A Convolutional and Long Short-time Memory Network Configuration to Predict the Remaining Useful Life of Rotating Machinery

Hélcio Ferreira Sarabando, and Eurípedes Guilherme de Oliveira Nobrega

Abstract—**Recently, several machine learning approaches have been proposed to provide predictions of the remaining useful life of rotating machine. This study presents a strong framework that employs machine learning algorithms to predict the useful life of rotating machines by evaluating their vibration signals. In this approach, the raw vibration signal undergoes feature extraction through auxiliary methods, trend analysis through statistical methods, and time-dependent feature extraction through a specialized hybrid neural network algorithm. The architecture is composed of three distinct phases: feature analysis, where the raw vibration data are processed to extract important characteristics for the definition of the signal trend creating a time series; modeling, where the training data is processed in a hybrid convolutional neural network, which returns a degradation model aiming at estimating the instant of total failure; and prediction, where the future failure trend of the test data is identified, using the failure threshold extracted from the training data. A neural network is used to analyze test data and identify the moment just prior to the occurrence of failure. We used the architecture to predict the remaining useful life of rotating machines in various cases, and the results error ranged between 3 and 4%, which is considered a satisfactory result.**

Link to graphical and video abstracts, and to code: https://latamt.ieeer9.org/index.php/transactions/article/view/9110

Index Terms—**Convolutional network, Recurrent neural network, Wavelet transform, Short-time Fourier transform, Remaining useful life, Hybrid neural network.**

I. Introduction

Condition monitoring (CM) is the most common maintenance approach to predicting machine faults and system failures through vibration measurement. Continuous analysis of a system's operational life-cycle can bring several benefits, such as improving machine availability and productivity, and decreasing maintenance expenses. [1]. To achieve these objectives, a CM program, based on diagnostics and prognostics, can minimize the number of maintenance operations using three main steps: data acquisition, processing and informed decision making [2]. Maintenance programming based on data-driven models represent up-to-date solutions for CM and predictive management.

The associate editor coordinating the review of this manuscript and approving it for publication was Guillermo Valencia-Palomo (*Corresponding author: Hélcio Ferreira Sarabando*).

H. F. Sarabando, and E. G. de O. Nobrega are with Universidade Estadual de Campinas, Campinas, Brazil (e-mails:h028340@dac.unicamp.br, and egon@unicamp.br).

Reliable estimation of Remaining Useful Life (RUL) for industrial rotating machinery is crucial for scheduling corrective maintenance and preventing unscheduled downtime [3], [4]. However, optimizing operational efficiency in this context remains a challenging research objective. The main task in machine prognosis is to predict the RUL based on condition analysis [5]. A significant amount of research has been published on this topic in recent years. In [6] the RUL is calculated following a four-step process that includes data acquisition, definition of health indicator, degradation model construction and RUL prediction. Features extracted from machine vibration data series can reveal trends corresponding to specific machine behavior leading to failure, permitting estimation of RUL prognostics [7]. Degradation models that use health indicators are a useful way to track changes in bearing behavior, which can help predict future failures and estimate RUL [8], [9]. Machine diagnosis and prognosis are tradicionally performed using signal processing techniques to build health indicators. To estimate the operational condition of a machine it is necessary to build indicators that model the degradation of its components [10]. The respective data must be preprocessed to extract these indicators [11], providing crucial information for a proper assessment of the system's integrity.

Wavelet Transform processed vibration signals are used by data-driven models to diagnose faults in rotating machines, specifically evaluating vibration signals. According to [12], a research trend was observed focusing on wavelet applications in the field of fault diagnosis of rotating machines. Applying the Continuous Wavelet Transform (CWT) to a signal produces a series of wavelet coefficients at different scales, where various indicators can be extracted and used as input to data-driven algorithms to characterize machine healthy status. This method can detect early gear faults and is insensitive to variable loads [13]. CWT-based adaptive bandpass filters are used to track energy and reduce noise contamination in frequency bands related to time-varying faults in monitored machine signals [14], [15]. The architecture suggested in [16] aims the automatic identification of bearing failures, preprocessing the vibration signals using Wavelet Packet followed by a neural network to create degradation models.

In the past few years deep learning models have been used to predict future values for fault diagnosis using measurable variables in mechanical systems with complex fluctuations in the measured data [17]. So far, we have seen algorithms performing fault diagnosis, classifying degradation, recognizing patterns and predicting RUL of mechanical components. The challenge is to integrate diagnostic and prognostic technologies to predict and isolate impending failures to help maintain system performance cost effectively. The deep learning approach using neural networks has been used for system health monitoring to model high-level data representation and predict patterns by stacking multiple layers of information processing modules into hierarquical structures. These structures are able to process massive amounts of data in a new bottom-up diagnostic paradigm. In [18] the vibration data is pre-processed by scaled Discrete Fourier Transform. The signal is collected on a mechanical experimental platform and fed into a Convolutional Neural Network (CNN) capable of autonomously learn useful features for bearing failure detection. In [19] a CNN based on LeNet-5 with two alternating convolutional-pooling layers and two fully connected layers is used to perform bearing fault diagnosis. In [20] an intelligent method, named Deep Recurrent Neural Network (DRNN), constructed through the stacking of Long Short-Term Memory (LSTM) cells, is used to automaticaly learn valuable features from the input spectrum sequences of vibration bearing data.

In [21], a LSTM and a Gated Recurrent Unit (GRU) are used to analyze the time series properties of vibration data to predict the abnormal states of electrical motors in a drone. The LSTM network is successfully applied in [22] to predict the RUL for mechanical components of turboprop aircraft engines, using multidimensional data and adding life labels in the data preprocessing stage.

A framework based on a hybrid CNN-LSTM network is used in [23] for Fault Detection and Diagnosis (FDD) on rotating machinery, aided by Fast Fourier Transform (FFT), CWT and raw signal statistical analysis to provide a deeper understanding of the fault's identity.

The proposed method for predicting RUL of a machine has three important modules: feature analysis, modeling, and prediction. Firstly, the feature analysis module decomposes vibration signals using function transforms to extract local spectral and temporal information with noise reduction. In modeling, a degradation model is generated using a hybrid convolutional and recurrent neural network and an autoregressive signal, enabling spatio-temporal sequence prediction. Finally, in prediction, post-processing and filtering functions are used to detect shifts in the data, and a high-degree function optimization curve is used to determine the RUL by extrapolating the trend of the post-processed data. The main contribution of this method is the combination of transforms and a hybrid convolutional and recurrent neural network aided by statistical techniques based on time-domain and signal energy determination to discover trends, which enhances the accuracy of the modeling and prediction modules. Overall, this three-module method is effective in predicting the RUL of rotating machines.

The other sections of this article are organized as follows: Section II substantiates the definition of RUL, the fundamental and theoretical bases of the techniques used in the algorithms developed for the proposed architecture. Section III presents the proposed architecture to determine the RUL in detail, with its feature analysis, modeling and prediction phases. In Section IV a case study is presented, including experimental outcomes, using a publicly accessible dataset. In Section V, the conclusions about the study are presented. Finally, in Subsection V-A some discussions about the performed experiments are developed.

II. Theoretical Foundation

The RUL of a machine consists of the estimation of its remaining operating time, generally used to schedule a corrective maintenance. Indicators are extracted from analyzed machine data, which are then used as inputs for degradation models [24]. These models predict the RUL by detecting trends in machine behavior analysis. The method used to determine the RUL depends on how the data is made available, namely: lifetime data, indicating how long it took for similar machines to reach failure; run-to-failure data derived from past records of machines that share similarities with the one under analysis; and threshold data, known from threshold values of a condition indicator also extracted from previous histories. This work focuses on the run-to-failure data acquisition method, where the used database contains behavior data of similar components. Data from a standard component is registered until the moment the failure occurs, then we process degradation profiles and compare them with new data from similar components to define the future point in time for a failure to occur.

Wavelet analysis and short-time Fourier transform (STFT) are employed to analyze non-stationary signals due to their higher sensitivity compared to conventional Fourier techniques. They excel in detecting signal components, identifying discontinuities and irregularities, accomplishing this by multiplying the signal with an analysis function and integrating it in the time domain [25], [26]. The Wavelet is a small wave, with energy concentrated in time, and a tool for the simultaneous analysis of time and frequency, of transient, non-stationary or time-varying phenomena [27].

CNNs are commonly used for image recognition and are composed of 3 types of layers: convolutional, pooling, and fully connected [28], [29]. In the convolutional layer, filters calculate the dot product between weights and regions of the input volume to produce a n-dimensional activation map that learns specific features. The output volume of the layer is formed by stacking these activation maps along the depth dimension [28]–[30].

The convolution layer produces an activation map, which is submitted to the pooling layer for dimension reduction. After one or more of these layers, the last map is presented to a fully connected layer for final data classification [28]–[30].

The proposed architecture combines a CNN and an LSTM network to model the degradation of a failure. The CNN extracts features from the input data, which are then fed into the LSTM network [31] to predict future behavior.

The LSTM network is made up of memory cells [32], which have three inputs (input gate, output gate, and forget gate). The state of the network will then be stored by this recurrent backpropagation, in the chain of modules by the signal C_t. The input port (i_t) is responsible for deciding whether new data flows internally to the cell or not. Output port (o_t) defines whether the state of cell *(C_t)* has any effect on other cells. The forget gate *(f_t)* defines whether the cell remembers the previous state *(C_{t-1})* [31], [32]. To analyze an LSTM cell we need to define an input data set *(x_t)* at a given time instant *t* [28], [33]. We get an output vector *(h_t)* by inserting the data vector *(x_t)* into the cell. Internally weight matrices, *($\boldsymbol{W}_i, \boldsymbol{W}_f, \boldsymbol{W}_c, \boldsymbol{W}_o, \boldsymbol{U}_i, \boldsymbol{U}_f, \boldsymbol{U}_c, \boldsymbol{U}_o, \boldsymbol{V}_o$)*, is stored, necessary for the activation of the internal gates, and the polarization vectors or *bias* *($\boldsymbol{b}_i, \boldsymbol{b}_f, \boldsymbol{b}_c$ and $\boldsymbol{b}_o$)*, both used in the equations that define the values of the gates of the LSTM cell. We can then calculate the values of the input port *(i_t)* and the possible (candidate) values of the cell state *($\widetilde{C}_t$)*, using Equations (1) and (2).

$$i_t = \sigma(\mathbf{W}_i.\mathbf{x}_t + \mathbf{U}_i.\mathbf{h}_{t-1} + \mathbf{b}_i) \tag{1}$$

$$\widetilde{C}_t = \tanh(\mathbf{W}_c.\mathbf{x}_t + \mathbf{U}_c.\mathbf{h}_{t-1} + \mathbf{b}_c), \tag{2}$$

where σ is a *sigmoid function* and tanh is a *hyperbolic tangent function*, both gate activation functions.

Equation 3 defines the activation of the forget gate *(f_t)*.

$$f_t = \sigma(\mathbf{W}_f.\mathbf{x}_t + \mathbf{U}_f.\mathbf{h}_{t-1} + \mathbf{b}_f) \tag{3}$$

With the values of gates i_t, $\widetilde{C}_t$ (cell state candidate) and f_t, defined by Equations (1), (2) and (3), it is now possible to determine the value of the current state of the cell *(C_t)*, using Equation (4). The value of the output port *(o_t)* can be obtained from Equation (5), right after determining the cell state value, defined by Equation (4). Then, it is possible to determine the output value of cell *(h_t)* in possession of the current state value of the cell, using Equation (6).

$$C_t = i_t * \widetilde{C}_t + f_t * C_{t-1} \tag{4}$$

$$o_t = \sigma(\mathbf{W}_o.\mathbf{x}_t + \mathbf{U}_o.\mathbf{h}_{t-1} + \mathbf{V}_o.C_t + \mathbf{b}_i) \tag{5}$$

$$h_t = o_t.\tanh(C_t) \tag{6}$$

The internal matrix of weights is updated during network training through a Backpropagation Through Time algorithm (BPTT), a gradient-based technique for training the LSTM network, unwinding and determining the gradient at all time steps.

Exponentially Weighted Moving Average (EWMA) is a type of Infinite Impulse Response (IIR) filter, where filtered data is represented as a weighted sum of previous measurements, which gives greater weight and meaning to more recent points of the data with the aim of smoothing the data series [34]. Thus, EWMA can be used to remove high-frequency noise, exponentially averaging a point with previous measurements, in a defined window of data, as we see in the Equation (7)

$$\widehat{x}_t = \alpha x_t + (1-\alpha)\widehat{x}_{t-1}, \tag{7}$$

where $\widehat{x}_t$ is the EWMA of the data window, x_t is the sample data of the data window, $\widehat{x}_{t-1}$ is the exponentially weighted average of the data window immediately preceding it, α is a smoothing parameter, adjustable between zero and one, which defines the cutoff frequency above which values are dropped.

The EWMA filter weights show that the filter coefficients drop exponentially depending on the α parameter, giving more importance to the most recent measurements. Higher α values cause the filter coefficients to fall faster, increasing the cutoff frequency, however, lower α values decrease the exponential fall of the coefficients, decreasing the cutoff frequency [34].

Statistical indicators such as Root Mean Square (RMS) and Kurtosis (KU), given respectively by Equation (8) and Equation (9), are used to calculate the overall energy and shape of the supplied signal, as auxiliary methods of processing raw data to extract a trend [11], [35].

$$RMS = \sqrt{\frac{1}{N}.\sum_{i=1}^{N}(x_i)^2}, \tag{8}$$

$$KU = \frac{\sum_{i=1}^{N}(x_i-\mu)^4}{(N-1)\sigma^4}, \tag{9}$$

where x_i is the *i-th* point of the sample data, in the time domain, of length *N*. Support Vector Regression (SVR) is a supervised learning algorithm used here to predict future values [36], [37]. The basic idea is to find the hyperplane in the feature space that best represents the data in a consistent way. Having a vector of training data $\{(x_1, y_1), \cdots, (x_l, y_l)\} \subset X\ x\ \Re$, where X denotes the space of input patterns (e.g., $X = \Re^d$), the regression *ϵ-SV* must find a function *f(x)* that has at most ϵ deviation from the actually obtained targets y_i, for all training data. Its primary objective is to minimize the error or distance between the predicted values and the actual values.

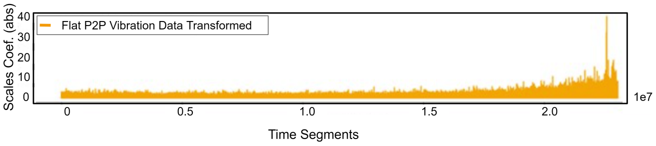
III. Methodology

The proposed architecture is composed of 3 phases called feature analysis, modeling and prediction, linked in a serial, sequential way. The goal is to predict when a catastrophic failure might occur by processing training data with the target architecture's algorithms to determine a threshold. The test data is then processed with the same algorithms and the trend of the truncated data is extrapolated to find the future time when the failure threshold is reached. The RUL is defined as the time until this point.

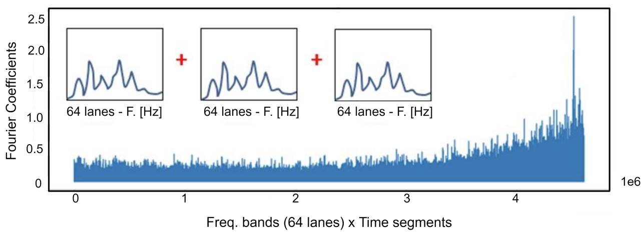
A. Feature Analysis

The vibration signal is firstly analyzed using STFT and Discrete Wavelet Transform (DWT). The Wavelet transform adopts a Mexican hat function to extract a centered time series with positive values. This time series only considers the differences between the maximum and minimum values of the original data in the time domain, resulting in a vector with values ranging from zero to the peak-to-peak value of the signal, as can be seen in Fig. 1a.

The STFT is applied to vibration signals from a test dataset to study changes in frequency over time. The signal sampling frequency is 25.6 kHz, and the STFT, considering only the positive frequencies and calculated with a 50% overlap, results in 64 frequencies bands, as can be seen in Fig. 1b. It creates a time series of the signal transformed by the STFT, the windows is connected consecutively in time, resulting in a periodogram. This can be seen in the close-up illustration shown in Fig. 1b. It is important to note that the processing of the vibration data from the training set was done using the same methods and with the same parameters as described above.



(a) Series composed by the transformed Wavelet Ricker



(b) Series composed by the STFT transform

Fig. 1. Time series of vibration measurements, extracted from test data

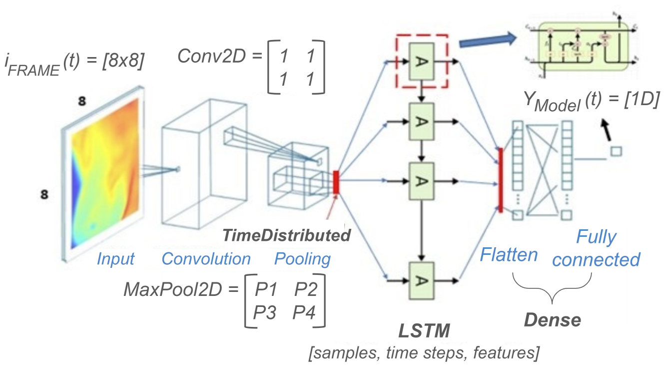
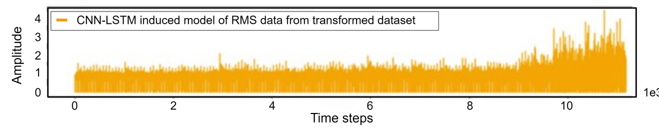


Fig. 2. Schematic diagram of the proposed CNN-LSTM hybrid neural network

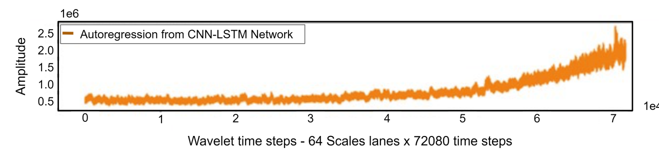
B. *Modeling*

A degradation model is built based on the detected evolution of each STFT along time, feeding the decomposed data into a CNN-LSTM to extract hidden features in the analyzed data. Fig. 2 presents the architecture of the proposed neural network. The convolutional layer applies filters to extract local features from an input frame of size 8x8. Activation functions introduce non-linearity, pooling layers reduce spatial dimensions while retaining crucial features, time distributed layers help reshape the data into a two-dimensional format suitable for LSTM layers across multiple time steps. Finally, dense layers are utilized as the last layers of the network, enabling feature extraction, non-linear transformations, and serving as the output layers.

Before entering the transformed vibration data to the convolutional layer, the windowed RMS is calculated, using Equation (8) where each data window has 64 sample points giving rise to a new time series $\mathbf{Y}(t)$. This new time series forms a two-dimensional frame with dimension 8 by 8 sample points, to be inserted in the convolutional layer of the neural network. The frames were processed by convolution with 2×2 kernels and the ReLU activation function, followed by a 2×2 max pooling. The flattened result was transformed into a 1D vector and becomes a 3D tensor $(n \times 3)$, with the help of the TimeDistributed layer to fit the input format of the LSTM submodel. The network is trained with 1024 data batches over 5 epochs, using a single LSTM layer of 128 cells. One network was trained on the full dataset to build the degradation model, while the other network, trained on 70% of the data set, was used for prediction, reserving 30% for testing. The models were built by auto-predicting the input data using a series of one-step ahead predictions from the training data and five-step ahead predictions from the test data. The training data (Bearing1_1) was transformed by the Wavelet method and is shown in Fig. 3a while the test data (Bearing1_3) is shown in Fig. 3b. These datasets were obtained from the PRONOSTIA platform [38], as described in Section IV-A



(a) Training data degradation model



(b) Test data degradation model

Fig. 3. Time series produced by autoregression of data on the CNN-LSTM network

C. *Prediction*

From the degradation model it is possible to extract a threshold for the occurrence of a catastrophic failure in the training data and project a possible occurrence of failure in the future time in the test data. Both the training data degradation model was transformed by calculating the energy of its time series in 64-sample point windows using the equation: $Energy_{window} = \sum_{i=1}^{N}(U_i^2)$, where N is the number of sample points in the window (64) and U_i is the *i-th* sample point in the data window. The EWMA of the entire time series was then calculated using the energy. The training results are shown in Fig. 4, which displays the time series of the signal energy modeled with the transformed training data, and the progression of the data in the last 1280 sample points in three different modes: simple moving average, linearly weighted moving average, and EWMA

The operating limit or threshold of a rotating machine is determined from training data and represents the point of total failure. The highest point in the mean amplitude is considered the failure threshold. Test data is evaluated against this threshold, with the goal of detecting a failure before it occurs. Fig. 5 shows the time series transformed by Wavelet, described from the signal energy modeled with the test data, and the moving averages. Moving average window size and last data window details are shown in the legend of Fig. 4.

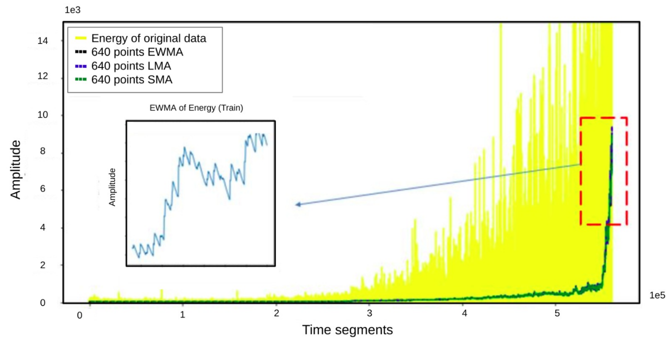


Fig. 4. Energy time series and EWMA of the modeled signal (training - Bearing1_1), highlighted in the last data window

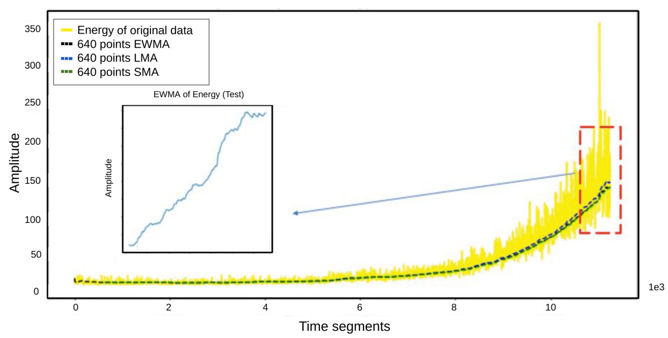


Fig. 5. Energy time series and EWMA of the modeled signal (test - Bearing1_3), highlighted in the last data window

IV. Results

A. *Experimental platform*

Fig. 6 presents the PRONOSTIA test bench. The experimental setup features a 250 W asynchronous motor connected to a gearbox and two shafts: one near the motor and the other positioned near an incremental encoder. The motor transfers rotational motion through the gearbox, and shaft couplings ensure the motion is transmitted to the shaft support bearing. The bearing support shaft is fixed with a shoulder on one side and a threaded locking ring on the other.

The PRONOSTIA platform presents two datasets. One is labeled as the training set, which contains vibration data from the moment the bearing starts operating until a catastrophic failure occurs. This dataset is known as "run-to-failure". The training dataset is named Bearing1_1 and represents the vibration data for bearing 1 in operating condition 1 with 1800 RPM speed and 4000 N of force applied to the shaft. The second dataset, called the test set, is intended to be used to adopt the previously trained models to determine the RUL. For this reason, vibration measurement sequences are truncated at some point before reaching the failure level. The test set is named Bearing1_3 and represents the vibration data for bearing 3 in operating condition 1. The vibration signals were captured using two accelerometers positioned at 90° from each other, one on the vertical axis (Y) and one on the horizontal axis (X) of the bearing. This study used only the vibration data in the vertical direction and made no assumptions about the type or origin of failure. The tests were ended when the vibration signal amplitude exceeded 20g (training set).

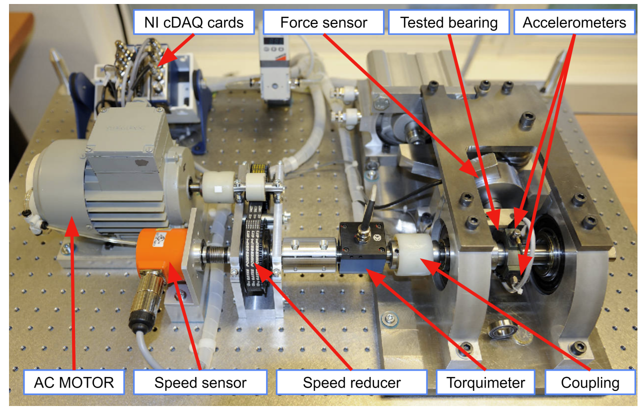


Fig. 6. Overview of the PRONOSTIA experimental platform, adapted from [38]

The accelerometer signals were acquired during 0.1 s with a sampling frequency of 25.6 kHz, which results in 2560 discrete points, with measurements repeated at intervals of 10 seconds.

B. *Experimental Outcomes*

The data trend was projected using a combination of EWMA curve, exponential function, and a third degree polynomial function. The regression was optimized based on the trend of the data in the test dataset, and the results were transformed using Wavelet. The failure threshold was defined using the training data, and the estimated point of failure was found. The results are shown in the Fig. 7 and include the polynomial and exponential curves, the estimated point of failure, and the RUL provided by the PRONOSTIA platform.

The RUL is calculated by subtracting the time between the last recorded measurement in the model and the predicted future failure. However, caution must be taken as the dataset only includes sample data and the measurement cycle's latency times are missing. The time series data was also compressed using EWMA, Linear Weighted Moving Average (LWMA) e Simple Moving Average (SMA), but this compression was taken into account in the RUL calculation.

It is worth noting that there was a late prediction, with a smaller time difference between the predicted value and the actual value, which was made with test data pre-processed by STFT, with the regression of the EWMA data trend performed by an SVR algorithm, and the extrapolation of future data made from a third degree polynomial curve, having its coefficients optimized using the trend regression curve of the measured data. Fig. 8 shows the result of this regression and extrapolation of the data, indicating the RUL.

The proposed method was compared with similar works carried out by [39]–[43] on the same dataset. The results of the comparison are shown in Table I, which presents the RUL estimation absolute errors (in seconds) and the respective percentages for each method in the analysis of the Bearing1_3 test. The proposed method obtained the best performance among the compared methods, exhibiting the lowest RUL estimation error.

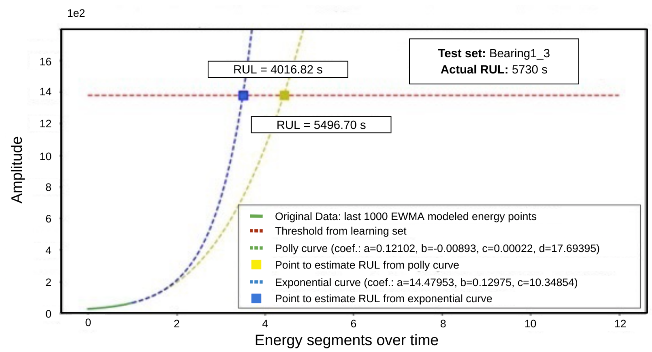


Fig. 7. Data extrapolation curves with the curves molded to the trend of the progression of the moving average of the energy of the signal transformed by the Wavelet

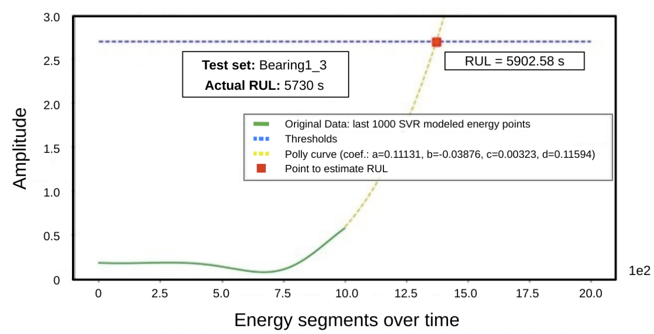


Fig. 8. Extrapolation curves of the data with the curve shaped to the trend of the progression of the SVR regression of the energy of the transformed signal by the STFT

TABLE I
COMPARISON TABLE OF SIMILAR WORKS

Method	Actual RUL (s)	Estimated RUL (s)	Error (s)	Error (%)
Proposed Method	5730	5902	172	3.0
[39] Method 2	5730	5440	290	5.1
[40]	5730	5293.5	436.5	7.6
[41]	5730	3604	2126	37.1
[42]	5730	Not provided	Not provided	4
[43]	5730	4731	989	17.3

V. CONCLUSIONS

The proposed architecture has three phases: feature analysis, modeling, and prediction. Each phase acts as a filter or feature extractor on the data. This chain of filters is designed to process the data and extract specific information at each step. To create a prognostic indicator, time series compression was necessary between phases or during signal processing.

Non-linear function transforms were applied to the data to extract local and temporal features from the vibration signal spectrum. Hybrid neural networks were used to create models for fault diagnosis and predicting failures. Vibration data was processed using an autoregression prediction model, followed by an EWMA filter and extrapolation using a polynomial function. This resulted in a RUL of forecast error between 3% and 4%, based on the previously established threshold value.

The proposed architecture proved to be robust and accurate in determining the RUL in relation to other similar methods tested with the same dataset, as compared in Subsection Subsection IV-B, confirming its effectiveness.

A. Discussions

The best-performing architecture in predicting the RUL of a rotating machine on the PRONOSTIA platform achieved 5496 seconds for case A and 5902 seconds for case B. It involved preprocessing the raw vibration signal using Ricker-type Wavelet transform and using a CNN-LSTM hybrid neural network to extract the degradation model (Bearing1_1) from a dataset with run-to-failure type. The model extracted the failure occurrence threshold, and the test data (Bearing1_3) was preprocessed using the same Wavelet transform. Using the CNN-LSTM hybrid network with the same configuration, the degradation model was built from the test data. To detect potential equipment failure, two methods were used:

- In Case A, data was postprocessed using an EWMA filter and modeled with exponential and third-degree polynomial functions, then extrapolated to predict failure time.

- In Case B, the data was pre-processed using STFT and modeled using an SVR on energy data obtained from a CNN-LSTM hybrid neural network. The resulting curve was used to predict failure time.

The PRONOSTIA platform proponents measured a remaining lifetime of 5730 seconds after data truncation for the test dataset bearing (Bearing1_3).

- In Case A, the best prediction was 5496 seconds using a 3rd degree polynomial curve, with an early prediction of 234 seconds (4% of the data range).

- In Case B, using the same method, the late RUL was 5902 seconds with a delay of 172 seconds (3% of the data range), starting with energy data degradation model regression.

In this case, although the early forecast was preferred, the late forecast had a smaller error. To optimize the forecast error, new sample data can be used to adjust the forecast each time new operating data is collected. Processing a new dataset and adjusting the forecast can decrease the confidence interval and improve the forecast.

ACKNOWLEDGMENTS

The authors thank CAPES for financial support (Process nº 88882.435215/2019-01), and the AS2M1 department of the FEMTO-ST Institute, Besançon – France, for making the datasets of the *PRONOSTIA* experimental platform public.

REFERENCES

[1] A. K. Jardine, D. Lin, and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance," *Mech Syst Signal Process*, vol. 20, pp. 1483–1510, Oct. 2006. doi: 10.1016/j.ymssp.2005.09.012.

[2] L. Polverino, R. Abbate, P. Manco, D. Perfetto, F. Caputo, R. Macchiaroli, and M. Caterino, "Machine learning for prognostics and health management of industrial mechanical systems and equipment: A systematic literature review," *International Journal of Engineering Business Management*, vol. 15, p. 18479790231186848, Jul. 2023. doi: 10.1177/18479790231186848.

[3] F. Ahmadzadeh and J. Lundberg, "Remaining useful life estimation: review," *International Journal of System Assurance Engineering and Management*, vol. 5, pp. 461–474, Sep. 2013. doi: 10.1007/s13198-013-0195-0.

[4] X.-S. Si, W. Wang, C.-H. Hu, and D.-H. Zhou, "Remaining useful life estimation – a review on the statistical data driven approaches," *European Journal of Operational Research*, vol. 213, pp. 1–14, Aug. 2011. doi: 10.1016/j.ejor.2010.11.018.

[5] S. Xiang, P. Li, Y. Huang, J. Luo, and Y. Qin, "Single gated rnn with differential weighted information storage mechanism and its application to machine rul prediction," *Reliability Engineering & System Safety*, vol. 242, p. 109741, Feb. 2024. doi: 10.1016/j.ress.2023.109741.

[6] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, and J. Lin, "Machinery health prognostics: A systematic review from data acquisition to rul prediction," *Mech Syst Signal Process*, vol. 104, pp. 799–834, May 2018. doi: 10.1016/j.ymssp.2017.11.016.

[7] M. J. Roemer, C. S. Byington, G. J. Kacprzynski, G. Vachtsevanos, and K. Goebel, *Prognostics*, ch. 17, pp. 281–295. John Wiley & Sons, Ltd, May 2011. doi: 10.1002/9781119994053.ch17.

[8] M. Kaji, J. Parvizian, and H. W. van de Venn, "Constructing a reliable health indicator for bearings using convolutional autoencoder and continuous wavelet transform," *Applied Sciences*, vol. 10, p. 8948, Dec. 2020. doi: 10.3390/app10248948.

[9] B. P. Duong, S. A. Khan, D. Shon, K. Im, J. Park, D.-S. Lim, B. Jang, and J.-M. Kim, "A reliable health indicator for fault prognosis of bearings," *Sensors*, vol. 18, p. 3740, Oct. 2018. doi: 10.3390/s18113740.

[10] J. C. A. Jauregui Correa and A. A. Lozano Guzman, "Chapter eight - condition monitoring," in *Mechanical Vibrations and Condition Monitoring* (J. C. A. Jauregui Correa and A. A. Lozano Guzman, eds.), pp. 147–168, Academic Press, 2020. doi: 10.1016/B978-0-12-819796-7.00008-1.

[11] V. Atamuradov, K. Medjaher, F. Camci, N. Zerhouni, P. Dersin, and B. Lamoureux, "Machine health indicator construction framework for failure diagnostics and prognostics," *J Signal Process Syst*, vol. 92, pp. 591–609, Jun. 2020. doi: 10.1007/s11265-019-01491-4.

[12] R. Yan, R. X. Gao, and X. Chen, "Wavelets for fault diagnosis of rotary machines: A review with applications," *Signal Processing*, vol. 96, pp. 1–15, Mar. 2014. doi = 10.1016/j.sigpro.2013.04.015.

[13] X. Wang, V. Makis, and M. Yang, "A wavelet approach to fault diagnosis of a gearbox under varying load conditions," *Journal of Sound and Vibration*, vol. 329, pp. 1570–1585, Apr. 2010. doi: 10.1016/j.jsv.2009.11.010.

[14] W. Yang, P. J. Tavner, C. J. Crabtree, and M. Wilkinson, "Cost-effective condition monitoring for wind turbines," *IEEE Transactions on industrial electronics*, vol. 57, pp. 263–271, Jan. 2010. doi: 10.1109/TIE.2009.2032202.

[15] B. Tang, W. Liu, and T. Song, "Wind turbine fault diagnosis based on morlet wavelet transformation and wigner-ville distribution," *Renewable Energy*, vol. 35, pp. 2862–2866, Dec. 2010. doi: 10.1016/j.renene.2010.05.012.

[16] H. Shao, H. Jiang, F. Wang, and Y. Wang, "Rolling bearing fault diagnosis using adaptive deep belief network with dual-tree complex wavelet packet," *ISA Transactions*, vol. 69, pp. 187–201, Jul. 2017. doi: 10.1016/j.isatra.2017.03.017.

[17] S. Khan and T. Yairi, "A review on the application of deep learning in system health management," *Mechanical Systems and Signal Processing*, vol. 107, pp. 241–265, Jul. 2018. doi: 10.1016/j.ymssp.2017.11.024.

[18] O. Janssens, V. Slavkovikj, B. Vervisch, K. Stockman, M. Loccufier, S. Verstockt, R. Van de Walle, and S. Van Hoecke, "Convolutional neural network based fault detection for rotating machinery," *J Sound Vib*, vol. 377, pp. 331–345, Sep. 2016. doi: 10.1016/j.jsv.2016.05.027.

[19] L. Wen, X. Li, L. Gao, and Y. Zhang, "A new convolutional neural network-based data-driven fault diagnosis method," *IEEE Transactions on Industrial Electronics*, vol. 65, pp. 5990–5998, Jul. 2018. doi: 10.1109/TIE.2017.2774777.

[20] H. Jiang, X. Li, H. Shao, and K. Zhao, "Intelligent fault diagnosis of rolling bearings using an improved deep recurrent neural network," *Measurement Science and Technology*, vol. 29, p. 065107, May 2018. doi: 10.1088/1361-6501/aab945.

[21] J.-K. Hong, "Vibration prediction of flying iot based on lstm and gru," *Electronics*, vol. 11, no. 7, pp. 1052 – 1068, 2022. DOI: 10.3390/electronics11071052.

[22] L. Tang, S. Zhang, X. Yang, and S. Hu, "Research on prognosis for engines by lstm deep learning method," in *2019 Prognostics and System Health Management Conference (PHM-Qingdao)*, pp. 1–8, IEEE, 2019. doi: 10.1109/PHM-Qingdao46334.2019.8942976.

[23] M. Jalayer, C. Orsenigo, and C. Vercellis, "Fault detection and diagnosis for rotating machinery: A model based on convolutional lstm, fast fourier and continuous wavelet transforms," *Computers in Industry*, vol. 125, p. 103378, Feb. 2021. doi: 10.1016/j.compind.2020.103378.

[24] M. Soualhi, K. T. Nguyen, and K. Medjaher, "Explainable rul estimation of turbofan engines based on prognostic indicators and heterogeneous ensemble machine learning predictors," *Engineering Applications of Artificial Intelligence*, vol. 133, p. 108186, Feb. 2024. doi: 10.1016/j.engappai.2024.108186.

[25] T. Guo, T. Zhang, E. Lim, M. López-Benítez, F. Ma, and L. Yu, "A review of wavelet analysis and its applications: Challenges and opportunities," *IEEE Access*, vol. 10, pp. 58869–58903, Jun. 2022. doi: 10.1109/ACCESS.2022.3179517.

[26] N. Kehtarnavaz, "Chapter 7 - frequency domain processing," in *Digital Signal Processing System Design (Second Edition)* (N. Kehtarnavaz, ed.), pp. 175–196, Burlington: Academic Press, second edition ed., 2008. doi: 10.1016/B978-0-12-374490-6.00007-6.

[27] Y. T. Chan, *Wavelet Basics*. Springer US, 1995. doi: 10.1007/978-1-4615-2213-3.

[28] M. Tovar, M. Robles, and F. Rashid, "Pv power prediction, using cnn-lstm hybrid neural network model. case of study: Temixco-morelos, méxico," *Energies*, vol. 13, p. 6512, Dec. 2020. doi: 10.3390/en13246512.

[29] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. P. and, "A review of convolutional neural networks in computer vision," *Artificial Intelligence Review*, vol. 57, pp. 57–99, Mar. 2024. doi: 10.1007/s10462-024-10721-6.

[30] A. Ajit, K. Acharya, and A. Samanta, "A review of convolutional neural networks," in *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, (Vellore, India), pp. 1–5, IEEE, Feb. 2020. doi: 10.1109/ic-etite47903.2020.049.

[31] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, pp. 1735–1780, Nov. 1997. doi: 10.1162/neco.1997.9.8.1735.

[32] S. M. Al-Selwi, M. F. Hassan, S. J. Abdulkadir, A. Muneer, E. H. Sumiea, A. Alqushaibi, and M. G. Ragab, "Rnn-lstm: From applications to modeling techniques and beyond—systematic review," *Journal of King Saud University - Computer and Information Sciences*, vol. 36, pp. 1224–1236, Sep. 2024. doi: 10.1016/j.jksuci.2024.102068.

[33] X. Ran, Z. Shan, Y. Fang, and C. Lin, "An lstm-based method with attention mechanism for travel time prediction," *Sensors*, vol. 19, p. 861, Feb. 2019. doi: 10.3390/s19040861.

[34] M. N. Nounou and B. R. Bakshi, *Chapter 5 - Multiscale Methods for Denoising and Compression*, vol. 22 of *Data Handling in Science and Technology*. Elsevier, 2000. doi: 10.1016/S0922-3487(00)80030-1.

[35] V. Kannan, T. Zhang, and H. Li, "A review of the intelligent condition monitoring of rolling element bearings," *Machines*, vol. 12, p. 484, Jul. 2024. doi: 10.3390/machines12070484.

[36] M. Awad and R. Khanna, "Support vector regression," in *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*, pp. 67–80, Berkeley,CA: Apress, Apr. 2015. doi: 10.1007/978-1-4302-5990-9_4.

[37] L. Kou, M. Sysyn, J. Liu, S. Fischer, O. Nabochenko, and W. He, "Prediction system of rolling contact fatigue on crossing nose based on support vector regression," *Measurement*, vol. 210, p. 112579, Mar. 2023. doi: 10.1016/j.measurement.2023.112579.

[38] P. Nectoux, R. Gouriveau, K. Medjaher, E. Ramasso, B. Chebel-Morello, N. Zerhouni, and C. Varnier, "Pronostia: An experimental platform for bearings accelerated degradation tests.," in *IEEE International Conference on Prognostics and Health Management, PHM'12.*, (Denver, Colorado, United States.), pp. 1–8, IEEE Catalog Number: CPF12PHM-CDR, 2012.

[39] B. Wang, Y. Lei, N. Li, and N. Li, "A hybrid prognostics approach for estimating remaining useful life of rolling element bearings," *IEEE Transactions on Reliability*, vol. 69, pp. 401–412, Jan. 2020. doi: 10.1109/TR.2018.2882682.

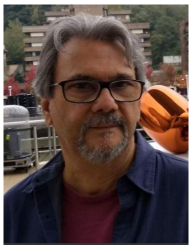
[40] Y. Chen, G. Peng, Z. Zhu, and S. Li, "A novel deep learning method based on attention mechanism for bearing remaining useful life prediction," *Applied Soft Computing*, vol. 86, p. 105919, Jan. 2020. doi: 10.1016/j.asoc.2019.105919.

[41] E. Sutrisno, H. Oh, A. S. S. Vasan, and M. Pecht, "Estimation of remaining useful life of ball bearings using data driven methodologies," in *2012 IEEE Conference on Prognostics and Health Management*, pp. 1–7, Jul. 2012. doi: 10.1109/ICPHM.2012.6299548.

[42] L. Ren, W. Lv, and S. Jiang, "Machine prognostics based on sparse representation model," *Journal of Intelligent Manufacturing*, vol. 29, pp. 277–285, Feb. 2018. doi: 10.1007/s10845-015-1107-8.

[43] J. Zhu, N. Chen, and W. Peng, "Estimation of bearing remaining useful life based on multiscale convolutional neural network," *IEEE Transactions on Industrial Electronics*, vol. 66, pp. 3208–3216, Jun. 2019. doi: 10.1109/TIE.2018.2844856.

Hélcio Ferreira Sarabando is a Ph.D. candidate in Mechatronics Engineering at UNICAMP, specializing in mechanical component prognosis using machine learning techniques. He holds a Master's degree in Aeronautical and Mechanical Engineering (ITA, 2015) and a Bachelor's degree in Electronic Engineering (Santa Cecília University, 2001). Bringing expertise in Artificial Intelligence and Machine Learning, he previously served as an Instructional Technician at SENAI and as a university lecturer at Anhanguera University, teaching engineering courses.



Eurípedes Guilherme de Oliveira Nóbrega holds a degree in Electronic Engineering from the Technological Institute of Aeronautics (1973), a MsC in Mechanical Engineering from the Federal University of Paraíba (1979) and a Ph.D. in Electrical Engineering from the University of Campinas (1992). He also developed a postdoctoral project at the University of Houston (1999/2001). He was an Invited Professor, for one didactic semester, in 2003 at ENSIM (Engineering School of University of Maine, at Le Mans), in 2010 at the Grenoble INP (Polytech Institute), and in 2012 at the Arts et Métiers ParisTech (ENSAM), at Paris, France. He is a Full Professor at the Department of Computational Mechanics, University of Campinas, Brazil. His research areas are multidiciplinary, with an emphasis on Robust and Fault-Tolerant Control Systems and Condition Monitoring, based on Artificial Intelligence, Data-Driven Modelling and Digital Signal Processing methods, applied to Fault Detection and Diagnosis and Structural Health Monitoring. In the area of Mechatronics, he has been working with autonomous robots and flexible mechanical structures, designing of FPGAbased embedded systems for control and monitoring, based on the analysis of vibrations and wave propagation. He has also developed methods for interpretation of electrocardiograms and worked on the development of diaphragm micropumps with pneumatic and electromagnetic actuation.