin.2020.04.033](https://doi.org/10.1016/j.jfranklin.2020.04.033).
- [28] S. Roy, S. Biswas, and S. S. Chaudhuri, "Nature-inspired swarm intelligence and its applications," *Int. J. Modern Educ. Comput. Sci.*, vol. 6, no. 12, pp. 55–65, Dec. 2014, doi: [10.5815/ijmecs.2014.12.08](https://doi.org/10.5815/ijmecs.2014.12.08).
- [29] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intell.*, vol. 1, no. 1, pp. 33–57, Jun. 2007, doi: [10.1007/s11721-007-0002-0](https://doi.org/10.1007/s11721-007-0002-0).
- [30] (2018). *Condition Monitoring of Hydraulic Systems Data Set*. Accessed: Sep. 22, 2021. [Online]. Available: <https://archive.ics.uci.edu/ml/datasets/Condition+monitoring+of+hydraulic+systems>
- [31] F. Lu, J. Jiang, J. Huang, and X. Qiu, "Dual reduced kernel extreme learning machine for aero-engine fault diagnosis," *Aerosp. Sci. Technol.*, vol. 71, pp. 742–750, Dec. 2017, doi: [10.1016/j.ast.2017.10.024](https://doi.org/10.1016/j.ast.2017.10.024).
- [32] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006, doi: [10.1109/TIT.2005.862083](https://doi.org/10.1109/TIT.2005.862083).
- [33] Stanford Statistics. *LMagic*. Accessed: May 10, 2021. [Online]. Available: <https://statweb.stanford.edu/candes/software/lmagic/#code>
- [34] G.-B. Huang, L. Chen, and C.-K. Siew, "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *IEEE Trans. Neural Netw.*, vol. 17, no. 4, pp. 879–892, Jul. 2006, doi: [10.1109/TNN.2006.875977](https://doi.org/10.1109/TNN.2006.875977).
- [35] G.-B. Huang, "An insight into extreme learning machines: Random neurons, random features and kernels," *Cognit. Comput.*, vol. 6, no. 3, pp. 376–390, Sep. 2014, doi: [10.1007/s12559-014-9255-2](https://doi.org/10.1007/s12559-014-9255-2).
- [36] N. C. Thompson, K. Greenewald, K. Lee, and G. F. Manso, "The computational limits of deep learning," 2020, *arXiv:2007.05558*.



TAREK BERGHOUT received the M.Sc. and Ph.D. degrees in industrial informatics and manufacturing from the University of Batna 2, Algeria, in 2015 and 2021, respectively.

His research interests include condition monitoring of industrial processes using machine learning tools while his main interests include artificial neural networks, deep learning least squares methods, and extreme learning machine.



MOHAMED BENBOUZID (Fellow, IEEE) received the B.Sc. degree in electrical engineering from the University of Batna, Batna, Algeria, in 1990, the M.Sc. and Ph.D. degrees in electrical and computer engineering from the National Polytechnic Institute of Grenoble, Grenoble, France, in 1991 and 1994, respectively, and the Habilitation à Diriger des Recherches degree from the University of Picardie Jules Verne, Amiens, France, in 2000.

After receiving the Ph.D. degree, he joined the Professional Institute of Amiens, University of Picardie Jules Verne, where he was an Associate Professor of electrical and computer engineering. Since September 2004, he has been with the University of Brest, Brest, France, where he is currently a Full Professor of electrical engineering. He is also a Distinguished Professor and a 1000 Talent Expert at Shanghai Maritime University, Shanghai, China. His main research interests and experience include analysis, design, and control of electric machines, variable-speed drives for traction, propulsion, and renewable energy applications, and fault diagnosis of electric machines.

Prof. Benbouzid is a fellow of IET. He is the Editor-in-Chief of the *International Journal on Energy Conversion* and the *Applied Sciences* (MDPI) Section on Electrical, Electronics and Communications Engineering. He is a Subject Editor for the *IET Renewable Power Generation*.



S. M. MUYEEN (Senior Member, IEEE) received the B.Sc. (Eng.) degree in electrical and electronic engineering from the Rajshahi University of Engineering and Technology (RUET), Bangladesh, formerly known as the Rajshahi Institute of Technology, in 2000, and the M.Eng. and Ph.D. degrees in electrical and electronic engineering from the Kitami Institute of Technology, Japan, in 2005 and 2008, respectively.

He is currently working as a Full Professor with the Electrical Engineering Department, Qatar University. He has published more than 250 articles in different journals and international conferences. He has published seven books as an author or an editor. His research interests include power system stability and control, electrical machine, FACTS, energy storage systems (ESS), renewable energy, and HVDC systems. He has been a keynote speaker and an invited speaker at many international conferences, workshops, and universities. He is a fellow of Engineers Australia. He is serving as an Editor/Associate Editor for many prestigious journals from IEEE, IET, and other publishers, including IEEE TRANSACTIONS ON ENERGY CONVERSION, IEEE POWER ENGINEERING LETTERS, *IET Renewable Power Generation*, and *IET Generation, Transmission and Distribution*.



LEILA-HAYET MOUSS received the B.Sc. degree in electrical engineering from the National Polytechnic School of Algiers, Algeria, in 1979, and the M.Sc. and Ph.D. degrees in electrical and computer engineering from the University of Bordeaux—ENSERB, Bordeaux, France, in 1982 and 1985, respectively.

After graduation, she joined the University of Batna 2, Algeria, where she is currently a Full Professor of electrical and computer engineering. She is the Head of the Automatic and Computer Integrated Manufacturing Laboratory, University of Batna 2. Her current research interests include industrial diagnosis of production system using artificial intelligence techniques.

Prof. Mouss is a member of the New York Science Academy.

• • •

TOUFIK BENTRCIA received the D.Sc. degree in industrial engineering from the University of Batna 2, Algeria, in 2017. He is currently a Researcher within the LTM laboratory, University of Batna 2. He has coauthored many scientific papers. His main interests include fractional modeling, numerical simulation, and industrial applications. He served as a reviewer for numerous journals and conferences.