

Integrated Bayesian Framework for Remaining Useful Life Prediction

A. Mosallam*, K. Medjaher and N. Zerhouni
FEMTO-ST Institute, AS2M department,
University of Franche-Comté/CNRS/ENSMM/UTBM,
24 rue Alain Savary, 25000 Besançon, France.
*Email: ahmed.mosallam@femto-st.fr

Abstract—In this paper, a data-driven method for remaining useful life (RUL) prediction is presented. The method learns the relation between acquired sensor data and end of life time (EOL) to predict the RUL. The proposed method extracts monotonic trends from offline sensor signals, which are used to build reference models. From online signals the method represents the uncertainty about the current status, using discrete Bayesian filter. Finally, the method predicts RUL of the monitored component using integrated method based on K-nearest neighbor (k-NN) and Gaussian process regression (GPR). The performance of the algorithm is demonstrated using two real data sets from NASA Ames prognostics data repository. The results show that the algorithm obtain good results for both application.

I. INTRODUCTION

The need of computer systems that are constantly monitoring the health status of critical systems, by implementing prognostics and health management (PHM) processes, is particularly important for increasing the reliability while decreasing the maintenance costs [1]. PHM consists of three main processes: fault detection, diagnostics and prognostics. Fault detection can be defined as the process of recognizing that a problem has occurred regardless of the root cause [2]. Fault diagnostics is the process of identifying the faults and their causes [3]. Fault prognostics can be defined as the prediction of when a failure might take place [4]. Prognostics has recently attracted significant research interest due to the need of models for accurate RUL prediction for different applications. RUL prediction of critical components is a non trivial task for many reasons. Sensory signals for instance are usually hidden by noise and it is very challenging to process and to extract informative representation of the remaining useful life [5]. Another problem is the prediction uncertainty due to the variation of the end of life time that can differ for two components made by the same manufacturer and operating under the same conditions. Therefore, proposed models should include such uncertainties and represent them in a probabilistic form [6].

The proposed methods for RUL prediction can be broadly divided into physics based and data-driven methods [7]. Physics based methods build physical models of the desired critical components by the means of state-space models [8] and dynamic ordinary or partial differential equations [9]. These models require extensive experimentation and model verification [10]. However, it will be very reliable at least until

the system is upgraded or changed [11]. Data-driven methods can be used to model critical components' degradation behavior when the first principles are complex or not available and sufficient historical data are available [12]. Data-driven methods employ pattern recognition and machine learning techniques to characterize the desired critical components' degradation behavior. For example, different regression models have been proposed in the literature such as the auto regressive model and the multivariate adaptive regression splines [4]. A drawback of using regression methods is that when available component degradation history is incomplete the extrapolation may lead to large errors. There have been more interests lately on various types of neural networks and neural-fuzzy systems [13]. However, these methods generate black box models and it is difficult to select the structure of the network.

Traditionally, data-driven approaches have mostly built on estimating the current health status of the desired component and when the degradation exceeds the alarm threshold, the algorithms start predicting the RUL [14], [15]. A method based on k-NN and belief function theory to estimate the RUL of turbofan engines is presented in [16]. The method manually annotates the health status of the offline data sets and predicts the RUL when the degradation level reaches a predefined alarm threshold. Another algorithm presented in [13] proposed an integrated prognostics model based on summation wavelet extreme learning machine and subtractive-maximum entropy fuzzy clustering. The algorithm automatically identifies the states of degrading machinery and sets dynamically the alarm threshold. Alternatively, instead of learning the degradation from the data and predict the RUL, direct RUL prediction algorithms learn the relation between the observable parameters and the end of life criteria (EOL) to predict the RUL [6].

In this work, integrated direct RUL prediction method is presented. The main contribution of this work is to model the relation between sensor data and EOL to predict the RUL without the need for predefined alarm threshold using integrated Bayesian framework. Also, the uncertainties about the current state and the predicted RUL are presented in probabilistic forms using Bayesian approaches. This paper is structured as follows. In Section II the proposed method is explained formally. The Experimental results are presented in Section III and Section IV concludes the paper.

II. INTEGRATED BAYESIAN FRAMEWORK FOR REMAINING USEFUL LIFE PREDICTION

Measurements observed from monitored components are usually noisy multidimensional time series signals. Thus, it is important to first extract information that represent the degradation evolution through time. The relation between the extracted information and EOL should be modeled to predict the RUL. To do this, the proposed method learns the model from the offline data set. It estimates the current status from the unseen online data and predict the RUL by measuring the similarity to the offline data. The method is summarized in Figure 1 and will be explained hereafter.

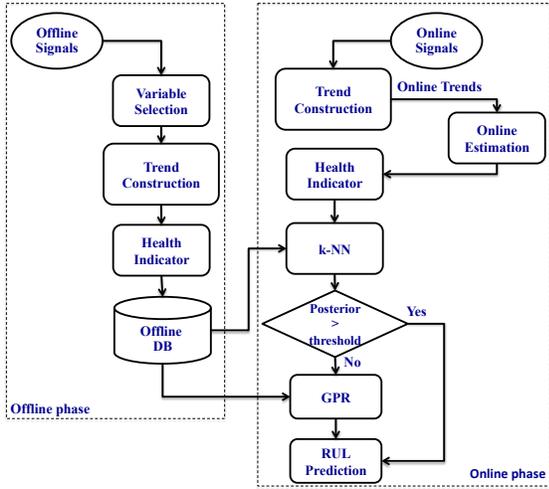


Fig. 1: The method's general scheme.

A. Offline phase

1) *Unsupervised variable selection*: not all signals from the monitored component are informative. Signals that have non random relationships contain information about system degradation. To select such signals, an unsupervised variable selection algorithm based on information theory is performed [17]. The algorithm first calculates pairwise symmetrical uncertainty (SU) for all the input signals, defined by:

$$SU(X, Y) = 2 \times \frac{I(X, Y)}{H(X) + H(Y)} \quad (1)$$

where, $I(X, Y)$ is the mutual information between two random variables X and Y ; $H(X)$ and $H(Y)$ are information entropy values of the random variables X and Y respectively. Then, the algorithm clusters the variables based on the SU distance using hierarchical clustering. The algorithm finally ranks the resulting clusters according to the quality of the included signals in representing interesting relationships using normalized self-organizing map distortion measure.

2) *Trend construction*: after selecting the interesting variables, the task is to extract smooth monotonic signals that are correlated with EOL. To do this, a trend extraction algorithm is used [5]. The algorithm is divided in two main steps:

a) *Variable compression*: the goal of this step is to compress the n signals selected in the previous step onto one-dimensional space. In this work, standard principal component analysis method is used. Thus, the first principle component is used to represent the health status evolution with respect to time. The compressed features are then further processed to get a monotonic trend.

b) *Trend extraction*: the aim of this step is to extract the internal structure of the data in a way that best explains the variation of EOL in a monotonic signal. To do that, empirical mode decomposition algorithm (EMD) is employed [18]. EMD is a method used to decompose a signal into successive intrinsic mode functions (IMF) and residual signal $r_n(t)$ which should be constant or monotonic signal that can be represented as:

$$r_n(t) = X(t) - \sum_{i=1}^n imf_i(t) \quad (2)$$

where, $X(t)$ is the input signal, imf_i is the IMF and n is maximum number of IMFs (Figure 2).

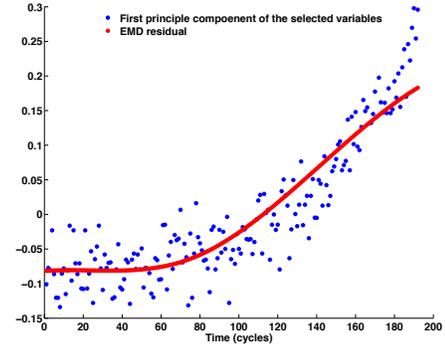


Fig. 2: Trend construction at time t=192 cycles for Engine 1.

3) *Health indicator*: in order to represent the degradation evolution, discriminant features should be extracted from acquired monotonic trends. In this work, a feature vector $F = [a, b, \bar{x}, s^2]$ is extracted from each trend at each time, where, a and b are the slope and the constant term of a linear curve fit of the input trend respectively, \bar{x} and s^2 are the mean and the variance of the input trend respectively. The resulting features are then used to represent each trend according to its EOL time. Each group of trends with similar EOL time is considered as a class and saved in the offline data base.

B. Online phase

In this phase, new sensor data are collected online from the critical component(s) from the same sensor signals that have been selected in the offline phase. Then, these signals are processed to extract monotonic trends in the same manner as in the offline phase. Before using these trends to predict the RUL, the method represents the uncertainty of the input online signal using recursive discrete Bayesian filter. The method then extracts feature vector F over the estimated online trend. K-NN classifier is then used to classify new online inputs based

on both offline and online features extracted from the trends. If the class posterior is less than predefined threshold the method uses GPR model to predict the RUL.

1) *Online Estimation*: the sensor data acquired from the monitored component is prone to be affected by different sources of noise. Thus, the uncertainty about the true or unobserved signal at the present time should be represented recursively in a probabilistic form. To do that, a recursive discrete Bayesian filter is used to estimate the uncertainty from the online trends. This filter, decomposes the state space into finitely many regions and represents the cumulative posterior for each region by probability values, see Algorithm 1.

Input : $\{p_{k,t-1}\}, z_t$

Output: $\{p_{k,t}\}$

forall the k do

$$\begin{cases} \bar{p}_{k,t} = \sum_i p(X_t = x_k | X_{t-1} = x_i) p_{i,t-1} \\ p_{k,t} = \eta p(z_t | X_t = x_k) \bar{p}_{k,t} \end{cases}$$

end

Algorithm 1: Discrete Bayesian filter.

where the input to the algorithm is a discrete probability distribution $\{p_{k,t}\}$ along with the recent measurement z_t ; $\bar{p}_{k,t}$ is prediction probability; $p_{k,t}$ is the posterior probability; $p(X_t = x_k | X_{t-1} = x_i)$ is the state transition model and $p(z_t | X_t = x_k)$ is the measurement transition model. Algorithm 1 first calculates the prediction for the new state based on previous state uncertainty and state transition model. The prediction is then updated in the second line so as to incorporate the measurement. Discrete Bayes filters apply to problems with finite state space, where the random variable X_t can take many values:

$$X_t = x_{1,t} \cup x_{2,t} \cup \dots \cup x_{k,t} \quad (3)$$

A straightforward decomposition of X_t is a multidimensional grid, where each $x_{k,t}$ is a bin or region. The size of each bin dx can be calculated as follows:

$$dx = \frac{x_{max} - x_{min}}{n} \quad (4)$$

Where, x_{max} is the maximum state value, x_{min} is the minimum state value and n is the number of bins. Each bin can then be represented as a Gaussian function with a mean value at each state and a common variance:

$$p(X_t | X_{t-1}) = \|dx \times \mathcal{N}(X_{k,t}, \sigma^2)\| \quad (5)$$

where, $p(X_t = x_k | X_{t-1})$ is the state transition model, dx is the bin size and $\mathcal{N}(X_{k,t}, \sigma^2)$ is the normal distribution at state $X_{k,t}$. Moreover, (5) is normalised to turn this quantity into a probability distribution. Similarly, measurement probability model can be calculated in the same manner as the transition model. Figure 3 depicts the overall Bayesian filter scheme.

2) *Classification using k-Nearest Neighbours*: in order to build the predictive model, k-NN classifier is applied in this work. The estimated feature vector F at time t is passed to k-NN to find the closest offline trend in the data base. The

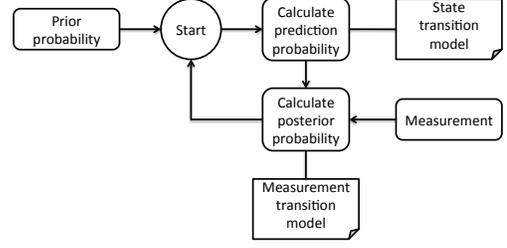


Fig. 3: General scheme of Bayesian filter.

classification decision is based on largest posterior probability of the tested sample, therefore, a probability value will be assigned to the prediction output:

$$p(C_k | \alpha) = \frac{p(\alpha | C_k) \times p(C_k)}{p(\alpha)} \quad (6)$$

where, α is the new online feature vector, C_k is the class or the group of trends that has similar EOL value, $p(\alpha | C_k)$ is the probability of observing α given C_k , is also known as the likelihood, $p(C_k)$ is class priors and $p(\alpha)$ is the marginal likelihood. In case the value of the posterior probability of a class membership, $p(C_k | \alpha)$, is less than a predefined threshold the algorithm neglects this value and infers the class of the online trend by using GPR model.

3) *Gaussian process regression*: GPR is a flexible, powerful and probabilistic approach for Bayesian inference over functions. In order to perform regression task, which aims to approximate the input online samples x to the most similar group of offline trends $y = f(x)$, GPR defines the prior for output $f(x)$ in form of distribution over functions specified by Gaussian process (GP).

$$y = f(x) + \mathcal{N}(0, \sigma_n^2) \quad (7)$$

where, y is the estimated RUL, x is the input feature vector, $f(x)$ is the underlying GP and $\mathcal{N}(0, \sigma_n^2)$ is the Gaussian noise with zero mean and σ_n^2 variance. GP function, $f(x)$, is specified by a mean function $m(x)$ and covariance function $k(x, x')$ collected for all possible pairs of the input vector x .

$$f(x) = \mathcal{GP}(m(x), k(x, x')) \quad (8)$$

Given new observation y_* , the goal is to compute the posterior probability $p(y_* | y)$, i.e., the probability of the test data y_* given the training data y . Since the assumption in GP is that data can be represented as samples from multivariate Gaussian distribution, the posterior probability distribution can be written as:

$$\begin{bmatrix} y \\ y_* \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \mu \\ \mu_* \end{bmatrix}, \begin{bmatrix} K & K_*^T \\ K_* & K_{**} \end{bmatrix} \right)$$

where, $\mu = m(x)$ for training set, $\mu_* = m(x_*)$ for test set, K is the covariance for training set, K_* is the covariance for training-test, K_{**} is the covariance for test set and T is the matrix transpose. The posterior probability for y_* follows a Gaussian distribution.

$$y_* | y \sim \mathcal{N}(\mu_* + K_* K^{-1}(y - \mu), K_{**} - K_* K^{-1} K_*^T) \quad (9)$$

where, the best estimate for y_* is the mean of this distribution which represents the RUL.

$$\bar{y}_* = \mu_* + K_* K^{-1} (y - \mu) \quad (10)$$

The uncertainty of the estimate is represented in the variance.

$$\text{var}(y_*) = K_{**} - K_* K^{-1} K_*^T \quad (11)$$

Figure 4 shows the final result of the proposed method. The uncertainty about the online trend is estimated. Then, the probability of the RUL prediction is calculated.

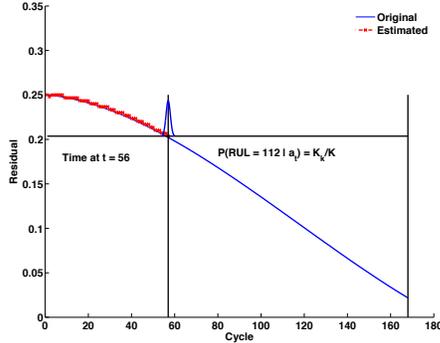


Fig. 4: Final prediction result for Battery online trend at cycle 56.

III. EXPERIMENTS

Two real life data sets were used in the experiments: a turbofan engine data sets and a lithium-ion battery aging.

A. Turbofan engine data

This is turbofan engine data set generated using commercial modular aero-propulsion system simulation (C-MAPSS) [19]. It consists of four training files, four testing files and four RUL values files. The training files contain run to failure sensor records of fleet of engines generated under different combinations of operational conditions and fault modes. Each engine is operating normally and it develops a fault at some point during the operation until finally it reaches the system failure and the engine stops. The test files are generated in the same way, however, the sensor readings are omitted prior to system failure. The RUL files contains vector of true RUL values for the test data. Each training and test file contains 26 columns of numbers that represent different variable. The first two columns represent engine number and the time in cycles respectively. The next three columns represent the operational settings. The last 21 columns, or variables, represent different time series sensor data such as total temperature at fan inlet, pressure at fan inlet, physical fan speed, etc. Each row represents a data snapshot taken during a single cycle. In this work, the data file “train_FD001.txt” is used for offline training and “test_FD001.txt” is used for online testing. Each file contains data for 100 engines and the objective is to predict the number of remaining operational cycles before failure in the test set. The true RUL values for the test data are presented in the data file “RUL_FD001.txt”.

1) *Variable selection*: one of the results of the selection algorithm is the pair of sensors number {8,13}, i.e. physical fan speed and corrected fan speed respectively, see Figure 5. The selected group is interesting as the two variables are correlated and both are related to the fan speed. Then, the

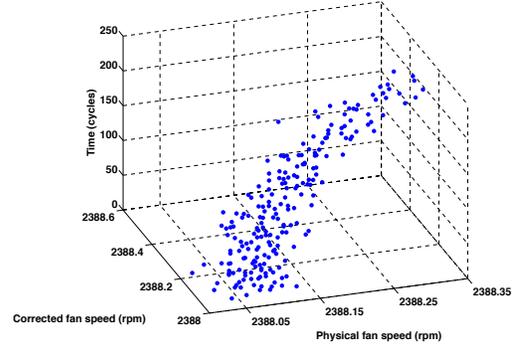


Fig. 5: The selected pair of sensors from turbofan data set.

algorithm starts constructing the monotonic trends iteratively from each pair at each time. As can be seen in Figure 2, the result of the EMD is monotonic which is an important property that help to extract health indicators.

2) *Health indicator*: as mentioned before, four features are extracted from each trend at each time and labeled with EOL time to be saved in the offline database. The features represent the relation between the extracted trends. Figure 6 shows one of the four health indicators for the Engine number 1. The indicator is monotonic and shows how the relation between the EOL and the extracted trend changes through the time. Each trend is then saved in offline database and labeled with the EOL time and will be used for predicting the RUL of new trends.

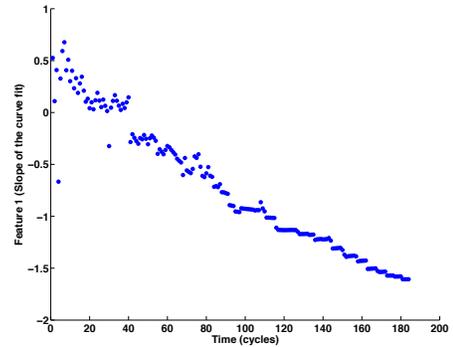


Fig. 6: The slope health indicator for Engine 1.

3) *Prediction results*: to assess the performance of the proposed method, mean absolute percentage error (MAPE) is calculated for all 100 online predictions:

$$E = \frac{100\%}{n} \times \sum_{i=1}^n \left| \frac{RUL_i - RUL_i^*}{RUL_i} \right| \quad (12)$$

where, RUL and RUL^* are the actual and predicted RUL values respectively and n is the number of total predictions.

TABLE I: Training data sets with three folds

Fold #1	Fold #2	Fold #3	EOL
B0006	B0005	B0005	168
B0007	B0007	B0006	168
B0026	B0025	B0025	28
B0027	B0026	B0026	28
B0028	B0027	B0028	28
B0030	B0029	B0029	40
B0031	B0031	B0030	40
B0032	B0032	B0031	40
B0034	B0033	B0033	197
B0036	B0036	B0034	197
B0039	B0038	B0038	47
B0040	B0040	B0039	47
B0043	B0042	B0042	112
B0044	B0044	B0043	112
B0045	B0045	B0045	72
B0047	B0046	B0046	72
B0048	B0048	B0047	72
B0050	B0049	B0049	25
B0051	B0050	B0050	25
B0052	B0051	B0052	25
B0055	B0054	B0054	102
B0056	B0056	B0055	102

The error is calculated only for the last cycles of all 100 test signals. The MAPE over the 100 test data is 11.41%. The proposed method outperforms the method presented in [13] in which the MAPE value is 15.5% for only the first 15 test engines.

B. Lithium-ion battery data

This data is collected on 34 lithium-ion batteries run through different operational profiles (e.g. charge, discharge and impedance) at different temperatures [20]. In this work only charge and discharge data are used. Each data set, corresponding to one experiment, consists of 11 variables such as charging voltage, charging current, temperature, discharging current, discharging voltage and capacity. The aging of the batteries was accelerated and the experiments continued until the batteries reached their EOL time. Each cycle is presented by the mean value to reduce the processing time. In order to validate the proposed method a 3-fold cross-validation is performed, i.e. the available data sets are partitioned into three groups of equal size. Each group is then divided into training and testing data set depicted in Table I and Table II respectively.

1) *Variable selection*: one of the results of the selection algorithm is the pair {6, 11}, i.e. the voltage measured at discharge and the capacity of the battery (Figure 7). The selected group is interesting as the two variables are correlated. Also, the capacity is related to the battery health as the decrease in the capacity indicates health degradation.

2) *Health indicator*: four features are extracted from each trend at each time and labeled with EOL time to be saved in

TABLE II: Testing data sets with three folds

Fold #1	Fold #2	Fold #3	EOL
B0005	B0006	B0007	168
B0025	B0028	B0027	28
B0029	B0030	B0032	40
B0033	B0034	B0036	197
B0038	B0039	B0040	47
B0042	B0043	B0044	112
B0046	B0047	B0048	72
B0049	B0052	B0051	25
B0054	B0055	B0056	102
B0018	B0018	B0018	132
B0041	B0041	B0041	67
B0053	B0053	B0053	55

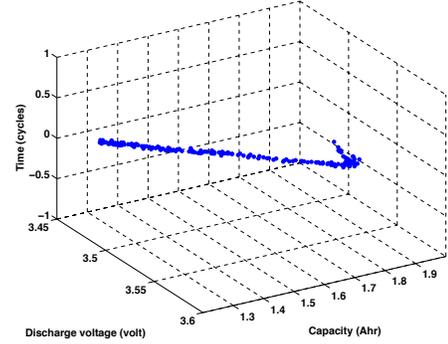


Fig. 7: Selected pair of variables from Battery data set.

the offline database. Figure 8 shows two of the four health indicators for battery B0005. The indicators are monotonic and show how the relation between the EOL and the extracted trend changes through the time.

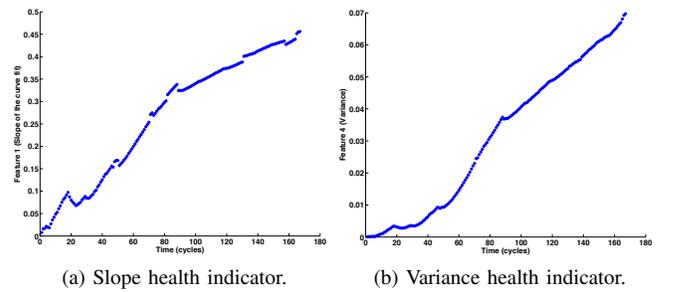


Fig. 8: Two health indicators for battery B0005.

3) *Prediction results*: to assess the performance of the proposed method, MAPE is calculated for all cycles of each battery, see Figure 9. The total MAPE per fold is calculated as follows:

$$MAPE_f = \frac{1}{n} \times \sum_{i=1}^n MAPE_{i,f} \quad (13)$$

where $MAPE_f$ is the average MAPE for a complete fold, $MAPE_{i,f}$ is the MAPE for test battery i in fold f .

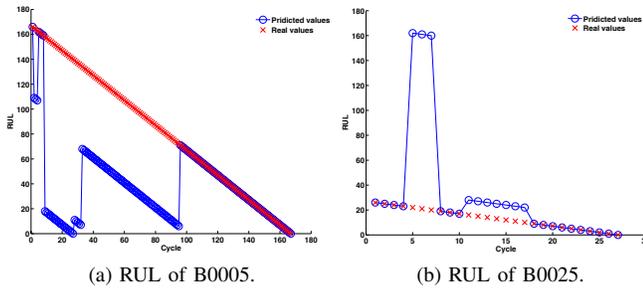


Fig. 9: Results of predicting the RUL at all cycles.

TABLE III: Mean absolute percentage error for battery data sets

Fold #1	Fold #2	Fold #3	Average
33.0966%	32.2086%	35.2726%	33.5259%

The final results are calculated and summarized in Table III.

IV. CONCLUSION

In this paper a data driven method for RUL prediction based on Bayesian approaches is proposed. The method builds on unsupervised selection of interesting variables from the input offline signals. It construct representative features that can be used as health indicators. The method represents the current status of the online signals as well as the uncertainty about the predictions in a probabilistic form. The performance of the prediction is enhanced by integrating two models, namely, k-NN and GPR. The selected variables are shown to be interesting. Moreover, the prediction results show low MAPE error for both applications and the turbofan engine results outperform the prediction of another method.

For future work, the proposed method should be tested using data sets with variable operating conditions and after introducing maintenance interventions. Also, new classification/regression models should be tested in the proposed framework.

ACKNOWLEDGMENT

The authors would like to acknowledge the European Regional Development Fund (FEDER) who funded this work as a part of a project named ALTIDE: Aid to Lifecycle Traceability for Intelligently Developed Equipment.

REFERENCES

[1] Andrew K.S. Jardine, Daming Lin, Dragan Banjevic. "A review on machinery diagnostics and prognostics implementing condition-based maintenance." *Mechanical Systems and Signal Processing*, volume 20, issue 7, pages 1483-1510, October 2006. DOI: dx.doi.org/10.1016/j.ymssp.2005.09.012.

[2] Jianfei Dong, Michel Verhaegen, Fredrik Gustafsson. "Robust Fault Detection With Statistical Uncertainty in Identified Parameters," *IEEE Transactions on Signal Processing*, volume 60, number 10, pages 5064-5076, October. 2012. DOI: 10.1109/TSP.2012.2208638.

[3] Kihoon Choi, Satnam Singh, Anuradha Kodali, Krishna R. Pattipati, John W. Sheppard, Setu Madhavi Namburu, Shunsuke Chigusa, Danil V. Prokhorov, Liu Qiao. "Novel Classifier Fusion Approaches for Fault Diagnosis in Automotive Systems," *IEEE Transactions on Instrumentation and Measurement*, volume 58, number 3, pages 602-611, March 2009. DOI: 10.1109/TIM.2008.2004340.

[4] J.Z. Sikorska, M. Hodkiewicz, L. Ma. "Prognostic modelling options for remaining useful life estimation by industry." *Mechanical Systems and Signal Processing*, volume 25, issue 5, pages 1803-1836, July 2011. DOI: dx.doi.org/10.1016/j.ymssp.2005.09.012.

[5] Mosallam A., Medjaher K., Zerhouni, N. "Nonparametric time series modelling for industrial prognostics and health management." *The International Journal of Advanced Manufacturing Technology*, volume 69, issue 5, page 1685-1699, November 2013. DOI: 10.1007/s00170-013-5065-z.

[6] Mosallam A., Medjaher K., Zerhouni N., 2013, Bayesian approach for remaining useful life prediction, *ChemicalEngineering Transactions*, 33, 139-144 DOI: 10.3303/CET1333024.

[7] Aiwina Heng, Sheng Zhang, Andy C.C. Tan, Joseph Mathew. "Rotating machinery prognostics: State of the art, challenges and opportunities." *Mechanical Systems and Signal Processing*, volume 23, issue 3, pages 724-739, April 2009. DOI: dx.doi.org/10.1016/j.ymssp.2008.06.009.

[8] R. Isermann, *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. Heidelberg: Springer-Verlag, Heidelberg, 2006.

[9] G. Vachtsevanos, F. Lewis, M. Roemer, A. Hess, B. Wu, *Intelligent Fault Diagnosis and Prognosis for Engineering Systems*, John Wiley and Sons Inc., Hoboken, New Jersey, 2006.

[10] J. Luo, M. Namburu, K. Pattipati, L. Qiao, M. Kawamoto, S. Chigusa, *Model-Based Prognostic Techniques*, Anaheim, CA, United States: 2003, Institute of Electrical and Electronics Engineers Inc., Piscataway, NJ, United States, 2003, Pages 330-340.

[11] Fakhre Chaari, Tahar Fakhfakh, Mohamed Haddar. "Analytical modelling of spur gear tooth crack and influence on gearmesh stiffness." *European Journal of Mechanics - A/Solids*, volume 28, issue 3, pages 461-468, May/June 2009. DOI: dx.doi.org/10.1016/j.euromechsol.2008.07.007.

[12] M.A. Schwabacher, *A Survey of Data-Driven Prognostic*, Infotech@Aerospace, 26 - 29 September 2005, Arlington, Virginia.

[13] Kamran Javed, Rafael Gouriveau, Nouredine Zerhouni. "Novel failure prognostics approach with dynamic thresholds for machine degradation." 39th Annual Conference of the IEEE Industrial Electronics Society, (IECON), pages 4404-4409, 10-13 November 2013 DOI: 10.1109/IECON.2013.6699844

[14] G.P. Zhang. Time series forecasting using a hybrid ARIMA and neural network model, *Neurocomputing* 50 (2003) 159-175.

[15] N. Gorjian, L. Ma, M. Mittinty, P. Yarlagadda, and Y. Sun. Review on Degradation Models in Reliability Analysis, Proceedings of the 4th World Congress on Engineering Asset Management, 2009, 28-30 Sept, Athens, Greece.

[16] Emmanuel Ramasso, Michle Rombaut, and Nouredine Zerhouni. "Joint Prediction of Continuous and Discrete States in Time-Series Based on Belief Functions." *IEEE Transactions on Cybernetics*, volume 43, number 1, pages 37-50, Feb. 2013. DOI: 10.1109/TSMCB.2012.2198882.

[17] Mosallam A., Byttner S., Svensson M., Rognvaldsson, T. "Nonlinear Relation Mining for Maintenance Prediction," *IEEE Aerospace Conference*, pages 1-9, 5-12 March 2011 DOI: 10.1109/AERO.2011.5747581.

[18] Huang, N.E., Shen, Z., Long, S.R., Wu, M.C., Shih, H.H., Zheng, Q., Yen, N.C., Tung, C.C., Liu, H.H. "The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis." *Proceedings of the Royal Society of London Series A. Mathematical, Physical and Engineering Sciences* Pages 903-995, 1998.

[19] A. Saxena and K. Goebel (2008). "C-MAPSS Data Set", NASA Ames Prognostics Data Repository, [http://ti.arc.nasa.gov/project/prognostic-data-repository], NASA Ames, Moffett Field, CA.

[20] B. Saha and K. Goebel (2007). "Battery Data Set", NASA Ames Prognostics Data Repository [http://ti.arc.nasa.gov/project/prognostic-data-repository], NASA Ames, Moffett Field, CA