

A mixture of gaussians hidden markov model for failure diagnostic and prognostic.

Diego Tobon-Mejia, Kamal Medjaher, Noureddine Zerhouni, Gérard Tripot

▶ **To cite this version:**

Diego Tobon-Mejia, Kamal Medjaher, Noureddine Zerhouni, Gérard Tripot. A mixture of gaussians hidden markov model for failure diagnostic and prognostic.. 6th Annual IEEE Conference on Automation Science and Engineering, CASE'10., Aug 2010, Toronto, Canada. pp.338-343. hal-00525073

HAL Id: hal-00525073
https://hal.science/hal-00525073

Submitted on 11 Oct 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Mixture of Gaussians Hidden Markov Model for failure diagnostic and prognostic

Diego A. Tobon-Mejia, Kamal Medjaher, Noureddine Zerhouni and Gerard Tripot

Abstract— **This paper deals with a data-driven diagnostic and prognostic method based on a Mixture of Gaussians Hidden Markov Model. The prognostic process of the proposed method is made in two steps. In the first step, which is performed off-line, the monitoring data provided by sensors are processed to extract features, which are then used to learn different models that capture the time evolution of the degradation and therefore of the system's health state. In the second step, performed on-line, the learned models are exploited to do failure diagnostic and prognostic by estimating the asset's current health state, its remaining useful life and the associated confidence degree. The proposed method is tested on a benchmark data related to several bearings and simulation results are given at the end of the paper.**

I. INTRODUCTION

Failure prognostic activity aims at anticipating the failure date by predicting the future health state of a given component, sub-system or system and its Remaining Useful Life (RUL). According to the International Standard Organization [1], failure prognostic corresponds to the "estimation of the time to failure and the risk for one or more existing and future failure modes". Several other definitions have been proposed in the literature [2]–[6] during this last decade. All the reported definitions agree about a prediction step and the estimation of the time before the failure. This time is called RUL in some of them, Estimated Time To Failure (ETTF) in the ISO's one [7] and in a small number of publications [8], [9] it is defined as a probability that a machine operates without a fault up to some future time.

Contrary to fault diagnostic which is a mature activity, well developed and spread within the research and the industrial communities, failure prognostic is a new research domain [4], [10], [11]. However, the increasing interest accorded to failure prognostic has led to numerous methods, tools and applications during the last decade. According to what is reported in the literature [4], [6], [12], failure prognostic methods can be classified into three main approaches: model-based, experience-based and data-driven prognostic.

Model-based prognostic approach relies on the use of an analytical model (set of algebraic or differential equations) to represent the behavior of the system including the degradation phenomenon. The advantage of the methods used in this approach is that they provide more precise prognostic results. However, their drawback dwells in the fact that real systems are often nonlinear and the degradation mechanisms are generally stochastic and difficult to obtain in the form of analytical models. Consequently, the applicability of this approach may be limited in practice.

Experience-based prognostic methods use mainly the data of the experience feedback gathered during a significant period of time (maintenance and operating data, failure times, etc.) in order to adjust the parameters of some reliability models (Weibull law, exponential law, etc.). These latter are then used to estimate the time to failure or the RUL. The main advantage of these methods is that they are not subordinated to on-line sensors' data, neither to complex mathematical models of the system's behavior. Instead, they are rather based on the use of simple reliability functions. However, the prognostic results they offer are less precise than those provided by model-based and data-driven approaches, especially when the operating context is variable or in the case of new systems because of a lack of experience data.

Data-driven methods aim at transforming the raw monitoring data into relevant information and behavior models of the system including the degradation. They use artificial intelligence tools (neuronal networks, Bayesian networks, Markovian processes, etc.) or statistical methods to learn the degradation model and to predict the future health state and the corresponding RUL of the system. These methods can be considered as a trade-off between the model-based and the experience-based methods. This is because, in one hand, in real industrial applications getting reliable data is easier than constructing physical or analytical behavior models. And in the other hand, the generated behavioral models from real monitoring data lead to more precise prognostic results than those obtained from experience feedback data.

In this paper a data-driven prognostic method based upon the use of Mixture of Gaussians Hidden Markov Model (MoG-HMM) is proposed. The use of this tool is motivated by the fact that it permits to handle complex emission probability density functions (pdfs) generated by a set of continuous features extracted from raw monitoring signals. The method, presented in section 3, performed in two steps: an off-line step where the raw data are used to learn a behavioral model of the physical component's condition, and an on-line step in which the learned model is used to identify the current condition of the component and to estimate its RUL. In section 4 the method is tested on real operating data related to bearings, and simulation results are given.

Diego A. Tobon-Mejia, Kamal Medjaher and Noureddine Zerhouni are with FEMTO-ST Institute, AS2M department, UMR CNRS 6174 - UFC / ENSMM / UTBM, 25000 Besançon, France diego.tobon@femto-st.fr

Diego A. Tobon-Mejia and Gerard Tripot are with ALSTOM Transport, 7, avenue De Lattre De Tassigny, BP 49, 25290 Ornans, France gerard.tripot@transport.alstom.com

II. Mixture of Gaussians Hidden Markov Models

A MoG-HMM is primarily an Hidden Markov Model (HMM) [13]. Thus, a brief introduction of HMMs would help the reader getting an insight about the MoG-HMM and understanding the proposed prognostic method. An HMM [14] is a statistical model used to represent stochastic processes. An HMM is a Markov chain in which the states are not directly observed. It can be of three types: ergodic, left-right or parallel left-right. An example of a three states left-right HMM is shown in Fig. 1. An HMM is completely defined

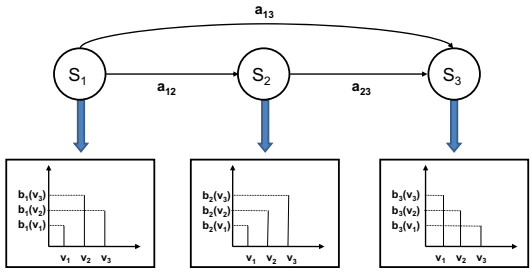


Fig. 1. A three state left-to-right HMM [15].

by the following parameters [14]:

- N: number of states in the model. The individual states are $1, 2, ..., N$, and the state at time t is defined as s_t.
- M: the number of distinct observations for each state. The observation symbols correspond to the physical output of the modeled system. The individual observation symbols are denoted as $V = v_1, v_2, ..., v_M$.
- A: the state transition probability distribution, $A = a_{ij}$, where $a_{ij} = P\left[s_{t+1} = j \,|s_t = i\right], 1 \leq i, j \leq N$.
- B: the observation probability distribution of a state i, $B = b_i(k)$, where $b_i\left(k\right) = P\left[v_k \,|s_t = i\right], 1 \leq i \leq N,\ 1 \leq k \leq M$.
- π: the initial state distribution $\pi = \pi_i$, where $\pi_i = P\left[s_l = i\right], 1 \leq i \leq N$.

For simplicity and clarity of presentation, a compact notation $(\lambda = \pi, A, B)$ is used for each HMM. In practice, HMMs are used to solve the following typical problems [14]:

- Problem 1: given a model λ and an observation sequence $O = (O_1, O_2, ..., O_T)$, compute the probability $P(O|\lambda)$ of the sequence given the model. This problem is known as the evaluation problem (in failure prognostic, this is considered as a detection or a diagnostic problem). The solution of this problem is obtained by using the forward-backward algorithm [16].
- Problem 2: given an observation sequence $O = (O_1, O_2, ..., O_T)$, find the hidden state sequence $S = (S_1, S_2, ..., S_T)$ that have most likely produced the observation sequence. This is the decoding problem and is solved by using the Viterbi algorithm [17].
- Problem 3: find the model parameters (π, A, B) that better fit the observation sequence O, i.e., that maximize the probability $P(O|\lambda)$. This is the learning problem and is solved by using the Baum-Welch algorithm [18].

Discrete HMMs consider the observations as discrete symbols chosen from a finite alphabet and use discrete probability densities to model the transition and the observation probabilities. The problem with this approach, at least for some applications, is that the observations are continuous signals. In order to use a continuous observation density, some restrictions must be placed to insure that the parameters of the probability density function (pdf) can be re-estimated. The most general representation of the pdf, for which a re-estimation procedure has been formulated, is a finite mixture of the form [14]:

$$b_i(O) = \sum_{m=1}^{M} C_{jm} \xi(O, \mu_{jm}, U_{jm}), 1 \leq j \leq N \qquad (1)$$

where O is the observation vector being modeled, C_{jm} is the mixture coefficient for the m^{th} mixture in state i and ξ is any log concave or elliptically symmetric density (e.g. Gaussian) with mean vector μ_{jm} and covariance matrix U_{jm} for the m^{th} mixture component in state j. Usually a Gaussian density is used for ξ and the corresponding model is called a MoG-HMM. Similarly to an HMM, a MoG-HMM is completely defined by four parameters: the number of states, the A matrix, the B matrix and the initial probability distribution π. For a MoG-HMM the observation matrix B is modeled by a Gaussian density with a mean μ, a standard deviation σ and a mixture matrix M. An illustration of a MoG-HMM is given in Fig. 2.

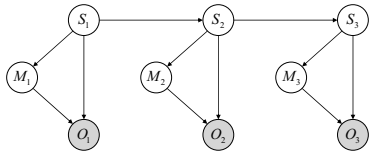


Fig. 2. A Mixture of Gaussians HMM [13].

III. MoG-HMM based prognostic method

An integrated diagnostic and prognostic method to estimate the current health state of a physical component and its remaining useful life is proposed in this section. The method is based on a nondestructive control and uses the data provided by sensors installed to monitor the component's condition. The originality of the proposed method dwells in the fact that in the generated MoG-HMM the states' stay durations are not assumed to be a geometrically decaying functions [19] (which is the case in traditional HMM based prognostic methods), but are learned from the monitoring data (note that multiple observations are considered for both learning and simulation phases, instead of the traditional mono observation approach). Moreover, in the proposed method there is no limitation for the type of the generated MoG-HMM (the model can be an ergodic, a left-right or a parallel left-right). The principle of the proposed method relies on two main phases, as shown in Fig. 3: a learning phase and an exploitation phase.

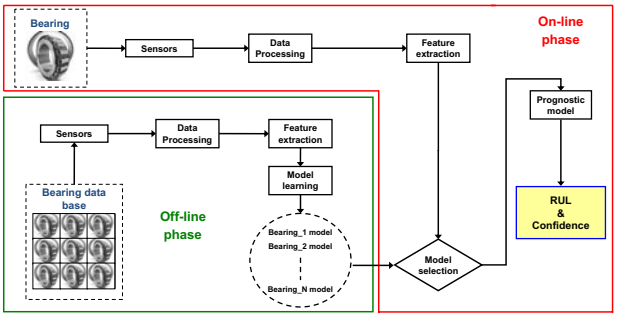


Fig. 3. MoG-HMM prognostic implementation steps.

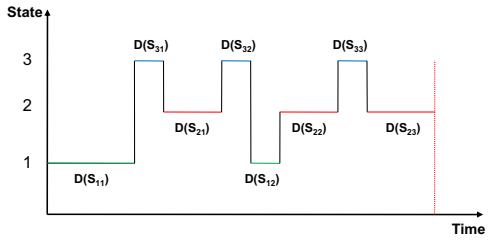


Fig. 4. Example of Viterbi decoding state.

In the first phase, which is executed off-line, the raw data recorded by sensors during the condition monitoring of the physical component are processed in order to extract appropriate features. These features are then used to learn several behavior models (in the form of MoG-HMMs) corresponding to different initial states and operating conditions of the component. Indeed, each raw signal (or data history) corresponding to a given component's condition is transformed to a feature matrix F, by using a feature extraction procedure (F.E). In the matrix F, each column vector (of C features at time t) corresponds to a snapshot on the raw signal, and each cell f_{ct} represents the feature c at time t (2).

$$\begin{aligned} &\text{Raw signal } \xrightarrow{F.E} F = (f_{1t}\ f_{2t} \cdots f_{ct})' \\ &\text{with } 1 \leqslant t \leqslant T \text{ and } 1 \leqslant c \leqslant C \end{aligned} \tag{2}$$

The extracted features are then used to estimate the parameters (π, A and B) and the temporal parameters (stay duration in each state) of the MoG-HMMs. The advantage of using several features instead of only one dwells in the fact that it can happen that a single feature may not capture all the information related to the behavior of the component. The parameters (π, A and B) of each MoG-HMM are learned by using the well known Baum-Welch algorithm [16], whereas the temporal ones are estimated by using the Viterbi algorithm [17]. In addition, this latter permits to obtain the state sequence and to compute the time duration for which the component has been in each state of the corresponding MoG-HMM (Fig. 4). Thus, by assuming that the state duration in each state follows a normal law, it is possible to estimate the mean duration (3) and the corresponding standard deviation (4) by computing the duration and the number of visits in each state. Moreover, the Viterbi algorithm permits to identify the final state which represents the physical component's failure state.

$$\mu\left(D\left(S_i\right)\right) = \frac{\sum_{w=1}^{\Omega} D\left(S_{iw}\right)}{\Omega} \tag{3}$$

$$\sigma\left(D\left(S_i\right)\right) = \frac{\sum_{w=1}^{\Omega} \left[D\left(S_{iw}\right) - \mu\left(D\left(S_i\right)\right)\right]^2}{\Omega} \tag{4}$$

In (3) and (4) $D(\cdot)$ stands for the visit duration, i is the state index, w is the visit index and Ω corresponds to the total of visits. A compact representation of each learned MoG-HMM used to perform diagnostic and prognostic is given by the following expression:

$$\lambda = \left(\pi,\ A,\ B,\ \mu\left(D\left(S_i\right)\right),\ \sigma\left(D\left(S_i\right)\right),\ S_{final}\right) \tag{5}$$

where S_{final} is the final state (corresponding to the end of the considered condition monitoring history), $\mu(D(S_i))$ is the mean state duration for the state i and $\sigma(D(S_i))$ is the standard deviation over the state duration for the state i.
The second phase, which is performed on-line, consists in exploiting the learned models to detect the component's current condition (using the Viterbi algorithm) and to compute the corresponding RUL. The processed data and the extracted features are thus continuously fed to the learned models in order to select the one that best represents the observed data and therefore the corresponding component's condition. The selection process is based on the calculation of a likelihood $P(O|\lambda)$ of the model over the observations. Finally, by knowing the current condition and by using the stay durations learned in the off-line phase, the component's RUL and its associated confidence value can be estimated.
The generated MoG-HMMs are used during the on-line phase to estimate the RUL and the associated confidence value of the physical component by using a dedicated procedure whose steps are the following:

- The first step consists in detecting the appropriate MoG-HMM that best fits and represents the on-line observed sequence of features. Indeed, the features are continuously fed to the set of learned models (completely defined) and a likelihood is calculated in order to select the appropriate model. The selected model is then used to compute the RUL (Fig. 5).
- The second step of this procedure concerns the identification of the current state of the component (diagnostic phase). The Viterbi algorithm is thus applied on the selected model in order to find the state sequence, which corresponds to the observation sequence, and to identify the current state of the component by choosing the most persistent state in the last observations (6).

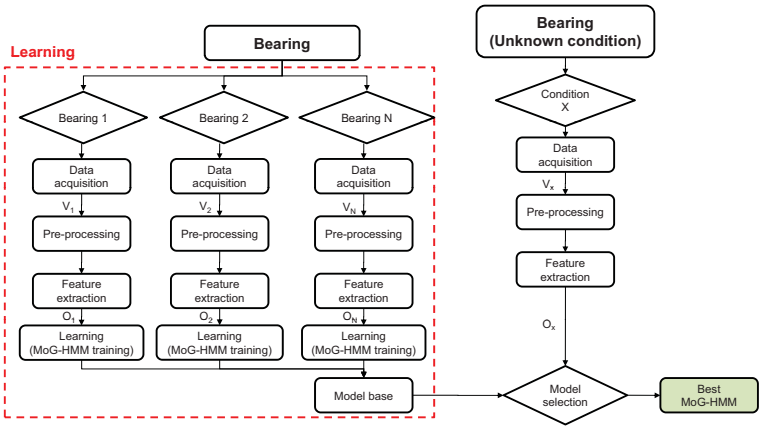


Fig. 5. Competitive model selection.

$$\begin{aligned}&\text{state sequence} = (S_1,\ S_2, \cdots,\ S_t),\\ &\text{with } t = \text{current time.}\\ &\text{Last states} = (S_{t-l}, \cdots,\ S_{t-2},\ S_{t-1},\ S_t),\\ &\text{with } l = \text{past observations factor.}\end{aligned} \tag{6}$$

- The third step consists in using the current identified state, the final state (the failure state) and the probability transition matrix A of the selected MoG-HMM to find the critical path, which goes from the current state to the end state. The idea is to identify all the non-zero probabilities in the transition matrix as potential transitions, and then to choose the minimal path among all the possible ones (Fig. 6) with only one visit per state. In the same way, it is possible to find the longest path by considering a maximum number of states in the path with only one visit per state. The shortest path is assimilated to the pessimistic path (rapid evolution to the failure), whereas the longest path is taken as the optimistic scenario.

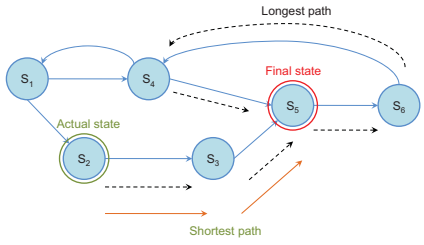


Fig. 6. Path estimation.

- Finally, in the fourth step the paths identified previously are used to estimate the RUL. This latter is obtained using the temporal parameters of the stay duration in each state. In addition, a confidence value over the RUL is calculated based on the standard deviation values of the stay durations. Thus, three values are calculated for each path (7): the upper RUL ($\mu + \sigma$), the mean RUL, and the lower RUL ($\mu - \sigma$).

$$\begin{aligned}RUL_{upper} &= \sum_{i=\text{current state}}^{N} [\mu(D(S_i)) + n \cdot \sigma(D(S_i))],\\ RUL_{mean} &= \sum_{i=\text{current state}}^{N} \mu(D(S_i)),\\ RUL_{lower} &= \sum_{i=\text{current state}}^{N} [\mu(D(S_i)) - n \cdot \sigma(D(S_i))],\\ &\forall i \in \text{ state in the active path, } n = \text{ confidence coef.}\end{aligned} \tag{7}$$

IV. Application and Simulation Results

The failure prognostic method presented previously is tested on a rich condition monitoring data base taken from [20] and containing several bearings tested until the failure. The choice of bearings can be explained by the fact that these components are considered as the most common mechanical elements in industry and are present in almost all industrial processes, especially in those using rotating elements and machines. Moreover, bearings are the main components which most frequently fail in rotating machines [21] (Fig. 7). Thus, the prediction of the time to failure of these components may help improving the reliability, the availability and the safety of the rotating machines while reducing their maintenance costs.

The test data extracted from [20] correspond to several tests

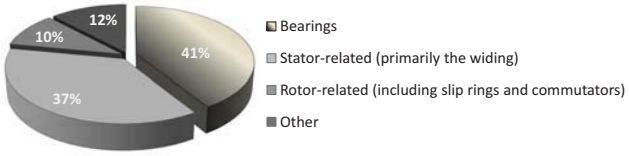


Fig. 7. Failure distribution of motors of power greater than 200 hp [21].

under constant conditions. Four bearings were installed on one shaft. The angular velocity was kept constant at 2000

rpm and a 6000 lb radial load was applied onto the shaft and bearings (Fig. 8). For simulation purposes (learning and on-line failure prognostic) twelve condition monitoring data histories are used (eleven for learning and one for test), each bearing was considered failed at the end of its associated history. The simulation results shown hereafter are obtained from the data corresponding to three tests.

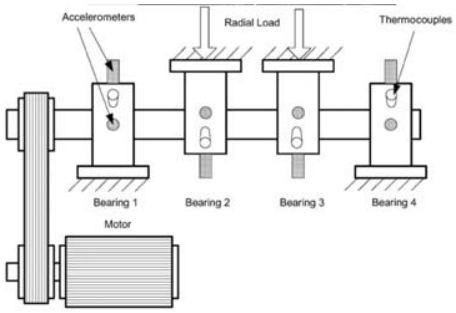


Fig. 8. The bearing benchmark [20].

For both learning and prognostic phases, temporal features (mean, root mean square, skew and kurtosis) [22] have been extracted from the raw signals (vibration signals). The principle of the procedure for feature extraction with an example of rms feature are shown in Fig. 9. During the learning

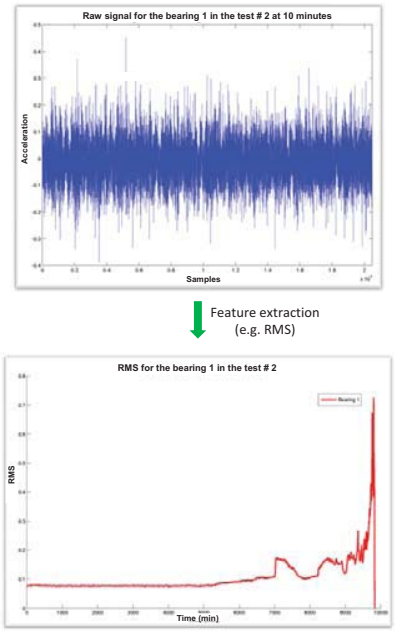


Fig. 9. Feature extraction principle.

phase, three states are defined for each MoG-HMM. The parameters of each MoG-HMM are first randomly initialized and then the continuous extracted features are fed to the learning algorithms in order to re-estimate the initialized parameters (π, A and B). The number of mixtures in each MoG-HMM is set to two, which allows a trade-off between precision and computation time. Eleven MoG-HMMs are thus obtained by using the Baum-Welch algorithm. The re-estimated numerical values of the parameters π, A and M of a MoG-HMM related to bearing one in the test $\#1$ are given below:

$$\pi = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

$$A = \begin{pmatrix} 0.9973 & 0.0013 & 0.0013 \\ 0.0014 & 0.9973 & 0.0014 \\ 0.0014 & 0.0001 & 0.9985 \end{pmatrix}$$

$$M = \begin{pmatrix} 0.3883 & 0.6117 \\ 0.4607 & 0.5393 \\ 0.5188 & 0.4812 \end{pmatrix}$$

The corresponding decoded state sequence of the bearing one in test $\#1$ is shown in Fig. 10. The mean state duration, the standard deviation and the final state for this history are presented below.

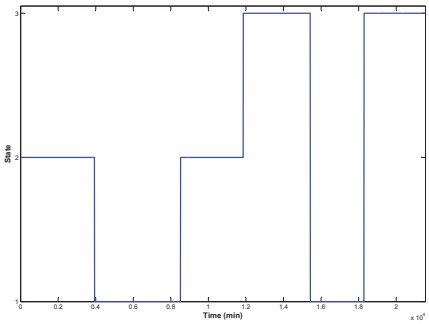


Fig. 10. Decoded state sequence.

$$\begin{bmatrix} \mathrm{S}_1 \\ \mathrm{S}_2 \\ \mathrm{S}_3 \end{bmatrix} = \begin{bmatrix} \mu(\mathrm{S}_1), \sigma(\mathrm{S}_1) \\ \mu(\mathrm{S}_2), \sigma(\mathrm{S}_2) \\ \mu(\mathrm{S}_3), \sigma(\mathrm{S}_3) \end{bmatrix} = \begin{bmatrix} 3720, 860 \\ 3635, 295 \\ 3425, 145 \end{bmatrix} min$$

$$S_{final} = S_3$$

In order to simulate an on-line failure prognostic a condition monitoring data history is taken randomly from the five existing ones. The selection process shown in Fig. 5 is then applied on the data history corresponding to the bearing 2 in the test $\#2$ in order to identify the "best MoG-HMM" and to estimate the RUL. A simulation result of the predicted RUL is shown in Fig. 11, where the data history used to perform the test correspond to a faulty bearing that ends at $9840\ min$. The associated error is shown in Fig. 12.

From Fig. 11 and Fig. 12 one can see that the precision of the estimated RUL increases as the prediction time is approaching the real failure time. Similarly, after 3470 min the mean estimation error drops below 24.5 % and continues to decrease as the real failure time approaches. After 8070 min the mean error stabilizes arround the value of 2.03 %, whereas when considering the upper RUL limit the error drops to 0.6 %. The 68 % confidence interval keeps the RUL estimation limits near acceptable

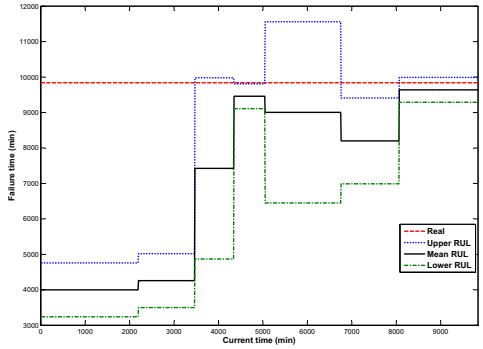


Fig. 11. RUL estimation for the bearing 2 in test #2.

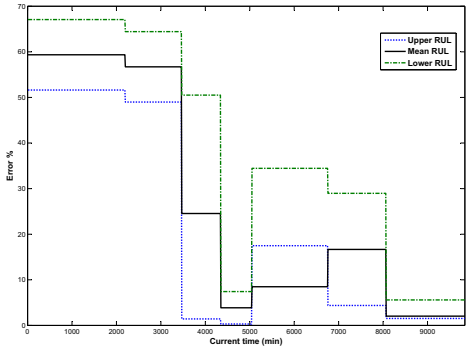


Fig. 12. RUL error associated to the bearing 2 in test #2.

values, a wide confidence intervals (95 % and 99.5 %) will give more sparse limits and will increase the prediction error.

V. Conclusion

A failure prognostic method is proposed in this paper. It deals with real monitoring data provided by sensors installed on a physical component. The raw signals are processed in order to extract useful information and transform them to relevant models that are used to represent the behavior of the system and to learn its degradation phenomenon. The derived models are then used to perform predictions and to estimate the RUL of the system. The choice of MoG-HMMs instead of traditional discrete HMMs is motivated by the fact that a set of continuous features allow to keep a maximum information of the raw signals and thus lead to more precise results. Moreover, the stay durations in each state of the model are not considered as constant and are learned from the monitoring data. Indeed, in real systems, the degradation phenomenon is strongly stochastic and the stay duration in each state is not necessarily governed by an exponential law.

Future works include a comparison of the proposed method with existing ones using the same data in order to assess its performance and limits. Furthermore, it would be interesting to test the method on a real industrial system with a real possibility of on-line failure prognostic and integration of maintenance actions.

References

[1] AFNOR, "Condition monitoring and diagnostics of machines - prognostics - part 1: General guidelines. NF ISO 13381-1," 2005.

[2] G. Provan, "Prognosis and condition-based monitoring: an open systems architecture," in *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, no. 5, 2003, pp. 57–62.

[3] W. Q. Wang, M. F. Golnaraghi, and F. Ismail, "Prognosis of machine health condition using neuro-fuzzy systems," *Mechanical Systems and Signal Processing*, vol. 18, no. 4, pp. 813 – 831, 2004.

[4] A. K. Jardine, D. Lin, and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance," *Mechanical Systems and Signal Processing*, vol. 20, no. 7, pp. 1483 – 1510, 2006.

[5] A. Muller, M.-C. Suhner, and B. Iung, "Formalisation of a new prognosis model for supporting proactive maintenance implementation on industrial system," *Reliability Engineering & System Safety*, vol. 93, no. 2, pp. 234 – 253, 2008.

[6] A. Heng, S. Zhang, A. C. Tan, and J. Mathew, "Rotating machinery prognostics: State of the art, challenges and opportunities," *Mechanical Systems and Signal Processing*, vol. 23, no. 3, pp. 724 – 739, 2009.

[7] D. A. Tobon-Mejia, K. Medjaher, and N. Zerhouni, "The iso 13381-1 standard's failure prognostics process through an example," in *IEEE - Prognostics & System Health Management Conference*, University of Macau, Macau, China, 12-14 January 2010 2010, in Press.

[8] C. Farrar, F. Hemez, G. Park, A. Robertson, H. Sohn, and T. Williams, "A coupled approach to developing damage prognosis solutions," in *5th International Conference on Damage Assessment of Structures*, Southampton, England, July 1-3 2003.

[9] L. D and M. V, "Recursive filters for a partially observable system subject to random failure," *Adv. in Applied Probability*, pp. 207 – 227, 2003.

[10] R. Isermann, "Model-based fault-detection and diagnosis - status and applications," *Annual Reviews in Control*, vol. 29, no. 1, pp. 71 – 85, 2005.

[11] V. Venkatasubramanian, "Prognostic and diagnostic monitoring of complex systems for product lifecycle management: Challenges and opportunities," *Computers & Chemical Engineering*, vol. 29, no. 6, pp. 1253 – 1263, 2005.

[12] M. Lebold and M. Thurston, "Open standards for condition-based maintenance and prognostic systems," in *Maintenance and Reliability Conference (MARCON)*, 2001.

[13] K. P. Murphy, "Dynamic bayesian networks: Representation, inference and learning," Ph.D. dissertation, University of California, 2002.

[14] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," in *Proceedings of the IEEE*, vol. 77 (2), 1989, pp. 257–286.

[15] H. Ocak, K. A. Loparo, and F. M. Discenzo, "Online tracking of bearing wear using wavelet packet decomposition and probabilistic modeling: A method for bearing prognostics," *Journal of sound and vibration*, vol. 302, pp. 951–961, 2007.

[16] L. Baum and J. Egon, "An inequality with applications to statistical estimation for probabilistic functions of a markov process and to a model for ecology," *Bulletin of the American Mathematical Society*, vol. 73, pp. 360 – 363, 1967.

[17] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimal decoding algorithm," *IEEE Transaction on Information Theory*, vol. 13, pp. 260 – 269, 1967.

[18] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, vol. 39, pp. 1 – 38, 1977.

[19] 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, pp. 2248–2266, 2007.

[20] NSF I/UCRC Center for Intelligent Maintenance Systems, "Prognostic data repository: Bearing data set," in *http://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/*, online in 2010.

[21] P. O'Donnell, "Report of large motor reliability survey of industrial and commercial installations, part I, II & III," *IEEE Transactions on Industry Applications*, vol. 21, pp. 853–872, 1985.

[22] T. Kurfess, S. Billington, and S. Liang, "Advanced diagnostic and prognostic techniques for rolling element bearings," *Springer Series in Advanced Manufacturing*, pp. 137–165, 2006.