Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 278120, 23 pages
http://dx.doi.org/10.1155/2015/278120

Research Article

Hidden Semi-Markov Models for Predictive Maintenance

Francesco Cartella,[1] Jan Lemeire,[1] Luca Dimiccoli,[1] and Hichem Sahli[1,2]

[1] *Electronics and Informatics Department (ETRO), Vrije Universiteit Brussel (VUB), Plainlaan 2, 1050 Brussels, Belgium*
[2] *Interuniversity Microelectronics Center (IMEC), Kapeldreef 75, 3001 Leuven, Belgium*

Correspondence should be addressed to Francesco Cartella; fcartell@etro.vub.ac.be

Received 9 October 2014; Accepted 28 December 2014

Academic Editor: Hang Xu

Realistic predictive maintenance approaches are essential for condition monitoring and predictive maintenance of industrial machines. In this work, we propose Hidden Semi-Markov Models (HSMMs) with (i) no constraints on the state duration density function and (ii) being applied to continuous or discrete observation. To deal with such a type of HSMM, we also propose modifications to the learning, inference, and prediction algorithms. Finally, automatic model selection has been made possible using the Akaike Information Criterion. This paper describes the theoretical formalization of the model as well as several experiments performed on simulated and real data with the aim of methodology validation. In all performed experiments, the model is able to correctly estimate the current state and to effectively predict the time to a predefined event with a low overall average absolute error. As a consequence, its applicability to real world settings can be beneficial, especially where in real time the Remaining Useful Lifetime (RUL) of the machine is calculated.

1. Introduction

Predictive models that are able to estimate the current condition and the Remaining Useful Lifetime of an industrial equipment are of high interest, especially for manufacturing companies, which can optimize their maintenance strategies. If we consider that the costs derived from maintenance are one of the largest parts of the operational costs [1] and that often the maintenance and operations departments comprise about 30% of the manpower [2, 3], it is not difficult to estimate the economic advantages that such innovative techniques can bring to industry. Moreover, predictive maintenance, where in real time the Remaining Useful Lifetime (RUL) of the machine is calculated, has been proven to significantly outperforms other maintenance strategies, such as corrective maintenance [4]. In this work, RUL is defined as the time, from the current moment, that the systems will fail [5]. *Failure*, in this context, is defined as a deviation of the delivered output of a machine from the specified service requirements [6] that necessitate maintenance.

Models like Support Vector Machines [7], Dynamic Bayesian Networks [8], clustering techniques [9], and data mining approaches [10] have been successfully applied to condition monitoring, RUL estimation, and predictive maintenance problems [11, 12]. State space models, like Hidden Markov Models (HMMs) [13], are particularly suitable to be used in industrial applications, due to their ability to model the latent state which represents the health condition of the machine.

Classical HMMs have been applied to condition assessment [14, 15]; however, their usage in predictive maintenance has not been effective due to their intrinsic modeling of the state duration as a geometric distribution.

To overcome this drawback, a modified version of HMM, which takes into account an estimate of the duration in each state, has been proposed in the works of Tobon-Mejia et al. [16–19]. Thanks to the explicit state sojourn time modeling, it has been shown that it is possible to effectively estimate the RUL for industrial equipment. However, the drawback of their proposed HMM model is that the state duration is always assumed as Gaussian distributed and the duration parameters are estimated empirically from the Viterbi path of the HMM.

A complete specification of a duration model together with a set of learning and inference algorithms has been given firstly by Ferguson [20]. In his work, Ferguson allowed the underlying stochastic process of the state to be a semi-Markov chain, instead of a simple Markov chain of a HMM. Such model is referred to as Hidden Semi-Markov Model (HSMM) [21]. HSMMs and explicit duration modeles have been proven beneficial for many applications [22–25]. A complete overview of different duration model classes has been made by Yu [26]. Most state duration models, used in the literature, are nonparametric discrete distributions [27–29]. As a consequence, the number of parameters that describe the model and that have to be estimated is high, and consequently the learning procedure can be computationally expensive for real complex applications. Moreover, it is necessary to specify a priori the maximum duration allowed in each state.

To alleviate the high dimensionality of the parameter space, parametric duration models have been proposed. For example, Salfner [6] proposed a generic parametric continuous distribution to model the state sojourn time. However, in their model, the observation has been assumed to be discrete and applied to recognize failure-prone observation sequence. Using continuous observation, Azimi et al. [30–32] specified an HSMM with parametric duration distribution belonging to the Gamma family and modeled the observation process by a Gaussian.

Inspired by the latter two approaches, in this work we propose a generic specification of a parametric HSMM, in which no constraints are made on the model of the state duration and on the observation processes. In our approach, the state duration is modeled as a generic parametric density function. On the other hand, the observations can be modeled either as a discrete stochastic process or as continuous mixture of Gaussians. The latter has been shown to approximate, arbitrarily closely, any finite, continuous density function [33]. The proposed model can be generally used in a wide range of applications and types of data. Moreover, in this paper we introduce a new and more effective estimator of the time spent by the system in a determinate state prior to the current time. To the best of our knowledge, a part from the above referred works, the literature on HSMMs applied to prognosis and predictive maintenance for industrial machines is limited [34]. Hence, the present work aims to show the effectiveness of the proposed duration model in solving condition monitoring and RUL estimation problems.

Dealing with state space models, and in particular of HSMMs, one should define the number of states and correct family of duration density, and in case of continuous observations, the adequate number of Gaussian mixtures. Such parameters play a prominent role, since the right model configuration is essential to enable an accurate modeling of the dynamic pattern and the covariance structure of the observed time series. The estimation of a satisfactory model configuration is referred to as *model selection* in literature.

While several state-of-the-art approaches use expert knowledge to get insight on the model structure [15, 35, 36], an automated methodology for model selection is often required. In the literature, model selection has been deeply studied for a wide range of models. Among the existing methodologies, information based techniques have been extensively analyzed in literature with satisfactory results. Although Bayesian Information Criterion (BIC) is particularly appropriate to be used in finite mixture models [37, 38], Akaike Information Criterion (AIC) has been demonstrated to outperform BIC when applied to more complex models and when the sample size is limited [39, 40], which is the case of the target application of this paper.

In this work AIC is used to estimate the correct model configuration, with the final goal of an automated HSMMs model selection, which exploits only the information available in the input data. While model selection techniques have been extensively used in the framework of Hidden Markov Models [41–43], to the best of our knowledge, the present work is the first that proposes their appliance to duration models and in particular to HSMMs.

In summary, the present work contributes to condition monitoring, predictive maintenance, and RUL estimation problems by

(i) proposing a general Hidden Semi-Markov Model applicable for continuous or discrete observations and with no constraints on the density function used to model the state duration;

(ii) proposing a more effective estimator of the state duration variable $d_t(i)$, that is, the time spent by the system in the ith state, prior to current time t;

(iii) adapting the learning, inference and prediction algorithms considering the defined HSMM parameters and the proposed $d_t(i)$ estimator;

(iv) using the Akaike Information Criterion for automatic model selection.

The rest of the paper is organized as follows: in Section 2 we introduce the theory of the proposed HSMM together with its learning, inference, and prediction algorithms. Section 3 gives a short theoretical overview of the Akaike Information Criterion. Section 4 presents the methodology used to estimate the Remaining Useful Lifetime using the proposed HSMM. In Section 5 experimental results are discussed. The conclusion and future research directions are given in Section 6.

2. Hidden Semi-Markov Models

Hidden Semi-Markov Models (HSMMs) introduce the concept of variable duration, which results in a more accurate modeling power if the system being modeled shows a dependence on time.

In this section we give the specification of the proposed HSMM, for which we model the state duration with a parametric state-dependent distribution. Compared to nonparametric modeling, this approach has two main advantages:

(i) the model is specified by a limited number of parameters; as a consequence, the learning procedure is computationally less expensive;

(ii) the model does not require the a priori knowledge of the maximum sojourn time allowed in each state, being inherently learnt through the duration distribution parameters.

2.1. Model Specification. A Hidden Semi-Markov Model is a doubly embedded stochastic model with an underlying stochastic process that is not observable (hidden) but can only be observed through another set of stochastic processes that produce the sequence of observations. HSMM allows the underlying process to be a semi-Markov chain with a variable duration or sojourn time for each state. The key concept of HSMMs is that the semi-Markov property holds for this model: while in HMMs the Markov property implies that the value of the hidden state at time t depends exclusively on its value of time $t-1$, in HSMMs the probability of transition from state S_j to state S_i at time t depends on the duration spent in state S_j prior to time t.

In the following we denote the number of states in the model as N, the individual states as $S=\{S_1,\dots,S_N\}$, and the state at time t as s_t. The semi-Markov property can be written as

$$\begin{aligned}\mathbb{P}\left(s_{t+1}=S_i \mid s_t=S_j,\dots,s_1=S_k\right) \\ = \mathbb{P}\left(s_{t+1}=i \mid s_t=j, d_t(j)\right) \quad 1\le i,j,k\le N,\end{aligned} \tag{1}$$

where the duration variable $d_t(j)$ is defined as the time spent in state S_j prior to time t.

Although the state duration is inherently discrete, in many studies [44, 45] it has been modeled with a continuous parametric density function. Similar to the work of Azimi et al. [30–32], in this paper, we use the discrete counterpart of the chosen parametric probability density function (pdf). With this approximation, if we denote the pdf of the sojourn time in state S_i as $f(x,\boldsymbol{\theta}_i)$, where $\boldsymbol{\theta}_i$ represents the set of parameters of the pdf relative to the ith state, the probability that the system stays in state S_i for exactly d time steps can be calculated as $\int_{d-1}^{d} f(x,\boldsymbol{\theta}_i)dx$. Considering the HSMM formulation, we can generally denote the state dependent duration distributions by the set of their parameters relative to each state as $\Theta=\{\boldsymbol{\theta}_1,\dots,\boldsymbol{\theta}_N\}$.

Many related works on HSMMs [31, 32, 44, 45] consider $f(x,\boldsymbol{\theta}_i)$ within the exponential family. In particular, Gamma distributions are often used in speech processing applications. In this work, we do not impose a type of distribution function to model the duration. The only requirement is that the duration should be modeled as a positive function, being negative durations physically meaningless.

HSMMs require also the definition of a "dynamic" transition matrix, as a consequence of the semi-Markov property. Differently from the HMMs in which a constant transition probability leads to a geometric distributed state sojourn time, HSMMs explicitly define a transition matrix which, depending on the duration variable, has increasing probabilities of changing state as the time goes on. For convenience, we specify the state duration variable in a form of a vector $\mathbf{d}_t$ with dimensions $N\times 1$ as

$$\mathbf{d}_t = \begin{cases} d_t(j) & \text{if } s_t = S_j \\ 1 & \text{if } s_t \neq S_j. \end{cases} \tag{2}$$

The quantity $d_t(j)$ can be easily calculated by induction from $d_{t-1}(j)$ as

$$d_t(j) = s_t(j)\cdot s_{t-1}(j)\cdot d_{t-1}(j)+1, \tag{3}$$

where $s_t(j)$ is 1 if $s_t=S_j$, 0 otherwise.

If we assume that at time t the system is in state S_i, we can formally define the duration-dependent transition matrix as $\mathbf{A}_{\mathbf{d}_t}=[a_{ij}(\mathbf{d}_t)]$ with

$$a_{ij}(\mathbf{d}_t) = \mathbb{P}\left(s_{t+1}=S_j \mid s_t=S_i, d_t(i)\right) \quad 1\le i,j\le N. \tag{4}$$

The specification of the model can be further simplified by observing that, at each time t, the matrix $\mathbf{A}_{\mathbf{d}_t}$ can be decomposed in two terms: the recurrent and the nonrecurrent state transition probabilities.

The recurrent transition probabilities $\mathbf{P}(\mathbf{d}_t)=[p_{ij}(\mathbf{d}_t)]$, which depend only on the duration vector $\mathbf{d}_t$ and the parameters Θ, take into account the dynamics of the self-transition probabilities. It is defined as the probability of remaining in the current state at the next time step, given the duration spent in the current state prior to time t:

$$\begin{aligned} p_{ii}(\mathbf{d}_t) &= \mathbb{P}\left(s_{t+1}=S_i \mid s_t=S_i, d_t(i)\right) \\ &= \mathbb{P}\left(s_{t+1}=S_i \mid \right. \\ &\qquad \left. s_t=S_i, s_{t-1}=S_i,\dots,s_{t-d_t(i)+1}=S_i, s_{t-d_t(i)}\neq S_i\right) \\ &= \left(\mathbb{P}\left(s_{t+1}=S_i, s_t=S_i,\dots,s_{t-d_t(i)+2}=S_i \mid \right.\right. \\ &\qquad \left.\left. s_{t-d_t(i)+1}=S_i, s_{t-d_t(i)}\neq S_i\right)\right) \\ &\quad \cdot \left(\mathbb{P}\left(s_t=S_i, s_{t-1}=S_i,\dots,s_{t-d_t(i)+2}=S_i \mid \right.\right. \\ &\qquad \left.\left. s_{t-d_t(i)+1}=S_i, s_{t-d_t(i)}\neq S_i\right)\right)^{-1}. \end{aligned} \tag{5}$$

The denominator in (5) can be expressed as $\sum_{k=1}^{\infty}\mathbb{P}(s_{t+k}\neq S_i, s_{t+k-1}=S_i,\dots,s_{t-d_t(i)+2}=S_i \mid s_{t-d_t(i)+1}=S_i, s_{t-d_t(i)}\neq S_i)$, which is the probability that the system, at time t, has been staying in state S_i for at least $d_t(i)-1$ time units. The above expression is equivalent to $1-F(d_t(i)-1,\boldsymbol{\theta}_i)$, where $F(\cdot,\boldsymbol{\theta}_i)$ is the duration cumulative distribution function relative to the the state S_i, that is, $F(d,\boldsymbol{\theta})=\int_{-\infty}^{d} f(x,\boldsymbol{\theta})dx$. As a consequence, from (5) we can define the recurrent transition probabilities as a diagonal matrix with dimensions $N\times N$, as

$$\mathbf{P}(\mathbf{d}_t) = \left[p_{ij}(\mathbf{d}_t)\right] = \begin{cases} \dfrac{1-F(d_t(i),\boldsymbol{\theta}_i)}{1-F(d_t(i)-1,\boldsymbol{\theta}_i)} & \text{if } i=j \\ 0 & \text{if } i\neq j. \end{cases} \tag{6}$$

The usage of the cumulative functions in (6), which tend to 1 as the duration tends to infinity, suggests that the probability of self-transition tends to decrease as the sojourn time increases, leading the model to always leave the current state if time approaches infinity.

The nonrecurrent state transition probabilities, $\mathbf{A}^0=[a_{ij}^0]$, rule the transitions between two different states. It is represented by a $N \times N$ matrix with the diagonal elements equal to zero, defined as

$$\mathbf{A}^0 = \left[a_{ij}^0\right] = \begin{cases} 0 & \text{if } i = j \\ \mathbb{P}\left(s_{t+1} = S_j \mid s_t = S_i\right) & \text{if } i \neq j. \end{cases} \quad (7)$$

$\mathbf{A}^0$ must be specified as a stochastic matrix; that is, its elements have to satisfy the constraint $\sum_{j=1}^{N} a_{ij}^0 = 1$ for all i.

As a consequence of the above decomposition, the dynamic of the underlying semi-Markov chain can be defined by specifying only the state-dependent duration parameters Θ and the nonrecurrent matrix $\mathbf{A}^0$, since the model transition matrix can be calculated, at each time t, using (6) and (7):

$$\mathbf{A}_{\mathbf{d}_t} = \mathbf{P}\left(\mathbf{d}_t\right) + \left(\mathbf{I} - \mathbf{P}\left(\mathbf{d}_t\right)\right)\mathbf{A}^0, \quad (8)$$

where $\mathbf{I}$ is the identity matrix. If we denote the elements of the dynamic transition matrix $\mathbf{A}_{\mathbf{d}_t}$ as $a_{ij}(\mathbf{d}_t)$, the stochastic constraint $\sum_{j=1}^{N} a_{ij}(\mathbf{d}_t) = 1$ for all i and t is guaranteed from the fact that $\mathbf{P}(\mathbf{d}_t)$ is a diagonal matrix and $\mathbf{A}^0$ is a stochastic matrix.

For several applications it is necessary to model the *absorbing state* which, in the case of industrial equipment, corresponds to the "broken" or "failure" state. If we denote the absorbing state as S_k with $k \in [1, N]$, we must fix the kth row of the nonrecurrent matrix $\mathbf{A}^0$ to be $a_{kk}^0 = 1$ and $a_{ki}^0 = 0$ for all $1 \leq i \leq N$ with $i \neq k$. By substituting such $\mathbf{A}^0$ matrix in (8), it is easy to show that the element $a_{kk}(\mathbf{d}_t) = 1$ and remains constant for all t, while the duration probability parameters $\boldsymbol{\theta}_k$ are not influent for the absorbing state S_k. An example of absorbing state specification will be given in Section 5.

With respect to the input observation signals, in this work we consider both continuous and discrete data, by adapting the suitable observation model depending on the observation nature. In particular, for the continuous case, we model the observations with a multivariate mixture of Gaussians distributions. This choice presents two main advantages: (i) a multivariate model allows to deal with multiple observations at the same time; this is often the case of industrial equipments modeling since, at each time, multiple sensors' measurements are available, and (ii) mixture of Gaussians has been proved to closely approximate any finite and continuous density function [33]. Formally, if we denote by $\mathbf{x}_t$ the observation vector at time t and the generic observation vector being modeled as $\mathbf{x}$, the observation density for the jth state is represented by a finite mixture of M gaussians

$$b_j(\mathbf{x}) = \sum_{m=1}^{M} c_{jm} \mathbb{N}\left(\mathbf{x}, \boldsymbol{\mu}_{jm}, \mathbf{U}_{jm}\right), \quad 1 \leq j \leq N, \quad (9)$$

where c_{jm} is the mixture coefficient for the mth mixture in state S_j, which satisfies the stochastic constraint $\sum_{m=1}^{M} c_{jm} = 1$ for $1 \leq j \leq N$ and $c_{jm} \geq 0$ for $1 \leq j \leq N$ and $1 \leq m \leq M$, while $\mathbb{N}$ is the Gaussian density, with mean vector $\boldsymbol{\mu}_{jm}$ and covariance matrix $\mathbf{U}_{jm}$ for the mth mixture component in state j.

In case of discrete data, we model the observations within each state with a nonparametric discrete probability distribution. In particular, if L is the number of distinct observation symbols per state and if we denote the symbols as $X = \{X_1, \ldots, X_L\}$ and the observation at time t as x_t, the observation symbol probability distribution can be defined as a matrix $B = [b_j(l)]$ of dimensions $N \times L$ where

$$b_j(l) = \mathbb{P}\left[x_t = X_l \mid s_t = S_j\right] \quad 1 \leq j \leq N; \ 1 \leq l \leq L. \quad (10)$$

Since the system in each state at each time step can emit one of the possible L symbols, the matrix B is stochastic; that is, it is constrained to $\sum_{l=1}^{L} b_j(l) = 1$ for all $1 \leq j \leq N$.

Finally, as in the case of HMMs, we specify the initial state distribution $\pi = \{\pi_i\}$ which defines the probability of the starting state as

$$\pi_i = \mathbb{P}\left[s_1 = S_i\right], \quad 1 \leq i \leq N. \quad (11)$$

From the above considerations, two different HSMM models can be considered. In the case of continuous observation, $\lambda = (\mathbf{A}^0, \Theta, C, \mu, U, \pi)$, and in the case of discrete observation the HSMM is characterized by $\lambda = (\mathbf{A}^0, \Theta, B, \pi)$. An example of continuous HSMM with 3 states is shown in Figure 1.

2.2. Learning and Inference Algorithms. Let us denote the generic sequence of observations, being indiscriminately continuous vectors or discrete symbols, as $\mathbf{x} = \mathbf{x}_1\mathbf{x}_2 \cdots \mathbf{x}_T$; in order to use the defined HSMM model in practice, similarly to the HMM, we need to solve three basic problems.

(1) Given the observation $\mathbf{x}$ and a model λ, calculate the probability that the sequence $\mathbf{x}$ has been generated by the model λ, that is, $\mathbb{P}(\mathbf{x} \mid \lambda)$.

(2) Given the observation $\mathbf{x}$ and a model λ, calculate the state sequence $S = s_1 s_2 \cdots s_T$ which have most probably generated the sequence $\mathbf{x}$.

(3) Given the observation $\mathbf{x}$ find the parameters of the model λ which maximize $\mathbb{P}(\mathbf{x} \mid \lambda)$.

As in case of HMM, solving the above problems requires using the forward-backward [13], decoding (Viterbi [46] and Forney [47]) and Expectation Maximization [48] algorithms, which will be adapted to the HSMM introduced in Section 2.1. In the following, we also propose a more effective estimator of the state duration variable $d_t(i)$ defined in (2).

2.2.1. The Forward-Backward Algorithm. Given a generic sequence of observations $\mathbf{x} = \mathbf{x}_1\mathbf{x}_2 \cdots \mathbf{x}_T$, the goal is to calculate the model likelihood, that is, $\mathbb{P}(\mathbf{x} \mid \lambda)$. This quantity is useful for the training procedure, where the parameters that locally maximize the model likelihood are chosen, as well as for classification problems. The latter is the case in which the observation sequence $\mathbf{x}$ has to be mapped to one of a finite set of C classes, represented by a set of HSMM parameters $L = \{\lambda_1, \ldots, \lambda_C\}$. The class of $\mathbf{x}$ is chosen such that $\lambda(\mathbf{x}) = \arg\max_{\lambda \in L} \mathbb{P}(X \mid \lambda)$.

To calculate the model likelihood, we first define the forward variable at each time t as

$$\alpha_t(i) = \mathbb{P}\left(\mathbf{x}_1\mathbf{x}_2 \cdots \mathbf{x}_t, s_t = S_i \mid \lambda\right) \quad 1 \leq i \leq N. \quad (12)$$

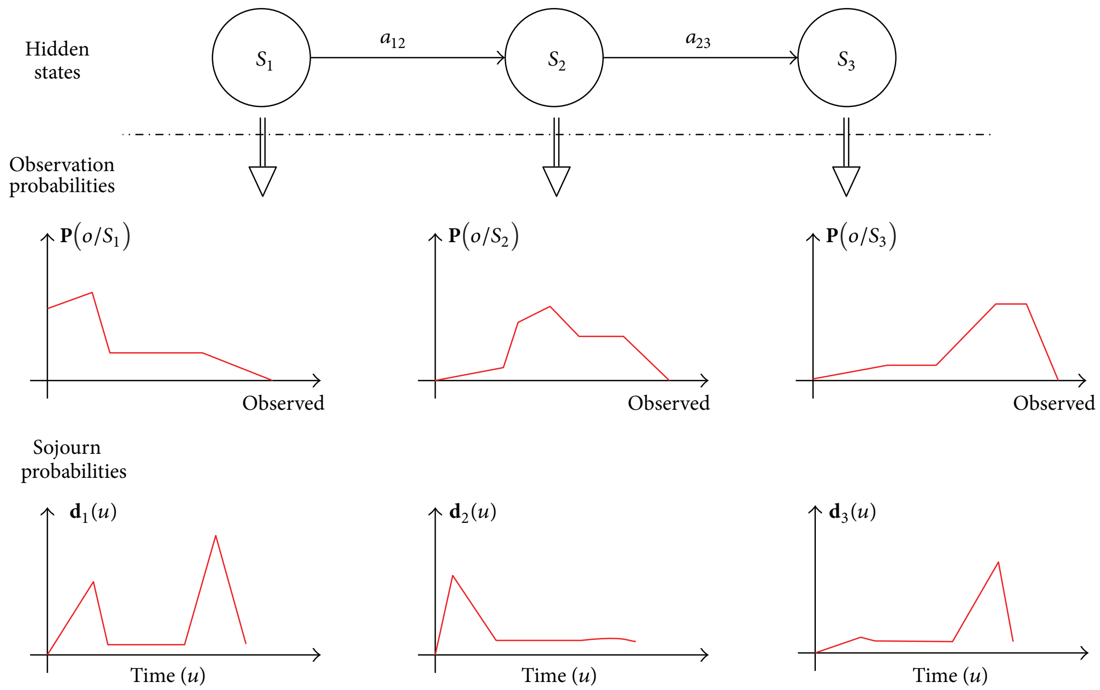


Figure 1: Graphical representation of an HSMM.

Contrarily to HMMs, for HSMMs the state duration must be taken into account in the the forward variable calculation. Consequently, Yu [26] proposed the following inductive formula:

$$\alpha_t(j,d) = \sum_{d'=1}^{D}\sum_{i=1}^{N}\left(\alpha_{t-d'}(i,d')\, a_{ij}^{0} p_{jj}(d') \prod_{k=t-d+1}^{t} b_j(\mathbf{x}_k)\right) \qquad 1 \le j \le N, \quad 1 \le t \le T \tag{13}$$

that is, the sum of the probabilities of being in the current state S_j (at time t) for the past d' time units, (with $1 \le d' \le D$ and D the maximum allowed duration for each state), coming from all the possible previous states S_i, $1 \le i \le N$, and $i \ne j$.

The disadvantage of the above formulation is that, as discussed in Introduction, the specification of the maximum duration D represents a limitation to the modeling generalization. Moreover, from (13), it is clear that the computation and memory complexities drastically increase with D, which can be very large in many applications, in particular for online failure prediction.

To alleviate this problem, Azimi et al. [30–32] introduced a new forward algorithm for HSMMs that, by keeping track of the estimated average state duration at each iteration, has a computational complexity comparable to the forward algorithm for HMMs [13]. However, the average state duration represents an approximation. Consequently, the forward algorithm of Azimi, compared with (13), pays the price of a lower precision in favor of a (indispensable) better computational efficiency.

To calculate the forward variable $\alpha_t(j)$ using Azimi's approach, the duration-dependent transition matrix, defined in (8), is taken in consideration in the induction formula of (13), which becomes [30]

$$\alpha_t(j) = \left[\sum_{i=1}^{N} \alpha_{t-1}(i)\, a_{ij}(\hat{\mathbf{d}}_{t-1})\right] b_j(\mathbf{x}_t). \tag{14}$$

To calculate the above formula, the average state duration of (2) must be estimated, for each time t, by means of the variable $\hat{\mathbf{d}}_t = [\hat{d}_t(i)]$, defined as

$$\hat{d}_t(i) = \mathbb{E}(d_t(i) \mid \mathbf{x}_1\mathbf{x}_2\cdots\mathbf{x}_t, s_t = S_i, \lambda) \qquad 1 \le i \le N, \tag{15}$$

where $\mathbb{E}$ denotes the expected value. To calculate the above quantity, Azimi et al. [30–32] use the following formula:

$$\hat{\mathbf{d}}_t = \gamma_{t-1} \odot \hat{\mathbf{d}}_{t-1} + 1, \tag{16}$$

where $\odot$ represents the element by element product between two matrices/vectors and the vector $\boldsymbol{\gamma}_\mathbf{t} = [\gamma_t(i)]$ (the probability of being in state S_i at time t given the observation sequence and the model parameters) with dimensions $N \times 1$ is calculated in terms of $\alpha_t(i)$ as

$$\gamma_t(i) = \mathbb{P}(s_t = S_i \mid \mathbf{x}_1\mathbf{x}_2\cdots\mathbf{x}_t, \lambda) = \frac{\alpha_t(i)}{\sum_{j=1}^{N}\alpha_t(j)} \qquad 1 \le i \le N. \tag{17}$$

Equation (16) is based on the following induction formula [30–32] that rules the dynamics of the duration vector when the system's state is known:

$$d_t(i) = s_{t-1}(i) \cdot d_{t-1}(i) + 1, \tag{18}$$

where, for each t, $s_t(i)$ is 1 if $s_t = S_i$, 0 otherwise.

A simple example shows that (18) is incorrect: assuming an HSMM with three states and considering the state sequence $(S_1, S_1, S_2, \ldots)$ the correct sequence of the duration vector is $\mathbf{d}_1 = [1,1,1]^T$, $\mathbf{d}_2 = [2,1,1]^T$, and $\mathbf{d}_3 = [1,1,1]^T$, where the superscript ^{T} denotes vector transpose. If we apply (18), we obtain $\mathbf{d}_1 = [1,1,1]^T$, $\mathbf{d}_2 = [2,1,1]^T$, and $\mathbf{d}_3 = [1,2,1]^T$, which is in contradiction with the definition of the state duration vector given in (2).

To calculate the average state duration variable $\widehat{d}_t(i)$ we propose a new induction formula that estimates, for each time t, the time spent in the ith state prior to t as

$$\begin{aligned} \widehat{d}_t(i) &= \mathbb{P}\left(s_{t-1} = S_i \mid s_t = S_i, \mathbf{x}_1, \ldots, \mathbf{x}_t\right) \cdot \left(\widehat{d}_{t-1}(i) + 1\right) && (19) \\ &= \frac{a_{ii}\left(\widehat{\mathbf{d}}_{t-1}\right) \cdot \alpha_{t-1}(i) \cdot b_i\left(\mathbf{x}_t\right)}{\alpha_t(i)} \cdot \left(\widehat{d}_{t-1}(i) + 1\right), && (20) \\ & \qquad 1 \le i \le N. \end{aligned}$$

The derivation of (20) is given in Appendix. The intuition behind (19) is that the current average duration is the previous average duration plus one, weighted with the "amount" of the current state that was already in state S_i in the previous step.

Using the proposed (20), the forward algorithm can be specified as follows:

(1) initialization, with $1 \le i \le N$:

$$\begin{gathered} \alpha_1(i) = \pi_i b_i\left(\mathbf{x}_1\right), \\ \widehat{d}_1(i) = 1, \\ \mathbf{A}_{\widehat{\mathbf{d}}_1} = \mathbf{P}\left(\widehat{\mathbf{d}}_1\right) + \left(\mathbf{I} - \mathbf{P}\left(\widehat{\mathbf{d}}_1\right)\right)\mathbf{A}^0, \end{gathered} \tag{21}$$

where $\mathbf{P}(\widehat{\mathbf{d}}_i)$ is estimated using (6);

(2) induction, with $1 \le j \le N$ and $1 \le t \le T-1$:

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i)\, a_{ij}\left(\widehat{\mathbf{d}}_t\right)\right] b_j\left(\mathbf{x}_{t+1}\right), \tag{22}$$

$$\widehat{d}_{t+1}(i) = \frac{a_{ii}\left(\widehat{\mathbf{d}}_t\right) \cdot \alpha_t(i) \cdot b_i\left(\mathbf{x}_{t+1}\right)}{\alpha_{t+1}(i)} \cdot \left(\widehat{d}_t(i) + 1\right), \tag{23}$$

$$\mathbf{A}_{\widehat{\mathbf{d}}_{t+1}} = \mathbf{P}\left(\widehat{\mathbf{d}}_{t+1}\right) + \left(\mathbf{I} - \mathbf{P}\left(\widehat{\mathbf{d}}_{t+1}\right)\right)\mathbf{A}^0, \tag{24}$$

where $a_{ij}(\widehat{\mathbf{d}}_t)$ are the coefficients of the matrix $\mathbf{A}_{\widehat{\mathbf{d}}_t}$;

(3) termination:

$$\mathbb{P}(\mathbf{x} \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i). \tag{25}$$

Similar considerations as the forward procedure can be made for the backward algorithm, which is implemented by defining the variable $\beta_t(i)$ as

$$\beta_t(i) = \mathbb{P}\left(\mathbf{x}_{t+1}\mathbf{x}_{t+2}\cdots\mathbf{x}_T \mid s_t = S_i, \lambda\right) \quad 1 \le i \le N. \tag{26}$$

Having estimated the dynamic transition matrix $\mathbf{A}_{\widehat{\mathbf{d}}_t}$ for each $1 \le t \le T$ using (24), the backward variable can be calculated inductively as follows.

(1) Initialization:

$$\beta_T(i) = 1, \quad 1 \le i \le N. \tag{27}$$

(2) Induction:

$$\begin{gathered} \beta_t(i) = \sum_{j=1}^{N} a_{ij}\left(\widehat{\mathbf{d}}_t\right) b_j\left(\mathbf{x}_{t+1}\right) \beta_{t+1}(j), \\ t = T-1, T-2, \ldots, 1, \quad 1 \le i \le N. \end{gathered} \tag{28}$$

Although the variable $\beta_t(i)$ is not necessary for the calculation of the model likelihood, it will be useful in the parameter reestimation procedure, as it will be explained in Section 2.2.3.

2.2.2. The Viterbi Algorithm. The Viterbi algorithm [46, 47] (also known as *decoding*) allows determining the best state sequence corresponding to a given observation sequence.

Formally, given a sequence of observation $\mathbf{x} = \mathbf{x}_1\mathbf{x}_2\cdots\mathbf{x}_T$, the best state sequence $S^* = s_1^* s_2^* \cdots s_T^*$ corresponding to $\mathbf{x}$ is calculated by defining the variable $\delta_t(i)$ as

$$\delta_t(i) = \max_{s_1, s_2, \ldots, s_{t-1}} \mathbb{P}\left(s_1 s_2, \ldots, s_t = S_i, \mathbf{x}_1\mathbf{x}_2\cdots\mathbf{x}_t \mid \lambda\right). \tag{29}$$

The procedure to recursively calculate the variable $\delta_t(i)$ and to retrieve the target state sequence (i.e., the arguments which maximize the $\delta_t(i)$'s) for the proposed HSMM is a straightforward extension of the Viterbi algorithm for HMMs [13]. The only change is the usage, in the recursive calculation of $\delta_t(i)$, of the dynamic transition matrix $\mathbf{A}_{\widehat{\mathbf{d}}_t} = [a_{ij}(\widehat{\mathbf{d}}_t)]$, calculated through (24). The Viterbi algorithm for the introduced parametric HSMMs can be summarized as follows:

(1) initialization, with $1 \le i \le N$:

$$\begin{gathered} \delta_1(i) = \pi_i b_i\left(\mathbf{x}_1\right), \\ \psi_1(i) = 0, \end{gathered} \tag{30}$$

(2) recursion, with $1 \le j \le N$ and $2 \le t \le T$:

$$\delta_t(j) = \max_{1 \le i \le N}\left[\delta_{t-1}(i)\, a_{ij}\left(\widehat{\mathbf{d}}_t\right)\right] b_j\left(\mathbf{x}_t\right), \tag{31}$$

$$\psi_t(j) = \arg\max_{1 \le i \le N}\left[\delta_{t-1}(i)\, a_{ij}\left(\widehat{\mathbf{d}}_t\right)\right], \tag{32}$$

(3) termination:

$$P^* = \max_{1 \le i \le N}\left[\delta_T(i)\right], \tag{33}$$

$$s_T^* = \arg\max_{1 \le i \le N}\left[\delta_T(i)\right], \tag{34}$$

where we keep track of the argument maximizing (31) using the vector ψ_t, which, tracked back, gives the desired best state sequence:

$$s_t^* = \psi_{t+1}\left(s_{t+1}^*\right), \quad t = T-1, T-2, \ldots, 1. \tag{35}$$

2.2.3. The Training Algorithm. The training algorithm consists of estimating the model parameters from the observation data. As discussed in Section 2.1, a parametric HSMM is defined by $\lambda = (\mathbf{A}^0, \Theta, C, \mu, U, \pi)$ if the observations are continuous, or $\lambda = (\mathbf{A}^0, \Theta, B, \pi)$ if the observations are discrete. Given a generic observation sequence $\mathbf{x} = \mathbf{x}_1\mathbf{x}_2 \cdots \mathbf{x}_T$, referred to as *training set* in the following, the training procedure consists of finding the model parameter set λ^* which locally maximizes the model likelihood $\mathbb{P}(\mathbf{x} \mid \lambda)$.

We use the modified Baum-Welch algorithm of Azimi et al. [30–32]. However, in our implementation we do not make assumption on the density function used to model the state duration, and we consider both continuous and discrete observations.

Being a variant of the more general Expectation-Maximization (EM) algorithm, Baum-Welch is an iterative procedure which consists of two steps: (i) the expectation step, in which the forward and backward variables are calculated and the model likelihood is estimated and (ii) the maximization step, in which the model parameters are updated and used in the next iteration. This process usually starts from a random guess of the model parameters λ^0 and it is iterated until the likelihood function does not improve between two consecutive iterations.

Similarly to HMMs, the reestimation formulas are derived by firstly introducing the variable $\xi_t(i, j)$, which represents the probability of being in state S_i at time t, and in state S_j at time $t + 1$, given the model and the observation sequence, as

$$\xi_t(i, j) = \mathbb{P}\left(s_t = S_i, s_{t+1} = S_j \mid \mathbf{x}, \lambda\right). \tag{36}$$

However, in the HSMM case, the variable $\xi_t(i, j)$ considers the duration estimation performed in the forward algorithm (see Equation (24)). Formulated in terms of the forward and backward variables, it is given by

$$\begin{aligned}
\xi_t(i, j) &= \mathbb{P}\left(s_t = S_i, s_{t+1} = S_j \mid \mathbf{x}, \lambda\right) \\
&= \frac{\mathbb{P}\left(s_t = S_i, s_{t+1} = S_j, \mathbf{x} \mid \lambda\right)}{\mathbb{P}(\mathbf{x} \mid \lambda)} \\
&= \frac{\alpha_t(i)\, a_{ij}\left(\widehat{\mathbf{d}}_t\right) b_j\left(\mathbf{x}_{t+1}\right) \beta_{t+1}(j)}{\mathbb{P}(\mathbf{x} \mid \lambda)} \\
&= \frac{\alpha_t(i)\, a_{ij}\left(\widehat{\mathbf{d}}_t\right) b_j\left(\mathbf{x}_{t+1}\right) \beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_t(i)\, a_{ij}\left(\widehat{\mathbf{d}}_t\right) b_j\left(\mathbf{x}_{t+1}\right) \beta_{t+1}(j)}.
\end{aligned} \tag{37}$$

From $\xi_t(i, j)$ we can derive the quantity $\gamma_t(i)$ (already defined in (17)) representing the probability of being in state S_i at time t given the observation sequence and the model parameters:

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i, j). \tag{38}$$

Finally, the the reestimation formulas for the parameters π and $\mathbf{A}^0$ are given by

$$\overline{\pi}_i = \gamma_1(i), \tag{39}$$

$$\overline{a}_{ij}^0 = \frac{\left(\sum_{t=1}^{T-1} \xi_t(i, j)\right) \odot G}{\sum_{j=1}^{N}\left(\sum_{t=1}^{T-1} \xi_t(i, j)\right) \odot G}, \tag{40}$$

where $G = [g_{ij}]$ is a square matrix of dimensions $N \times N$ where $g_{ij} = 0$ for $i = j$ and $g_{ij} = 1$ for $i \neq j$, $\odot$ represents the element by element product between two matrices, $\sum_{t=1}^{T-1} \gamma_t(i)$ is the expected number of transitions from state S_i, and $\sum_{t=1}^{T-1} \xi_t(i, j)$ is the expected number of transitions from state S_i to state S_j.

Equation (39) represents the expected number of times that the model starts in state S_i, while (40) represents the expected number of transitions from state S_i to state S_j with $i \neq j$ over the total expected number of transitions from state S_i to any other state different from S_i.

For the matrix $\overline{\mathbf{A}}^0$, being normalized, the stochastic constraints are satisfied at each iteration, that is, $\sum_{j=1}^{N} \overline{a}_{ij}^0 = 1$ for each $1 \leq i \leq N$, while the estimation of the prior probability $\overline{\pi}_i$ inherently sums up to 1 at each iteration, since it represents the expected frequency in state S_i at time $t = 1$ for each $1 \leq i \leq N$.

With respect to the reestimation of the state duration parameters Θ, firstly, we estimate the mean $\mu_{i,d}$ and the variance $\sigma_{i,d}^2$ of the ith state duration for each $1 \leq i \leq N$, from the forward and backward variables and the estimation of the state duration variable

$$\mu_{i,d} = \frac{\sum_{t=1}^{T-1} \alpha_t(i)\left(\sum_{j=1, j\neq i}^{N} a_{ij}\left(\widehat{d}_t(i)\right) b_j\left(\mathbf{x}_{t+1}\right) \beta_{t+1}(j)\right) \widehat{d}_t(i)}{\sum_{t=1}^{T-1} \alpha_t(i)\left(\sum_{j=1, j\neq i}^{N} a_{ij}\left(\widehat{d}_t(i)\right) b_j\left(\mathbf{x}_{t+1}\right) \beta_{t+1}(j)\right)}, \tag{41}$$

$$\begin{aligned}
\sigma_{i,d}^2 = &\left(\sum_{t=1}^{T-1} \alpha_t(i)\left(\sum_{j=1, j\neq i}^{N} a_{ij}\left(\widehat{d}_t(i)\right) b_j\left(\mathbf{x}_{t+1}\right) \beta_{t+1}(j)\right)\right. \\
&\left.\cdot\left(\widehat{d}_t(i) - \mu_{i,d}\right)^2\right) \\
&\cdot\left(\sum_{t=1}^{T-1} \alpha_t(i)\left(\sum_{j=1, j\neq i}^{N} a_{ij}\left(\widehat{d}_t(i)\right) b_j\left(\mathbf{x}_{t+1}\right) \beta_{t+1}(j)\right)\right)^{-1},
\end{aligned} \tag{42}$$

where (41) can be interpreted as the probability of transition from state S_i to S_j with $i \neq j$ at time t weighted by the duration of state S_i at t, giving the desired expected value, while in (42) the same quantity is weighted by the squared distance of the duration at time t from its mean, giving the estimation of the variance.

Then, the parameters of the desired duration distribution can be estimated from $\mu_{i,d}$ and $\sigma_{i,d}^2$. For example, if a Gamma distribution with shape parameter ν and scale parameter η is chosen to model the state duration, the parameters ν_i and η_i for each $1 \leq i \leq N$ can be calculated as $\nu_i = \mu_{i,d}^2/\sigma_{i,d}^2$ and $\eta_i = \sigma_{i,d}^2/\mu_{i,d}$.

Concerning the observation parameters, once the modified forward and backward variables, accounting for the state duration, are defined as in (22) and (28), the reestimation formulas are the same as for Hidden Markov Models [13].

In particular, for continuous observations, the parameters of the Gaussians' mixture defined in (9) are reestimated by firstly defining the probability of being in state S_j at time t with the probability of the observation vector $\mathbf{x}_t$ evaluated by the kth mixture component, as

$$\gamma_t(j,k) = \left[\frac{\alpha_t(j)\,\beta_t(j)}{\sum_{j=1}^{N}\alpha_t(j)\,\beta_t(j)}\right] \cdot \left[\frac{c_{jk}\mathbb{N}\left(\mathbf{x}_t, \boldsymbol{\mu}_{jk}, \mathbf{U}_{jk}\right)}{\sum_{m=1}^{M} c_{jm}\mathbb{N}\left(\mathbf{x}_t, \boldsymbol{\mu}_{jm}, \mathbf{U}_{jm}\right)}\right]. \tag{43}$$

By using the former quantity, the parameters c_{jk}, $\boldsymbol{\mu}_{jk}$, and $\mathbf{U}_{jk}$ are reestimated through the following formulas:

$$\begin{aligned} \bar{c}_{jk} &= \frac{\sum_{t=1}^{T}\gamma_t(j,k)}{\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_t(j,m)}, \\ \bar{\boldsymbol{\mu}}_{jk} &= \frac{\sum_{t=1}^{T}\gamma_t(j,k)\cdot\mathbf{x}_t}{\sum_{t=1}^{T}\gamma_t(j,k)}, \\ \overline{\mathbf{U}}_{jk} &= \frac{\sum_{t=1}^{T}\gamma_t(j,k)\cdot\left(\mathbf{x}_t-\boldsymbol{\mu}_{jk}\right)\left(\mathbf{x}_t-\boldsymbol{\mu}_{jk}\right)^{T}}{\sum_{t=1}^{T}\gamma_t(j,k)}, \end{aligned} \tag{44}$$

where superscript ^{T} denotes vector transpose.

For discrete observations, the reestimation formula for the observation matrix $b_j(l)$ is

$$\bar{b}_j(l) = \frac{\sum_{t=1,\,\text{with } x_t=X_l}^{T}\gamma_t(j)}{\sum_{t=1}^{T}\gamma_t(j)}, \tag{45}$$

where the quantity $\gamma_t(j)$, which takes into account the duration dependent forward variable $\alpha_t(j)$, is calculated through (17).

The reader is referred to Rabiner's work [13] for the interpretation on the observation parameters reestimation formulas.

3. AIC-Based Model Selection

In the framework of the proposed parametric HSMMs, the model selection procedure aims to select the optimal number of hidden states N, the right duration distribution family, and, in the case of mixture observation modeling, the number of Gaussian mixtures M to be used. In this work, we make use of the Akaike Information Criterion (AIC). Indeed, it has been seen that in case of complex models and in presence of a limited number of training observations, AIC represents a satisfactory methodology for model selection, outperforming other approaches like Bayesian Information Criterion.

In general, information criteria are represented as a two-term structure. They account for a compromise between a measure of model fitness, which is based on the likelihood of the model, and a penalty term which takes into account the model complexity. Usually, the model complexity is measured in terms of the number of parameters that have to be estimated and in terms of the number of observations.

The Akaike Information Criterion is an estimate of the asymptotic value of the expected distance between the unknown true likelihood function of the data and the fitted likelihood function of the model. In particular, the AIC can be expressed as

$$\text{AIC} = \frac{-\log L\left(\widehat{\lambda}\right) + p}{T}, \tag{46}$$

where $L(\widehat{\lambda})$ is the likelihood of the model with the estimated parameters $\widehat{\lambda}$, as defined in (25), p is the number of model parameters, and T is the length of the observed sequence. The best model is the one minimizing equation (46).

Concerning p, the number of parameters to be estimated for a parametric HSMM with N states is $p = p_h + p_o$, where p_h are the parameters of the hidden states layer, while p_o are those of the observation layer.

In particular $p_h = (N-1) + (N-1)\cdot N + z\cdot N$ where

(i) $N-1$ accounts for the prior probabilities π;

(ii) $(N-1)\cdot N$ accounts for the nonrecurrent transition matrix $\mathbf{A}^0$;

(iii) $z\cdot N$ accounts for the duration probability, being z the number of parameters $\boldsymbol{\theta}$ of the duration distribution.

Concerning p_o, a distinction must be made between discrete and continuous observations:

(i) in the case of discrete observations with L possible observable values, $p_o = (L-1)\cdot N$, which accounts for the elements of the observation matrix B;

(ii) if the observations are continuous and a multivariate mixture of M Gaussians with O variates is used as observation model, $p_o = [O\cdot N\cdot M] + [O\cdot O\cdot N\cdot M] + [(M-1)\cdot N]$ where each term accounts, respectively, for the mean vector μ, the covariance matrix U, and the mixture coefficients C.

4. Remaining Useful Lifetime Estimation

One of the most important advantages of the time modeling of HSMMs is the possibility to effectively face the prediction problem. The knowledge of the state duration distributions allows the estimation of the remaining time in a certain state and, in general, the prediction of the expected time $\widetilde{D}$ before entering in a determinate state.

As already mentioned, an interesting application of the prediction problem is the Remaining Useful Lifetime (RUL) estimation of industrial equipments. Indeed, if each state of an HSMM is mapped to a different condition of an industrial machine and if the state S_k that represents the *failure* condition is identified, at each moment, the RUL can be defined as the expected time $\widetilde{D}$ to reach the failure state

S_k. If we assume that the time to failure is a random variable D following a determinate probability density, we define the RUL at the current time t as

$$\mathrm{RUL}_t = \widetilde{D} = \mathbb{E}(D) : s_{t+\widetilde{D}} = S_k, \qquad s_{t+\widetilde{D}-1} = S_i \\ 1 \leq i, k \leq N, \quad i \neq k, \tag{47}$$

where $\mathbb{E}$ denotes the expected value.

Having fixed the *failure* state, the estimation of the RUL is performed, in two steps, every time a new observation is acquired (online):

(1) estimation of the current state;

(2) projection of the future state transitions until the failure state is reached and estimation of the expected sojourn time.

The estimation of the current state is performed via the Viterbi path, that is, the variable $\boldsymbol{\delta}_t = [\delta_t(i)]_{1\leq i\leq N}$ defined in (29). To correctly model the uncertainty of the current state estimation, we use the normalized variable $\bar{\delta}_t(i)$ obtained as

$$\bar{\delta}_t(i) = \max_{s_1, s_2, \ldots, s_{t-1}} \mathbb{P}(s_t = S_i \mid s_1 s_2 \cdots s_{t-1}, \mathbf{x}_1 \mathbf{x}_2 \cdots \mathbf{x}_t, \lambda) \\ = \frac{\delta_t(i)}{\sum_{j=1}^{N} \delta_t(j)}, \quad 1 \leq i \leq N \tag{48}$$

that is, an estimate of the probability of being in state S_i at time t.

Together with the normalized variable $\bar{\delta}_t(i)$, the maximum a posteriori estimate of the current state $\widehat{s}_t^*$ is taken into account according to (34). If $\widehat{s}_t^*$ coincides with the failure state, the desired event is detected by the model and the time to this event is obviously zero. Otherwise, an estimation of the average remaining time in the current state $\widetilde{d}_{\mathrm{avg}}(\widehat{s}_t^*)$ is calculated as

$$\widetilde{d}_{\mathrm{avg}}(\widehat{s}_t^*) = \sum_{i=1}^{N} \left(\mu_{d_i} - \widehat{d}_t(i)\right) \odot \bar{\delta}_t(i), \tag{49}$$

where with μ_{d_i} we denote the expected value of the duration variable in state S_i according to the duration distribution specified by the parameters $\boldsymbol{\theta}_i$. Equation (49) thus estimates the remaining time in the current state by subtracting the estimated states duration, $\widehat{d}_t(i)$, at time t from the expected sojourn time of state S_i, and weighting the result using the uncertainty about the current state, $\bar{\delta}_t(i)$, and, finally, by summing up all the contributions from each state.

In addition to the average remaining time, a lower and an upper bound value can be calculated based on the standard deviation, σ_{d_i}, of the duration distribution for state S_i:

$$\widetilde{d}_{\mathrm{low}}(\widehat{s}_t^*) = \sum_{i=1}^{N} \left(\mu_{d_i} - \sigma_{d_i} - \widehat{d}_t(i)\right) \odot \bar{\delta}_t(i), \tag{50}$$

$$\widetilde{d}_{\mathrm{up}}(\widehat{s}_t^*) = \sum_{i=1}^{N} \left(\mu_{d_i} + \sigma_{d_i} - \widehat{d}_t(i)\right) \odot \bar{\delta}_t(i). \tag{51}$$

Once the remaining time in the current state is estimated, the probability of the next state is calculated by multiplying the transpose of the nonrecurrent transition matrix by the current state probability estimation as follows:

$$\bar{\boldsymbol{\delta}}_{\mathrm{next}} = \left[\bar{\delta}_{t+\widetilde{d}}(i)\right]_{1\leq i\leq N} = \left(\mathbf{A}^0\right)^T \cdot \bar{\boldsymbol{\delta}}_t, \tag{52}$$

while the maximum a posteriori estimate of the next state $\widehat{s}_{\mathrm{next}}^*$ is calculated as

$$\widehat{s}_{\mathrm{next}}^* = \widehat{s}_{t+\widetilde{d}}^* = \arg\max_{1\leq i\leq N} \bar{\delta}_{t+\widetilde{d}}(i). \tag{53}$$

Again, if $\widehat{s}_{t+\widetilde{d}}^*$ coincides with the failure state, the failure will happen after the remaining time at the current state is over and the average estimation of the failure time is $\widetilde{D}_{\mathrm{avg}} = \widetilde{d}_{\mathrm{avg}}(\widehat{s}_t^*)$ calculated at the previous step, with the bound values $\widetilde{D}_{\mathrm{low}} = \widetilde{d}_{\mathrm{low}}(\widehat{s}_t^*)$ and $\widetilde{D}_{\mathrm{up}} = \widetilde{d}_{\mathrm{up}}(\widehat{s}_t^*)$. Otherwise the estimation of the sojourn time of the next state is calculated as follows:

$$\widetilde{d}_{\mathrm{avg}}\left(\widehat{s}_{t+\widetilde{d}}^*\right) = \sum_{i=1}^{N} \mu_{d_i} \odot \bar{\delta}_{t+\widetilde{d}}(i), \tag{54}$$

$$\widetilde{d}_{\mathrm{low}}\left(\widehat{s}_{t+\widetilde{d}}^*\right) = \sum_{i=1}^{N} \left(\mu_{d_i} - \sigma_{d_i}\right) \odot \bar{\delta}_{t+\widetilde{d}}(i), \tag{55}$$

$$\widetilde{d}_{\mathrm{up}}\left(\widehat{s}_{t+\widetilde{d}}^*\right) = \sum_{i=1}^{N} \left(\mu_{d_i} + \sigma_{d_i}\right) \odot \bar{\delta}_{t+\widetilde{d}}(i). \tag{56}$$

This procedure is repeated until the failure state is encountered in the prediction of the next state. The calculation of the RUL is then simply obtained by summing all the estimated remaining time in each intermediate state before encountering the failure state:

$$\widetilde{D}_{\mathrm{avg}} = \sum \widetilde{d}_{\mathrm{avg}}, \tag{57}$$

$$\widetilde{D}_{\mathrm{low}} = \sum \widetilde{d}_{\mathrm{low}}, \tag{58}$$

$$\widetilde{D}_{\mathrm{up}} = \sum \widetilde{d}_{\mathrm{up}}. \tag{59}$$

Finally, Algorithm 1 details the above described RUL estimation procedure.

5. Experimental Results

To demonstrate the effectiveness of the proposed HSMM models, we make use of a series of experiments, performed both on simulated and real data.

The simulated data were generated by considering a left-right HSMM and adapting the parameters of the artificial example reported in the work of Lee et al. [15]. The real case data are monitoring data related to the entire operational life of bearings, made available for the IEEE PHM 2012 data challenge (http://www.femto-st.fr/en/Research-departments/AS2M/Research-groups/PHM/IEEE-PHM-2012-Data-challenge.php).

```
function RulEstimation($\mathbf{x}_t$, $S_k$)                ▷ $\mathbf{x}_t$: The last observation acquired
                                                             ▷ $S_k$: The failure state
Initialization:
    $\widetilde{D}_{\text{avg}} \leftarrow 0$
    $\widetilde{D}_{\text{low}} \leftarrow 0$
    $\widetilde{D}_{\text{up}} \leftarrow 0$
Current state estimation:
    Calculate $\bar{\boldsymbol{\delta}}_t$                  ▷ Using (48)
    Calculate $\widehat{s}_t^*$                              ▷ Using (34)
    Calculate $\widehat{\mathbf{d}}_t$                       ▷ Using (20)
    $\widehat{S} \leftarrow \widehat{s}_t^*$
Loop:
    while $\widehat{S} \neq S_k$ do
        Calculate $\widetilde{d}_{\text{avg}}$               ▷ Using (49) or (54)
        Calculate $\widetilde{d}_{\text{low}}$               ▷ Using (50) or (55)
        Calculate $\widetilde{d}_{\text{up}}$                ▷ Using (51) or (56)
        $\widetilde{D}_{\text{avg}} \leftarrow \widetilde{D}_{\text{avg}} + \widetilde{d}_{\text{avg}}$
        $\widetilde{D}_{\text{low}} \leftarrow \widetilde{D}_{\text{low}} + \widetilde{d}_{\text{low}}$
        $\widetilde{D}_{\text{up}} \leftarrow \widetilde{D}_{\text{up}} + \widetilde{d}_{\text{up}}$
        Calculate $\boldsymbol{\delta}_{\text{next}}$        ▷ Using (52)
        Calculate $\widehat{s}_{\text{next}}^*$              ▷ Using (53)
        $\widehat{S} \leftarrow \widehat{s}_{\text{next}}^*$
    end
    return $\widetilde{D}_{\text{avg}}$, $\widetilde{D}_{\text{low}}$, $\widetilde{D}_{\text{up}}$
```

Algorithm 1: Remaining Useful Lifetime estimation (Pseudo-Code).

5.1. Simulated Experiment. Data have been generated with the idea of simulating the behavior of an industrial machine that, during its functioning, experiences several degradation modalities until a failure state is reached at the end of its lifetime. The generated data are used to test the performances of our methodology for (i) automatic model selection, (ii) online condition monitoring, and (iii) Remaining Useful Lifetime estimation, considering both continuous and discrete observations.

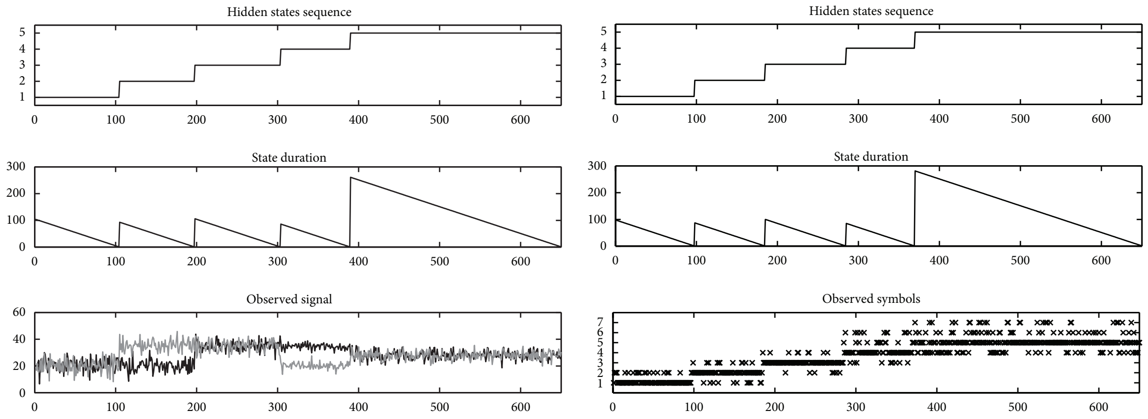
5.1.1. Data Generation. The industrial machine subject of these experiments has been modeled as a left-right parametric HSMM, with $N = 5$ states, having state S_5 as absorbing (failure) state. The choice of a left-right setting has been made for simplicity reasons, since the primary goal of this work is to demonstrate that the proposed model specification coupled with the Akaike Information Criterion is effective to solve automatic model selection, online condition monitoring, and prediction problems. At this purpose, we divided the experiments in two cases according to the nature of the observation, being continuous or discrete.

For each of the continuous and the discrete cases, three data sets have been generated by considering the following duration models: Gaussian, Gamma, and Weibull densities. For each of the three data sets, 30 series of data are used as training set and 10 series as testing set. Each time series contains $T = 650$ observations. The parameters used to generate the data are taken from the work of Lee et al. [15] and are adapted to obtain an equivalent left-right parametric HSMM as follows:

$$\pi = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \qquad \mathbf{A}^0 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\begin{gathered} \Theta_{\mathcal{N}} = \{\boldsymbol{\theta}_1 = [100;\ 20], \boldsymbol{\theta}_2 = [90;\ 15], \boldsymbol{\theta}_3 = [100;\ 20], \\ \boldsymbol{\theta}_4 = [80;\ 25], \boldsymbol{\theta}_5 = [200;\ 1]\}, \\ \Theta_{\mathcal{G}} = \{\boldsymbol{\theta}_1 = [500;\ 0.2], \boldsymbol{\theta}_2 = [540;\ 0.1667], \\ \boldsymbol{\theta}_3 = [500;\ 0.2], \boldsymbol{\theta}_4 = [256;\ 0.3125], \\ \boldsymbol{\theta}_5 = [800;\ 0.005]\}, \\ \Theta_{\mathcal{W}} = \{\boldsymbol{\theta}_1 = [102;\ 28], \boldsymbol{\theta}_2 = [92;\ 29], \boldsymbol{\theta}_3 = [102;\ 28], \\ \boldsymbol{\theta}_4 = [82;\ 20], \boldsymbol{\theta}_5 = [200;\ 256]\}, \end{gathered} \tag{60}$$

where $\Theta_{\mathcal{N}}$, $\Theta_{\mathcal{G}}$, and $\Theta_{\mathcal{W}}$ are the different distribution parameters used for the Gaussian, Gamma, and Weibull duration models, respectively. More precisely, they represent the values of the mean μ_d and the the variance σ_d^2 of the Gaussian distribution, the shape ν_d and the scale η_d of the Gamma distribution, and the scale a_d and the shape b_d of



(a) Example of simulated data for the continuous case

(b) Example of simulated data for the discrete case

Figure 2: The data generated with the parameter described in Section 5.1.1, both for the continuous and the discrete case.

the Weibull distribution. It must be noticed that, as explained in Section 2.1, for state S_5, being the absorbing state, the duration parameters $\boldsymbol{\theta}_5$ have no influence on the data, since, once the state S_5 is reached, the system will remain there forever.

Concerning the continuous observation modeling, a bivariate Gaussian distribution has been used with the following parameters [15]:

$$\mu_1 = \begin{bmatrix} 20 \\ 20 \end{bmatrix}, \qquad \mu_2 = \begin{bmatrix} 20 \\ 35 \end{bmatrix}, \qquad \mu_3 = \begin{bmatrix} 35 \\ 35 \end{bmatrix},$$
$$\mu_5 = \begin{bmatrix} 28 \\ 28 \end{bmatrix},$$
$$U_1 = \begin{bmatrix} 20 & 0 \\ 0 & 20 \end{bmatrix}, \qquad U_2 = \begin{bmatrix} 15 & 0 \\ 0 & 15 \end{bmatrix}, \qquad U_3 = \begin{bmatrix} 15 & -2 \\ -2 & 15 \end{bmatrix},$$
$$U_4 = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}, \qquad U_5 = \begin{bmatrix} 10 & 3 \\ 3 & 10 \end{bmatrix} \tag{61}$$

while for the discrete case, $L = 7$ distinct observation symbols have been taken into consideration with the following observation probability distribution

$$B = \begin{bmatrix} 0.8 & 0.2 & 0 & 0 & 0 & 0 & 0 \\ 0.1 & 0.8 & 0.1 & 0 & 0 & 0 & 0 \\ 0 & 0.1 & 0.8 & 0.1 & 0 & 0 & 0 \\ 0 & 0 & 0.1 & 0.7 & 0.1 & 0.1 & 0 \\ 0 & 0 & 0 & 0.2 & 0.6 & 0.1 & 0.1 \end{bmatrix}. \tag{62}$$

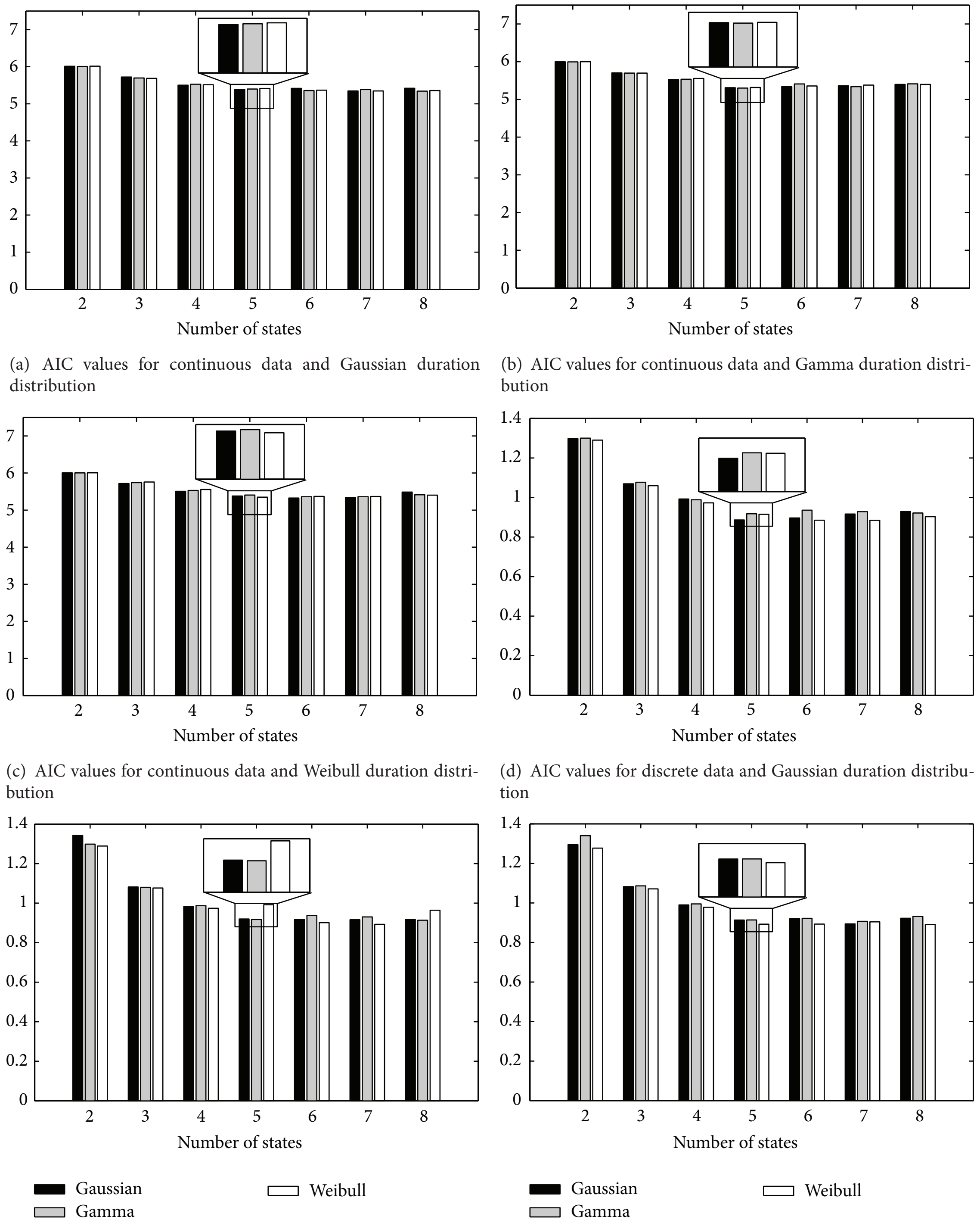
An example of simulated data both for the continuous and the discrete cases is shown in Figure 2, where a Gaussian duration model has been used.

5.1.2. Training and Model Selection. The goal of this experimental phase is to test the effectiveness of the AIC in solving the automatic model selection. For this purpose, the training sets of the 6 data sets (continuous/discrte observation with Gaussian, Gamma, and Weibull duration models) have been taken individually and, for each one of them, a series of learning procedure has been run, each one with a variable HSMM structure. In particular we took into account all the combinations of the duration distribution families (Gaussian, Gamma, and Weibull), an increasing number of states, N, from 2 to 8 and, for the continuous observation cases, an increasing number of Gaussian mixtures, M, in the observation distribution from 1 to 4.

As accurate parameter initialization is crucial for obtaining a good model fitting [14], a series of 40 learning procedures corresponding to 40 random initializations of the initial parameters λ^0 have been executed for each of the considered HSMM structures. For each model structure, the AIC value, as defined in (46), has been evaluated. The final trained set of parameters λ^* corresponding to the minimum AIC value has been retained, resulting in 7 HSMMs with a number of states from 2 to 8.

The obtained results are shown in Figure 3, for both, the continuous and discrete observation data. As it can be noticed, for all the 6 test cases of Figure 3 the AIC values do not improve much for a number of states higher than 5, meaning that adding more states does not add considerable information to the HSMM modeling power. Hence 5 states can be considered as an optimal number of states. Moreover, as shown in the zoomed sections of Figure 3, for the HSMMs with 5 states, the minimum AIC values are obtained for the duration distributions corresponding to the ones used to generate the data. As a consequence AIC can be considered as an effective approach to perform model selection for HSMM, as well as selecting the appropriate parametric distribution family for the state duration modeling.

5.1.3. Condition Monitoring. The optimal parameters λ^* obtained in the previous phase have been used to perform the online condition monitoring experiment on the 10 testing cases for all the 6 considered HSMM configurations. In this experiment, we simulate online monitoring by considering all the testing observations up to the current time, that is, $\{\mathbf{x}_1 \mathbf{x}_2 \cdots \mathbf{x}_t\}$. Each time a new data point is acquired, the Viterbi algorithm is used to estimate the current state $s_t^* = \arg\max_{1 \le i \le N} [\delta_t(i)]$ as specified in (34).



(a) AIC values for continuous data and Gaussian duration distribution

(b) AIC values for continuous data and Gamma duration distribution

(c) AIC values for continuous data and Weibull duration distribution

(d) AIC values for discrete data and Gaussian duration distribution

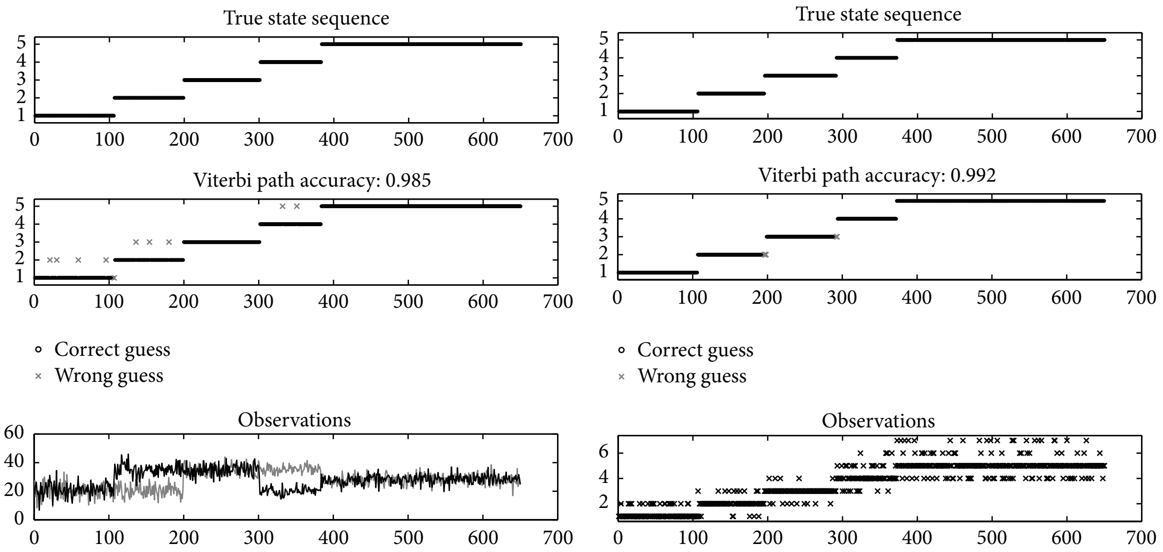
(e) AIC values for discrete data and Gamma duration distribution

(f) AIC values for discrete data and Weibull duration distribution

Figure 3: Akaike Information Criterion (AIC) applied to continuous and discrete observations data. AIC is effective for automatic model selection, since its minimum value provides the same number of states and duration model used to generate the data.

An example of execution of the condition monitoring experiment is shown in Figure 4, for both, continuous and discrete observations, respectively. In Figure 4(a) the state duration has been modeled with a Gamma distribution, while in Figure 4(b) with a Gaussian distribution. In Figures 4(a) and 4(b), the first display represents the true state of the HSMM and the second display represents the estimated state from the Viterbi algorithm, while the third display represents the observed time series.

Knowing the true state sequence we calculated the accuracy, defined as the percentage of correctly estimated states over the total length of the state sequence, for each of



(a) State estimation with Viterbi path for continuous data and Gamma duration distribution

(b) State estimation with Viterbi path for discrete data and Gaussian duration distribution

Figure 4: Condition monitoring using the Viterbi path. HSMMs can be effective to solve condition monitoring problems in time-dependent applications due to their high accuracy in hidden state recognition.

the testing cases. The results are summarized in Table 1(a) for the continuous observations and in Table 1(b) for the discrete observations. The high percentage of correct classified states shows that HSMMs can be effectively used to solve condition monitoring problems for applications in which the system shows a strong time and state duration dependency.

5.1.4. Remaining Useful Lifetime Estimation. In this experimental phase we considered the state S_5 as the failure state and the trained parameters λ^* of Section 5.1.2 for each HSMM configuration. As the online RUL estimation procedure is intended to be used in real time, we simulated condition monitoring experiment where we progressively consider the observations $\{\mathbf{x}_1\mathbf{x}_2\cdots\mathbf{x}_t\}$ up to time t. When a new observation is acquired, after the current state probability $\overline{\delta}_t(i)$ is estimated (Equation (48)), the calculation of the average, upper, and lower RUL ((57), (58), and (59)) is performed.

Examples of RUL estimation are illustrated in Figure 5. In particular Figure 5(a) represents the case of continuous observations and duration modeled by a Weibull distribution, while Figure 5(b) shows the case of discrete observations and duration modeled by a Gamma distribution. From the figures one can notice that the average, as well as the lower and the upper bound estimations, converges to the real RUL as the failure time approaches. Moreover, as expected, the uncertainty about the estimation decreases with the time, since predictions performed at an early stage are more imprecise. As a consequence, the upper and the lower bound become more narrow as the failure state approaches, and the estimation becomes more precise until it converges to the actual RUL value, with the prediction error tending to zero at the end of the evaluation.

To quantitatively estimate the performance of our methodology for the RUL estimation, we considered at each time t the absolute prediction error (APE) between the real RUL and the predicted value defined as

$$\mathrm{APE}(t) = \left|\mathrm{RUL}_{\mathrm{real}}(t) - \mathrm{RUL}(t)\right|, \tag{63}$$

where $\mathrm{RUL}_{\mathrm{real}}(t)$ is the (known) value of the RUL at time t, while $\mathrm{RUL}(t)$ is RUL predicted by the model. To evaluate the overall performance of our methodology, we considered the average absolute prediction error of the RUL estimation, defined as

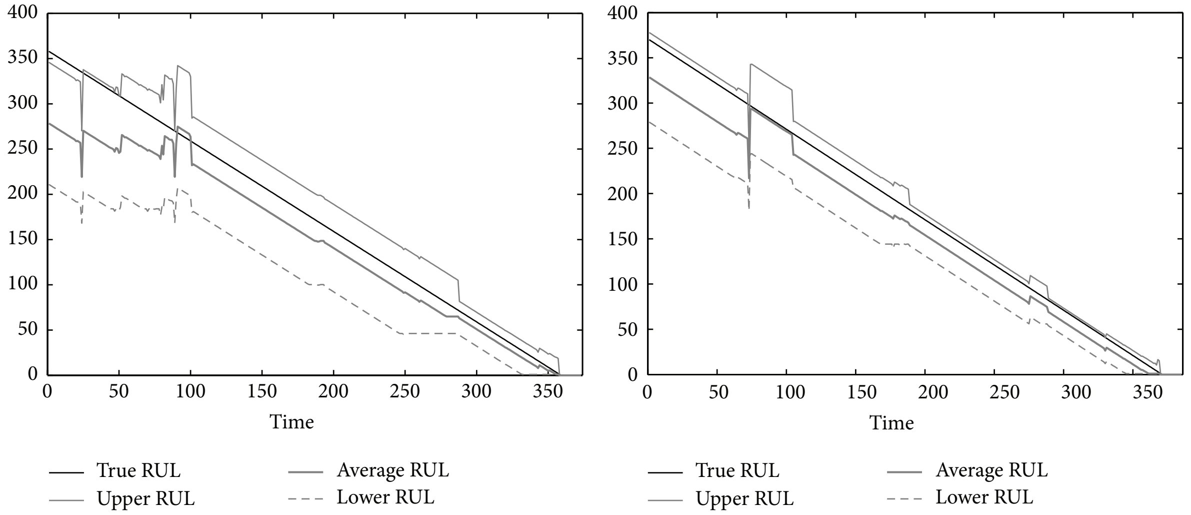
$$\overline{\mathrm{APE}} = \frac{\sum_{t=1}^{T} \mathrm{APE}(t)}{T}, \tag{64}$$

where T is the length of the testing signal. $\overline{\mathrm{APE}}$ being a prediction error average, values of (64) close to zero correspond to good predictive performances.

The results for each of the 10 testing cases and the different HSMM configurations are reported in Tables 2(a) and 2(b), for the continuous and the discrete observation cases, respectively. As it can be noticed, the online Remaining Useful Lifetime estimation and in general the online prediction of the time to a certain event can be effectively faced with HSMMs, which achieve a reliable estimation power with a small prediction error.

Finally, we tested our RUL estimation methodology using the state duration estimation of (16) introduced by Azimi et al. [30–32]. The results are shown in Tables 3(a) and 3(b), in which, respectively, the prediction errors obtained for continuous and discrete observations are reported.

Comparing Table 2 and Table 3, one can notice that the proposed RUL method outperforms the one of Azimi. This



(a) Remaining Useful Lifetime estimation for continuous data and Weibull duration distribution

(b) Remaining Useful Lifetime estimation for discrete data and Gamma duration distribution

Figure 5: HSMMs effectively solve RUL estimation problems. The prediction converges to the actual RUL value and its uncertainty decreases as the real failure time approaches.

Table 1: State recognition accuracy.

(a) Continuous observations

Test case	Duration distribution		
	Gaussian	Gamma	Weibull
1	99.4%	98.5%	99.2%
2	99.7%	98.6%	99.5%
3	99.4%	99.2%	99.7%
4	98.9%	98.9%	99.7%
5	98.2%	98.9%	100%
6	99.1%	98.8%	99.7%
7	98.5%	99.4%	99.7%
8	99.2%	99.1%	99.5%
9	99.2%	98.6%	99.7%
10	99.2%	99.1%	99.5%
Average	**99.1%**	**98.9%**	**99.6%**

(b) Discrete observations

Test case	Duration distribution		
	Gaussian	Gamma	Weibull
1	97.4%	96.7%	97.4%
2	97.2%	97.6%	96.5%
3	99.4%	95.8%	96.6%
4	98.2%	95.3%	97.7%
5	99.1%	97.4%	97.5%
6	97.8%	97.7%	97.8%
7	95.8%	97.2%	96.6%
8	97.7%	96.4%	97.2%
9	98.9%	97.2%	98.5%
10	99.2%	95.6%	96.9%
Average	**98.1%**	**96.7%**	**97.3%**

is mainly due to the proposed average state duration of (20), compared to the one of Azimi given by (16).

5.2. Real Data. In this section we apply the proposed HSMM-based approach for RUL estimation to a real case study using bearing monitoring data recorded during experiments carried out on the Pronostia experimental platform and made available for the IEEE Prognostics and Health Management (PHM) 2012 challenge [49]. The data correspond to normally degraded bearings, leading to cases which closely correspond to the industrial reality.

The choice of testing the proposed methodology on bearings derives from two facts: (i) bearings are the most critical components related to failures of rotating machines [50] and (ii) their monotonically increasing degradation pattern justifies the usage of left-right HSMM models.

5.2.1. Data Description. Pronostia is an experimental platform designed and realized at the Automatic Control and Micro-Mechatronic Systems (AS2M) Department of Franche-Comté Electronics, Mechanics, Thermal Processing, Optics-Science and Technology (FEMTO-ST) Institute (http://www.femto-st.fr/) (Besançon, France), with the aim of collecting real data related to accelerated degradation of bearings. Such data are used to validate methods for bearing condition assessment, diagnostic and prognostic [19, 51–59].

The Pronostia platform allows to perform run-to-failure tests under constant or variable operating conditions. The operating conditions are determined by two factors that can be controlled online: (i) the rotation speed and (ii) the radial force load. During each experiment, temperature and vibration monitoring measurements are gathered online, through two type of sensors placed in the bearing

Table 2: Average absolute prediction error ($\overline{\text{APE}}$) of the RUL estimation using the proposed state duration estimator of (20).

(a) $\overline{\text{APE}}$ of the RUL estimation for the continuous observation test cases

	Duration distribution								
Test case	Gaussian			Gamma			Weibull		
	$\overline{\text{APE}}$ avg	$\overline{\text{APE}}$ up	$\overline{\text{APE}}$ low	$\overline{\text{APE}}$ avg	$\overline{\text{APE}}$ up	$\overline{\text{APE}}$ low	$\overline{\text{APE}}$ avg	$\overline{\text{APE}}$ up	$\overline{\text{APE}}$ low
1	5.1	17.0	6.7	14.0	29.0	0.91	4.5	17.0	8.1
2	7.6	19.0	5.0	6.1	21.0	8.5	6.6	19.0	6.1
3	7.7	5.4	19.0	2.9	12.0	17.0	16.0	29.0	3.0
4	9.0	21.0	2.9	7.5	22.0	6.8	6.0	19.0	6.7
5	7.3	19.0	4.7	2.2	14.0	14.0	3.9	17.0	8.7
6	6.5	18.0	5.6	5.1	18.0	10.0	14.0	27.0	2.7
7	4.7	16.0	7.5	4.8	17.0	11.0	1.2	13.0	12.0
8	10.0	22.0	2.9	5.2	18.0	10.0	9.2	22.0	3.9
9	3.1	9.2	14.0	2.0	16.0	13.0	8.2	21.0	4.9
10	6.4	18.0	5.6	7.5	22.0	6.9	3.3	12.0	13.0
Average	**6.8**	**17.0**	**7.4**	**5.7**	**19.0**	**9.9**	**7.3**	**20.0**	**7.0**

(b) $\overline{\text{APE}}$ of the RUL estimation for the discrete observation test cases

	Duration distribution								
Test case	Gaussian			Gamma			Weibull		
	$\overline{\text{APE}}$ avg	$\overline{\text{APE}}$ up	$\overline{\text{APE}}$ low	$\overline{\text{APE}}$ avg	$\overline{\text{APE}}$ up	$\overline{\text{APE}}$ low	$\overline{\text{APE}}$ avg	$\overline{\text{APE}}$ up	$\overline{\text{APE}}$ low
1	2.1	11.0	14.0	3.1	8.8	14.0	2.4	12.0	13.0
2	2.1	11.0	13.0	11.0	22.0	3.3	19.0	32.0	7.1
3	5.1	17.0	7.6	6.6	18.0	5.1	2.3	14.0	11.0
4	5.9	6.5	18.0	5.2	17.0	6.7	4.2	16.0	9.0
5	3.2	14.0	10.0	8.3	19.0	3.4	12.0	24.0	2.9
6	12.0	24.0	2.7	6.2	18.0	5.2	4.1	8.4	16.0
7	2.9	15.0	9.7	9.3	21.0	2.3	19.0	31.0	6.6
8	15.0	27.0	7.0	7.4	18.0	4.3	4.3	17.0	9.4
9	5.9	18.0	7.7	11.0	23.0	5.5	3.9	16.0	8.8
10	3.5	11.0	14.0	5.5	6.0	16.0	5.2	17.0	7.1
Average	**5.7**	**15.0**	**10.0**	**7.4**	**17.0**	**6.6**	**7.7**	**19.0**	**9.0**

Table 3: Average absolute prediction error ($\overline{\text{APE}}$) of the RUL estimation using the state duration estimator of (16) introduced by Azimi et al. [30–32].

(a) $\overline{\text{APE}}$ of the RUL estimation for the continuous observation test cases

	Duration distribution								
Test case	Gaussian			Gamma			Weibull		
	$\overline{\text{APE}}$ avg	$\overline{\text{APE}}$ up	$\overline{\text{APE}}$ low	$\overline{\text{APE}}$ avg	$\overline{\text{APE}}$ up	$\overline{\text{APE}}$ low	$\overline{\text{APE}}$ avg	$\overline{\text{APE}}$ up	$\overline{\text{APE}}$ low
1	57.8	51.0	66.8	26.2	9.7	52.7	25.9	28.4	64.6
2	50.2	44.4	57.7	21.3	17.0	46.9	29.0	19.2	70.8
3	50.3	44.7	57.3	27.1	8.7	56.5	34.5	13.9	73.4
4	51.8	46.0	60.4	21.3	14.3	45.9	34.9	17.1	78.7
5	59.4	53.7	66.2	29.0	9.5	55.4	33.4	15.6	74.9
6	58.0	51.7	67.1	25.8	8.3	54.1	23.1	25.8	66.5
7	59.4	53.6	66.9	18.2	12.5	47.7	36.0	17.1	74.4
8	63.4	55.6	72.3	19.4	15.7	44.1	34.8	17.8	77.0
9	49.1	43.5	57.0	14.5	17.1	43.2	25.1	26.7	67.0
10	54.4	48.4	62.8	23.2	7.9	52.7	24.1	24.5	67.4
Average	**55.4**	**49.3**	**63.5**	**22.6**	**12.1**	**49.9**	**30.1**	**20.6**	**71.5**

(b) $\overline{\text{APE}}$ of the RUL estimation for the discrete observation test cases

	Duration distribution								
Test case	Gaussian			Gamma			Weibull		
	$\overline{\text{APE}}$ avg	$\overline{\text{APE}}$ up	$\overline{\text{APE}}$ low	$\overline{\text{APE}}$ avg	$\overline{\text{APE}}$ up	$\overline{\text{APE}}$ low	$\overline{\text{APE}}$ avg	$\overline{\text{APE}}$ up	$\overline{\text{APE}}$ low
1	51.4	41.0	62.4	42.4	31.8	53.0	32.6	26.4	73.6
2	49.6	39.9	60.4	59.5	48.3	70.8	31.3	27.6	69.3
3	50.2	38.6	62.3	46.5	35.7	57.4	32.4	25.7	70.2
4	42.2	31.5	53.8	50.1	40.5	60.6	23.7	36.1	60.3
5	44.3	33.9	55.8	47.8	37.4	59.1	36.0	25.6	76.5
6	52.2	43.2	62.7	55.2	44.3	66.9	27.2	31.6	64.3
7	55.0	43.9	66.8	56.0	45.7	67.0	34.7	23.2	74.4
8	50.3	39.0	62.0	60.4	50.5	71.0	35.1	26.4	72.4
9	55.5	47.4	64.0	48.0	37.2	59.5	31.8	22.2	73.6
10	49.0	38.2	60.7	52.1	41.2	63.1	29.4	28.9	68.7
Average	**50.0**	**39.7**	**61.1**	**51.8**	**41.3**	**62.9**	**31.4**	**27.4**	**70.4**

housing: a temperature probe and two accelerometers (one on the vertical and one on the horizontal axis).

The platform is composed of three main parts: a rotating part, a load profile generation part, and a measurement part, as illustrated in Figure 6.

The rotating part is composed of an asynchronous motor which develops a power equal to 250 W, two shafts, and a gearbox, which allows the motor to reach its rated speed of 2830 rpm. The motor's rotation speed and the direction are set through a human machine interface.

The load profiles part issues a radial force on the external ring of the test bearing through a pneumatic jack connected to a lever arm, which indirectly transmits the load through a clamping ring. The goal of the applied radial force is to accelerate the bearing's degradation process.

The measurement part consists of a data acquisition card connected to the monitoring sensors, which provides the user with the measured temperature and vibration data. The vibration measurements are provided in snapshots of 0.1 s collected each 10 seconds at a sampling frequency of 25.6 kHz (2560 samples per each snapshot), while the temperature has been continuously recorded at a sampling frequency of 10 Hz (600 samples collected each minute).

Further details on the Pronostia test rig can be found on the data presentation paper [49] and on the web page of the data challenge (http://www.femto-st.fr/en/Research-departments/AS2M/Research-groups/PHM/IEEE-PHM-2012-Data-challenge.php).

Table 4: Lifetime duration (in seconds) and operating conditions of the bearings tested in the IEEE PHM 2012 Prognostic Challenge dataset [49].

Condition 1		Condition 2		Condition 3	
1800 rpm and 4000 N		1650 rpm and 4200 N		1500 rpm and 5000 N	
Bearing	Lifetime [s]	Bearing	Lifetime [s]	Bearing	Lifetime [s]
Bearing1_1	28030	Bearing2_1	9110	Bearing3_1	5150
Bearing1_2	8710	Bearing2_2	7970	Bearing3_2	16370
Bearing1_3	23750	Bearing2_3	19550	Bearing3_3	4340
Bearing1_4	14280	Bearing2_4	7510		
Bearing1_5	24630	Bearing2_5	23110		
Bearing1_6	24480	Bearing2_6	7010		
Bearing1_7	22590	Bearing2_7	2300		

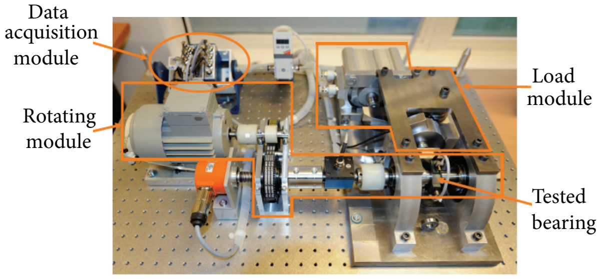


Figure 6: Global overview of the Pronostia experimental platform [19].

Regarding the data provided for the PHM 2012 challenge, 3 different operating conditions were considered:

(i) first operating conditions: speed of 1800 rpm and load of 4000 Newton;

(ii) second operating conditions: speed of 1650 rpm and load of 4200 Newton;

(iii) third operating conditions: speed of 1500 rpm and load of 5000 Newton.

Under the above operating conditions, a total of 17 accelerated life tests were realized on bearings of type NSK 6804 DD, which can operate at a maximum speed of 13000 rpm and a load limit of 4000 N. The tests were stopped when the amplitude of the vibration signal was higher than 20 g; thus this moment was defined as the bearing failure time. An example of bearing before and after the experiment is shown in Figure 7 together with the corresponding vibration signal collected during the whole test.

Table 4 reports how the 17 tested bearings were separated into the three operating conditions. Moreover, the duration of each experiment, being the RUL to be predicted for each bearing, is also given. We performed two sets of experiments by considering, respectively, the bearings relative to the first and the second operating condition (i.e., *Bearing1_1, Bearing1_2, . . ., Bearing1_7* and *Bearing2_1, Bearing2_2, . . ., Bearing2_7*).

As already mentioned, the available data correspond to normally degraded bearings, meaning that the defects were not initially induced and that each degraded bearing contains almost all the types of defects (balls, rings, and cage), resembling faithfully a common real industrial situation. Moreover, no assumption about the type of failure to be occurred is provided with the data and, since the variability in experiment durations is high (from 1 h to 7 h), performing good estimates of the RUL is a difficult task [49].

In our experiments we considered, as input to our model, the horizontal channel of the accelerometer. We preprocessed the raw signals by extracting two time-domain features, that is, root mean square (RMS) and kurtosis, within windows of the same length as the given snapshots ($L = 2560$). Let $r_w(t)$ be the raw signal of the wth window; for each w we estimate RMS as $x_w^{\mathrm{RMS}} = \sqrt{(1/L)\sum_{t=1}^{L} r_w^2(t)}$ and kurtosis as $x_w^{\mathrm{KURT}} = ((1/L)\sum_{t=1}^{L}(r_w(t)-\overline{r}_w)^4)/((1/L)\sum_{t=1}^{L}(r_w(t)-\overline{r}_w)^2)^2$, where $\overline{r}_w$ is the mean of r_w. An example of feature extraction for *Bearing1_1* is shown in Figure 8.

To assess the performance of the proposed HSMM, after the model selection procedure, we implemented a leave-one-out cross validation scheme: by considering separately conditions 1 and 2, we performed the online RUL estimation for each of the 7 bearings, using an HSMM trained with the remaining 6 bearing histories. Similarly to the simulated case, we considered the average absolute prediction error, defined in (64), to quantitatively evaluate our method.

5.2.2. Bearings RUL Estimation. We performed our experiments in two steps: firstly we applied model selection in order to determine an optimal model structure, and secondly, we estimated the RUL of the bearings. The full procedure is detailed in the following.

(A) HSMM Structure. To determine an appropriate HSMM structure for effectively modeling the considered data, we considered several HSMM structures, characterized by (i) the duration distribution family (being Gaussian, Gamma, or Weibull), (ii) an increasing number of states, N, from 2 to 6, and (iii) an increasing number of Gaussian mixtures, M, in the observation density from 1 to 4. For each model structure, obtained by systematically considering all the combinations of (i) to (iii), we run 120 parameter learnings, corresponding to 120 random initializations, λ^0, on the data sets (*Bearing1_1*,

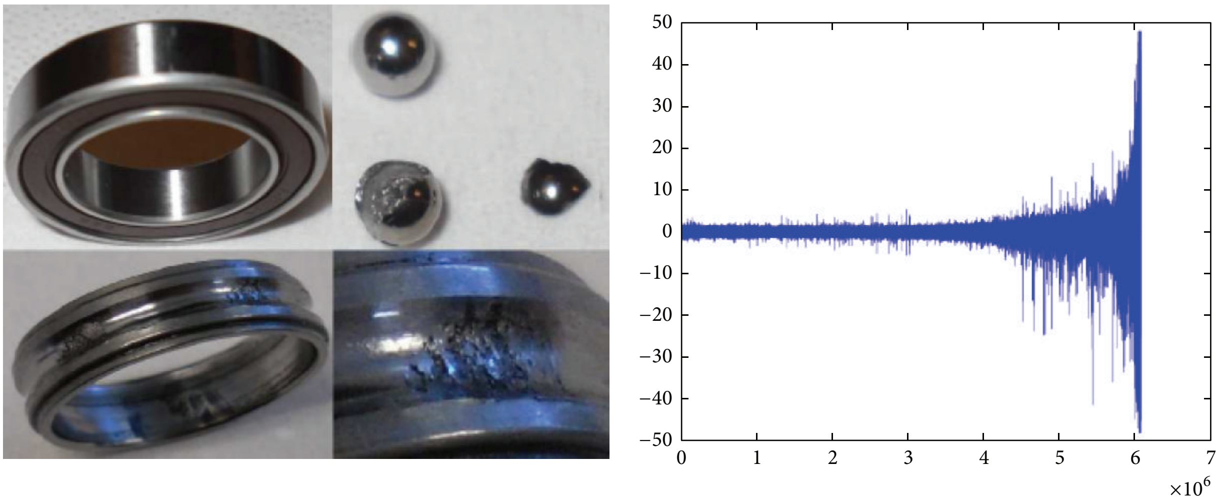
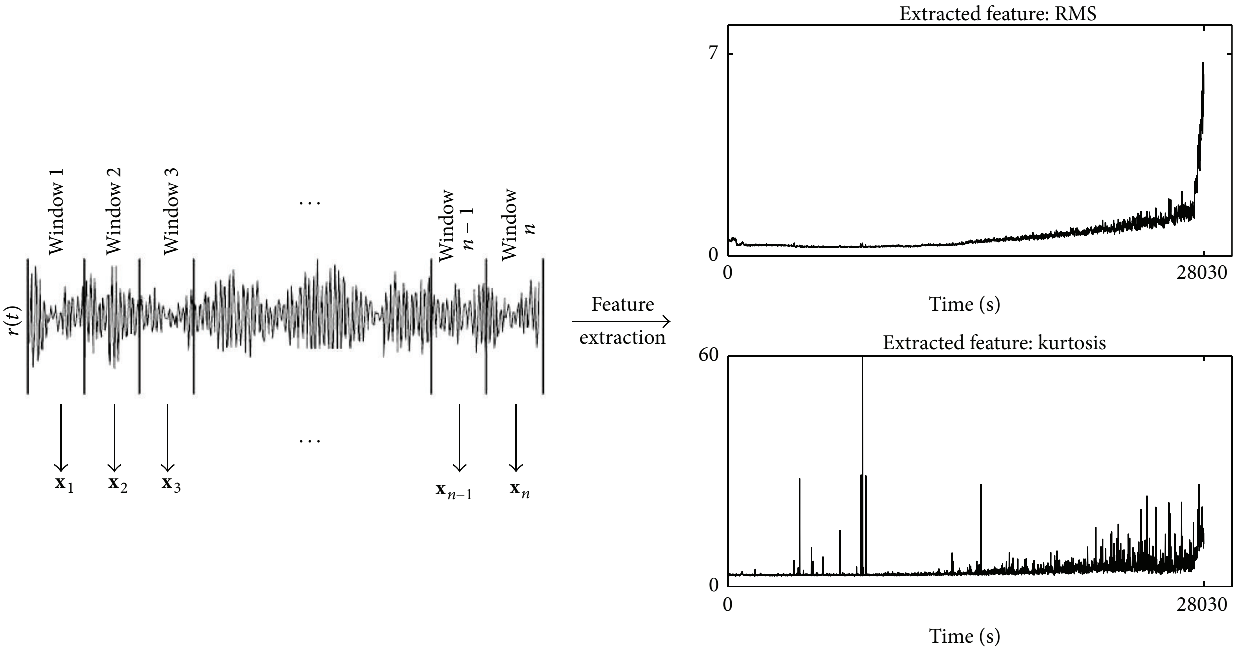


FIGURE 7: A tested bearing before and after the experiment with its recorded vibration signal [49].



(a)

(b)

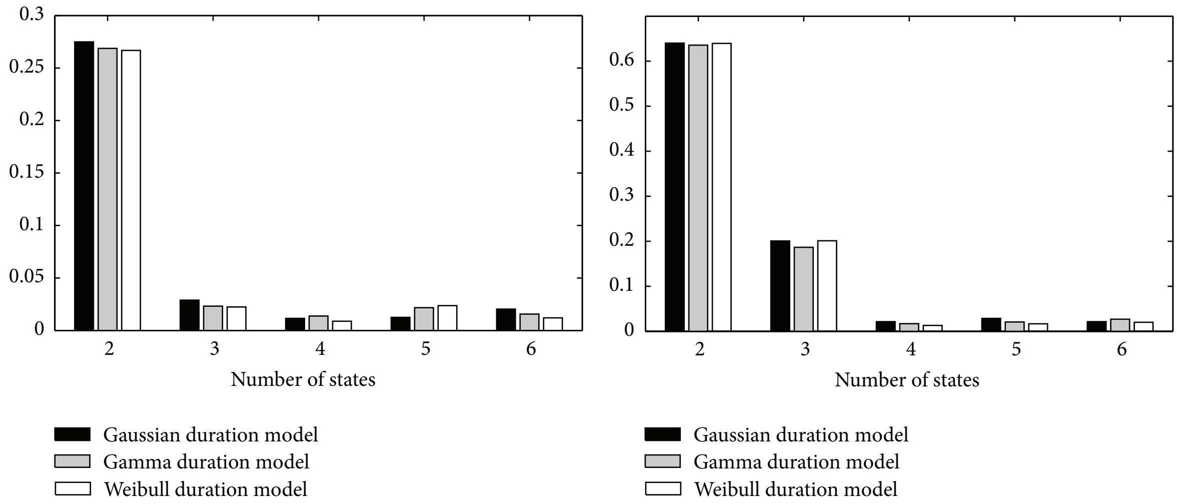
FIGURE 8: Raw vibration data (a) versus RMS and kurtosis features (b) for *Bearing1_1*.

Bearing1_2, ..., *Bearing1_7* and *Bearing2_1*, *Bearing2_2*, ..., *Bearing2_7*).

Similar to Section 5.1.2, at the end of each learning we evaluated the AIC value (Equation (46)) as reported in Figures 9(a) and 9(b) for conditions 1 and 2, respectively. In both cases, the global minimum AIC value corresponds to an HSMM with $N = 4$ states, a Weibull duration model, and a $M = 1$ Gaussians mixture for the observation density.

(B) RUL Estimation. Using the above obtained optimal HSMM structure, we trained it via a leave-one-out cross validation scheme by using for condition 1, at each iteration, *Bearing1_i*, $1 \leq i \leq 7$, as the testing bearing, while the remaining six bearings were used for training. Once the trained parameters λ_i^* were estimated for the ith testing bearing, we progressively collected the observations of the tested *Bearing1_i* to calculate, at each time t, the average, lower, and upper RUL, as specified in (57), (58), and (59), respectively, considering the state S_4 as the failure state. The same procedure has been performed for the bearings in condition 2.

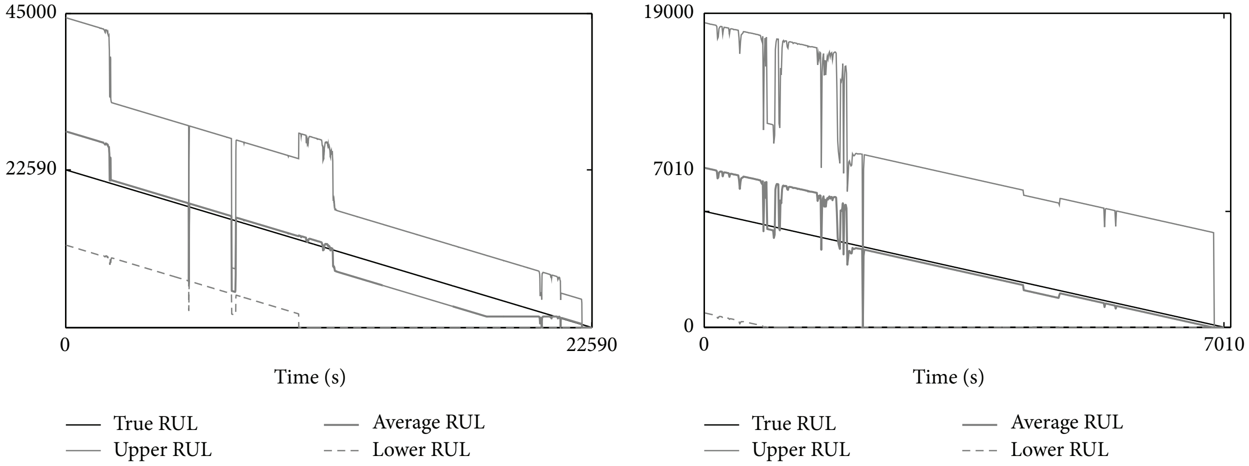
Examples of RUL estimation for *Bearing1_7* and *Bearing2_6* are shown in Figures 10(a) and 10(b), respectively, where the black solid line represents the real RUL which goes to zero as the time goes on. As it can be seen, the average as



(a) AIC values for Condition 1

(b) AIC values for Condition 2

Figure 9: In both cases the minimum AIC value corresponds to an HSMM with $N = 4$ states, a Weibull duration model, and $M = 1$ mixture in the observation density.



(a) RUL estimation for Bearing1_7

(b) RUL estimation for Bearing2_6

Figure 10: By obtaining a low average absolute prediction error, the proposed parametric HSMM is effective for estimating the Remaining Useful Lifetime of bearings.

well as the lower and the upper bound estimations converge to the real RUL as the real failure time approaches and the uncertainty about the estimation decreases with time.

Concerning the quantitative estimation of the predictive performances, we report in Table 5 the average absolute prediction error of the RUL estimation (see Equation (64)), expressed in seconds. As it can be noticed, the average absolute prediction error of the average RUL is, respectively, 1 hour and 15 minutes for condition 1, and 1 hour and 5 minutes for condition 2, which are good values, considering the high variability in the training set durations and the fact that the performance metric takes into account also the less accurate predictions performed at an early stage of the bearings life. Moreover, for both conditions, the average prediction errors of 5 tests out of 7 are below the average, while the best average error of the mean RUL is only 23 minutes for condition 1 while it further decreases to 14 minutes for condition 2.

6. Conclusion and Future Work

In this paper, we introduced an approach based on Hidden Semi-Markov Models (HSMM) and Akaike Information Criteria (AIC) to perform (i) automatic model selection, (ii) online condition monitoring, and (iii) online time to event estimation.

The proposed HSMM models the state duration distribution with a parametric density, allowing a less computationally expensive learning procedure due to the few required

Table 5: Average absolute prediction error ($\overline{\text{APE}}$) of the RUL estimation, expressed in seconds.

(a) Condition 1

Test Bearings	$\overline{\text{APE}}_{\text{avg}}$	$\overline{\text{APE}}_{\text{low}}$	$\overline{\text{APE}}_{\text{up}}$
Bearing1_1	10571.6	12723.0	9414.6
Bearing1_2	4331.2	3815.6	3821.3
Bearing1_3	2997.0	9730.9	6091.2
Bearing1_4	6336.3	2876.6	14871.9
Bearing1_5	1968.9	7448.4	10411.5
Bearing1_6	4253.0	9896.4	9793.7
Bearing1_7	1388.0	7494.3	10088.1
Average	**4549.4**	**7712.2**	**9213.2**

(b) Condition 2

Test Bearings	$\overline{\text{APE}}_{\text{avg}}$	$\overline{\text{APE}}_{\text{low}}$	$\overline{\text{APE}}_{\text{up}}$
Bearing2_1	2475.9	5006.5	7287.5
Bearing2_2	1647.3	4497.2	8288.6
Bearing2_3	8877.1	9508.3	7962.1
Bearing2_4	1769.8	4248.6	4982.5
Bearing2_5	8663.1	10490.0	10730.0
Bearing2_6	877.1	3504.7	6687.0
Bearing2_7	3012.5	3866.4	6651.9
Average	**3903.3**	**5874.5**	**7512.8**

parameters to estimate. Together with the provided general model specification, the modified learning, inference, and prediction algorithms allow the usage of any parametric distribution to model the state duration, as well as continuous or discrete observations. As a consequence, a wide class of different applications can be modeled with the proposed methodology.

This paper highlights, through experiments performed on simulated data, that the proposed approach is effective in (i) automatically selecting the correct configuration of the HSMM in terms of number of states and correct duration distribution family, (ii) performing online state estimation, and (iii) correctly predict the time to a determinate event, identified as the entrance of the model in a target state. As a consequence, the proposed parametric HSMM combined with AIC can be used in practice for condition monitoring and Remaining Useful Lifetime applications.

As the targeted application of the proposed methodology is failure prognosis in industrial machines, combining the proposed HSMM model with online learning procedure, capable of adapting the model parameter to new conditions, would be considered in a future extension.

Appendix

In this appendix we give the derivation of the state duration variable, introduced in (20) as

$$\hat{d}_{t+1}(i) = \frac{a_{ii}\left(\hat{\mathbf{d}}_t\right)\cdot\alpha_t(i)\cdot b_i\left(X_{t+1}\right)}{\alpha_{t+1}(i)}\cdot\left(\hat{d}_t(i)+1\right). \quad \text{(A.1)}$$

The random variable $d_t(i)$ has been defined in Section 2.1 *as the duration spent in state i prior to current time t, assuming that the state at current time t be i*. $d_t(i)$ is sampled from an arbitrary distribution:

$$d_t(i) \sim f(d). \quad \text{(A.2)}$$

We can specify the probability that the system has been in state i for d time units prior to current time t, giving the observations and the model parameters λ and knowing that the current state is i as

$$\begin{aligned}\mathbb{P}\left(d_t(i)=d\right) = \mathbb{P}\big(&s_{t-d-1}\neq S_i, s_{t-d}=S_i,\dots,s_{t-1}=S_i \mid \\ &s_t=S_i,\mathbf{x}_1,\dots,\mathbf{x}_t,\lambda\big).\end{aligned} \quad \text{(A.3)}$$

We omit the conditioning to the model parameters λ in the following equations, being inherently implied. We are interested to derive the estimator $\hat{d}_t(i)$ of $d_t(i)$, defined as its expected value (see Equation (15)):

$$\hat{d}_t(i) = \mathbb{E}\left(d_t(i)\mid s_t=S_i,\mathbf{x}_1\mathbf{x}_2\cdots\mathbf{x}_t\right) \qquad 1\leq i\leq N. \quad \text{(A.4)}$$

From the definition of expectation we have

$$\begin{aligned}\hat{d}_t(i) &= \sum_{d=1}^{t} d\cdot\mathbb{P}\left(d_t(i)=d\right) \\ &= \sum_{d=1}^{t} d\cdot\mathbb{P}\big(s_{t-d-1}\neq S_i,s_{t-d}=S_i,\dots,s_{t-1}=S_i \mid \\ &\qquad s_t=S_i,\mathbf{x}_1,\dots,\mathbf{x}_t\big).\end{aligned} \quad \text{(A.5)}$$

For $\hat{d}_{t+1}(i)$, we have

$$\begin{aligned}\hat{d}_{t+1}(i) &= \sum_{d=1}^{t+1} d\cdot\mathbb{P}\big(s_{t-d}\neq S_i,s_{t-d+1}=S_i,\dots,s_t=S_i \mid \\ &\qquad s_{t+1}=S_i,\mathbf{x}_1,\dots,\mathbf{x}_{t+1}\big)\end{aligned} \quad \text{(A.6)}$$

$$\begin{aligned}&= \underbrace{\mathbb{P}\left(s_{t-1}\neq S_i,s_t=S_i\mid s_{t+1}=S_i,\mathbf{x}_1,\dots,\mathbf{x}_{t+1}\right)}_{(a)} \\ &\quad+\sum_{d=2}^{t+1} d\cdot\mathbb{P}\big(s_{t-d}\neq S_i,s_{t-d+1}=S_i,\dots,s_t=S_i \mid \\ &\qquad s_{t+1}=S_i,\mathbf{x}_1,\dots,\mathbf{x}_{t+1}\big).\end{aligned} \quad \text{(A.7)}$$

By noticing that

$$\begin{aligned}&\mathbb{P}\big(s_{t-d}\neq S_i,s_{t-d+1}=S_i,\dots,s_{t-1}=S_i \mid \\ &\quad s_t=S_i,s_{t+1}=S_i,\mathbf{x}_1,\dots,\mathbf{x}_{t+1}\big) \\ &\quad= \big(\mathbb{P}\big(s_{t-d}\neq S_i,s_{t-d+1}=S_i,\dots,s_{t-1}=S_i,s_t=S_i \mid \\ &\qquad s_{t+1}=S_i,\mathbf{x}_1,\dots,\mathbf{x}_{t+1}\big)\big) \\ &\qquad\cdot\left(\mathbb{P}\left(s_t=S_i\mid s_{t+1}=S_i,\mathbf{x}_1,\dots,\mathbf{x}_{t+1}\right)\right)^{-1}\end{aligned} \quad \text{(A.8)}$$

we can replace the probability of the second term of (A.7) with

$$\begin{aligned}\mathbb{P}\left(s_{t-d} \neq S_i, s_{t-d+1} = S_i, \dots, s_{t-1} = S_i, s_t = S_i \mid\right.&\\ \left. s_{t+1} = S_i, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right)&\\ = \underbrace{\mathbb{P}\left(s_t = S_i \mid s_{t+1} = S_i, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right)}_{(b)}&\end{aligned} \tag{A.9}$$

$$\begin{aligned}\cdot\, \mathbb{P}\left(s_{t-d} \neq S_i, s_{t-d+1} = S_i, \dots, s_{t-1} = S_i \mid\right.&\\ \left. s_t = S_i, s_{t+1} = S_i, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right).&\end{aligned} \tag{A.10}$$

In the last factor of (A.10), we can omit the information about the current state and observation by observing that

$$\begin{aligned}\mathbb{P}\left(s_{t-d} \neq S_i, s_{t-d+1} = S_i, \dots, s_{t-1} = S_i \mid\right.&\\ \left. s_t = S_i, s_{t+1} = S_i, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right)&\\ \approx \underbrace{\mathbb{P}\left(s_{t-d} \neq S_i, s_{t-d+1} = S_i, \dots, s_{t-1} = S_i \mid s_t = S_i, \mathbf{x}_1, \dots, \mathbf{x}_t\right)}_{(c)}&\end{aligned} \tag{A.11}$$

if the following independencies hold:

$$\begin{aligned} s_{t+1} &\perp s_{t-d+1}, \dots, s_{t-1} \mid s_t, \mathbf{x}_1, \dots, \mathbf{x}_t,\\ X_{t+1} &\perp s_{t-d+1}, \dots, s_{t-1} \mid s_t, \mathbf{x}_1, \dots, \mathbf{x}_t,\end{aligned} \tag{A.12}$$

where with $\perp$ we denote independency. Equation (A.12) holds for HMMs (even without conditioning on $\mathbf{x}_1, \dots, \mathbf{x}_t$), but they do not hold for HSMMs since the state duration (expressed by $s_{t-d+1}, \dots, s_{t-1}$) determines the system evolution. On the other hand, state duration is partially known by the observtions, $\mathbf{x}_1, \dots, \mathbf{x}_t$. Thus, the approximation is reasonable as long as the uncertainty on the states is within limits.

From (A.6), (A.9), and (A.11) we obtain

$$\begin{aligned}\widehat{d}_{t+1}(i) &= (a) + \sum_{d=2}^{t+1} d \cdot (b) \cdot (c)\\ &= \underbrace{\mathbb{P}\left(s_{t-1} \neq S_i, s_t = S_i \mid s_{t+1} = S_i, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right)}_{\mathbb{P}(A,B|C) = \mathbb{P}(A|B,C)\cdot\mathbb{P}(B|C)}\\ &\quad + \sum_{d=2}^{t+1} d \cdot \underbrace{\mathbb{P}\left(s_t = S_i \mid s_{t+1} = S_i, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right)}_{\text{it does not depend on } d}\\ &\quad \cdot \mathbb{P}\left(s_{t-d} \neq S_i, s_{t-d+1} = S_i, \dots, s_{t-1} = S_i \mid\right.\\ &\qquad \left. s_t = S_i, \mathbf{x}_1, \dots, \mathbf{x}_t\right)\\ &= \left\{\mathbb{P}\left(s_{t-1} \neq S_i \mid s_t = S_i, s_{t+1} = S_i, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right)\right.\\ &\quad \left. \cdot\, \mathbb{P}\left(s_t = S_i \mid s_{t+1} = S_i, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right)\right\}\\ &\quad + \mathbb{P}\left(s_t = S_i \mid s_{t+1} = S_i, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right)\\ &\quad \cdot \sum_{d=2}^{t+1} d \cdot \mathbb{P}\left(s_{t-d} \neq S_i, s_{t-d+1} = S_i, \dots, s_{t-1} = S_i \mid\right.\\ &\qquad \left. s_t = S_i, \mathbf{x}_1, \dots, \mathbf{x}_t\right)\\ &= \mathbb{P}\left(s_t = S_i \mid s_{t+1} = S_i, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right)\\ &\quad \cdot \left[\underbrace{\mathbb{P}\left(s_{t-1} \neq S_i \mid s_t = S_i, \cancel{s_{t+1} = S_i}, \mathbf{x}_1, \dots, \mathbf{x}_t, \cancel{\mathbf{x}_{t+1}}\right)}_{\text{for the approximation of (A.11)}}\right.\\ &\quad + \sum_{d=2}^{t+1} d \cdot \mathbb{P}\left(s_{t-d} \neq S_i, s_{t-d+1} = S_i, \dots, s_{t-1} = S_i \mid\right.\\ &\qquad \left.\left. s_t = S_i, \mathbf{x}_1, \dots, \mathbf{x}_t\right)\right]\\ &= \mathbb{P}\left(s_t = S_i \mid s_{t+1} = S_i, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right)\\ &\quad \cdot \left[\mathbb{P}\left(s_{t-1} \neq S_i \mid s_t = S_i, \mathbf{x}_1, \dots, \mathbf{x}_t\right) + \sum_{d'=1}^{t} \left(d' + 1\right)\right.\\ &\quad \cdot \mathbb{P}\left(s_{t-d'-1} \neq S_i, s_{t-d'} = S_i, \dots, s_{t-1} = S_i \mid\right.\\ &\qquad \left.\left. s_t = S_i, \mathbf{x}_1, \dots, \mathbf{x}_t\right)\right].\end{aligned} \tag{A.13}$$

Noticing that

$$\begin{aligned}\sum_{d'=1}^{t} \mathbb{P}\left(s_{t-d'-1} \neq S_i, s_{t-d'} = S_i, \dots, s_{t-1} = S_i \mid\right.&\\ \left. s_t = S_i, \mathbf{x}_1, \dots, \mathbf{x}_t\right)&\\ + \mathbb{P}\left(s_{t-1} \neq S_i \mid s_t = S_i, \mathbf{x}_1, \dots, \mathbf{x}_t\right) = 1&\end{aligned} \tag{A.14}$$

because it represents the sum of the probabilities of all the possible combinations of state sequences up to the current time t, we can rewrite (A.13) as follows:

$$\widehat{d}_{t+1}(i) = \mathbb{P}\left(s_t = S_i \mid s_{t+1} = S_i, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right) \cdot \left(\widehat{d}_t(i) + 1\right). \tag{A.15}$$

The intuition behind the latter induction formula is that the current average duration is the previous average duration plus 1 weighted with the "amount" of the current state that was already in state i in the previous step.

In order to transform (A.15) in terms of model parameters for an easy numerical calculation of the induction for $\widehat{d}_{t+1}(i)$, we can consider the following equality:

$$\begin{aligned}\mathbb{P}&\left(s_t = S_i \mid s_{t+1} = S_i, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right)\\ &= \frac{\mathbb{P}\left(s_t = S_i, s_{t+1} = S_i \mid \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right)}{\underbrace{\mathbb{P}\left(s_{t+1} = S_i \mid \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\right)}_{\gamma_{t+1}(i)}}.\end{aligned} \tag{A.16}$$

If we consider the terms involved in the probability at the numerator of the right-hand side of (A.16), we have that

$$\underbrace{\mathbf{x}_1, \dots, \mathbf{x}_t}_{B} \perp \underbrace{\mathbf{x}_{t+1}}_{C} \mid \underbrace{s_t = S_i, s_{t+1} = S_i}_{A}. \tag{A.17}$$

If $B \perp C \mid A$, for the Bayes rule, we have that

$$\mathbb{P}(A \mid C, B) = \frac{\mathbb{P}(C \mid A, \cancel{B}) \cdot \mathbb{P}(A \mid B)}{\mathbb{P}(C \mid B)}. \tag{A.18}$$

Hence, we can rewrite the numerator of the right-hand side of (A.16) as follows:

$$\begin{aligned}
&\mathbb{P}(s_t = S_i, s_{t+1} = S_i \mid \mathbf{x}_1, \dots, \mathbf{x}_{t+1}) \\
&\quad = \Bigg(\mathbb{P}(s_t = S_i, s_{t+1} = S_i \mid \mathbf{x}_1, \dots, \mathbf{x}_t) \\
&\qquad \cdot \mathbb{P}\Bigg(\mathbf{x}_{t+1} \mid \overbrace{\cancel{s_t = S_i}}^{\mathbf{x}_{t+1} \perp s_t \mid s_{t+1}}, s_{t+1} = S_i \Bigg) \Bigg) \\
&\qquad \cdot \left(\mathbb{P}(\mathbf{x}_{t+1} \mid \mathbf{x}_1, \dots, \mathbf{x}_t)\right)^{-1} \\
&\quad = \Bigg(\mathbb{P}(s_{t+1} = S_i \mid s_t = S_i, \mathbf{x}_1, \dots, \mathbf{x}_t) \\
&\qquad \cdot \overbrace{\mathbb{P}(s_t = S_i \mid \mathbf{x}_1, \dots, \mathbf{x}_t)}^{\gamma_t(i)} \cdot \overbrace{\mathbb{P}(\mathbf{x}_{t+1} \mid s_{t+1} = S_i)}^{b_i(\mathbf{x}_{t+1})} \Bigg) \\
&\qquad \cdot \left(\mathbb{P}(\mathbf{x}_{t+1} \mid \mathbf{x}_1, \dots, \mathbf{x}_t)\right)^{-1}.
\end{aligned} \tag{A.19}$$

The first probability in the numerator of (A.19) is the state transition which can be approximated by considering the average duration as

$$\begin{aligned}
&\mathbb{P}(s_{t+1} = S_i \mid s_t = S_i, \mathbf{x}_1, \dots, \mathbf{x}_t) \\
&\quad = \sum_{d_t} a_{ii}(\mathbf{d}_t) \cdot \mathbb{P}(d_t \mid \mathbf{x}_1, \dots, \mathbf{x}_t) \\
&\quad \approx a_{ii}(\widehat{\mathbf{d}}_t)
\end{aligned} \tag{A.20}$$

while the denominator of (A.19) can be expressed as follows:

$$\mathbb{P}(\mathbf{x}_{t+1} \mid \mathbf{x}_1, \dots, \mathbf{x}_t) = \frac{\mathbb{P}(\mathbf{x}_1, \dots, \mathbf{x}_t, \mathbf{x}_{t+1})}{\mathbb{P}(\mathbf{x}_1, \dots, \mathbf{x}_t)} = \frac{\sum_{i=1}^{N} \alpha_{t+1}(i)}{\sum_{i=1}^{N} \alpha_t(i)}. \tag{A.21}$$

By substituting (A.20) and (A.21) in (A.19) we obtain

$$\begin{aligned}
&\mathbb{P}(s_t = S_i, s_{t+1} = S_i \mid \mathbf{x}_1, \dots, \mathbf{x}_{t+1}) \\
&\quad = \frac{a_{ii}(\widehat{\mathbf{d}}_t) \cdot \gamma_t(i) \cdot \sum_{i=1}^{N} \alpha_t(i) \cdot b_i(\mathbf{x}_{t+1})}{\sum_{i=1}^{N} \alpha_{t+1}(i)}
\end{aligned} \tag{A.22}$$

and then, by combining (A.22) and (A.16) we obtain

$$\begin{aligned}
&\mathbb{P}(s_t = S_i \mid s_{t+1} = S_i, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}) \\
&\quad = \frac{a_{ii}(\widehat{\mathbf{d}}_t) \cdot \gamma_t(i) \cdot \sum_{i=1}^{N} \alpha_t(i) \cdot b_i(\mathbf{x}_{t+1})}{\gamma_{t+1}(i) \sum_{i=1}^{N} \alpha_{t+1}(i)}.
\end{aligned} \tag{A.23}$$

Finally, by substituting (A.23) in (A.15) and considering that

$$\gamma_t(i) = \frac{\alpha_t(i)}{\sum_{i=1}^{N} \alpha_t(i)} \tag{A.24}$$

we derive the induction formula for $\widehat{d}_{t+1}(i)$ in terms of model parameters as

$$\widehat{d}_{t+1}(i) = \frac{a_{ii}(\widehat{\mathbf{d}}_t) \cdot \alpha_t(i) \cdot b_i(\mathbf{x}_{t+1})}{\alpha_{t+1}(i)} \cdot \left(\widehat{d}_t(i) + 1\right). \tag{A.25}$$

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] L. Solomon, "Essential elements of maintenance improvement programs," in *Proceedings of the IFAC Workshopon Production Control in the Process Industry, Osaka, Japan and Kariya, Japan, October-November 1989*, E. Oshima and C. van Rijn, Eds., pp. 195–198, Pergamon Press, Oxford, UK, 1989.

[2] T. Honkanen, *Modelling industrial maintenance systems and the effects of automatic condition monitoring [Ph.D. dissertation]*, Helsinki University of Technology, Information and Computer Systems in Automation, 2004.

[3] R. Dekker, "Applications of maintenance optimization models: a review and analysis," *Reliability Engineering & System Safety*, vol. 51, no. 3, pp. 229–240, 1996.

[4] H. Wang, "A survey of maintenance policies of deteriorating systems," *European Journal of Operational Research*, vol. 139, no. 3, pp. 469–489, 2002.

[5] AFNOR, "Condition monitoring and diagnostics of machines—prognostics—part 1: generalguidelines," Tech. Rep. NF ISO 13381-1, 2005.

[6] F. Salfner, *Event-based failure prediction: an extended hidden markov model approach [Ph.D. thesis]*, Humboldt-Universität zu, Berlin, Germany, 2008.

[7] C. Domeniconi, C.-S. Perng, R. Vilalta, and S. Ma, "A classification approachfor prediction of target events in temporal sequences," in *Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD '02)*, pp. 125–137, Springer, London, UK, 2002, http://dl.acm.org/citation.cfm?id=645806.670309.

[8] K. Medjaher, J.-Y. Moya, and N. Zerhouni, "Failure prognostic by using dynamic Bayesian networks," in *Dependable Control of Discrete Systems: 2nd IFAC Workshop on Dependable Control of Discrete Systems (DCDS '09), June 2009, Bari, Italy*, M. P. Fanti and M. Dotoli, Eds., vol. 1, pp. 291–296, International Federation of Accountants, New York, NY, USA, 2009, http://hal.archives-ouvertes.fr/hal-00402938/en/.

[9] A. Sfetsos, "Short-term load forecasting with a hybrid clustering algorithm," *IEE Proceedings: Generation, Transmission and Distribution*, vol. 150, no. 3, pp. 257–262, 2003.

[10] R. Vilalta and S. Ma, "Predicting rare events in temporal domains," in *Proceedings of the 2nd IEEE International Conference on Data Mining (ICDM '02)*, pp. 474–481, December 2002.

[11] E. Sutrisno, H. Oh, A. S. S. Vasan, and M. Pecht, "Estimation of remaining useful life of ball bearings using data driven

methodologies," in *Proceedings of the IEEE Conference on Prognostics and Health Management (PHM '12)*, pp. 1–7, Denver, Colo, USA, June 2012.

[12] K. Goebel, B. Saha, and A. Saxena, "A comparison of three data-driven techniques for prognostics," in *Proceedings of the 62nd Meeting of the Society for Machinery Failure Prevention Technology*, April 2008.

[13] L. R. Rabiner, "Tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[14] F. Cartella, T. Liu, S. Meganck, J. Lemeire, and H. Sahli, "Online adaptive learning of left-right continuous HMM for bearings condition assessment," *Journal of Physics: Conference Series*, vol. 364, Article ID 012031, 2012, http://iopscience.iop.org/1742-6596/364/1/012031.

[15] S. Lee, L. Li, and J. Ni, "Online degradation assessment and adaptive fault detection usingmodified hidden markov model," *Journal of Manufacturing Science and Engineering*, vol. 132, no. 2, Article ID 021010, 11 pages, 2010.

[16] D. A. Tobon-Mejia, K. Medjaher, N. Zerhouni, and G. Tripot, "A mixture of gaussians hiddenmarkov model for failure diagnostic and prognostic," in *Proceedings of the IEEE International Conference on Automation Science and Engineering (CASE '10)*, pp. 338–343, Toronto, Canada, August 2010.

[17] D. A. Tobon-Mejia, K. Medjaher, N. Zerhouni, and G. Tripot, "Estimation of the remaining useful life by using wavelet packet decomposition and HMMs," in *Proceedings of the IEEE Aerospace Conference (AERO '11)*, pp. 1–10, IEEE Computer Society, AIAA, Big Sky, Mont, USA, March 2011.

[18] D. A. Tobon-Mejia, 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, pp. 491–503, 2012.

[19] K. Medjaher, D. A. Tobon-Mejia, and N. Zerhouni, "Remaining useful life estimation of critical components with application to bearings," *IEEE Transactions on Reliability*, vol. 61, no. 2, pp. 292–302, 2012.

[20] J. Ferguson, "Variable duration models for speech," in *Proceedings of the Symposium on the Application of Hidden Markov Models to Text and Speech*, pp. 143–179, October 1980.

[21] K. P. Murphy, "Hidden semi-Markov models (hsmms)," Tech. Rep., University of British Columbia, 2002, http://www.cs.ubc.ca/~murphyk.

[22] A. Kundu, T. Hines, J. Phillips, B. D. Huyck, and L. C. Van Guilder, "Arabic handwriting recognition using variable duration HMM," in *Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR '07)*, vol. 2, pp. 644–648, IEEE, Washington, DC, USA, September 2007.

[23] M. T. Johnson, "Capacity and complexity of HMM duration modeling techniques," *IEEE Signal Processing Letters*, vol. 12, no. 5, pp. 407–410, 2005.

[24] J.-T. Chien and C.-H. Huang, "Bayesian learning of speech duration models," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 6, pp. 558–567, 2003.

[25] K. Laurila, "Noise robust speech recognition with state duration constraints," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '97)*, vol. 2, pp. 871–874, IEEE Computer Society, Munich, Germany, April 1997.

[26] S.-Z. Yu, "Hidden semi-Markov models," *Artificial Intelligence*, vol. 174, no. 2, pp. 215–243, 2010.

[27] S.-Z. Yu and H. Kobayashi, "An efficient forward-backward algorithm for an explicit-duration hidden Markov model," *IEEE Signal Processing Letters*, vol. 10, no. 1, pp. 11–14, 2003.

[28] S.-Z. Yu and H. Kobayashi, "Practical implementation of an efficient forward-backward algorithm for an explicit-duration hidden Markov model," *IEEE Transactions on Signal Processing*, vol. 54, no. 5, pp. 1947–1951, 2006.

[29] N. Wang, S.-D. Sun, Z.-Q. Cai, S. Zhang, and C. Saygin, "A hidden semi-markov model with duration-dependent state transition probabilities for prognostics," *Mathematical Problems in Engineering*, vol. 2014, Article ID 632702, 10 pages, 2014.

[30] M. Azimi, P. Nasiopoulos, and R. K. Ward, "Online identification of hidden semi-Markov models," in *Proceedings of the 3rd International Symposium on Image and Signal Processing and Analysis (ISPA '03)*, vol. 2, pp. 991–996, Rome, Italy, September 2003.

[31] M. Azimi, *Data transmission schemes for a new generation of interactive digital television [Ph.D. dissertation]*, Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, Canada, 2008.

[32] M. Azimi, P. Nasiopoulos, and R. K. Ward, "Offline and online identification of hidden semi-Markov models," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 2658–2663, 2005.

[33] J. Q. Li and A. R. Barron, "Mixture density estimation," in *Advances in Neural Information Processing Systems 12*, pp. 279–285, MIT Press, Boston, Mass, USA, 1999.

[34] M. Dong and D. He, "A segmental hidden semi-Markov model (HSMM)-based diagnostics and prognostics framework and methodology," *Mechanical Systems and Signal Processing*, vol. 21, no. 5, pp. 2248–2266, 2007.

[35] T. Liu, J. Chen, and G. Dong, "Application of continuous hide markov model to bearing performance degradation assessment," in *Proceedings of the 24th International Congress on Condition Monitoring and Diagnostics Engineering Management (COMADEM '11)*, pp. 166–172, 2011.

[36] H. Ocak and K. A. Loparo, "A new bearing fault detection and diagnosis scheme based onhidden markov modeling of vibration signals," in *Proceedings of the IEEE International Conference on Acoustics, Speech, andSignal Processing*, pp. 3141–3144, IEEE Computer Society, Washington, DC, USA, 2001.

[37] C. Fraley and A. E. Raftery, "Model-based clustering, discriminant analysis, and density estimation," *Journal of the American Statistical Association*, vol. 97, no. 458, pp. 611–631, 2002.

[38] K. L. Nylund, T. Asparouhov, and B. O. Muthén, "Deciding on the number of classes in latent class analysis and growth mixture modeling: a Monte Carlo simulation study," *Structural Equation Modeling*, vol. 14, no. 4, pp. 535–569, 2007.

[39] T. H. Lin and C. M. Dayton, "Model selection information criteria for non-nested latent class models," *Journal of Educational and Behavioral Statistics*, vol. 22, no. 3, pp. 249–264, 1997.

[40] O. Lukociene and J. K. Vermunt, "Determining the number of components in mixture models for hierarchical data," in *Advances in Data Analysis, Data Handling and Business Intelligence, Studies in Classification, Data Analysis, and Knowledge Organization*, pp. 241–249, Springer, New York, NY, USA, 2008.

[41] O. Cappé, E. Moulines, and T. Ryden, *Inference in Hidden Markov Models*, Springer Series in Statistics, Springer, New York, NY, USA, 2005.

[42] I. L. MacDonald and W. Zucchini, *Hidden Markov and Other Models for Discrete-Valued Time Series*, Chapman & Hall/CRC, 1997.

[43] R. J. MacKay, “Estimating the order of a hidden Markov model,” *The Canadian Journal of Statistics*, vol. 30, no. 4, pp. 573–589, 2002.

[44] S. E. Levinson, “Continuously variable duration hidden Markov models for automatic speechrecognition,” *Computer Speech and Language*, vol. 1, no. 1, pp. 29–45, 1986.

[45] C. D. Mitchell and L. H. Jamieson, “Modeling duration in a hidden Markov model with the exponential family,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP ’93)*, vol. 2, pp. 331–334, Minneapolis, Minn, USA, April 1993.

[46] A. Viterbi, “Error bounds for convolutional codes and an asymptotically optimum decoding algorithm,” *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, 2006.

[47] G. D. Forney Jr., “The viterbi algorithm,” *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, 1973.

[48] 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*, vol. 39, no. 1, pp. 1–38, 1977.

[49] P. Nectoux, R. Gouriveau, K. Medjaher et al., “Pronostia: an experimental platform for bearings accelerated life test,” in *Proceedings of the IEEE International Conference on Prognostics and Health Management*, Denver, Colo, USA, 2012.

[50] P. O’Donnell, “Report of large motor reliability survey of industrial and commercial installations, part I and II,” *IEEE Transactions on Industry Applications*, vol. 21, no. 4, pp. 853–872, 1985.

[51] P. Boškoski, M. Gašperin, D. Petelin, and Đ. Juričić, “Bearing fault prognostics using Rényi entropy based features and Gaussian process models,” *Mechanical Systems and Signal Processing*, vol. 52-53, pp. 327–337, 2015.

[52] B. Chouri, F. Montero, M. Tabaa, and A. Dandache, “Residual useful life estimation based on stable distribution feature extraction and SVM classifier,” *Journal of Theoretical and Applied Information Technology*, vol. 55, no. 3, pp. 299–306, 2013.

[53] K. Javed, R. Gouriveau, N. Zerhouni, and P. Nectoux, “A feature extraction procedure basedon trigonometric functions and cumulative descriptors to enhance prognostics modeling,” in *Proceedings of the IEEE Conference on Prognostics and Health Management (PHM ’13)*, pp. 1–7, June 2013.

[54] K. Medjaher, N. Zerhouni, and J. Baklouti, “Data-driven prognostics based on health indicatorconstruction: application to pronostia’s data,” in *Proceedings of the 12th European Control Conference (ECC ’13)*, pp. 1451–1456, Zürich, Switzerland, July 2013.

[55] A. Mosallam, K. Medjaher, and N. Zerhouni, “Nonparametric time series modelling for industrial prognostics and health management,” *International Journal of Advanced Manufacturing Technology*, vol. 69, no. 5–8, pp. 1685–1699, 2013.

[56] S. Porotsky and Z. Bluvband, “Remaining useful life estimation for systems with non-trendability behaviour,” in *Proceedings of the IEEE Conference on Prognostics and Health Management (PHM ’12)*, pp. 1–6, Denver, Colo, USA, June 2012.

[57] L. Serir, E. Ramasso, and N. Zerhouni, “An evidential evolving prognostic approach and itsapplication to pronostias data streams,” in *Annual Conference of the Prognostics and Health Management Society*, p. 9, 2012.

[58] F. Sloukia, M. El Aroussi, H. Medromi, and M. Wahbi, “Bearings prognostic using mixtureof gaussians hidden Markov model and support vector machine,” in *Proceedings of the ACS International Conference on Computer Systems and Applications (AICCSA ’13)*, pp. 1–4, May 2013.

[59] B. Zhang, L. Zhang, and J. Xu, “Remaining useful life prediction for rolling element bearing based on ensemble learning,” *Chemical Engineering Transactions*, vol. 33, pp. 157–162, 2013.