2019 16th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE) Mexico City, Mexico. September 11-13, 2019

Remaining Useful Life Prediction for Turbofan based on a Multilayer Perceptron and Kalman Filter*

1st Mario Alberto-Olivares
School of Engineering
Autonomous University of Queretaro
Queretaro, Mexico
malberto26@alumnos.uaq.mx

2nd Arturo Gonzalez-Gutierrez
School of Engineering
Autonomous University of Queretaro
Queretaro, Mexico
aglez@uaq.mx

3rd Saul Tovar-Arriaga
School of Engineering
Autonomous University of Queretaro
Queretaro, Mexico
saul.tovar@uaq.mx

4th Efren Gorrostieta-Hurtado
School of Engineering
Autonomous University of Queretaro
Queretaro, Mexico
efrengorrostieta@gmail.com

***Abstract*—Today, maintenance programs are required to guarantee the reliability and availability of engineering systems. In order to do so, a system needs a degradation model to predict its remaining useful life (RUL) and act before any failure occurs. Ideally, physics-based models are used as they are accurate, but they are difficult to develop, and in complex systems, where there are many interactions, it is practically impossible. This work presents a degradation model composed of a Multilayer Perceptron (MLP) and a Kalman Filter (KF) for a common complex system, consisting of an aircraft turbine or turbofan, in order to predict its RUL. The results indicate that the model outperforms other models which are even more complex.**

***Index Terms*—Turbofan, Prognosis Health Management, Multilayer Perceptron, Kalman Filter, Remaining Useful Life**

I. Introduction

As the technological development increases, the engineering systems are more complex, especially in the areas of electronics, nuclear energy, automotive and aerospace [1]. Although a good design is essential to obtain a high reliability, the deterioration due to time, wear and working conditions, impacts directly on the performance of the systems, which implies more efficient maintenance programs [2].

Prognosis Health Management (PHM) is a term introduced in the field of medicine as the prediction of the development and outcome of a disease [3]. Derived from the same concept, several prognosis methods have been developed for machinery maintenance during the last 14 years [4]. This is a field of research and application that uses data of the current condition of the system to predict when and how the system is more prone to fail during its useful life [5]. This basically refers to the prediction of the remaining useful life (RUL), which is an important concept for decision making in preventive maintenance.

Techniques based on machine learning are appropriate when there is not enough information to understand the behavior of a system, or when a system is complex enough to develop precise physics-based models [6]. The objective is to use the data from the measurements of important variables that account for the condition of the system. For example, pressure, temperature, speed, vibration, electric current, among others, and from them to create a model that correlates the behavior of these parameters with the level of degradation of the system with the objective of predicting its RUL [7].

In this work, the common complex system that presents an aircraft turbine or turbofan is used as a case study.

II. Data Set Description

For the test and validation of the prognosis algorithm proposed in this study, the Turbofan Engine Degradation Simulation Data Set by the Prognostic Center of Excellence of National Aeronautics and Space Administration (NASA) is used [8].

This data set was created by synthetic data collected from a thermodynamic simulation model called C-MAPSS (Commercial Modular Aero-Propulsion System Simulation).

The simulator consists of 14 input parameters and 58 output signals, and from the latter only 21 are reported in the data set. Each turbofan unit provides the following information:

- ID

Authors express their gratitude to the Mexican National Council for Science and Technology (CONACYT) for its support.

978-1-7281-4840-3/19/$31.00 ©2019 IEEE

- Operation Time (in flight cycles)
- 3 Operating Condition Parameters (altitude, mach number and throttle resolver angle)
- 21 Sensor Signals (4 temperatures, 4 pressures, 6 speeds and 7 others)

The data set consists of multiple time series divided into 4 training and test subsets, both identified by the names: FD001, FD002, FD003 and FD004 (see Table I). It also contains results of the test set for verification and validation of the algorithm.

The whole data set contains sensor data from several turbofans, which operate normally at the beginning of the recording and eventually they develop a failure. There are two failure modes: high-pressure compressor degradation and fan degradation.

In the training set, the data recording ends when the turbofan stops working completely due to the failure, representing a data base of turbofans that failed during operation. On the other hand, in the test set, the time series finishes sometime before the system stops working, representing a data base of turbofans in current operation. The objective is to predict the time (in flight cycles) in which those turbofans in current operation will completely fail, i.e., to predict their RUL.

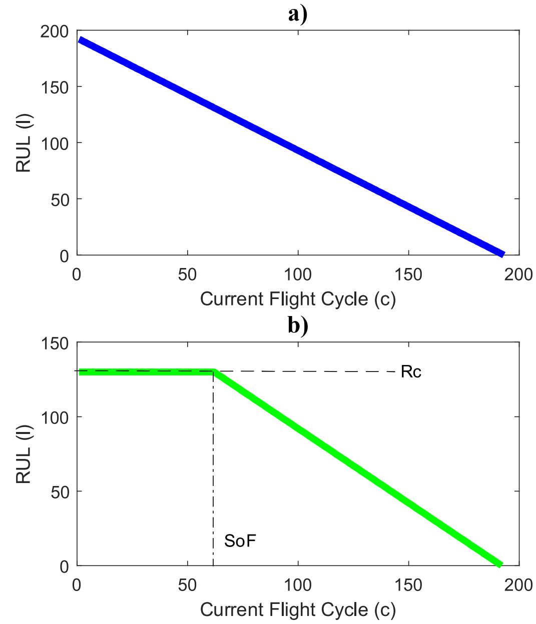

Fig. 1. a) Degradation behavior based on a linear function for unit 1 from the FD001 training set. b) Degradation behavior based on a piece-wise function for unit 1 from the FD001 training set.

TABLE I
DATA SET DESCRIPTION

	FD001	FD002	FD003	FD004
Training Units	100	260	100	249
Test Units	100	259	100	248
Number of Operating Conditions	1	6	1	6
Number of Failure Modes	1	1	2	2

III. PREPROCESSING

A. Data Labelling

In order to train an algorithm for RUL prognosis, it is necessary a set of input and output data. Where the input data is the information from several sensors and the output data is the RUL. However, databases for prognosis applications do not often contain the RUL information for training since, in many industrial applications, it is impossible to accurately assess the RUL information. Therefore, how to label the training data becomes part of the problem to solve.

A simple solution is to use directly, as RUL information, the time remaining before the end of each turbofan data recording [9] (see Figure 1, blue line). This labelling method, however, implies that the RUL as well as the health of the system decrease linearly throughout the whole life of each turbofan.

Another more appropriate labelling method is proposed by Heimes [10]. This method, based on a piece-wise function rather than a single linear function, represents the behavior of a system that begins to degrade after a certain time of operation, when some failure has occurred (see Figure 1, green line). It should be mentioned that this method has been validated as adequate for labelling this training data set [11], [12]. The piece-wise function, that this labelling method describes, is as follows:

$$l = \begin{cases} R_c & \text{if } 0 \leq c \leq SoF \\ EoL - c & \text{if } SoF < c \leq EoL \end{cases} \tag{1}$$

Where l is the label which corresponds to the RUL in number of flight cycles in operation; EoL is the number of the last cycle or end of life for the unit; c is the current flight cycle number; R_c is the initial constant value of RUL which varies from 120 to 130 cycles [10]; and SoF is the start of failure which is equal to $EoL - R_c$. In this work, the piece-wise degradation model has been used to label the data, and after several tests, R_c is set to 130.

B. Data Normalization

One of the most commonly used normalization techniques is called Z-Score, which is based on the mean and the standard deviation of the data set to scale it (equation 2). In our study the data corresponds to the signal values from sensors of temperature, speed, pressure and others. Thus the normalization of each data recovered from each sensor f is performed according to the following equation:

$$N(x^f) = \frac{x^f - \mu^f}{\sigma^f} \tag{2}$$

For the sensor f, x^f represents the data, μ^f and σ^f are the mean and standard deviation of the data set, respectively.

The creators of the data set indicate that there are three variables (altitude, mach number and throttle resolver angle) that refer to the operating conditions of the turbines and these have a strong impact on the performance of the system.

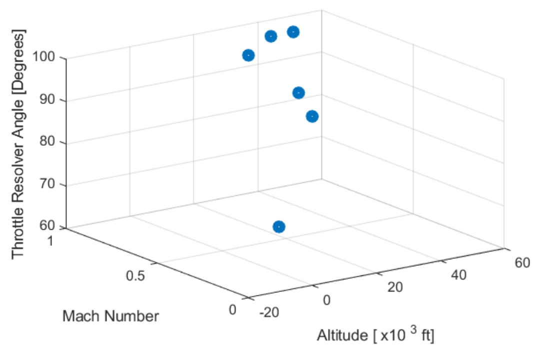

Fig. 2. Data clustering in 6 operating regimes.

When plotting the three operation variables (see Figure 2) it is important to point out that the data are concentrated in six clusters, indicating six operating regimes.

After the normalization of the data set using equation 2, the operating regimes are preserved (see Figure 3). However, Peel and Gold indicate that there is no relationship between the operating regimes and the RUL [9]. Thus, in order to ensure an equal contribution of each sensor in all regimes, it is better for the prediction task to incorporate the information of each regime in the normalization as follows [9], [13]:

$$N(x^{(r,f)}) = \frac{x^{(r,f)} - \mu^{(r,f)}}{\sigma^{(r,f)}} \tag{3}$$

Where r stands for the operating regime.

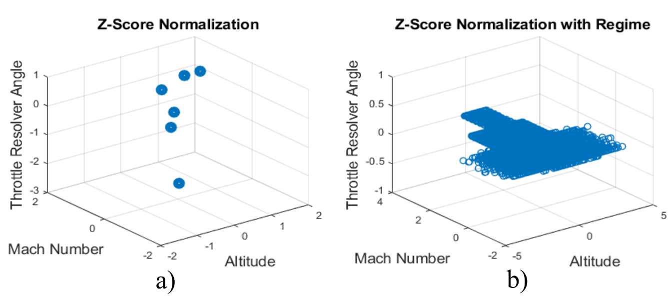

Fig. 3. Comparison of the regimes clustering after two normalization methods: a) applying Z-Score normalization equation; b) applying the Z-Score normalization along with the operating regime information.

Although the 21 sensor signals and the 3 operating condition parameters are normalized by equation 3, the flight cycles are simply divided by one thousand in order to keep their values within a range of 0 and 1.

IV. Feature Selection

The selection of sensors or components, for the input data of the algorithm, depends on the profile of the time series. In the literature referring to the Turbofan Degradation Data Set, the sensors commonly selected are those whose data has an increasing or decreasing behavior. These sensors are: 2, 3, 4, 7, 8, 9, 11, 12, 13, 14, 15, 17, 20 and 21. In addition, it has been proposed to incorporate six variables called HERI (Historical Engine Run Indicators), which are associated to the number of cycles that the turbofan has remained in each operation regimen [9], [14]. Moreover, the component of the flight cycles is not usually included for training.

The proposed model uses all the components: the 21 sensors, the 3 operating conditions and the flight cycles. For this work, the use of HERI does not improve the prediction accuracy, and this is not included for training.

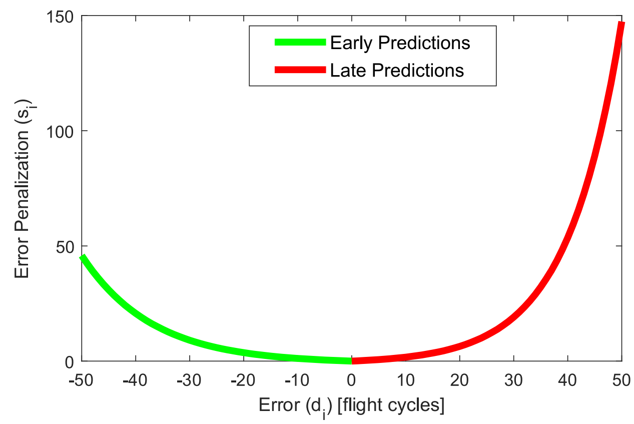

Fig. 4. EEP metric: late predictions are more penalized.

V. Metrics for Evaluation

To measure the performance of the algorithm, the Mean Square Error (MSE) is used, as follows:

$$MSE = \frac{1}{N}\Sigma_{i=1}^{N}(\hat{t}_i - t_i)^2 \tag{4}$$

Where N is the number of samples, $\hat{t}_i$ is the estimate of RUL and t_i is the true value of RUL.

In addition to the MSE, a metric called Exponential Error Penalization (EEP) is used to measure the algorithm performance. Saxena et al. proposed EEP for the competition of prognosis algorithms [15]. The EEP metric is characterized by an asymmetric function, penalizing late predictions more than early predictions (see Figure 4). An ideal algorithm, which predicts exactly the true RUL values, would obtain an evaluation of zero using this metric.

$$d_i = \hat{t}_i - t_i \tag{5}$$

$$s_i = \begin{cases} e^{-d_i/13} - 1 & \text{if } d_i \leq 0 \\ e^{d_i/10} - 1 & \text{if } d_i > 0 \end{cases} \tag{6}$$

$$S = \Sigma_{i=1}^{N} s_i \tag{7}$$

Where S is the score of the algorithm, d_i is the error and s_i is the error penalty.

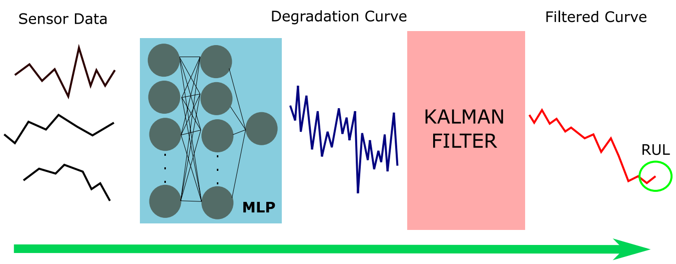

Fig. 5. Prognosis model of MLP + KF. The signals from the sensors feed the MLP, resulting in a degradation curve, which is filtered by the Kalman Filter.

VI. Prognosis Model

Recently, different prognosis models have arisen for the prediction of RUL of the turbofan, such that the Multilayer Perceptron (MLP), Recurrent Neural Networks (RNN), Long-Short Term Memory (LSTM), Convolutional Neural Network (CNN) and Deep Belief Networks (DBN). For this work a hybrid model based on an MLP and a Kalman Filter (MLP + KF) is proposed (see Figure 5).

Time series generally offer a mathematical structure whose measurement at time k is highly related to the measurement at a previous time $k-n$, where n is the delay time, which may be one or several. When the decision is to use neural networks, the best options are dynamic neural networks (LSTM or RNN). However, these are computationally more expensive for applications in embedded systems [14]. On the other hand, although the MLP is less computationally expensive, the predictions made by this model tend to be noisy and with low level of precision, due to their predictions are not related to those of the previous moment. To solve this problem, the Kalman Filter (KF) is used to correct the signal, including the prediction information at the previous instant [16].

VII. Kalman Filter

The KF is a signal processing technique based on an iterative process and data measurements with uncertainty to generate, in general, the best estimates of the variable of interest [17].

The KF algorithm basically consists of two stages: prediction and update [14].

Kalman Filter Algorithm for the Turbofan Degradation Data Set

1: $Z = \{z_0, z_1 ... z_n\}$ // Measurements from MLP
2: **for** $k \in \{1...n\}$ **do**
3: $\quad \hat{x}_k^- = A\hat{x}_{k-1}$
4: $\quad P_k^- = AP_{k-1}A^T + Q$
5: $\quad K_k = P_k^- H^T / (HP_k^- H^T + R)$
6: $\quad \hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-)$
7: $\quad P_k = (I - K_k H) * P_k^-$
8: **end for**

A. Prediction Stage

The prediction stage is defined in steps 3 and 4 of the algorithm. Where $\hat{x}_k^-$ is the *a priori* estimate of the state vector x_k at time k, A is the state transition matrix, P_k^- is the error covariance matrix associated with the *a priori* estimation and Q is the process noise covariance matrix.

Elattar et al. established the following assumptions to adapt the KF to the degradation of the turbofan [14]. First, the model of a moving object is used to calculate and update the estimate of the degradation state vector $\hat{x}_k$ and the error covariance matrix P_k. Second, it is assumed that the degradation (the "object") evolves under a constant acceleration, therefore the equation of the state vector is the following:

$$x_k = \begin{bmatrix} 1 & -\Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_{k-1} \\ \dot{v}_{k-1} \end{bmatrix} \tag{8}$$

Where v_k is the RUL, $\dot{v_k}$ is the degradation rate, and Δt, the sampling time, is equal to 1 flight cycle, since the measurements in the data set are collected at each flight cycle of the turbofan. Thus, from equation 8, $A = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}$.

The process noise covariance matrix Q, corresponding to the model of a moving object, is defined according to the following equation:

$$Q = \begin{bmatrix} \Delta t^4/4 & \Delta t^3/2 \\ \Delta t^3/2 & \Delta t^2 \end{bmatrix} \sigma_a^2 \tag{9}$$

Finally $Q = \begin{bmatrix} 1/4 & 1/2 \\ 1/2 & 1 \end{bmatrix} \sigma_a^2$, where $\sigma_a = 0.01$ is obtained by trial and error.

B. Update Stage

The update of the state vector $\hat{x}_k$ and the error covariance matrix P_k depends basically on the value of the gain K_k, calculated in step 5 of the algorithm.

The output of the MLP is used as the measurement state z_k, and this is described by the following equation:

$$z_k = H \begin{bmatrix} v_k \\ \dot{v}_k \end{bmatrix} \tag{10}$$

where $H = \begin{bmatrix} 1 & 0 \end{bmatrix}$ is the matrix that indicates the relationship between measurements and the state vector at time k, in the ideal assumption that there is no noise in the measurements. Finally, R, the noise covariance of the measurements, is equal to σ_z^2, where $\sigma_z = 0.3$ (obtained by trial and error).

C. Kalman Filter Results

The initial conditions for the KF are $v_0 = 1$ and the degradation rate $\dot{v}_0 = 1/209$, corresponding to 209 flight cycles in average.

In the Figure 6 two degradations curves, corresponding to two different turbines, are filtered using the KF.

TABLE II
COMPARISON OF DIFFERENT MLP MODELS

	Without Kalman Filter				With Kalman Filter			
	Without Flight Cycle		With Flight Cycle		Without Flight Cycle		With Flight Cycle	
Test Set	MSE	EEP	MSE	EEP	MSE	EEP	MSE	EEP
FD001	347.23	1292.46	337.89	611.78	346.14	1250.97	338.12	652.06
FD002	796.46	9544.71	742.42	8203.91	764.44	9573.01	680.17	7112.24
FD003	435.10	1664.82	337.42	798.72	409.33	1733.80	319.10	998.33
FD004	861.69	7797.96	764.66	5144.78	824.41	6184.73	742.61	5013.95
Total	704.69	20299.95	635.72	14759.19	676.08	18742.50	**602.62**	**13776.58**

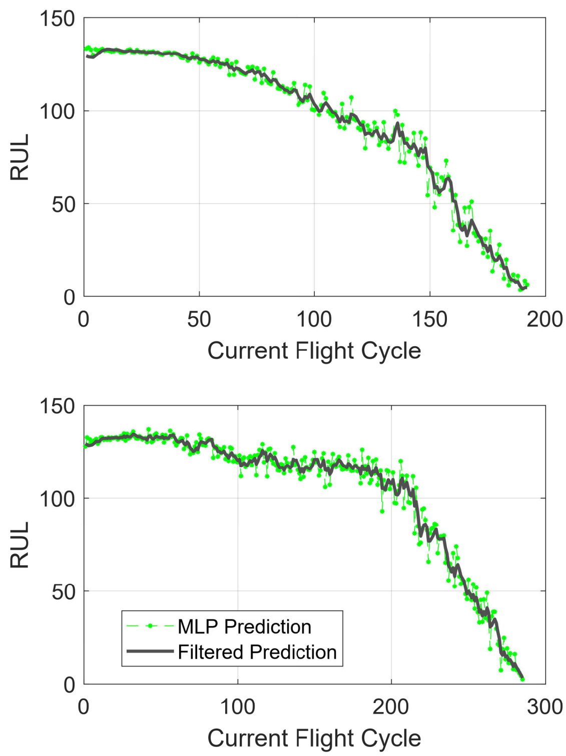

Fig. 6. KF applied to the MLP prediction (notice that the KF is applied on the normalized output but in this image the values are obtained multiplying the normalized output by $R_c = 130$).

VIII. NEURAL NETWORK ARCHITECTURE

The MLP architecture consists of three layers: (1) input layer, where the input neurons receive data corresponding to the flight cycle, 3 operating conditions and 21 sensor signals; (2) a hidden layer with 10 neurons containing a sigmoid activation function; (3) and an output layer with one neuron and its linear activation function. The output data for training are the labels generated by the piece-wise function (see equation 1). The labels are normalized to values between 0 and 1, which result from dividing them by the maximum value of $R_c = 130$.

The MATLAB Neural Network Toolbox (R2017b version) is used to create the model, along with the Levenberg Marquard training method and the Nguyen-Widrow initialization algorithm. It is used an initial learning rate of 0.001 with a decrease factor and increase factor of 0.1 and 10, respectively.

From the training set, only 70% of the training data set is used for training, while 15% is used for validation, and another 15% for tests. The MLP weight parameters are modified based on the error of the training examples, and the training process finishes when the generalization error increases. This error is calculated with the validation samples. The testing samples have no effect on the training process, as they just provide information of the MLP performance.

The method used to select the MLP hyper-parameters (number of neurons, activation function and so on) is based on trial and error. During these tests, it was noticed that increasing the number of neurons and hidden layers do not significantly improve the performance of the model.

A second MLP is trained differently by removing the flight cycles input parameter. This is done in order to show the impact of the variable on the prognosis results, since this parameter is not often incorporated for training but for this model it turned out to appreciably increase the prediction accuracy.

IX. RESULTS

Table II lists the results of different MLP proposals. The variations lie in the inclusion of the number of flight cycles in the training of the algorithm and the use of the KF in the output of the MLP. The impact of incorporating the component of the flight cycles to the training of the algorithm is clearly observed, increasing significantly the accuracy according to the EEP and the MSE. On the other hand, as it was expected, the use of the KF also increases the overall performance of the algorithm by reducing the noise of the MLP output.

A wide variety of works have been published with this turbofan data set. In order to compare the MLP + KF model, the following recent studies are selected: the work of Sateesh et al. with the first attempt of CNN for RUL prognosis [13] in 2016; a newer model of CNN by Li et al. in 2018 [12]; and two models from the work of Zhang et al. in 2017: a DBN network and a more complex model called Multiobjective Deep Belief Networks Ensemble (MODBNE) [18]. The comparisons are listed in Tables III and IV. In addition, Table V lists the inputs used by the authors for the training of each model.

The four models to be compared appear to be more sophisticated neural network architectures than the MLP of 10 neurons. However, the latter outperforms the two CNN and the DBN, although the MODBNE method continues to surpass the

TABLE III
MSE COMPARISON

Test Set	CNN [13]	DBN [18]	CNN [12]	MODBNE [18]	MLP+KF
FD001	340.32	231.34	N/A	226.20	338.12
FD002	917.72	735.49	N/A	627.50	680.17
FD003	392.72	216.38	N/A	156.50	319.10
FD004	849.72	892.81	N/A	821.39	742.61
Total	737.94	645.94	N/A	**572.13**	**602.62**

TABLE IV
EEP SCORE COMPARISON

Test Set	CNN [13]	DBN [18]	CNN [12]	MODBNE [18]	MLP+KF
FD001	1286.7	417.59	273.71	334.23	652.06
FD002	13570	9031.64	10412	5585.34	7112.24
FD003	1596.2	442.43	284.1	421.91	998.33
FD004	7886.4	7954.51	12466	6557.62	5013.95
Total	24339.3	17846.17	23435.81	**12899.1**	**13776.58**

MLP. Another aspect to note in the individual evaluations is that the proposed model, unfortunately, has higher numbers in MSE and EEP Score for the test sets FD001 and FD003, but the evaluations of the FD002 and FD004 sets have improved considerably. Previous works have shown that the evaluations in these last two sets are usually less favorable, since they have 6 operating conditions, complicating the relationships between the data.

Regarding the input data used for each model. Although the sensors that are commonly selected are those whose data have an increasing or decreasing behaviour, it was observed that the performance of the neural network increases when all features are used, which means that there is important information of those features that are generally discarded.

X. CONCLUSIONS

In this work, an RUL prognosis model for turbofan is proposed, using the Turbofan Engine Degradation Simulation Data Set by NASA. Although the use of dynamic networks such as LSTM and RNN is preferred for problems with time series, these models are more expensive computationally in embedded system applications. Thus, in this work the use of MLP and KF is adopted due to their low computational cost. The results indicate that the inclusion of the KF and a time component (flight cycles) to the MLP significantly enhances the prediction accuracy, overcoming more complex architectures such as CNN and DBN. Although this model has not managed to overcome the MODBNE, the difference in performance is relatively small.

In addition, it should be mentioned that although the labels by the piece-wise function turn out to be effective for training, they do not completely describe the pattern degradation for the training data set. Recent works begin to explore the use of Self-labelling techniques, which are used for semi-supervised learning problems, where there is a small amount of information labelled.

TABLE V
COMPARISON OF DATA INPUT USED FOR TRAINING

Model	Features Used
CNN [13]	21 Sensors and the 6 HERI
CNN [12]	Sensors: 2, 3, 4, 7, 8, 9, 11, 12, 13, 14, 15, 17, 20, 21
MODBNE [18]	Sensors: 2, 3, 4, 7, 8, 9, 11, 12, 13, 14, 15, 17, 20, 21
DBN [18]	Sensors: 2, 3, 4, 7, 8, 9, 11, 12, 13, 14, 15, 17, 20, 22
MLP+KF	21 Sensors, 3 Operating Conditions and Flight Cycles

REFERENCES

[1] G. Niu, B. S. Yang, and M. Pecht, "Development of an optimized condition-based maintenance system by data fusion and reliability-centered maintenance," *Reliability Engineering and System Safety*, vol. 95, no. 7, pp. 786–796, 2010.

[2] 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.

[3] A. Abu-Hanna and P. J. Lucas, "Prognostic models in medicine," *Methods of Information in Medicine-Methodik der Information in der Medizin*, vol. 40, no. 1, pp. 1–5, 2001.

[4] J. Lee, F. Wu, W. Zhao, M. Ghaffari, L. Liao, and D. Siegel, "Prognostics and health management design for rotary machinery systems - Reviews, methodology and applications," *Mechanical Systems and Signal Processing*, vol. 42, no. 1-2, pp. 314–334, 2014.

[5] C. Hu, B. D. Youn, P. Wang *et al.*, *Engineering Design under Uncertainty and Health Prognostics*. Springer, 2019.

[6] J. Liu, W. Wang, and F. Golnaraghi, "A multi-step predictor with a variable input pattern for system state forecasting," *Mechanical Systems and Signal Processing*, vol. 23, no. 5, pp. 1586–1599, 2009.

[7] H. M. Elattar, H. K. Elminir, and A. M. Riad, "Prognostics: a literature review," *Complex & Intelligent Systems*, vol. 2, no. 2, pp. 125–154, 2016.

[8] A. Saxena and K. Goebel, "Turbofan Engine Degradation Simulation Data Set," 2018. [Online]. Available: NASA Ames Prognostics Data Repository (http://ti.arc.nasa.gov/project/prognostic-data-repository), NASA Ames Research Center, Moffett Field, CA

[9] L. Peel, "Data driven prognostics using a kalman filter ensemble of neural network models," in *2008 International Conference on Prognostics and Health Management, PHM 2008*. IEEE, oct 2008, pp. 1–6.

[10] F. O. Heimes, "Recurrent neural networks for remaining useful life estimation," in *2008 International Conference on Prognostics and Health Management*. IEEE, oct 2008, pp. 1–6.

[11] E. Ramasso, "Investigating computational geometry for failure prognostics. Investigating computational geometry for failure prognostics," Tech. Rep. 1, 2014.

[12] X. Li, Q. Ding, and J. Q. Sun, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliability Engineering and System Safety*, vol. 172, no. December 2017, pp. 1–11, 2018.

[13] G. Sateesh Babu, P. Zhao, and X.-L. Li, "Deep Convolutional Neural Network Based Regression Approach for Estimation of Remaining Useful Life," pp. 214–228, 2016.

[14] H. M. Elattar, H. K. Elminir, and A. M. Riad, "Conception and implementation of a data-driven prognostics algorithm for safetycritical systems," *Soft Computing*, no. 2010, pp. 1–18, 2018.

[15] A. Saxena, K. Goebel, D. Simon, and N. Eklund, "Damage propagation modeling for aircraft engine run-to-failure simulation," in *2008 International Conference on Prognostics and Health Management*, vol. 41. IEEE, oct 2008, pp. 1–9.

[16] M. Baptista, E. M. Henriques, I. P. de Medeiros, J. P. Malere, C. L. Nascimento, and H. Prendinger, "Remaining useful life estimation in aeronautics: Combining data-driven and Kalman filtering," *Reliability Engineering and System Safety*, vol. 000, pp. 1–12, 2018.

[17] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, vol. 82, no. 1, p. 35, 1960.

[18] C. Zhang, P. Lim, A. K. Qin, and K. C. Tan, "Multiobjective Deep Belief Networks Ensemble for Remaining Useful Life Estimation in Prognostics," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2306–2318, 2017.