Asymmetric HMMs for online ball-bearing health assessments

Carlos Puerto-Santana, *Aingura IIoT and Universidad Politécnica de Madrid,*
Concha Bielza, *Universidad Politécnica de Madrid*, Javier Diaz-Rozo, *Aingura IIoT*,
Guillem Ramirez-Gargallo, *Barcelona Supercomputing Center*,
Filippo Mantovani, *Barcelona Supercomputing Center*, Gaizka Virumbrales, *Aingura IIoT*,
Jesús Labarta, *Barcelona Supercomputing Center*, and Pedro Larrañaga ,*Universidad Politécnica de Madrid*

Abstract—The degradation of critical components inside large industrial assets, such as ball-bearings, has a negative impact on production facilities, reducing the availability of assets due to an unexpectedly high failure rate. Machine learning-based monitoring systems can estimate the remaining useful life (RUL) of ball-bearings, reducing the downtime by early failure detection. However, traditional approaches for predictive systems require run-to-failure (RTF) data as training data, which in real scenarios can be scarce and expensive to obtain as the expected useful life could be measured in years. Therefore, to overcome the need of RTF, we propose a new methodology based on online novelty detection and asymmetrical hidden Markov models (As-HMM) to work out the health assessment. This new methodology does not require previous RTF data and can adapt to natural degradation of mechanical components over time in data-stream and online environments. As the system is designed to work online within the electrical cabinet of machines it has to be deployed using embedded electronics. Therefore, a performance analysis of As-HMM is presented to detect the strengths and critical points of the algorithm. To validate our approach, we use real life ball-bearing data-sets and compare our methodology with other methodologies where no RTF data is needed and check the advantages in RUL prediction and health monitoring. As a result, we showcase a complete end-to-end solution from the sensor to actionable insights regarding RUL estimation towards maintenance application in real industrial environments.

Index Terms—Machine Learning, Hidden Markov model, novelty detection, concept drift, health index, remaining useful life, predictive maintenance, industrial Internet of Things (IIoT).

I. INTRODUCTION

NOWADAYS, the successful application of machine learning to solve specific challenges in the industry has been enabled [1]. The goal of a manufacturing system is to stay in an 'in-control' state, avoiding the production of defective items due to failures or aging of the system itself [2]. As a result, the machine design process is aimed at maximizing reliability and availability, reducing failures, and increasing useful life.

Mechanical components suffer natural degradation as the operating conditions, along with aging, shape their behavior. Within industrial environments, ball-bearings are components which can be found in almost all rotating machinery, and they are commonly associated with premature failures. An unexpected failure of these components can be

C. Puerto-Santana, Aingura IIoT and Universidad Politécnica de Madrid, Spain
E-mail: ce.puerto@alumnos.upm.es

critical enough to stop a production system completely [3]. Depending on how critical they are within the mechanical design, a ball-bearing could have a useful life span measured in years. Some ball-bearings are designed to have an infinite useful life under specific operating conditions. Infinite means that, theoretically, the ball-bearing will never fail when working under certain conditions. However, if the operating conditions change during specific situations, it might affect the ball-bearing RUL [4].

In this scenario, the role of predictive capabilities is decisive to avoid unexpected failures. Intelligent systems that can estimate the remaining useful life of critical systems or components, such as ball-bearings without extensive and expensive training procedures, are among the essential technologies that enable smart factories [5]. Therefore, a machine learning-based ball-bearing health assessment system has the following specific challenge to overcome: to detect and adapt to low degradation rates, and to be applicable and robust for machines with extensive lifespans. Although such asset properties are desirable for a machine, it is a more challenging issue for an intelligent system.

Predictive machine learning algorithms need to deal with highly unbalanced data, as there are not enough failures to train predictive models to have a positive effect on their performance. Model accuracy is reduced as ball-bearings have different failure modes, increasing the difficulty of training these models before deployment. Additionally, ball-bearing degradation negatively affects the algorithm performance, reducing its fitting to the data stream [6].

Novelty detection refers to identifying new patterns, which were not observed in the training phrase, in testing data. The identification of such patterns is pertinent for a model or artificial intelligence performance in an online environment, where learned trends may become obsolete [7]. Hence, novelty detection methodologies have to be used to adapt the health assessment model to new trends that may appear during the monitoring of ball-bearings. However, in real-world industrial environments, data coming from ball-bearings are defined as a dynamic data stream that have to be acquired, pre-processed, and processed at a given sampling rate. Furthermore, the model used must run its learning, prediction, and novelty detection algorithms as quickly as data arrives to detect possible abnormal behavior. We define these types of methodologies in data streams as online analysis.

In this article, a novelty detection online methodol-

ogy based on asymmetric hidden Markov models (As-HMMs) [8] is introduced (online As-HMM), which determines the health of the ball-bearings and their remaining useful life (RUL). The proposed methodology assumes that no previous run-to-failure (RTF) data is available, which makes it suitable for cases where obtaining RTF data is not possible. On the other hand, As-HMMs are convenient to identify different trends in data, which is useful when non-stationary data is being modeled. Ball-bearing wear signal is a non-stationary process [9]; consequently, the use of As-HMMs is justified. Also, As-HMM can detect and estimate precisely when a novel trend appears in data as long as the model learns those changes from training data. Coupling As-HMMs with novelty detection methodologies generates a model capable of learning new trends as data arrives, and detecting the time when such trends change. Additionally, As-HMMs can minimize the number of parameters: a set of relationships among variables for each detected trend is defined instead of considering a full or Naïve dependency hypothesis like other HMMs do [10].

For the online analysis in this article, the Page sequential test and Chernoff-bounds are applied as a concept drift detection methodology as in [6]. Also, a novel health index (HI) is proposed for ball-bearing monitoring. Ball-bearing signals are used to validate the proposed methodology and compare it to other ball-bearing diagnosis and prognosis techniques.

As the scope of this article is to showcase an online monitoring pipeline, its deployment into embedded electronics that enable its application is analyzed. Therefore, the computational cost and the performance of the model implementation are taken into account in order to make novelty detection techniques matching the real-time constraints of real industrial environments. Our study includes an in-depth performance analysis of the proposed As-HMM methodology from a computational point of view. The code optimization of the proposed online As-HMM model allows us to deploy our solution on computational devices close to the sensors, and process their data streams in real-time, a process of this type is defined as Edge computing. The proposed methodology, based on As-HMM, has the following advantages:

- It has a novelty detection approach.
- It does not require previous ball-bearing RTF data to estimate the HI and RUL.
- It is capable of adapting itself when new data appears and learning dynamic probabilistic relationships among variables and their past values.
- It can be used in online data-stream environments inside embedded systems such as Edge computing devices.

The content of the paper is structured as follows. In Section II, the relevant contributions in the field of degradation and RUL prediction are reviewed. Section III summarizes the methodology used for our study, including details on the data processing and the model. Section IV explains the datasets used for validation and the experimental setup. In Section V, the results of the experiments combined with the prediction of the model are reported. The article closes with remarks and conclusions in Section VI.

II. RELATED WORK

In this section, we summarize current methodologies concerning the diagnosis and prognosis of mechanical components. We divide the state-of-the-art methodologies into three types: (A) requires RTF data, (B) requires RTF times, and (C) neither requires RTF data nor RTF times. The last type are more robust and ideal for many real and industrial applications since the failure data may be non-existing, limited (such as unbalanced data), or expensive in time and money to obtain.

In Table. I a list of the reviewed literature with a short description is presented. The literature is compared depending on:
(1.) Does it use HMM?
(2.) Can it be used in online environments?
(3.) Does it not require RTF data?
(4.) Can it estimate the tool RUL?
(5.) Can it update itself when new trends in data appear?

A. Requires RTF data

Concerning HMMs, we can find the following methodologies: [11] proposed an adaptive modified HMM (MHMM) for online tool wear monitoring (TWM). The authors used the forward-backward algorithm and the Hotteling multivariate control chart to detect changes in bearing wear. An online learning algorithm based on the EM algorithm was proposed to update the MHMM. [12] proposed an online mixure of Gaussians HMM (online MoG-HMM) for bearings prognosis and RUL prediction. Using the Viterbi algorithm, the distribution of the RUL for each hidden state in RTF data was learned. In the online phase, the learned model and RUL distribution was used to predict the RUL. Later, [13] used HMMs and neuro-fuzzy algorithms (NF) to predict future bearing conditions. First, the data were clustered to determine different levels of degradation. For each learned cluster, an HMM was learned; when new instances arrived, the most likely HMM was used with a neuro-fuzzy algorithm to predict the future wear of bearings.

[14] proposed a hidden semi-Markov model (HSMM) for the RUL prediction of bearings. A wrapper algorithm based on the Akaike information criterion was used to determine the best model to predict from a set of models where the time duration distribution, number of components, and number of hidden states change. More recently, [15] proposed an HMM to predict RUL for TWM. The HMM parameters were computed using a physical ball-bearing degradation model, whereas a multilayer perceptron was used to estimate the emission probabilities. The prognosis phase, or RUL prediction, was performed with a modified forward variable. Finally, [16] used a MoG-HMMs to determine the RUL of bearings. An initial HMM based on the first observations from all the signals was trained. The log-likelihoods were computed for further observations with a current HMM until the log-likelihood surpassed a certain threshold; then, another HMM was trained from the not-well fitted data. Then, RUL is predicted using a polynomial regression function of the log-likelihoods of the learned HMMs.

All the approaches presented here vary their way to predict the degradation or RUL and learn the HMM model.

Paper	Short Description	(1.)	(2.)	(3.)	(4.)	(5.)
MHMM [11]	Modified HMM for TWM	✓	✓	-	-	✓
Online MoG-HMM[12]	Learns RUL distributions with MoG-HMM	✓	✓	-	✓	-
HMM and NF[13]	Predicts degradation with Neuro-Fuzzy	✓	-	-	-	-
HSMM[14]	Uses HSMM to predict RUL	✓	-	-	✓	-
HMM for TWM [15]	Uses HMM to predict RUL	✓	-	-	✓	-
HMM ensemble[16]	HMM ensemble to predict RUL	✓	✓	-	✓	-
Neo Fuzzy[17]	Predicts RMS with Neuro-Fuzzy	-	-	-	-	-
FDFDA [18]	FD analysis and regression updating	-	✓	-	-	✓
ANN for TWM[19]	Convolutional ANN for wear classification	-	-	-	-	-
RNN with HI[20]	RNN for HI and RUL estimation	-	-	-	✓	-
Fault effects [21]	Uses fault effects to predict RUL	-	✓	-	✓	-
LSTM-SVM [22]	LSTM-SVM for RUL prediction	-	✓	-	✓	-
LSTM with PF [23]	Uses LSTM networks and PF to predict RUL	-	✓	-	✓	-
BDNN-RF[24]	BDNN and RF for ball-bearing fault prediction	-	✓	-	-	-
Regression[25]	Regression models for RUL estimation	-	✓	-	✓	✓
EKM for TWM[26]	EKF and regression to predict RUL	-	✓	-	✓	✓
WPD-HMM[27]	Log-likelihood of HMM as HI	✓	✓	✓	-	-
AHMM[28]	Adaptive HMM for TWM	✓	✓	✓	-	✓
Trigger regression [29]	Triggered regression for RUL estimation	-	✓	✓	✓	✓
APCMD[30]	Local and global regressions to predict RUL	-	✓	✓	✓	✓
HSIC [31]	Changes in dependencies as degradation	-	✓	✓	-	✓
Random Forest[32]	Uses random forest to detect anomalies	-	✓	✓	-	-
Genetic HMMs [33]	Learns HMMs with genetic algorithms	✓	✓	✓	-	-
AMBi-GAN[34]	Uses GAN to detect anomalies	-	✓	✓	-	✓
This article	Online AS-HMM for RUL prediction	✓	✓	✓	✓	✓

TABLE I: State-of-the-art models for prognosis and diagnosis of dynamical systems

Nevertheless, the need for previous RTF data limits the application of these models and in some cases it is critical for the learning phase as in [12], where the distribution of the RUL must be computed for each failure mode and hidden state, making it unfeasible in certain real industrial scenarios. It is also relevant to mention that the prognosis in these methods is usually driven by the Viterbi algorithm and the forward-backward algorithm, with the latter used as well to compute the log-likelihood from data.

Other methodologies not based on HMMs with the hypotheses of RTF data can be found in the state-of-the-art: [17] proposed a long prediction scheme for bearing condition prediction using a model-based on fuzzy logic and a single neuron artificial neuronal network (ANN) called neo-fuzzy neuron. Later, [18] proposed an online surveillance process to detect a failure in a dynamical process called FDFDA. A fisher discriminant (FD) analysis was performed to extract relevant components from the data. A ratio of stability factors of faulty data to normal data was computed for each component. Components that went over a certain threshold were eliminated from the analysis and used in an autoregressive regression model to predict the degradation process. Next, [19] proposed two 1-dimensional convolution ANNs to determine bearing degradation. One convolutional neural network was used to detect outer ring failures and another to detect inner ring failures. The online testing phase consisted of passing the vibrational signals through both models and classify every instance as healthy or faulty. [20] used a recurrent neural network (RNN) to predict the RUL of bearings. Time and frequency features were extracted to feed the RNN, and from its output, a linear regression was used to build a health index and predict the bearing RUL. [22] used a log short time memory network (LSTM) and a support vector regression for RUL prediction of aero-engine data. A risk-averse function was optimized to minimize RUL overestimation. Later, [21] proposed a methodology for RUL prediction of turbofan engines. For each feature a Wiener process was learned and a RUL distribution was estimated using a Bayesian approach considering fault effects. A copula was used to generate a joint RUL distribution. [23] used a LSTM network to extract features to determine a normalized health index that predicted the RUL for rollers in hot rolling production. For the RUL estimation a state space model was used with a particle filtering (PF) algorithm for inference and prediction. Whenever a new instance arrived, the state space model could self-update using all the historic data. In spite of that, the LSTM network had to be trained with run to failure data. Finally, [24] used binary deep neural network (BDNN) with random forest (RF) to classify ball-bearing faults. The BDNN-RF were designed to be used in edge devices to enable them for online analysis. However, the training phase was performed in a cloud device.

In these non-HMM-based techniques, it was observed that a relevant portion of them used ANN to classify the level of degradation; then, a health index was used to predict the degradation level, and regression is later used to perform the RUL or degradation prediction. Although the ANNs in the previous works were not so computationally demanding, the requirement of previous RTF data can make such methodologies unfeasible in many industrial scenarios.

B. Requires RTF times

Sometimes, previous RTF data is not available; however, the records of RTF times are available and can be used for prognosis purposes. However, not many works under these circumstances were found. For instance, [25] proposed two bearing online prognosis techniques based on exponential and linear regression. The methodology assumed that there was at least prior knowledge of RTF times; from these times, the parameters of a prior Bernstein distribution were estimated using the maximum likelihood method. A threshold from the standard normative was imposed in the

magnitude of the ball-bearing signals to indicate the start of the failure phase. Later, the residual life distribution was computed, and the RUL was extracted by computing the median of the resultant distribution. Later, [26] introduced a bearing online RUL prediction algorithm based on extended Kalman filters (EKF). Whenever a new instance arrived, relevant features were computed and the EKF were updated and used to extrapolate the current signal up to a failure threshold. For determining the threshold, previous RTF times were required.

In these methodologies, the run to failure times were used to compute the RUL distribution or indicate the prognosis failure thresholds. However, the prognosis phases were done using regression techniques. It must be recalled that these methodologies can be more flexible than those described in the previous subsection since if RTF data are available, the RTF times are also available; nevertheless, the opposite is not valid.

C. Does not require RTF data nor RTF times

First, some techniques found related to HMMs are summarized. For instance, [27] proposed an online algorithm to determine the level of degradation of bearings based on the wavelet packet decomposition (WPD) and HMMs. The methodology extracted its features applying the WPD to the bearing vibrational signals. Then, from the early stages of the bearing life, the signal measures were used to train an HMM. Next, the likelihood from incoming vibrational signals was computed. If the likelihood went below a certain threshold, the bearing behaviour was considered abnormal. Later, [28] used an online adaptive HMM (AHMM) to determine machine tool wearing. The author used a progressive learning algorithm and a split and merge operation for components in hidden states to update the model whenever a new instance arrived. A Hotteling multivariate control chart was used to detect outliers, and a threshold (set by trial and error) for consecutive out-of-chart instances was imposed to determine when a new hidden state had to be added. Finally, a health index based on the Cauchy-Schwarz correlation between the current HMM and the previous learned HMM was proposed as a flag for degradation.

For non-HMM-based techniques, much more developed work can be found, especially related to regression techniques and novelty detection in data flows. For instance, [29] proposed a regression method for RUL prediction. The authors applied an adaptive approach based on three standard deviation intervals to determine a time trigger to launch an exponential regression model for RUL prediction. Next, [30] proposed an online prognosis technique for computing ball-bearing RUL called APCMD. A principal component Mahalanobis distance algorithm was used to reduce the feature dimensionality and build a health index. The RUL prediction was divided into two phases; the first one built local exponential regressions to represent the local degradation trajectory; in the second one, an empirical Bayesian algorithm was built to predict a global RUL. In [31] a dependence analysis was used to detect bending and deformations in steel structures. In their model, the authors

computed, for each pair of captured features, the normalized Hilbert-Schmidt independence criterion (HSIC) to measure the level of dependency among them. From these measures, a dependency graph was computed. Changes in the health state of the steel structure were deduced from significant changes in the dependency graph.

More recently, [32] used random forest and data compression to detect abnormal behavior in industrial components. An abnormal bounded score was built from the random forest to determine the level of abnormality of incoming data. In the case of [33], the authors used HMMs to detect abnormal signal behavior in several industrial components. The authors used genetic algorithms to learn the HMM parameters from random sub-sequences of sequences of normal behavior data. The last offspring generation of HMMs were used to compute the fitness of incoming data. Abnormal behavior was detected observing changes in log-likelihood fitness. [34] proposed an ANN approach to detect anomalies based on generative adversarial network (GAN)[1] and bi-directional LSTM networks (AMBi-GAN). The model used the residuals of both networks to build a global score to detect anomalies and update the model. As a final detail, the property of the model of being LSTM enabled them to work and process faster than other ANNs e.g., RNNs.

It is worth mentioning that when no-run-to failure data is assumed, fewer works using ANN are observed (at least in our research) and more attention is given to detect and measure changes in distribution or parameters within a data stream (novelty detection and concept drifts). Additionally, in the reviewed articles, regression was the key for prognosis when no run to failure data is provided. Also, the signal processing phase plays a vital role in such methodologies, since it is needed to extract the relevant information to detect the before-mentioned changes in data distribution.

III. PROPOSED METHODOLOGY

Fig. 1 shows a flow diagram indicating the proposed methodology workflow. In short, the methodology consists of repeating the following steps as required:

1. Capture new data and process it using signal processing algorithms.
2. Compute the processed data fitness with the current As-HMM.
3. Use the Page test and Chernoff bounds to detect new trends in data.
4. Update the model and add new hidden states to the As-HMM if needed.
5. Use the Viterbi algorithm with the current As-HMM and compute the HI.
6. Use the HI historic to estimate the RUL.

Observe that in the diagram in Fig. 1 there are three shaded boxes, namely *Novelty detection*, *Model update* and *RUL prediction*. The *Novelty detection* box includes the algorithms of BIC computation, the Page test (Section III-C1) and

---

[1]Adversarial networks consist of two ANNs, called discriminator and generator. The idea of the generator ANN is to maximize the classification error of the discriminator ANN, whereas the generator ANN learns by minimizing the classification error obtained from the ouput generated by the generator ANN.
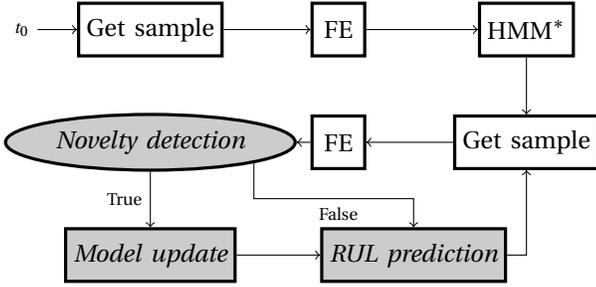
Fig. 1: Flow diagram of the novelty detection and RUL prediction strategy. FE stands for feature extraction. HMM* stands for training a new HMM. *Novelty detection* looks for new trends in data. *Model update* updates the model and the baseline health. *RUL prediction* computes the proposed health index and estimates the ball-bearing RUL

the Chernoff bounds (Section III-C2). The *Model update*, it consists of the model online training that is further explained in Section III-B. With respect to *RUL prediction*, a health index is proposed (Section III-D1) and a linear regression is used to estimate the RUL (Section III-D2). For a good performance of the proposed methodology, we make the following assumptions:

1. The sensors are sensitive enough to capture the ball-bearing degradation process to failure and not only the bearing failure phase.
2. No health recoveries are observed by the sensors.
3. No control affects the dynamical behavior during the data acquisition.

The first assumption refers to data quality. The Second assumption, as can be seen below, ensures a proper RUL prediction, and the third assumption prevents the controlled behaviors from being considered as degradation.

A. Data processing and feature extraction

When an acceleration sensor is put on ball-bearing support, the measured signal usually includes the noise of the complete system (i.e., fans, electrical noise, other nearby machines, etc.). The noise can be problematic for any machine learning algorithm; therefore, signal processing tools are needed to extract relevant features from raw data.
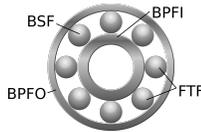
Fig. 2: Fundamental frequencies of the ball-bearings are related to the ball-bearing components

In this article, we use the frequency features of the ball-bearings. We extract the fundamental frequencies of the ball-bearings corresponding to the components of the ball-bearing (see Fig. 2), namely: ball pass frequency outer (BPFO, related to the outer ring of the ball bearings), ball pass frequency inner (BPFI, related to the inner ring of the ball-bearings), ball spin frequency (BSF, related to the rollers of the ball-bearings), and fundamental train frequency (FTF,

related to the cage of the ball-bearings). More details on this technique are presented in [35] and [36].

B. Hidden Markov models and model update

Hidden Markov models [37] are used to describe dynamic data. They are traditionally used for speech recognition problems or for analysing sequences of biological proteins. However, more recently HMMs have been used for machine-tools and energy surveillance. The flexibility of the model assumptions allows energy and machine-tool dynamical processes to be predicted and segmented.

The model can be seen as a double stochastic sequence. One of the sequences is hidden, $\boldsymbol{Q}^{1:T} = (Q^1, ..., Q^T)$, and the other is observable, $\boldsymbol{X}^{1:T} = (\boldsymbol{X}^1, ..., \boldsymbol{X}^T)$. $Q^t \in \{1, 2, ..., N\}$, $\boldsymbol{X}^t \in \mathbb{R}^M$ and M indicates the number of features. The probabilistic relationships among these two sequences can be described with a parameter $\boldsymbol{\lambda} = (\mathbf{A}, \boldsymbol{\pi}, \mathbf{B})$. $\mathbf{A} = [a_{ij}]_{i,j=1}^N$ is a matrix representing the transition probabilities among the hidden states i, j over time, i.e., $a_{ij} = P(Q^{t+1} = j | Q^t = i, \boldsymbol{\lambda})$; $\mathbf{B}$ is a vector representing the emission probabilities of the observations given the hidden state. $\mathbf{B} = [b_i(\mathbf{x}^t)]_{i=1}^N$, where $b_i(\boldsymbol{x}^t) = P(\boldsymbol{X}^t = \boldsymbol{x}^t | Q^t = i, \boldsymbol{\lambda})$ is a probability density function; $\boldsymbol{\pi}$ is the initial probability distribution of the hidden states, $\boldsymbol{\pi} = [\pi_j]_{j=1}^N$, where $\pi_j = P(Q^0 = j | \boldsymbol{\lambda})$.

In this article we use an As-HMM with continuous variables with auto-regressive factors (AR-AsLG-HMM) [10]. In this approach, a context-specific linear Gaussian Bayesian network [38] is used to model emission probabilities, where the context in this case is given by the hidden variable. Context-specific Bayesian networks can be useful for non-stationary data since they can provide a different network whenever an important change in the statistical parameters or relationships among variables arise. Each variable X_m^t in each context-specific Bayesian network has a, possibly different, set of k_{im} parents $\mathbf{Pa}_i(X_m^t) = (U_{im1}^t, ..., U_{imk_{im}}^t)$ in the associated graph and p_{im} auto-regressive variables $\mathbf{D}_i(X_m^t) = (X_m^{t-1}, ..., X_m^{t-p_{im}})$ which explain the variable X_m, $m = 1, ..., M$ and $i = 1, ..., N$. Then, the emission probabilities can be written as:

$$
\begin{aligned}
b_i(\boldsymbol{x}^t) &= P(\boldsymbol{x}^t | \boldsymbol{x}^{t-1:t-p^*}, Q^t = i, \boldsymbol{\lambda}) \\
&= \prod_{m=1}^M P(x_m^t | Q^t = i, \mathbf{Pa}_i(X_m^t), \mathbf{D}_i(X_m^t), \boldsymbol{\lambda}) \\
&= \prod_{m=1}^M \mathcal{N}(x_m^t | \boldsymbol{\beta}_{im} \cdot \mathbf{pa}_{im}^t + \boldsymbol{\eta}_{im} \cdot \mathbf{d}_{im}^t, \sigma_{im}^2), \quad (1) \\
&= \prod_{m=1}^M \frac{1}{\sqrt{2\pi\sigma_{im}^2}} e^{-\frac{(x_m^t - (\boldsymbol{\beta}_{im} \cdot \mathbf{pa}_{im}^t + \boldsymbol{\eta}_{im} \cdot \mathbf{d}_{im}^t))^2}{2\sigma_{im}^2}}
\end{aligned}
$$

where $\mathcal{N}$ denotes the normal probability density function, $\boldsymbol{\beta}_{im} = (\beta_{im0}, ..., \beta_{imk_{im}})$, $\mathbf{pa}_{im}^t = (1, u_{im1}^t, ..., u_{imk_{im}}^t)$, $\boldsymbol{\eta}_{im} = (\eta_{im1}, ..., \eta_{imp_{im}})$, $\mathbf{d}_{im}^t = (x_m^{t-1}, ..., x_m^{t-p_{im}})$ and p^* is a maximum lag bound such that $p_{im} \leq p^*$. Observe in particular that: $f(x_m^t) = \boldsymbol{\beta}_{im} \cdot \mathbf{pa}_{im}^t + \boldsymbol{\eta}_{im} \cdot \mathbf{d}_{im}^t$ is a mean that changes over time. Also, note that these emission probabilities defined as factorized Gaussian densities encoding a Bayesian network, allow to model conditional (in)dependencies among features and to show them explicitly.

As this algorithm is expected to be used in online environments, other emission probabilities such as mixtures or a multivariate normal Gaussian can compromise the model and data acquisition integrity since they are slower to compute due to the use of unnecessary parameters [10]. In [10] it is detailed how to estimate the parameters $\boldsymbol{\beta}_{im}$, $\boldsymbol{\eta}_{im}$ and σ_{im}^2 by the EM algorithm [39] for batch data or offline environments. Additionally, the selection of the context-specific Bayesian networks with the structural EM algorithm (SEM) [40] and other relevant issues such as the forward-backward algorithm and the Viterbi algorithm [37] are also explained. In particular, the networks are selected by their fitness computed via the Bayesian information criterion (BIC) score; whereas the search method is based on a greedy-forward explore scheme which verifies that the networks are directed acyclic graphs (DAGs).

As stated in Section II, the proposed methodology uses an As-HMM which evolves when a concept drift is detected. For the initial HMM, we train the model as explained in [10], with only one hidden state. However, for online data, the learning process must be different, and the SEM batch algorithm stated in [10] is manipulated to fit the proposed requirements. Additionally, the number of hidden states is not fixed a priori and can change with the data.

Assume that the novelty detection methodology detects an unobserved concept drift at time $t+L$ with $L \in \mathbb{N}$ and the current model is no longer valid to explain $\boldsymbol{x}^{t:t+L}$. In such cases, we propose the addition of a new hidden state to the current HMM. Suppose that the current model has parameter $\boldsymbol{\lambda}^0 = \{\boldsymbol{A}^0, \boldsymbol{B}^0, \boldsymbol{\pi}^0\}$ with $N \in \mathbb{N}$ hidden states. We generate a new prior model with parameter $\boldsymbol{\lambda}^1 = \{\boldsymbol{A}^1, \boldsymbol{B}^1, \boldsymbol{\pi}^1\}$ with $N+1$ hidden states. The parameter $\boldsymbol{\pi}^1$ is built as:

$$\boldsymbol{\pi}^1 = [\boldsymbol{\pi}^0 | 0] \qquad (2)$$

or append a 0 at the right of the vector $\boldsymbol{\pi}^0$. For the prior of $\boldsymbol{A}^1$, we build the following matrix $\boldsymbol{C} = [c_{ij}]_{i,j=1}^{N+1}$:

$$\boldsymbol{C} = \begin{bmatrix} & & & & 0 \\ & & & & 0 \\ & \boldsymbol{A}^0 & & & \vdots \\ & & & & 0 \\ & & & & 0.01 \\ \hline \frac{1}{N+1} & \frac{1}{N+1} & \cdots & & \frac{1}{N+1} \end{bmatrix} \qquad (3)$$

The value $c_{N,N+1} = 0.01$ is chosen for the EM to be aware that there is a non-null probability of transition among the current known hidden states to the latest discovered hidden state. The prior transition matrix $\boldsymbol{A}^1 = [a_{ij}^1]_{i,j=1}^{N+1}$ for the new model is $a_{ij}^1 = c_{ij}/\sum_{j=1}^{N+1} c_{ij}$ and $\boldsymbol{B}^1 = [\boldsymbol{B}^0 | b_{N+1}(\boldsymbol{x}^t)]$. It is assumed a priori that $b_{N+1}(\boldsymbol{x}^t)$ is represented by a naïve-Bayes graph with $p_{(N+1)m} = 0$ for $m = 1, ..., M$. The prior parameters of the corresponding new linear Gaussian Bayesian network are set as follows:

$$\beta_{(N+1)m0} = \frac{1}{L} \sum_{j=t}^{t+L} x_m^j$$

$$\sigma_{(N+1)m}^2 = |\max_{j=t,...,t+L} \{x_m^j\} - \min_{j=t,...,t+L} \{x_m^j\}| \qquad (4)$$

During the learning optimization phase of $\boldsymbol{\lambda}^1$, we restrict the optimization exposed in [10] to the parameters

$\{\boldsymbol{\beta}_{(n+1)m}\}_{m=1}^M$, $\{a_{(N+1)j}\}_{j=1}^{N+1}$ and $\{a_{j(N+1)}\}_{j=1}^{N+1}$. In this manner, we save the information obtained in previous hidden states and update the model to explain the novel incoming data.

Although a renovation of the model is performed with the new data, the newly learned parameter $\boldsymbol{\lambda}'$ must be better than the current parameter $\boldsymbol{\lambda}^0$. Therefore, a condition in the BIC score for a new model to be valid is imposed, i.e., if $\boldsymbol{x}^{t:t+L}$ is the current data window, then the new model or parameter is accepted if $\text{BIC}(\boldsymbol{x}^{t:t+L}|\boldsymbol{\lambda}') < \text{BIC}(\boldsymbol{x}^{t:t+L}|\boldsymbol{\lambda})$.

Finally, whenever a new hidden state is added, the base health indexes need to be updated, but that will be explained in Section III-D1.

C. Novelty detection

In Table. II the list of hyper-parameters used by this algorithm are mentioned. In the following subsections, they are detailed and it is explained how to tune and interpret them.

Parameter	Set Value	Description
L	128	Length of the processing window
ΔL	10	Window slice
Φ	3	Maximum ratio of BIC difference
γ_1	$128\ln(3)$	Threshold in Page test to detect outliers
ϵ	1×10^{-2}	Maximum error of population percentage
p	0.1	Maximum enabled population of outliers
γ_2	0.05	Reliability of the population estimation
ζ	-2.5	Threshold for RUL prediction

TABLE II: List of hyper-parameters used by the algorithm

1) Anomaly detection: We define a window size $L \in \mathbb{N}$, which receives new instances data and slides $\triangle L \in \mathbb{N}$ data. For the anomaly detection procedure, we use the Page sequential detection scheme [41]. Suppose that an As-HMM with parameter $\boldsymbol{\lambda}$ has been learned at time t. Let $\text{BIC}(\boldsymbol{x}^{t:t+L}|\boldsymbol{\lambda})$ stand for the Bayesian information criterion and S_l be:

$$S_l = \frac{1}{l} \sum_{j=1}^{l} \text{BIC}(\boldsymbol{x}^{t+j\triangle L:t+j\triangle L+L}|\boldsymbol{\lambda}), \qquad (5)$$

where l is the number of sliced windows up to current time and let $S_l^* = \min_{j=1,...,l}\{S_j\}$. Define a decision parameter $PH_l = S_l - S_l^*$ and compute:

$$r_l = \begin{cases} 1 & PH_l > \gamma_1 \\ 0 & PH_l \le \gamma_1 \end{cases} \qquad (6)$$

where $\gamma_1 > 0$ is a threshold that indicates the maximum permissible deterioration in BIC. Notice that the log-likelihood of the model could be used instead of the BIC for the Page test. Nevertheless, since the BIC is used during the training phase to determine the best set of context-specific Bayesian networks, this score is also used for anomaly detection for the sake of consistency. The hyper-parameter γ_1 is set as $\gamma_1 = L\ln(\Phi)$ where Φ can be seen as the maximum permissible quotient of likelihood per testing observation over the likelihood per training observation. Fig. 3 (a) shows the value of γ_1 for different combinations of L (axis X) and Φ (axis Y). The greater the value of γ_1, the less the algorithm will declare deviated data as abnormal; whereas a lower value of γ_1 implies that little deviations in data

will be considered as abnormal. Finding an equilibrium in this threshold is crucial for concept-drift detection. In Fig. 3 (a), it can be observed that after $\Phi > 1.8$, the change in L has more effect on the value of γ_1, but not extreme values are obtained. Therefore, in this study, $\Phi = 3$ is chosen as an intermediate point from the previous discussion. If $r_l = 1$, the data $x^{t+l\triangle L:t+l\triangle L+L}$ is considered an anomaly for the model. In this article, $\triangle L = 10$ and $L = 128$ are fixed from previous experiments. The fitness and training speed of the models were maximized when these parameters were acknowledged.

2) Novel concept drift detection criterion: The detection of a single anomaly is not enough to determine a novel trend in the data. Nevertheless, if many outliers are seen in a time window, the possibility of observing a novel trend is likely. Assume that the probability of observing anomalies in the last n^* BICs is p, and $\hat{p}_{n^*}$ is the estimation of such probability. To determine the length n^*, the sampling problem is used, i.e., given an error $\epsilon \in (0,1)$, a proportion $p \in (0,1)$ and a precision $\gamma_2 \in (0,1)$, n^* must be found such that:

$$P(|\hat{p}_{n^*} - p| < \epsilon) > 1 - \gamma_2. \tag{7}$$

Assuming that for the current model the normal level of anomalies is $\hat{p}_{n^*} \leq p$, using the Chernoff bounds (additive form), n^* must be at least:

$$n^* > \frac{-\ln(1-\gamma_2)}{D_{Be}(p-\epsilon|p)}. \tag{8}$$

In the previous equation, $D_{Be}(p-\epsilon|p)$ is the Kullback-Leibler divergence measure of two Bernoulli distributions with parameters $p-\epsilon$ and p. In Fig. 3 (b) the sensitivity of the bound n^* given by the Chernoff bounds for a precision level of $\gamma_2 = 0.05$ for different levels of ϵ and p is shown. As can be seen, the shorter the value of ϵ, the larger is n^* to ensure the bound. It is also noticeable that the curves are symmetric i.e., for a fixed ϵ and γ_2, the value n^* is the same for p and $1-p$. Since the goal is to detect novel trends as soon as possible, it is preferable to choose p closer to zero instead of one, and also greater values for ϵ. Therefore, in this article $p = 0.1$, $\epsilon = 10^{-2}$ and $\gamma_2 = 0.05$, in this manner a window size of $n^* = 87$ is obtained.

If R outliers are observed from the Page sequential test in a window of size n^*, $\hat{p}_{n^*} = R/n^*$ is computed. If $\hat{p}_{n^*} > p$, then the initial assumption is violated and a novel pattern is detected. Hence, an updating process of the HMM is performed.

D. RUL prediction

1) Proposed health index: It is plausible to think that the earlier the ball-bearing is measured and the lower the vibration amplitudes are, the healthier the ball-bearing is. Therefore, the first learned HMM from $x^{0:L}$ is taken as a healthy model, and its parameters are used as a base health index. The first As-HMM is trained with only one hidden state, the emission probability $b_1(x^t)$ can be transformed from a linear Gaussian Bayesian network into a multivariate normal distribution (see, for example, in [42]) with mean vector $\mu_1 = (\mu_{11}, ..., \mu_{1M})$ and covariance matrix
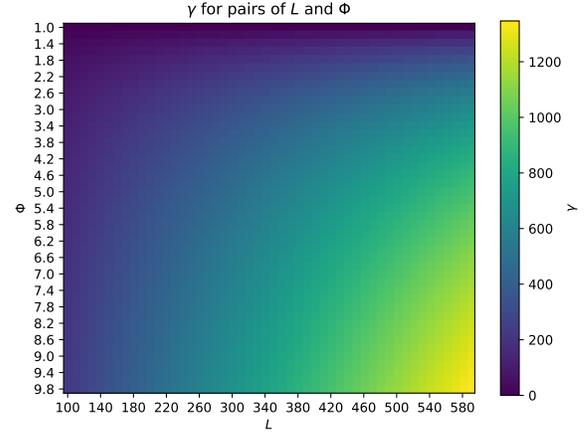
(a)
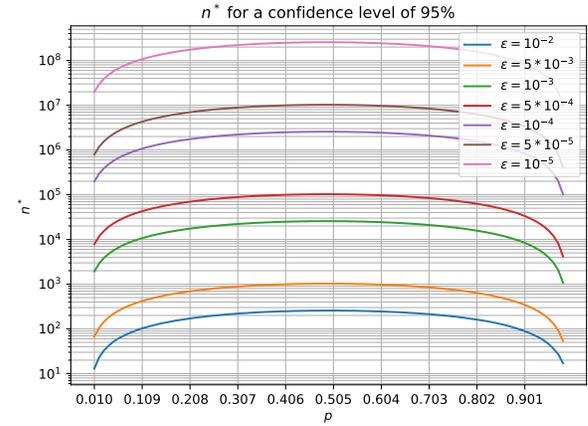
(b)

Fig. 3: (a) Sensitivity of γ_1, for different values of L and Φ. (b) Sensitivity of n^* for different values of p and ϵ for a fixed level of confidence of γ_2

$\Sigma_1 = \{\Sigma_{1jk}\}_{j,k=1}^{M}$. Define $\mu_{hm} := \mu_{1m}$ and $\Sigma_{hm} := \Sigma_{1mm}$ and compute the set of health indexes:

$$HI_m^\dagger(x^t) = \left\{ -\log\left(\frac{\mu_{q^t m}}{\mu_{hm}}\right), -\log\left(\frac{\Sigma_{q^t mm}}{\Sigma_{hm}}\right) \right\} \quad m = 1, ..., M. \tag{9}$$

Whenever a new hidden state is added (suppose it is the $N+1$ hidden state), the following assignment is performed:

$$\mu_{(N+1)m} \rightarrow \mu_{hm} \text{ if } \mu_{hm} > \mu_{(N+1)m}, \quad m = 1, ..., M$$
$$\Sigma_{(N+1)mm} \rightarrow \Sigma_{hm} \text{ if } \Sigma_{hm} > \Sigma_{(N+1)mm}, \quad m = 1, ..., M \tag{10}$$

The idea behind this health index assignment is that the higher the mean and variance of fundamental-frequencies-amplitudes, the more evident the ball-bearings wear. Also, a global health index is defined as:

$$HI^\dagger(x^t) = \min \bigcup_{m=1}^{M} HI_m^\dagger(x^t). \tag{11}$$

This global index takes the worst health index obtained among all the features.

Some final considerations: the latest q^t detected is saved in order to be used in further algorithms such as Viterbi

algorithm or forward-backward algorithm. The reason behind this is that during the traditional forward-backward algorithm or Viterbi algorithm, the parameter $\boldsymbol{\pi}$, which indicates the participation of the initial distribution, is always used. Nevertheless, since it is possible that further incoming data do not contain evidence of the first learned hidden state, the parameters $\{a_{q^t i}\}_{i=1}^N$ must be used as initial distribution for the algorithms. Finally, in order to obtain smoother health index curves, the health index historic pass through a moving average filter of order twenty.

2) RUL estimation: In this article, a time-dependent model is created to explain and predict the global health index. A regression model is used after the first concept drift is detected:

$$HI^{\dagger}(t) := a_0 + a_1 t + a_2 t^2 + w(t), \quad (12)$$

where $w(t)$ is a zero mean normal error term. The parameters of the regression models are estimated as in an ordinary least squares problem.

Once the regression has been estimated, RUL^t is determined as follows: find t_f such that $HI^{\dagger}(t_f) = \zeta$, where $0 > \zeta > -\infty$ is a maximum permissible magnitude order deviation from the good health estimation. In this article, $\zeta = -2.5$ or, in other words, a failure happens for this health index when the global process trend is 2.5 orders of magnitude away from the healthy trend. Nevertheless, depending on the mechanical application, the threshold may be changed. In applications where little deviations from the normal states imply a malfunction, the threshold must be closer to zero; otherwise, further away from zero. The RUL at time t is computed as:

$$RUL^t = t_f - t. \quad (13)$$

If $HI^{\dagger}(t_f) \neq \zeta$, $\forall t$, then $RUL^t = \infty$ and this can be interpreted as a non-degradation in the health of ball-bearings.

E. Theoretical computational complexity

In Table III theoretical bounds using big O notation for an iteration of the proposed algorithm are given. These bounds assume that the AR and context-specific Bayesian networks are dense. For the sake of space $K = p^* + M$. For the *Novelty detection* box, the complexity lies in the BIC computation, whereas the Page sequential test and Chernoff bounds have $O(1)$ complexity. The *Model update* box complexity relies principally on the SEM algorithm. Finally, for the *RUL regression* box, the complexity comes from the Viterbi algorithm and the $HI^{\dagger}$ computation.

The bound for the *Model update* box suggests that the algorithm is too complex to perform in online environments. Therefore, this box is further analyzed in Section V-C1 to analyze its behavior with real data.

Phase	Complexity
Novelty detection	$O(LN(NK + N^2))$
Model update	$O(M^2 K^2(K(L+K) + NL) + LN(N^2 + MK))$
RUL prediction	$O(T + N(MK + M^2 + L))$

TABLE III: Computational complexity of one iteration of the proposed online algorithm

IV. EXPERIMENTAL SET-UP

To evaluate and see the capabilities of the proposed methodology, two data-sets were used. The first one comes from the online FEMTO repository which has ball-bearing RTF data [43]. The second dataset comes from our own mechanical setup, where ball-bearings are run to failure. In this manner, we check the capabilities of the proposed method for common data (FEMTO) and specific daily data (ours).

The models WPD-HMM[27], AHMM [28] and APCMD [30], are used for comparison purposes in the case of the FEMTO dataset. AHMM is used for comparison since it can be considered as a simple application of HMM for online analysis where no novelty detection or model update is considered; WPD-HMM is an HMM where the model is updated whenever a new instance arrives. However, in none of these, there is a description or algorithm to compute the RUL. In particular, no HMM model with no RTF data assumption for RUL prediction was found. For that reason, APCMD is used to compare RUL prediction.

The previous methodologies use a different definition of HI, which hardens the comparison task. In the case of WPD-HMM [27], the HI is the log-likelihood/BIC of incoming data being evaluated by an HMM learned with normal or healthy data. The closer the log-likelihood/BIC to zero, the better the model; if the log-likelihood/BIC decreases/increases, the model fitness is worse and abnormal data is being processed. AHMM [28] proposed a HI given by the Cauchy-Schwarz correlation of the AHMM dynamic evolution. Here a correlation of 100% implies no change in the current distribution or model and 0% deduces a total drift from the current distribution or model. APCMD [30] proposed a HI based on the Mahalanobis distance of a reduced set of features after being processed by a principal component analysis (PCA). In this case, a distance of 0 implies no change in distribution, whereas a big distance suggests a drift from the normal data.

The proposed methodology will be compared as follows: In the case of WPD-HMM, the BIC curve works as HI; then it is compared with the one obtained by the proposed methodology. For AHMM, as their methodology uses a Cauchy-Schwarz correlation as health index, it is compared with $HI^{\dagger}$. APCMD and the proposed methodology are the only ones that can generate RUL predictions; therefore their results in HI and RUL are contrasted.

1) FEMTO dataset: The benchmark used to validate the model comes from the "FEMTO Bearing Data Set" [43][2]. This dataset consists of RTF bearings data under critical rotational speed and load conditions. With these conditions, the bearings are forced to fail faster. The idea is to estimate the ball-bearing health state up to failure and its RUL. Table IV shows a description of the dataset. In this dataset, a ball-bearing fails when the accelerometer records are above 20 g, where g is the gravity acceleration at sea level $(9.7805 \frac{m}{s^2})$.

Since our model assumes that a data stream is being received and no previous RTF data is available, the training-testing specification is ignored and those datasets which

[2] https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/ online status on May 13, 2022

Condition	Label	Purpose	# Samples
1	Bearing1_1	Training	2803
1	Bearing1_2	Training	871
1	Bearing1_3	Testing	1802
1	Bearing1_4	Testing	1139
1	Bearing1_5	Testing	2302
1	Bearing1_6	Testing	2302
1	Bearing1_7	Testing	1502
2	Bearing2_1	Training	911
2	Bearing2_2	Training	797
2	Bearing2_3	Testing	1202
2	Bearing2_4	Testing	612
2	Bearing2_5	Testing	2002
2	Bearing2_6	Testing	572
2	Bearing2_7	Testing	172
3	Bearing3_1	Training	515
3	Bearing3_2	Training	1637
3	Bearing3_3	Testing	352

TABLE IV: Training and test data sets of the FEMTO dataset
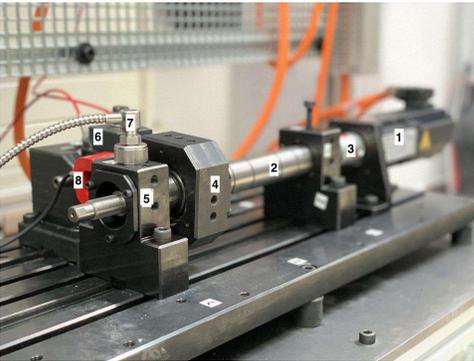
(a)

(b)

Fig. 4: Experimental testbed: (a) Ball-bearing RTF testbed details. (b) CMAI

show evidence of degradation and not a sudden failure are used. Bearings with labels Bearing1_1, Bearing1_3, Bearing2_1 and Bearing2_2 show degradation among the previous mentioned bearings. In this article, the bearings Bearing1_1, Bearing1_3 and Bearing2_2 are used to test the proposed methodology.

2) Mechanical set-up data-set: The purpose of this ball-bearing testing set-up is to monitor vibrations overtime during ball-bearing useful life. The experimental set-up is shown in Figure 4a. This testbed has a Bosch IndraDyn MS2N synchronous servomotor 1 that guarantees required speed during testing, a shaft 2 with different ball-bearing clamping positions and an elastic coupling 3 to the servomotor.
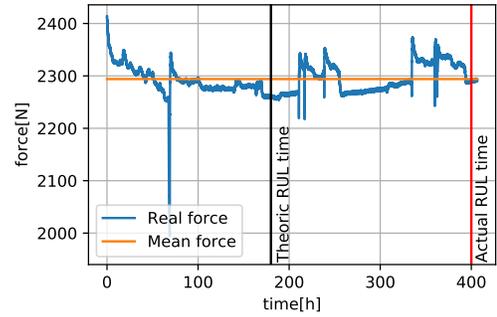
Fig. 5: Observed force evolution given by the actuator

The testbed has three ball-bearing supports, one for the axial force actuator 4 and two for support. The system is designed to affect the outside ball-bearing 5 useful life applying an axial force. The force actuator is a screw-based system fitted with a load sensor that measures the applied force 6. An IMI 607A61 accelerometer with a sensibility of 10.2 mV and vibrational-signal detection range between 0.5-10 kHz. 7 is used to measure vibrations, and a thermocouple 8 to monitor temperature and guarantee testbed integrity.

Figure 4b shows the data acquisition and pre-processing device called Computing Module Aingura Insights (CMAI). The CMAI compute node is an embedded system powered by a ZU3 Zynq® Ultrascale+™ System on Chip (SoC) housing four cores Cortex-A53 and a programmable logic. The node is customized with add-on modules collecting data from different sensors, and it can be coupled with other nodes to improve its computational capabilities.

For our experiments, we implement four channels for accelerometer readings up to 19.5 kHz and sensor fusion capabilities to guarantee signal synchronization. Data collected are stored in files as comma-separated values (CSV).

The CMAI is used to acquire, condition and pre-process the acceleration signal at a sampling rate of 19.5 kHz with 0.5 ns of maximum jitter. In a production setup, the CMAI has enabled processing capabilities to work online with the proposed technique. For validation and complete control over the proposed pipeline, in this paper we stored sensor data into CSV files for offline analysis, simulating the data-stream as in an online environment. The implementation of the model has been written in C++, compiled with GCC 10.2, and executed using a single core of the ZU3 SoC. Ball-bearing data are collected and pre-processed by a single CMAI.

For our test, a radial force of 2.3-2.4 kN and a rotational speed of 3180 RPM (53 Hz) were selected from the traditional mechanical analysis theory [44] in order to run an Eco 6004-2RS ball-bearing to failure between 180 hours (assuming 2.3 kN) and 156 hours (assuming 2.4 kN). The force measured profile is shown in Fig. 5. The mean force was around 2.3 kN which implies that the theoretical RUL was close to 180h. However, the total testbed operation time was over 400 hours, meaning 2.25 times beyond the theoretical useful life. As a safety measure, since the bearings operation time overpassed by several hours the theoretical RUL predictions given by the

provider, we stopped the acquisition when audible signals of degradation were evident. Finally, a signal smoothing processing was applied: a root-mean-square was computed every 50 samples from the FE box in Fig 1, equivalent to collapsing 85 seconds of data to a single data instance.

The CMAI was configured to keep the amplitudes of fundamental frequencies and discard the rest of vibrational data. Health index ($HI^\dagger$) and RUL predictions were stored.

V. RESULTS

A. Time Analysis

In this section a time consumption of the proposed algorithm is shown. Regarding the feature extraction (FE) box of Fig. 1, the CMAI required at least 1.6 seconds of measurements for a reasonable granularity in the signal spectrum in order to obtain the desired frequencies with precision. The signal processing and feature extraction follows the methodology proposed by [35] as described above. After the data acquisition, the CMAI required 1.5 seconds to extract the desired features.

In Table. V, the mean and standard deviation of the time execution algorithm of the three boxes in Fig. 1, say: *novelty detection*, *model update* and *RUL prediction* are shown for each tested data-set. Note that for the FEMTO data-sets, the maximum mean time needed for the *Novelty detection* process it was 1.16ms, for the *Model update* was 6.03ms and for *RUL prediction* it was 2.44ms. On the other hand, for the mechanical set-up, it is observed that the mean times for the *novelty detection* were 8.73 times higher than in the case of the FEMTO datasets, for the *model update* 15.1 times higher and for *RUL prediction* it was 9.37 times higher. This can be explained due to the length of the data and the higher number of hidden states. Finally, it is observed that the standard deviations of the times for all the cases were lower than one second which gives evidence of relatively stable algorithms.

These time results show that the algorithm can achieve a fair time response for an online data stream environment. Nonetheless a more detailed analysis is given in Section V-C1

Dataset	Phase	$\bar{t}$(ms)	S_t(ms)
Bearing_11	Novelty detection	0.960	0.713
	Model update	4.291	32.474
	RUL prediction	2.188	1.429
Bearing_13	Novelty detection	1.160	1.005
	Model update	6.025	39.298
	RUL prediction	2.440	2.201
Bearing_22	Novelty detection	0.982	0.417
	Model update	3.933	23.100
	RUL prediction	1.610	0.746
Set-up	Novelty detection	10.130	7.015
	Model update	91.049	292.751
	RUL prediction	22.884	16.450

TABLE V: Execution times statistics of the different phases of the algorithm in the CMAI, for the different datasets

B. FEMTO results

1) Bearing1_1 results: Fig. 6 summarizes the results obtained for the data-set Bearing1_1. We can see that the BIC score from the WPD-HMM methodology is drawn

(a) WPD BIC score

(b) Proposed BIC score

(c) AHMM health index

(d) APCMD health index

(e) APCMD RUL

(f) Proposed $HI^\dagger$

(g) Proposed RUL

(h) Regression uncertainty
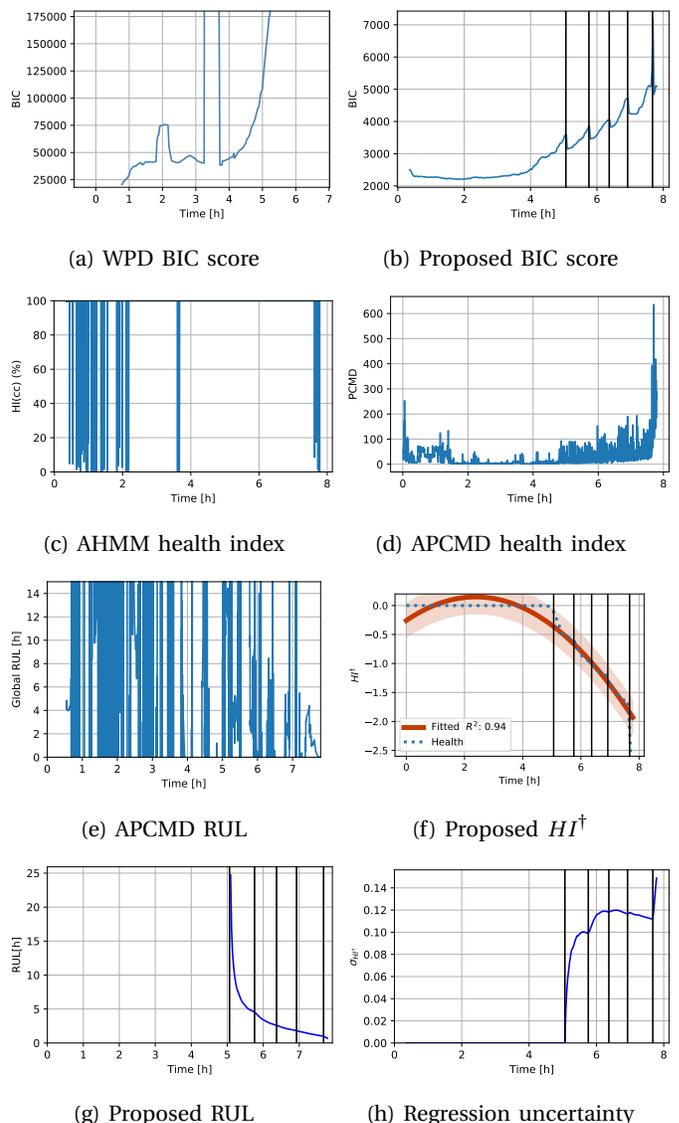
Fig. 6: Results of supervising Bearing1_1 with different methodologies: (a) shows the WPD BIC score evolution, (b) draws the As-HMM online model BIC score evolution, (c) gives the AHMM health index, (d) pictures the APCMD health index, (e) displays APCMD RUL prediction, (f) shows $HI^\dagger$ results, (g) is the obtained RUL curve from the proposed methodology, (h) is the standard deviation of the proposed health index regression

in Fig. 6a which is used as a health index. It can be observed that WPD-HMM obtains results different from those obtained by the online As-HMM which are displayed in Fig. 6b. In particular, it can be noticed that, arriving at the fourth operational hour, the BIC score in WPD-HMM had a discontinuity; later it returned back to a lower BIC score, but followed by exponential growth. For the WPD-HMM methodology, it can be argued that, before the fourth hour, an accelerated degradation process to failure can be observed. On the other hand, the proposed methodology uses a novel detection technique. When a novel trend was detected at time *t*, a vertical black line was displayed in the plot. It can be said that whenever a novel trend

was detected, the BIC score was improved and drastic changes in BIC (as observed in the case of WPD-HMM) were avoided. However, an essential controlled growth in BIC could be observed after the fourth hour.

The health index in the case of AHMM, which can be seen in Fig. 6c, reveals a noisy behavior at the beginning and at the end of the ball-bearing life which is not informative at all. In particular, the early decays to 0 would imply an early failure, which is not true. $HI^\dagger$ drawn on Fig. 6f remained unchanged until the first novel trend was detected. After that, $HI^\dagger$ decreased and showed evidence of degradation.

In this experimental set-up, the uncertainty of the RUL was exposed in Fig. 6h with the standard deviation of the health index prediction. This uncertainty is the level of fidelity of the health index regression; the higher this measure, the worse the fitness of the regression for RUL prediction. Regarding Bearing1_1, the uncertainty appeared after the first concept drift as expected. Note that the uncertainty or standard deviation was around 0.12 orders of magnitude, far from half an order of magnitude, which indicated a fair regression for RUL prediction. This can be visualized in the health index regression shaded curves in Fig. 6f It is remarkable to say that the uncertainty varied with time, because the regression was updated whenever a new $HI^\dagger$ instance was computed.

The proposed methodology and APCMD are capable of prognosis. Both use a regression process to predict the health index. However, in the case of APCMD, the health index is a Mahalanobis distance based on the PCA and, in the case of the proposed methodology, $HI^\dagger$ is based on logarithms of ratios. Both methodologies require a threshold on the health index to determine the RUL. The threshold in the case of APCMD was set to 200; this value was set after observing the evolution of the Mahalanobis distance exposed in Fig. 6d. In the case of APCMD, a noisy RUL prediction during almost all the ball-bearings life is drawn in Fig. 6e. In particular, it can be observed that at early times the RUL was zero, which would imply an early failure, which again was not true. Nevertheless, at the end of the process, after the seventh hour the accuracy was improved, for the given threshold. In the case of the proposed methodology, the RUL, pictured in Fig. 6g, was only computed after the first concept drift; after that, the RUL went towards zero. It is worth noting that the first predictions overestimated the RUL, and later, the regression gave better predictions.

From the point of view of applicability, a maintenance engineer will use the actionable insight from the proposed methodology as a decision support element to plan ball-bearing maintenance. It is important that the actionable insight provided is relevant where the ball-bearing is critical to operation, with a high risk of stopping the machine or production line if an unexpected failure occurs. Therefore, if the engineer on charge observes a decrease in $HI^\dagger$, the engineer should begin to plan a controlled maintenance stop with haste as indicated by the indicator trend.

As stated before, the proposed methodology is capable of giving an explanatory model, where temporal and instantaneous probabilistic relationships may appear. Fig. 7 displays a pair of generated Bayesian networks from the learned
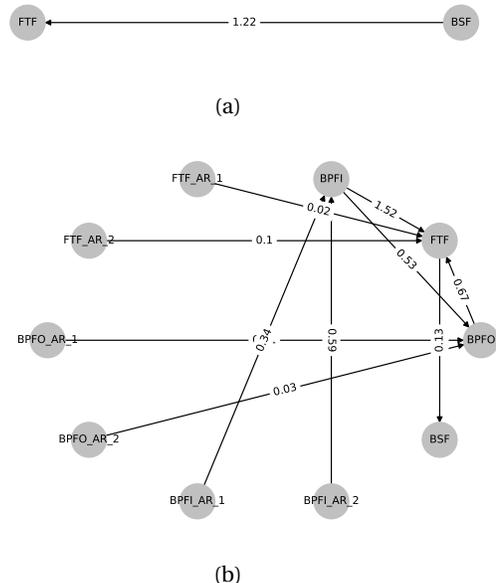
Fig. 7: Learned Bayesian networks from Bearing1_1

hidden states. Fig. 7a shows the Bayesian network generated from a low degradation state, whereas Fig. 7b draws a Bayesian network from a degenerated state. Observe that, in a low level of degradation, in this case, few probabilistic relationships appear, whereas in a degraded state, in this case, more relationships appear. Each arc represents a probabilistic dependency; in the case of a degraded state, it can be observed that the BPFI frequency-amplitude affects the BPFO and the FTF frequency-amplitudes; or in other words, in this ball-bearing, in a degraded state the ball-bearing inner ring behavior drives the ball-bearing outer ring and cage. Following the structure of the Bayesian network, it can also be observed that some AR dependencies appear e.g., the BPFI frequency-amplitude is affected by two of its previous values.

2) Bearing1_3 results: Fig. 8 shows the results obtained for the dataset Bearing1_3. The BIC score from the WPD-HMM methodology is pictured in Fig. 8a. In this case we can observe that after time 4 h, the BIC score grew abruptly, which indicates an accelerated degradation and failure. If it is compared with the BIC score obtained from the proposed methodology Fig. 8b, we can observe that a concept drift appeared at the fourth operational time and the BIC grew from this point in a controlled manner with the appearance of more concept drifts. This difference evident that our proposed methodology is capable of describing, in a refined manner, the ball-bearing degradation and failure states.

In the case of the AHMM health index, which is pictured in Fig. 8c, this time it shows a noisy behavior during all the ball-bearing operational time. Although the Cauchy-Schwarz distribution correlation is easy to interpret (0% implies zero correlation between distributions and 100% indicates perfect correlation), in this case, due to the noisy behavior and jumps between 100% and 0%, no insights can be extracted from the health index. Meanwhile $HI^\dagger$ in Fig. 8f, remained unchanged until the first novel trend

(a) WPD BIC score

(b) Proposed BIC score

(c) AHMM health index

(d) APCMD health index

(e) APCMD RUL

(f) Proposed $HI^{\dagger}$

(g) Proposed RUL

(h) Regression uncertainty
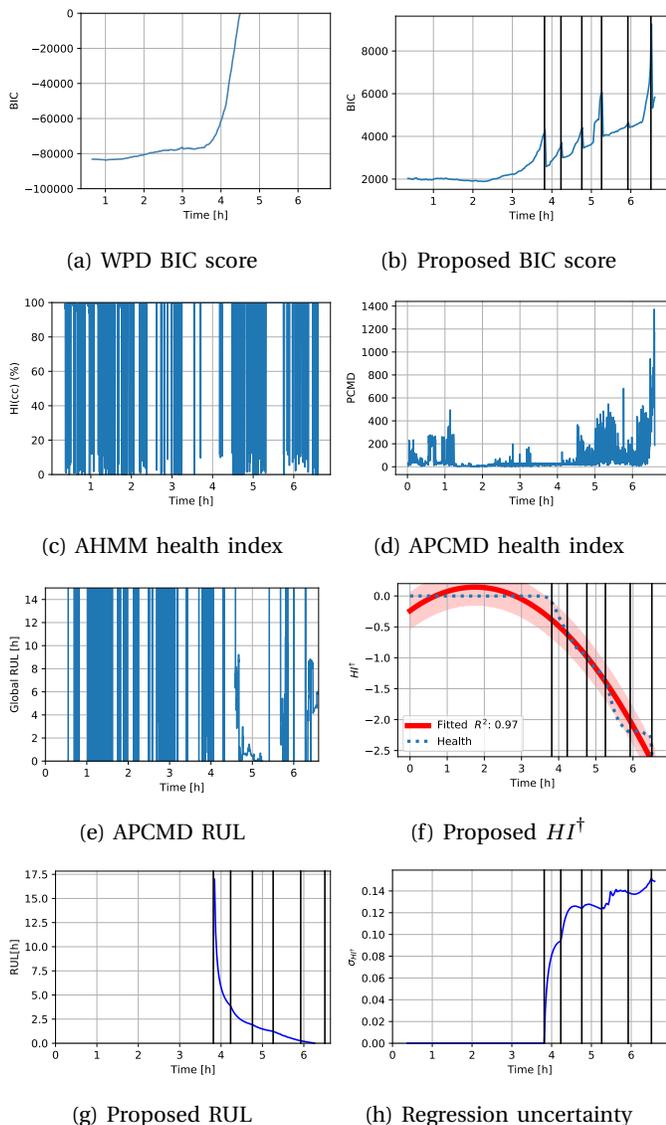
Fig. 8: Results of supervising Bearing1_3 with different methodologies: (a) shows the WPD BIC score evolution, (b) draws the As-HMM online model BIC score evolution, (c) gives the AHMM health index, (d) pictures the APCMD health index, (e) displays APCMD RUL prediction, (f) shows $HI^{\dagger}$ results, (g) is the obtained RUL curve from the proposed methodology, (h) is the standard deviation of the proposed health index regression

was detected as in the case of Bearing1_1. After that, $HI^{\dagger}$ decreased and showed evidence of degradation until it reached the failure threshold.

Now, concerning RUL, APCMD as before obtained a noisy prediction, as shown in Fig. 8e, of the ball-bearings RUL. It indicates failure during the first hours of the bearing operation, which is not true. Even during the last hours of the ball-bearing operation, the RUL prediction was highly inaccurate. On the other hand, the proposed model RUL prediction (see Fig. 8g) showed predictions only after the first concept drift was detected. As in the case of Bearing1_1, at the first prediction times, the RUL was overestimated; however, as time went, the RUL prediction became

more accurate until $HI^{\dagger}$ passed the failure threshold. Since the proposed methodology exposed a similar prediction as in the case of Bearing1_1, the maintenance policies that can be generated for the Bearing1_1 are also valid for this ball-bearing (Bearing1_3).

In this case, it is observed that the uncertainty of the regression pictured in Fig. 8h used for RUL prediction was different from that exposed in Bearing1_1. Observe that the level of uncertainty did not pass over 0.15 orders of magnitude, which implies that the fidelity of the RUL was fair.
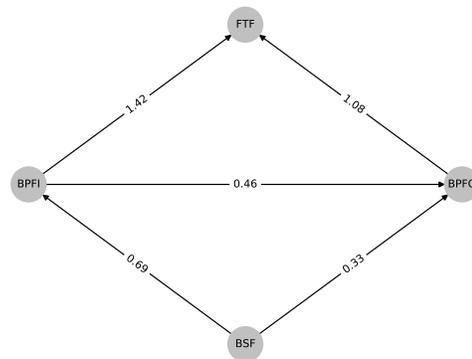
Fig. 9: Learned Bayesian networks from Bearing1_3

Here, we also expose one of the learned Bayesian networks learned by the asymmetric hidden Markov model. Fig. 9 displays one Bayesian network from an intermediate degraded state. For example, the FTF frequency-amplitude depends on the BPFI and BPFO frequency-amplitudes, the BPFI frequency-amplitude depends on the BSF frequency-amplitude, and the BPFO frequency-amplitude relies on the BPFI and BSF frequency-amplitude. From this graph we can say that the ball-bearing balls are the ones leading the mechanical behavior of the ball-bearing at an intermediate degradation level.

3) Bearing2_2 results: Fig. 10 shows the BIC score from the WPD-HMM methodology. It can be observed that, during the first hour of operation, the BIC score grew in an important manner, which for that methodology, implies an early fast degradation process in the ball-bearing. On the other hand, the proposed methodology observed a concept drift during the first hour of operation; nevertheless, the evolution of the BIC score did not grow as uncontrolled as in the WPD-HMM methodology. It can be said, as in the case of Bearing1_1, that whenever a novelty detection occurred, the BIC score was improved and this adaptation prevented the model from obtaining drastic changes in this score. However, in this case, a clear increasing trend in BIC was not seen as in the case of Bearing1_1 or Bearing1_3; on the contrary, from the first and a half hour to the second hour of operational time, a decrease in BIC was obtained.

The health index in the case of AHMM (Fig. 10c) reveals again a noisy behavior that does not provide any relevant information if it is compared to any of the other methodologies. In the case of APCMD, the health index (Fig. 10d) shows noisy growth indicating a worsening in the

(a) WPD BIC score

(b) Proposed BIC score

(c) AHMM health index

(d) APCMD health index

(e) APCMD RUL

(f) Proposed $HI^\dagger$

(g) Proposed RUL

(h) Regression uncertainty
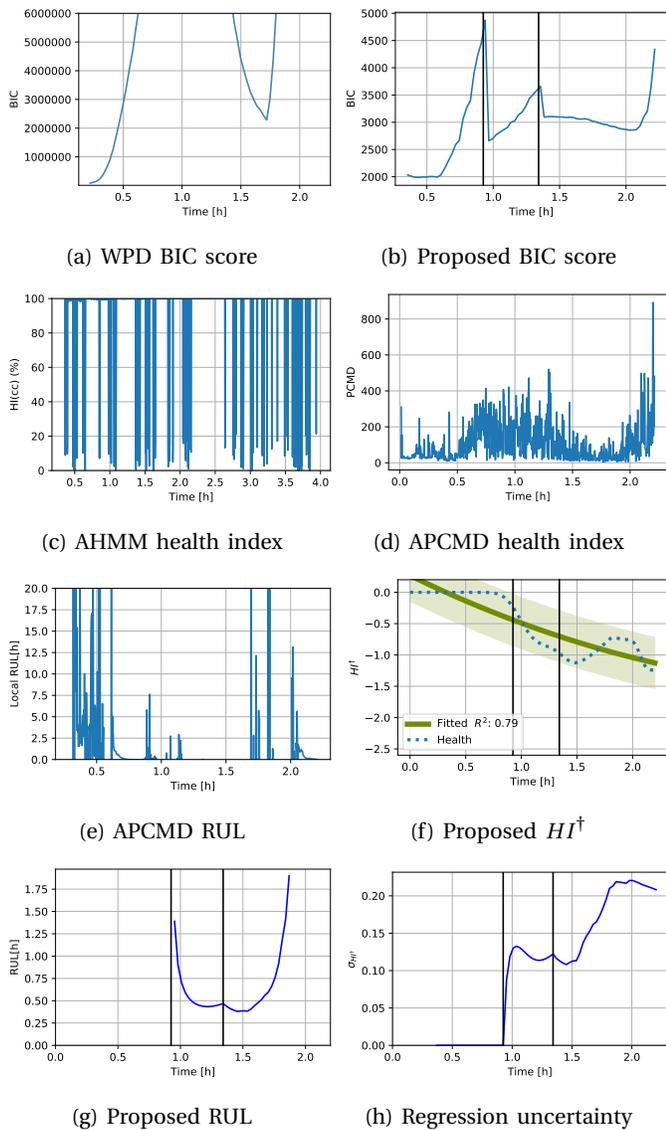
Fig. 10: Results of supervising Bearing2_2 with different methodologies: (a) shows the WPD BIC score evolution, (b) draws the As-HMM online model BIC score evolution, (c) gives the AHMM health index, (d) pictures the APCMD health index, (e) displays APCMD RUL prediction, (f) shows $HI^\dagger$ results, (g) is the proposed methodology RUL prediction, (h) is the standard deviation of the proposed health index regression

ball-bearing condition. Later, the health index decreased between the first hour and a half to the second hour of operational time which implies improvement in the ball-bearing condition. However, since this health index lies in the PCA projected space it is hard to tell if the improvement was significant or not. In the case of our methodology, the health index $HI^\dagger$ decreased after the first concept drift. Nonetheless, between the first hour and a half to the second hour of operational time (as the WPD and APCMD methodologies also detect) a slight (0.4 orders of magnitude) health improvement was observed; after that, $HI^\dagger$ decreased again.

In terms of RUL curve, the APCMD obtained a noisy prediction during almost all the ball-bearing life as shown in Fig. 10e. In this case, the local RUL had to be used instead of the global because it obtained poor results. Nevertheless, at the end of the process after the second hour of operation the accuracy was improved. Fig. 10g shows the obtained RUL curve from the proposed methodology. In this case, after the first novel detection flag, a decreasing tendency was observed. But, as mentioned before, an improvement in health was observed; this violates the second assumptions imposed in Section. III. As a consequence of this violation, the prediction curve sign convexity changed, which could be translated into the increasing trend in the RUL curve. Although the RUL prediction was poor in this case, thanks to $HI^\dagger$, there was statistical evidence of bearing degradation.

The uncertainty of the RUL prediction in this case is shown in Fig. 10h. In spite that the RUL prediction in this case was worse when compared with the other datasets from FEMTO, the standard deviation of the prediction obtained fair values below a quarter of order of magnitude. However it is relevant to remark that the change in the sign of concavity in the health index after the first hour and a half affected drastically and negatively to the confidence of the prediction.

In this case, from the point of view of application, these actionable insights are also useful to explore failure related to wrong installation procedure for the ball-bearing. That is, early problems could suggest that the ball-bearing has problems with actual operating conditions and must be reinstalled as soon as possible to avoid unexpected failures.
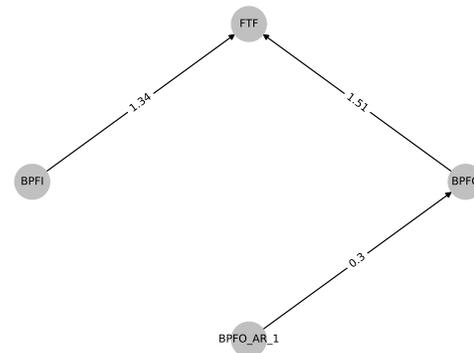
Fig. 11: Learned Bayesian networks from Bearing 2_2

As in the case of Bearing1_1 and Bearing1_3, for each learned hidden state, context-specific Bayesian networks can be extracted. In Fig. 11 the resulting Bayesian networks from a degraded state is drawn. In this scenario, the FTF frequency-amplitude dynamics can be explained by the BPFI and BPFO frequency-amplitude. Also, the BPFO depends on one previous value. In this case, the inner and outer race lead the dynamic behavior at a degraded state.

C. Mechanical set-up results

In Fig. 12, the worn ball-bearing inner ring is shown. Different degradation pieces of evidence are detected, such

as in Fig. 12a, where a rolling surface indentation is found, usually started by small particles generated by the ball cage or the lateral seals. In Fig. 12b and 12c, porosity has developed over the rolling surface, related to the lubrication issues created by lateral seal failure and temperature. As shown in Fig. 13a, most of the time, the temperature was between 40 and 45°C, which is not enough to damage the ball-bearing. However, this effect mixed with the failure of the seals can create lubrication problems in the balls.

Fig. 13b pictures the smoothed evolution of the ball-bearings fundamental frequencies (BPFO, BPFI, BSF, FTF) extracted from the FE algorithm. From this picture, it can be observed that there is evidence of ball-bearing degradation since the ball-bearing frequency-amplitudes increase over time; in particular, the FTF frequency-amplitude exhibits a more significant positive trend. Nevertheless, by just looking, it is not possible to extract a clear health index or bearings RUL estimation. This enables the use of the proposed methodology.

(a)

(b)

(c)
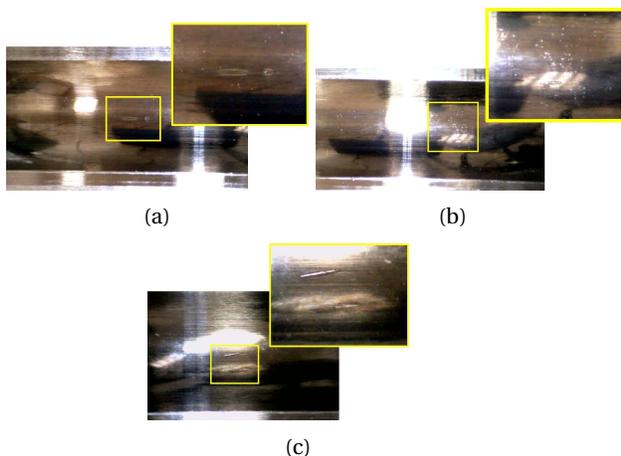
Fig. 12: Test results: (a), (b), (c) show evidence of degradation in the inner ring

(a)

(b)
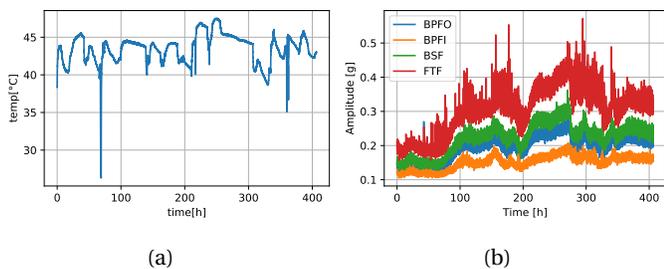
Fig. 13: a shows the Measured temperature evolution of the ball-bearing. b shows the Evolution of the ball-bearings fundamental frequencies

It is expected that the proposed methodology (Fig. 1) process data streams within embedded electronics. However, to analyze its performance, the batch SEM algorithm of the As-HMM [10] is studied from the computational point of view inside the CMAI module. In this way, memory and time constraints and opportunities can be deduced for the model

learning phase and can enable the model to be optimally employed in an online data stream.

1) Execution time and performance analysis: The As-HMM implementation has been evaluated using the following configuration: 25 variables, a window size of 5000 samples, and 3 hidden states. The C++ code makes use of two external libraries, igraph v0.8.4 and openblas v0.3.13 and it has been compiled with GCC v10.2. All computational measurements presented in this section have been gathered on one CMAI.
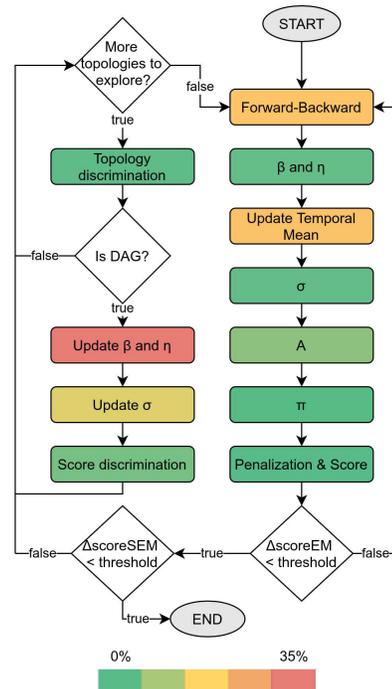
Fig. 14: Steps of the As-HMM algorithm color coded depending on the relative execution time

The first step for evaluating the experimental setup has been to study the execution time of our optimized implementation of the As-HMM algorithm. Fig. 14 shows the phases of the algorithm: the color code represents the percentage of the execution time taken by each phase, from green, corresponding to short execution time, to red, corresponding to steps taking up to 35% of the total execution time. From Fig. 14, the reader can conclude that `Update` $\boldsymbol{\beta}$ `and` $\boldsymbol{\eta}$, `Forward-Backward`, `Update Temporal Mean`, `Update` $\boldsymbol{\sigma}$ are the phases taking more time, summing up 87% of the total execution time.

Fig. 15 shows the execution time of the four phases on a single Cortex-A53 of the CMAI.
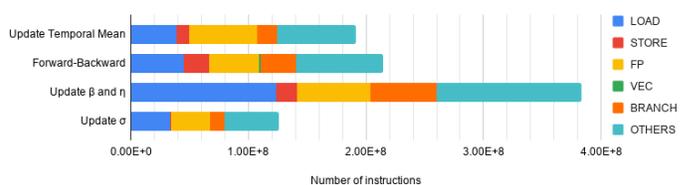
Fig. 15: Instruction mix of the four most time consuming phases of the As-HMM algorithm

Using hardware counters, it has been possible to count different types of instructions, which are also reported in Fig. 15. The classes of instructions identified are: memory accesses (LOAD and STORE or LD and ST for short respectively); floating-point instructions (FP as counting the scalar instructions and VEC as counting the instructions executed by the single instruction multiple data (SIMD) unit of the core); branches and jumps (BRANCH); integer operations and other instructions for the bookkeeping of the program (OTHERS).

From a first analysis of the type of instructions, it appears that the compiler was not able to vectorize the code. There were, in fact, very few vector instructions (VEC) in all phases so that they can be ignored in the rest of our study. Also, the phase Update σ had almost no store instructions: it had been verified that, indeed, the Update σ phase required loading on average more than 300 data to update/store one value to the memory.

	Instruction type per clock				
	IPC	LD + ST	FP	BRANCH	OTHERS
Update β and η	0.90	0.33	0.15	0.13	0.29
Forward-Backward	0.80	0.25	0.16	0.11	0.28
Update σ	0.74	0.20	0.20	0.07	0.27
Update Temporal Mean	0.69	0.18	0.21	0.06	0.24
Correlation with IPC	1.00	0.99	-0.94	0.96	0.91

TABLE VI: Density of the different type of instructions in four phases of the As-HMM algorithm

To understand if there are types of instructions that are harming the performance of our implementation, the density over time of each type of instruction has been computed. The global density of all instructions is called Instructions per Clock-Cycle (IPC), and it has been computed for each phase. The densities for all classes of instructions have also been computed: memory accesses (load and store instructions are aggregated, LD + ST) per cycle, floating-point (FP) per cycle, branch (BRANCH) per cycle, and other instructions (OTHERS) per cycle. All values are reported in Table VI. Phases are sorted by the decreasing value of the IPC, and the values of instruction densities have also been color-coded. Looking at the color gradients, it can be detected that there is a direct correlation between the IPC (yellow gradient) and all instruction densities (green gradients), except for the floating-point instructions. This means that increasing the number of floating-point instructions per unit of time implies a lower IPC resulting in performance degradation.

Table VI highlights that there is a direct correlation between the IPC and the density of memory accesses (LD + ST): this result is counterintuitive. Since the memory is the slowest of the resources in a compute node, one would expect the opposite. For this reason, the number of accesses to memory that were misses in the two-level of caches of the Cortex-A53 core have been studied. The measured number of misses per kilo-instructions (MPKI) are between 1.56 and 7.63 in the L1 cache and between 0.01 and 0.09 in the L2 cache (that is the last level cache of the Cortex-A53). This means that almost all the memory accesses of our implementation are hit on cache, explaining the positive

effect of memory instructions on the global IPC.

This effect depends on the configuration of the model: this would probably become worse, increasing the number of variables, hidden states, or the size of the window. Also, changing the configuration of the model would change the data structures storing the graph used for the model. The code is susceptible to changes in the data structures so that one can expect cache effects in the case of changes in the initial configurations. Our current study shows that we are able to spot weaknesses of the code, and we leave this for future research.

The data structures of the model are sparse and have an arithmetic intensity between 0.06 and 0.15 Floating Point Operations per Byte (FLOPs/Byte). These low values of the arithmetic intensity can be mitigated by reorganizing the data structures as adjacency lists. This has the benefit that the code can operate with more dense objects and maximize the cache reuse. As a price to pay, more integer instructions for pointer bookkeeping are needed. This is visible in Fig. 15 with the fact that almost 30% of the instructions of all phases are of type OTHERS.

2) Health states and RUL prediction: The weaknesses and strengths of the underlying model (As-HMM) of the proposed methodology have been explored and mentioned. The corresponding results from the methodology using a ball-bearing are now shown.
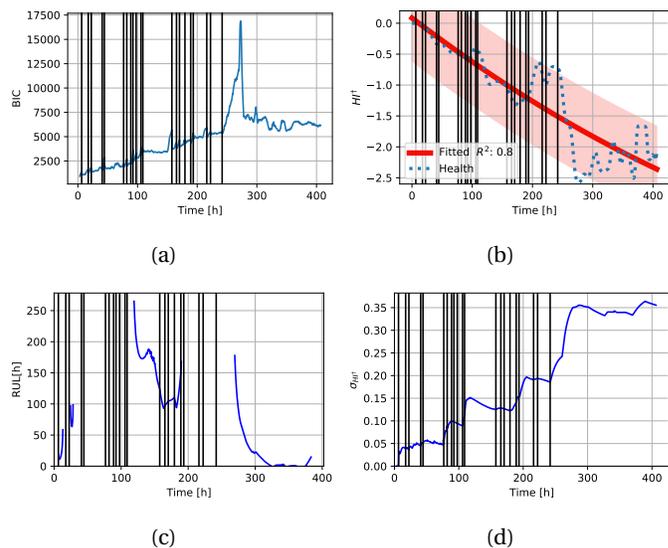
Fig. 16: Results of mechanical setup. (a) is the BIC evolution (b) is the health index $HI^{\dagger}$, (c) is the online As-HMM RUL predictions, (d) is the standard deviation of the health index regression

Recall that no RTF data (i.e., training data) is used for this analysis, and the health index and RUL are computed as the data arrives at the processor (i.e., the simulated data stream processing). In Fig. 16a the evolution of the BIC score is highlighted. Although the model is adapted, a clear increasing trend can be observed in the BIC score. At the end of the process, 21 hidden states were learned. On the other hand, the first concept drift was detected during the first hours of operation. In Fig. 16b the health index $HI^{\dagger}$ re-

sults are shown. In this case, $HI^\dagger$ shows a slow degradation process with some short health recovering phases, which violates the assumptions in Section. III. When observing the corresponding RUL prediction in Fig. 16c, it can be observed that the RUL went from being predictable to unpredictable at different times. This behavior is caused by the observed health recoveries as in the case of Bearing2_2. The health recoveries may cause a change in the sign convexity in the prediction curve; in such cases, the RUL cannot be computed; in spite of that, as observed in the latest RUL predictions, at the time 270 h, the RUL decreases and the regression predicts 170 hours of remaining useful life. As time goes, this prediction converges at 320 h as the time when the failure threshold is overpassed. Although the assumptions made in Section III were violated, the RUL predictions and $HI^\dagger$ gave insights of a bearing that was evolving towards a failure state. However, the extreme degradation from the beginning provides insights into the bearing installation on the machine or quality issues of the brand-new ball-bearing.

Regarding the uncertainty of the prediction for this data-set and mechanical set-up, Fig. 16d shows the evolution of the health index regression. In this case, at the first 100 hours of operation, the uncertainty level was around 0.05 orders of magnitude. However, since there was a change in the sign of the concavity of the health index, it caused an important increase in the uncertainty. At 200 hours, a change in the sign of the concavity in the health index caused again an increase in the uncertainty of the health index regression, reaching a maximum of 0.35 orders of magnitude. Notice that with this level of uncertainty, the dispersion of the prediction can be important and affect the RUL prediction as observed in the shaded curves in Fig. 16b, where the shaded area is bigger than in the FEMTO data-sets. Therefore, the condition of no health recoveries is important to minimize the RUL uncertainty.
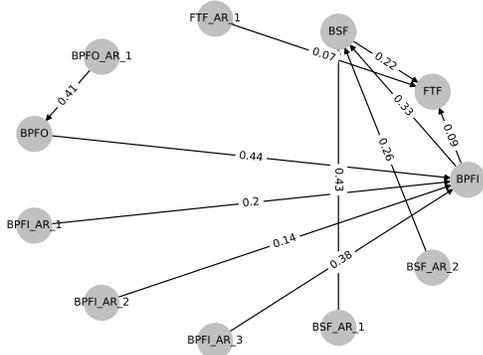
Fig. 17: Learned Bayesian networks from the own testbench testbed

As previously, the model can estimate the possible probabilistic relationships between variables. This step plays a pertinent role in the interpretation of the ball-bearing degradation. In particular, in this case, the resulting Bayesian network of the most degraded observable state in Fig. 17 is shown. A degradation in this ball-bearing caused

the BSF and FTF frequency-amplitudes to depend on the PBFI frequency-amplitude. The BPFI frequency-amplitude depended on the BPFO frequency-amplitude. In this way, it was interpreted that the outer ring of the ball-bearing drove the process, and the bearing rollers and cage depended totally on the behavior of the inner and outer ring. Finally, all the frequencies had a certain degree of dependency on the past, which is the BPFI frequency-amplitude, which required more past values to be explained (3 auto-regressive past values). In contrast, the BPFO frequency-amplitude had only one AR dependency.

VI. CONCLUSION

This paper presents a complete online predictive health assessment solution in terms of an algorithm and its deployment, to monitor ball-bearings in real IIoT environments. This machine learning-based solution can work without previous ball-bearing RTF data, which is one of the main challenges in the industry. Specifically, the proposed methodology's output is the ball-bearing health status, which is expressed in different orders of magnitude from a healthy state, and hours for the remaining useful life of the ball-bearing.

As natural degradation is expected in real industrial environments, we use a concept drift detection methodology to automatically update an adapatite asymmetric hidden Markov model when novel trends appear.

As the new methodology works at embedded devices, from the computational point of view, a performance analysis study of the code has been performed, highlighting that the code is highly dependent on how data structures are stored in the memory of the Edge device. Also, the performance of the code is directly correlated with the density of floating-point instructions. The obtained results gave to the authors the fundamental insights to design the algorithm parametrization strategy towards an optimum performance under limited computing power environments.

Different real applications of ball-bearings were tested under separate operating conditions to test the proposed methodology, showcasing useful actionable insights that maintenance practitioners can use.

The proposed methodology was compared with other state-of-the-art methodologies. Although every methodology has its own definition of health index, when our health index was compared to other health indexes such as Malahanobis distance from APCMD, BIC score from WPD or Cauchy-Schwarz divergence from AHMMM, it was observed that our health measure was more informative and interpretable for all the time series and was robust to noise and outliers. In terms of fitness, when compared to a non-adaptative methodology such as WPD, it was observed that the fitness or BIC score of our methodology was more stable and informative. In terms of remaining useful life prediction, it was observed that under certain conditions, the remaining useful life can be predicted; otherwise, the predictions are no longer accurate. Additionally, an operational threshold must be tuned depending on the application, influencing the RUL prediction. In spite of these drawbacks, the ball-bearing health reading alone can be a good indicator of performance. Finally, the asymmetric

hidden Markov model was capable of learning, for each data series, a set of Bayesian networks that are useful to give further insights into the evolution of the dynamic process, probabilistic dependencies and degradation process.

REFERENCES

[1] P. Larrañaga, D. Atienza, J. Diaz-Rozo, A. Ogbechie, C. Puerto-Santana, and C. Bielza, *Industrial Applications of Machine Learning*. CRC Press, 2018.

[2] S.-T. Tseng, "Optimal preventive maintenance policy for deteriorating production systems," *IIE transactions*, vol. 28, no. 8, pp. 687–694, 1996.

[3] R. K. Mobley, *An Introduction to Predictive Maintenance*. Butterworth-Heinemann, 2002.

[4] E. Sutrisno, H. Oh, A. S. S. Vasan, and M. Pecht, "Estimation of remaining useful life of ball bearings using data driven methodologies," in *2012 IEEE Conference on Prognostics and Health Management*. IEEE, 2012, pp. 1–7.

[5] A. Skariah, R. Pradeep, R. Rejith *et al.*, "Health monitoring of rolling element bearings using improved wavelet cross spectrum technique and support vector machines," *Tribology International*, p. 106650, 2020.

[6] J. Diaz-Rozo, C. Bielza, and P. Larrañaga, "Machine-tool condition monitoring with Gaussian mixture models-based dynamic probabilistic clustering," *Engineering Applications of Artificial Intelligence*, vol. 89, p. 103434, 2020.

[7] M. Markou and S. Singh, "Novelty detection: a review—part 1: statistical approaches," *Signal Processing*, vol. 83, no. 12, pp. 2481–2497, 2003.

[8] M. L. Bueno, A. Hommersom, P. J. Lucas, and A. Linard, "Asymmetric hidden Markov models," *International Journal of Approximate Reasoning*, vol. 88, pp. 169–191, 2017.

[9] J. Diaz-Rozo, C. Bielza, and P. Larrañaga, "Clustering of data streams with dynamic Gaussian mixture models: An IoT application in industrial processes," *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 3533–3547, 2018.

[10] C. Puerto-Santana, P. Larrañaga, and C. Bielza, "Autoregressive asymmetric linear Gaussian hidden Markov models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–17, 2021.

[11] S. Lee, L. Ji, and J. Jun, "Online degradation assessment and adaptive fault detection using modified hidden Markov model," *IEEE Transactions on Reliability*, vol. 132, no. 2, pp. 183–198, 2010.

[12] D. Tobon, K. Medjaher, N. Zerhouni, and G. Tripot, "A data-driven failure prognostics method based on mixture of gaussians hidden Markov models," *IEEE Transactions on Reliability*, vol. 61, no. 2, 2012.

[13] A. Soualhi, H. Razik, G. Clerc, and D. D. Doan, "Prognosis of bearing failures using hidden Markov models and the adaptive neuro-fuzzy inference system," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 6, pp. 2864–2874, 2014.

[14] D. L. Cartella F., Lemeire J. and S. H., "Hidden semi-Markov models for predictive maintenance," *Mathematical Problems in Engineering*, vol. 2015, 2015.

[15] W. Li and T. Liu, "Time varying and condition adaptive hidden Markov model for tool wear state estimation and remaining useful life prediction in micro-milling," *Mechanical Systems and Signal Processing*, vol. 131, pp. 689–702, 2019.

[16] A. Kumar, R. B. Chinnam, and F. Tseng, "An HMM and polynomial regression based approach for remaining useful life and health state estimation of cutting tools," *Computers and Industrial Engineering*, vol. 128, pp. 1008–1014, 2019.

[17] A. Soualhi, G. Clerc, H. Razik, and F. Rivas, "Long-term prediction of bearing condition by the neo-fuzzy neuron," in *2013 9th IEEE International Symposium on Diagnostics for Electric Machines, Power Electronics and Drives (SDEMPED)*, 2013, pp. 586–591.

[18] C. Zhao and F. Gao, "Critical-to-fault-degradation variable analysis and direction extraction for online fault prognostic," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 3, pp. 842–854, 2017.

[19] O. Abdeljaber, S. Sassi, O. Avci, S. Kiranyaz, A. A. Ibrahim, and M. Gabbouj, "Fault detection and severity identification of ball bearings by online condition monitoring," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 10, pp. 8136–8147, 2019.

[20] Y. Chen, G. Peng, Z. Zhu, and S. Li, "A novel deep learning method based on attention mechanism for bearing remaining useful life prediction," *Applied Soft Computing*, vol. 86, p. 105919, 2020.

[21] J. Lin, Z. Lin, G. Liao, and H. Yin, "A novel product remaining useful life prediction approach considering fault effects," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 11, pp. 1762–1773, 2021.

[22] C. Chen, N. Lu, B. Jiang, and C. Wang, "A risk-averse remaining useful life estimation for predictive maintenance," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 2, pp. 412–422, 2021.

[23] R. Jiao, K. Peng, and J. Dong, "Remaining useful life prediction for a roller in a hot strip mill based on deep recurrent neural networks," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 7, pp. 1345–1354, 2021.

[24] H. Li, G. Hu, J. Li, and M. Zhou, "Intelligent fault diagnosis for large-scale rotating machines using binarized deep neural networks and random forests," *IEEE Transactions on Automation Science and Engineering*, pp. 1–11, 2021.

[25] N. Gebraeel, A. Elwany, and J. Pan, "Residual life predictions in the absence of prior degradation knowledge," *IEEE Transactions on Reliability*, vol. 58, no. 1, pp. 106–117, 2009.

[26] R. K. Singleton, E. G. Strangas, and S. Aviyente, "Extended kalman filtering for remaining-useful-life estimation of bearings," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 3, pp. 1781–1790, 2015.

[27] H. Ocak, K. A. Loparo, and F. M. Discenzo, "Online tracking of bearing wear using wavelet packet decomposition and probabilistic modeling: A method for bearing prognostics," *Journal of Sound and Vibration*, vol. 302, no. 4, pp. 951–961, 2007.

[28] J. Yu, "Adaptive hidden Markov model-based online learning framework for bearing faulty detection and performance degradation monitoring," *Mechanical Systems and Signal Processing*, vol. 83, pp. 149–162, 2017.

[29] N. Li, Y. Lei, J. Lin, and S. X. Ding, "An improved exponential model for predicting remaining useful life of rolling element bearings," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7762–7773, 2015.

[30] J. Wu, C. Wu, S. Cao, S. W. Or, C. Deng, and X. Shao, "Degradation data-driven time-to-failure prognostics approach for rolling element bearings in electrical machines," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 1, pp. 529–539, 2019.

[31] R. Mohammadi-Ghazi, R. E. Welsch, and O. Büyüköztürk, "Kernel dependence analysis and graph structure morphing for novelty detection with high-dimensional small size data set," *Mechanical Systems and Signal Processing*, vol. 143, p. 106775, 2020.

[32] D. Liu, H. Zhen, D. Kong, X. Chen, L. Zhang, M. Yuan, and H. Wang, "Sensors anomaly detection of industrial internet of things based on isolated forest algorithm and data compression," *Scientific Programming*, vol. 2021, pp. 1–9, 2021.

[33] M. A. Khan and K. A. Abuhasel, "An evolutionary multi-hidden markov model for intelligent threat sensing in industrial internet of things," *The Journal of Supercomputing*, vol. 77, pp. 6236–6250, 2021.

[34] F. Kong, J. Li, B. Jiang, H. Wang, and H. Song, "Integrated generative model for industrial anomaly detection via bi-directional LSTM and attention mechanism," *IEEE Transactions on Industrial Informatics*, pp. 1–10, 2021.

[35] E. Bechhoefer, "Envelope bearing analysis: Theory and practice," *IEEE Aerospace Conference Proceedings*, vol. 2005, pp. 3658–3666, 2005.

[36] Y. Wang, J. Xiang, R. Markert, and M. Liang, "Spectral kurtosis for fault detection, diagnosis and prognostics of rotating machines: A review with applications," *Mechanical Systems and Signal Processing*, vol. 66-67, pp. 679–698, 2016.

[37] L. R. Rabiner, "A tutorial on hidden Markov Models and selected applications in Speech Recognition," in *Readings in speech recognition*. Morgan Kaufmann, 1990, pp. 267–296.

[38] C. Boutilier, N. Friedman, M. Goldszmidt, and D. Koller, "Context-specific independence in Bayesian networks," in *Proceedings of the Twelfth International Conference on Uncertainty in Artificial Intelligence*, ser. UAI'96. Morgan Kaufmann, 1996, pp. 115—-123.

[39] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[40] N. Friedman, "The Bayesian structural EM algorithm," in *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, 1998, pp. 129–138.

[41] E. S. Page, "Continuous inspection schemes," *Biometrika*, vol. 41, no. 1/2, pp. 100–115, 1954.

[42] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques.* The MIT Press, 2009.

[43] P. Nectoux, R. Gouriveau, K. Medjaher, E. Ramasso, B. Morello, N. Zerhouni, and C. Varnier, "Pronostia: An experimental platform for bearings accelerated life test," *IEEE International Conference on Prognostics and Health Management*, 2012.

[44] R. G. Budynas, J. K. Nisbett, and J. E. Shigley, *Shigley's Mechanical engineering design.* McGraw-Hill, 2011.

Carlos Puerto-Santana received his bachelor's degree in Mathematics from Universidad de los Andes, Bogota, Colombia, in 2016 and his master's degree in Artificial Intelligence from Universidad Politécnica de Madrid, Spain, in 2018. He is currently a doctoral student at Universidad Politécnica de Madrid, Spain and researcher and developer in Aingura IIoT at San Sebastián, Spain.

Concha Bielza received her M.S. degree in Mathematics from Universidad Complutense de Madrid, in 1989 and her Ph.D. degree in Computer Science from Universidad Politécnica de Madrid, in 1996 (extraordinary doctorate award). She is currently (since 2010) a Full Professor of Statistics and Operations Research with the Departamento de Inteligencia Artificial, Universidad Politécnica de Madrid. Her research interests are primarily in the areas of probabilistic graphical models, decision analysis, metaheuristics for optimization, data mining, classification models, and real applications, such as biomedicine, bioinformatics, neuroscience and industry. She has published more than 150 papers in impact factor journals and has supervised 20 PhD theses. She was awarded the 2014 UPM Research Prize and the 2020 machine learning award of Amity University (India).

Javier Diaz-Rozo is Chief Technology Officer at Aingura IIoT, leading R&D and the industrial data analytics product development. He received his M.Eng. degree in Mechanical Engineering from the University of Los Andes, Bogotá, Colombia in 2001, his M.Sc. in Advanced Manufacturing Technologies and Productive Systems from the University of Manchester, UK in 2003 and his PhD in Artificial Intelligence (Cum Laude) from the Universidad Politécnica de Madrid, Spain, in 2019.

Guillem Ramirez-Gargallo graduated at Universitat Poliécnica de Catalunya in the Computer Architecture course. He is enrolled at the Barcelona Supercomputing Center as a junior research engineer since July of 2018 in the *Mobile and embedded-based HPC* group. During his stay, he has worked in the Mont-Blanc project improving the performance of the TensorFlow code on Marvell ThunderX2 and benchmarking it.

Filippo Mantovani is senior research responsible for the Mobile and embedded-based HPC group at the Barcelona Supercomputing Center (BSC). He graduated in Mathematics and held a Ph.D. in Computer Science from the University of Ferrara, Italy. He has been a scientific associate at the DESY laboratory in Zeuthen, Germany, and the University of Regensburg, Germany. He spent most of his scientific career in computational physics and high-performance computing.

Gaizka Virumbrales received his B.S. and M.S. in computer science from the University of Deusto, Bilbao (with highest honors). He joined Aingura IIoT as a research engineer in 2017. He has been involved in the development of accurate and reliable data acquisition systems since then. His main interest are optimization of algorithms for embedded systems and data mining.

Jesus Labarta is full professor on Computer Architecture at the Technical University of Catalonia (UPC) since 1990. Since 1981 he has been lecturing on computer architecture, operating systems, computer networks and performance evaluation. Since 2005 he is responsible of the Computer Science Research Department within the Barcelona Supercomputing Center (BSC). He has been involved in research cooperation with many leading companies on HPC related topics. His major directions of current work relate to performance analysis tools, programming models and resource management. He is involved in the development of the OmpSs programming model and its different implementations for SMP, GPUs and cluster platforms.

Pedro Larrañaga received the M.Sc. degree in Mathematics (Statistics) from the University of Valladolid and the Ph.D. degree in Computer Science from the University of the Basque Country (excellence award). He has been a Full Professor in Computer Science and Artificial Intelligence with the Universidad Politécnica de Madrid (UPM), since 2007. Before moving to UPM, his academic career developed at the University of the Basque Country (UPV-EHU) at several faculty ranks: Assistant Professor, from 1985 to 1998, Associate Professor, from 1998 to 2004, and a Full Professor, from 2004 to 2007. He has published over 200 papers in high-impact factor journals, He has supervised over 30 PhD theses. His research interests include the areas of probabilistic graphical models, metaheuristics for optimization, data mining, classification models, and real applications, such as biomedicine, bioinformatics, neuroscience, industry 4.0, and sports. He is Fellow of the European Association for Artificial Intelligence, since 2012, and a Fellow of the Academia Europaea, since 2018. He has received the 2013 Spanish National Prize in computer science and the prize of the Spanish Association for Artificial Intelligence in 2018 and the 2020 machine learning award of Amity University (India).