POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Rotor Loads Prediction via Virtual Sensors - Blending Physics and AI

Laurea Magistrale in Mechanical Engineering - Ingegneria Meccanica

Author: Federico Midei

Advisor: Prof. Claudio Sbarufatti

Co-advisor: Alberto Angelo Trezzini

Academic year: 2022-2023

1. Introduction

This thesis proposes a novel approach for predicting rotor loads using virtual sensors that combine physics-based modelling and artificial intelligence (AI) techniques, specifically merging AI predictions coming from aeromechanical parameters with a rotor damper physical model. [19] Previous studies have shown the viability of data-driven approaches and the benefits of physics-informed predictions for virtual sensors. [2, 8, 11] By capturing the fundamental principles of rotorcraft component mechanics and leveraging historical data, the proposed approach improves the accuracy of blade displacement predictions. The thesis outlines the development process of virtual sensors, including input variable selection, model formulation, and parameter estimation. Simulation results demonstrate the effectiveness of the approach in predicting rotor component loads. The study concludes by highlighting the advantages of blending physics-based modelling and AI techniques in virtual sensor development for rotor loads prediction, with the potential for practical implementation of prognostics in the future.

2. Literature Review

Digital Twin, Virtual Sensors and Prognostics are interconnected concepts that have gained attention in engineering and technology. Prognostics estimate the Remaining Useful Life (RUL) of a system, leading to a shift in maintenance and operation strategies. Different approaches to prognostics include reliability-based, physics-based, data-driven and hybrid methods. [6]
Data-driven approaches face challenges such as noisy data, uncertainty and lack of physical knowledge, which can be addressed through hybrid approaches.
Digital Twins and Predictive Maintenance (PdM) are critical components of Industry 4.0, with DTs providing a virtual analysis of physical performance.
Coexistence of physical and virtual sensors is necessary to manage computational costs. Optimization of virtual sensors using methods like recurrent neural networks improves dynamic behaviour prediction. [1, 17]
The Digital Twin (DT) technology creates virtual replicas of physical assets or systems for real-time monitoring, analysis and optimization. It has gained significance in energy, healthcare, manufacturing and aerospace industries.
In the energy sector, the DT has been employed

to predict network loads in Italy, utilizing machine learning techniques like neural networks. ENEA's technical report highlights this application [3].
In healthcare, the DT is utilized to digitalize traditional medicine and enhance patient-centred experiences. Challenges encompass technical issues, infrastructure, communication, security and ethical concerns. Cardiology benefits from DT implementation by improving diagnosis, treatment guidelines and outcomes [4, 14].
Manufacturing, specifically CNC manufacturing, employs DTs for machinery prognostics and predictive maintenance. CNC machine tools (CNCMT) are critical for product quality and predictive maintenance using DTs aids in fault identification and prevention [16].
The aerospace industry demonstrates the DT's potential through Sikorsky's Rotorcraft Digital Twin (RDT). The RDT acts as a producer and consumer of data, supporting design engineers and technicians with access to DT data for analysis. It captures the unique experiences of individual machines and components, facilitating continuous analysis and improvement through online learning algorithms [12].
These examples illustrate the broad applications of the DT across industries, empowering better decision-making, optimization and predictive maintenance.
The field of predicting main rotor loads using neural networks has a rich history. In 1993, a study titled "Prediction of Helicopter Component Loads Using Neural Networks" utilized single-layer feed-forward neural networks (also referred to as Multi-Layer Perceptron, MLP) to predict vertical load, blade bending moment and blade damper load in the rotor system. The neural network model achieved high correlation coefficients, outperforming traditional regression approaches [11].
Subsequent research in 1997 and 1998 focused on rotor system load monitoring and oscillatory load prediction using neural networks. These studies demonstrated the potential of neural networks in predicting rotor system vibratory loads, improving safety and maintenance practices in the aviation industry [2, 8].
In recent years, attention has been drawn back to this subject. In 2019, a study proposed a neural network-based solution for real-time prediction of rotor loads on an AW609 tilt-rotor. The study introduced a harmonic decomposition approach to predict loads and feed-forward neural networks were trained to predict individual harmonics [7]. In 2020, another study employed neural networks to infer the relation between flight mechanics parameters and rotor loads, reconstructing load time history by combining predictions from multiple neural networks [10]. Additionally, a research paper from Airbus presented an end-to-end solution for load recognition and damage estimation, using Multi-Layer Perceptron algorithms [9].
The future of rotor load prediction is expected to involve the use of Transformers and CNN technologies, as researchers explore new applications of deep learning algorithms. It is crucial to consider data quality and explore techniques like Principal Component Analysis (PCA) to reduce data dimension and improve computational efficiency. Additionally, implementing network structures that predict single parameters per harmonic component could enhance prediction performance [10].
Overall, while challenges exist in implementing digital twin technology in the aerospace industry, utilizing neural networks for load prediction has shown to be promising. The use of virtual sensors and improved data management can facilitate the implementation of digital twin technology, leading to enhanced decision-making, reduced downtime and improved system performance.

3. Flight data analysis

A well-structured and organized data set, along with thorough data cleaning and analysis, is crucial for achieving reliable and reproducible results in data-intensive projects, particularly in machine learning applications, where the quality of the data provided significantly impacts the model's performance and outcomes. Therefore, following a structured arrangement including problem statement, data collection, data cleaning, exploratory data analysis (EDA), feature engineering and selection and modelling phases is essential for a successful data-driven project focused on predicting trim and loads in the rotor system of civil helicopter AW169.
The first step in the data collection is to define which data to collect, in this work, the data col-

lection was divided into two phases, the input and output data that are hereafter listed.
Input data:

- Main rotor mast torque
- Tail rotor mast torque
- Rotor RPM
- Main Rotor collective command pitch
- Main rotor cyclic longitudinal
- Main rotor cyclic lateral
- Pedal
- Pitch angle
- Roll angle
- Body pitch rate
- Body roll rate
- Body yaw rate
- Pressure corrected altitude
- Outside air temperature
- Airspeed
- Load factor
- True Airspeed, blended with ground speed (blend_speed)

Output data:

- Blade lag angle (white blade, yellow blade)
- Blade flap angle (white blade, yellow blade)
- Blade pitch angle (white blade, yellow blade)
- Main rotor lead-lag damper link axial load

Regarding the rotor trim data (the blade angle of the list above), an intermediary step was required due to the unavailability of the aforementioned data. Specifically, based on the investigation conducted by Alberto Trezzini et al. [5], all the necessary procedures for determining the blade parameters, namely lag, flap and pitch angles, were identified and verified.

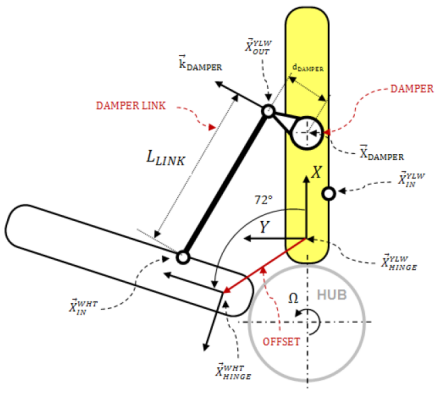


Figure 1: Damper inter blade architecture. [5]

The rotor system of the AW169 helicopter is characterized by an inter-blade architecture. In fact, in this model, the anti-ground resonance damper (lead-lag damper) is mounted between two consecutive blades as shown in 1. The need to monitor this component characterizes the main reason why it was considered necessary to have knowledge of blade movements.

Figure 2: Weight on Wheel signal applied to select portions of the flight, highlighted in orange

The input and output data in this study were collected only during actual flight periods using a "weight on wheel" sensor, which detects whether the helicopter is in contact with the ground or not, ensuring hole-free signals and facilitating the detection of flats.

4. Damper physical model

In this section, the physical model of the lead-lag damper, which predicts the force exerted on the damper link inter-blade component, is described. The transformation from blade angles to damper rotations is explained and an iterative process similar to Trezzini's work is adopted to calculate the damper system rotation based on the measured angles of adjacent blades. The geometry of the system is illustrated in Figure 1. The damper rotation can be obtained as an unknown parameter imposing the congruence between the damper link length L_{link} and the distance of the inboard and outboard points (X_{in}, X_{out} in the mentioned figure). The above results in:

$$\|\vec{X}^{w}_{in} - \vec{X}^{y}_{out}\| = L_{link} = \|\vec{X}^{*w}_{in} - \vec{X}^{*y}_{out}\| \quad (1)$$

In eq 1, the symbol $*$ stands for the undeformed case configuration.
A lead-lag damper is a mechanical system comprising a lead damper and a lag damper, where

the lead damper reduces vibration amplitude through a tuned spring-mass-damper system and the lag damper decreases phase lag by being placed in parallel with the vibrating component.

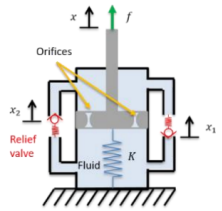


Figure 3: Scheme of the lead-lag damper

The damper considered in this work has the characteristics to mix two actions: the response of a high-viscosity fluid and the response of the deformation of an elastomeric material, the scheme in Fig 3 represents a simplified linear version of the rotative damper in exam. [19]
The overall $\hat{f}_w$ force exerted by the damper on the damper-link, is given by the force generated by the fluid pressure distributed over the piston area and a stiffness component:

$$\hat{f}_w = A_p\,\delta(\dot{x}) + kx \tag{2}$$

Be noted that this equation represents a linear spring acting in parallel with the piston. The reason is that as the elastomeric material deformation and fluid compressibility are governed by complex conditions but as demonstrated by Zilletti et al. [19], this simplified model configuration is enough to represent a good approximation of the damper response.

5. Data cleaning

The chosen method in this thesis for detecting spike samples involved computing the Z-score index over a moving window. The Z-score for each sample is calculated based on its mean and standard deviation, resulting in the equation:

$$z_i = \frac{(x_i - \mu_i)}{\sigma_i} \tag{3}$$

Where x_i is the generic sample of the i^{th} window, while μ_i and σ_i are the mean value and standard deviation of the mentioned window.
The outcome of the proposed method is hereafter presented on an input parameter time series data:

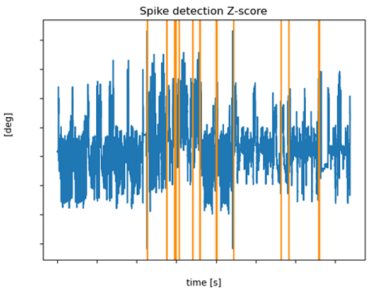


Figure 4: spike detected by the Z-score algorithm, MR cyclic lateral command parameter

Data cleaning is emphasized as a crucial step to address errors and inconsistencies, ensuring accuracy and reliability for obtaining precise insights from the data.

6. Harmonic time history

A method called Harmonic Time History (HTH) was implemented to decompose the dynamics of main rotor blade movements. The HTH algorithm divided the time series signal into chunks based on the main rotor revolution frequency and a harmonic decomposition was performed within each chunk to approximate the signal as a sum of harmonic components as shown in the following:

$$y^i(t) \approx A_0^i + \sum_{n=1}^{N} C_n^i cos(n\Omega t) + S_n^i cos(n\Omega t) \tag{4}$$

In eq 4, the coefficients A_0^i, C_n^i, S_n^i will be referred as "static coefficient", "sine and cosine terms coefficients" respectively. The signals were processed with the HTH algorithm and reconstructed back to the time domain for comparison. The accuracy of the reconstruction was evaluated using the normalized root mean squared error (NRMSE), which indicated how well the estimated signal matched the original signal. This algorithm was implemented since, as discovered in Trezzini and Graziani's previous works [7, 10], training an MLP capable to predict the complete dynamics of the main rotor blade movements (blade loads in the case of [7, 10]) could be too demanding. However, this involved an MLP architecture per harmonic considered. This results in multiple MLPs that predict the A_0, S and C coefficients reported in

eq 4. The NRMSE metric was then used as a pointer to indicate the minimum number of harmonics required to sufficiently reconstruct the original signal. An example of the white blade lag angle time series reconstruction involving the original signal decomposed taking into account a discrete number of harmonics is hereafter reported. As can be seen, the more harmonics were included in the HTH decomposition, the more the reconstructed signal overlapped the original time series.

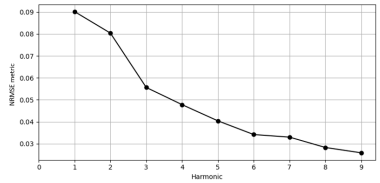
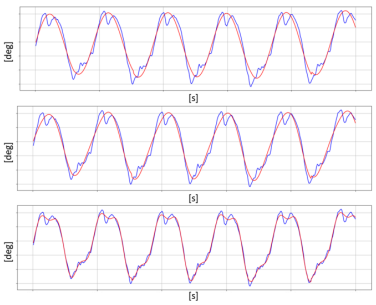


Figure 5: Accuracy in time series reconstruction, Lag angle

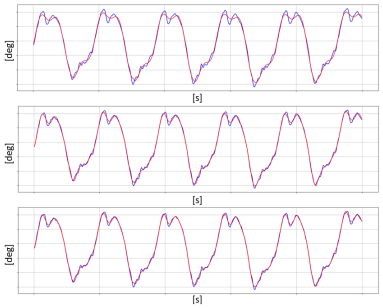
The NRMSE score proved to be effective in the final choice to have the following discretization hereafter reported for each output parameter:

- Lag : 3^{rd}harmonic
- Flap: 2^{nd}harmonic
- Pitch: 1^{st}harmonic

Additionally an example of the lag angle time reconstruction is proposed:



(a) Lag, from 1 to 3 harmonics



(b) Lag, from 4 to 6 harmonics

Figure 6: Lag, time reconstruction after HTH algorithm, from 1 to 6 harmonics

7. Feature engineering & selection

Feature engineering and selection is a vital component of machine learning and data analysis, focusing on the creation, identification, transformation and selection of relevant features (variables) to enhance the performance and effectiveness of predictive models. It plays a critical role in extracting valuable insights from raw data and improving the model's predictive capabilities.

7.1. Principal component analysis

PCA was used in this study to analyze the dataset and reduce its dimensionality while preserving the most significant patterns. Although PCA can provide valuable insights, it has the drawback of losing the physical interpretation of the features. Unfortunately, in this work, PCA was not successful in capturing the underlying structure of the dataset.

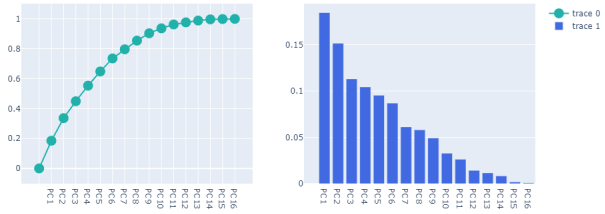


Figure 7: PCA Scree plot and cumulative variance

The Scree plot and cumulative distribution of principal components were analyzed in Fig 7, revealing that at least ten principal components would be needed to explain a significant amount of data variance. However, considering the drawbacks of PCA and the absence of the typical knee curve in the cumulative scree plot, the decision was made not to implement PCA in this study.

7.2. Feature importance

Feature importance is a metric used in feature selection methods to quantify the contribution of each feature in a dataset to the predictive power of a model, helping prioritize the most influential features. Feature importance and Shapley additive explanations are important concepts in

explainable artificial intelligence (XAI) research. Feature importance refers to the quantification of the importance of various features or covariates in a model, while Shapley additive explanations are a popular model-agnostic method for measuring feature importance in XAI. [13, 18] Shapley values are a set of axioms from cooperative game theory that provide a unique way to assign credit to each feature in a model's prediction. [13] In this work, SHAP [15] python library helped in obtaining a ranked order of the features participation in the models' decisions. Moreover, thanks to the Shapley values concept it was possible to show in a specific case how the single feature was affecting the final prediction of the MLP model in exam.
The Shapely value of a feature value is its contribution to a payout, weighted and summed over all possible feature value combinations:

$$\phi_j(val) = \sum_{S \subseteq 1,\dots,p \setminus j} \frac{|S|!(p-|S|-1)!}{p!} * \\ *(val(S \cup j) - val(S)) \quad (5)$$

Explaining the Shapley value equation (eq 5) with the coalition game theory, it's obtained that:

- $p!$: "number of ways" to form a coalition for player p (the features)
- $|S|$: number of players in coalition S (subset of features)
- $|S|!$: number of ways coalition S can form
- $(p - |S| - 1)!$: number of players that can join after player j has joined coalition S. The minus 1 is necessary since the player j has already joined the coalition
- the term $\frac{|S|!(p-|S|-1)!}{p!}$, defines the weight
- $val(S \cup j)$: value of the coalition including player j
- $val(S)$: value of the coalition excluding player j
- $\frac{|S|!(p-|S|-1)!}{p!}(val(S \cup j) - val(S))$: marginal contribution of player j to coalition S

SHAP feature importance provides a measure of the importance of each feature based on the magnitude of feature attributions. It computes the average absolute Shapley value per feature across the data and sorts the features accordingly. This approach differs from permutation feature importance, which relies on the reduction in model performance. Here is reported an example of the feature importance plot in the case of the static coefficient A_0 pitch parameter is reported.

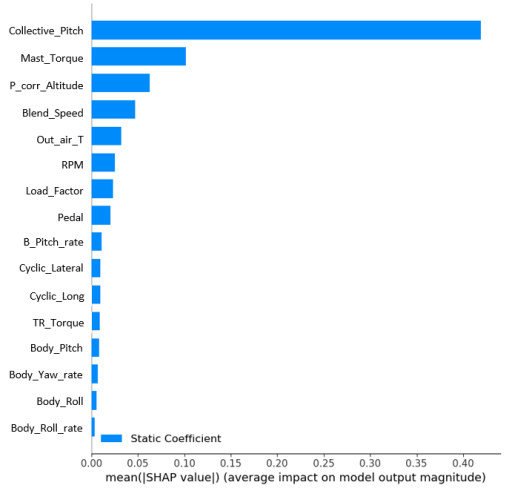


Figure 8: Feature importance summary plot, model that predicts pitch A_0 coefficient

Moreover, as anticipated, thanks to the decision plot in Fig 9 it is shown how the single feature was affecting the final prediction of the pitch A_0 MLP model examined.

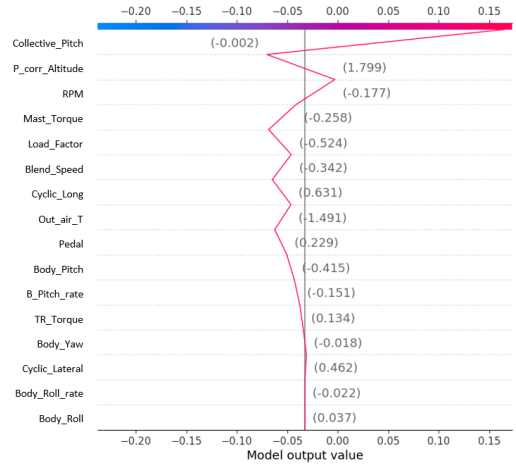


Figure 9: Decision plot, model that predicts pitch A_0 coefficient

7.3. Data Standardization

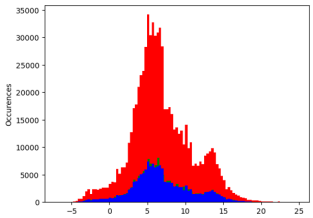
The data standardization process generally involves the implementation of two common al-

gorithms, normalization and standardization, whose implementation depends upon the specific case considered. The followings, represents the normalization min-max scaler and the data standardization equations respectively

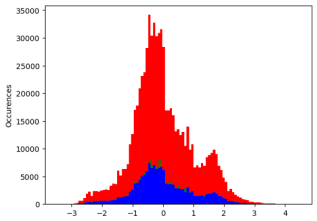
$$x_{new} = \frac{x - x_{min}}{x_{max} - x - min} \tag{6}$$

$$x_{std} = \frac{x - \mu}{\sigma} \tag{7}$$

The results of tests indicated that the min-max normalization scaler was not suitable for the problem at hand, leading to the adoption of the standardization process. Following best practices, the mean and standard deviation values were computed based on the training set and then applied to the test set, validation set and the training set itself. Test, training and validation sets will be explained later. The histograms in Fig 10a and Fig 10b display the distribution of data before and after the standardization process, respectively.



(a) Distribution of the A_0 coefficient data of an input parameter, before standardization



(b) Distribution of the A_0 coefficient data of an input parameter, after standardization

Figure 10: Histograms representing data standardization process

As expected, data presented a unit standard deviation and a zero-centred mean value after the standardization process.

8. Results

Python was selected as the working environment for this work. The training phase utilized several libraries, including Scikit-learn for tasks such as data set splitting into Training-Validation-Test subsets and data standardization. Additionally, Keras, built on top of TensorFlow, was employed for constructing the models.

In this master's thesis, the problem at hand was a direct continuous supervised learning regression problem. This led to the selection of appropriate loss and evaluation metrics. The loss function chosen was Mean Squared Error (MSE), while the evaluation metrics included MSE and Mean Absolute Error (MAE). The Mean Absolute Percentage Error (MAPE) was not considered as it is more suitable for different types of datasets.

Regarding regularization, the L2 regularizer, along with the early stopping callback, was employed. The optimizer used in this work was ADAM, which is an extension of the gradient descent algorithm. Last, the Rectified Linear Unit (ReLU) activation function was applied to the hidden layer units, while the output layer used a linear activation function.

To optimize the hyperparameters of the Multilayer Perceptrons (MLPs), an optimizer was implemented to explore different combinations of hyperparameter coefficients. For example, the L2 coefficient was set to 0.001, same holds for the ADAM coefficient. This random search optimization was also performed for the number of layers and the number of neurons per layer. However, a sensitivity analysis was conducted to avoid overfitting, as the search consistently resulted in six layers with a high number of neurons per layer (ranging from 100 to 400 neurons). The sensitivity analysis focused on the lag parameter MLPs, where a fixed number of well-performing neurons was chosen and the number of hidden layers was manually adjusted to achieve acceptable results. The results were evaluated relying on the learning plots (loss, MSE, MAE), regression plot dispersion and correlation coefficient. In table 1 the number of layers and neurons regarding the MLPs involved in this study were reported.

8.1. Model robustness

To assess the robustness of the defined and trained models, several tests were conducted. Typically, comparing the error metrics between the training and validation sets and evaluating the model's performance on the test set

are sufficient to assess robustness. However, in this thesis work, additional efforts were made to understand the generalization properties of the trained models.
The data set was split into training, test and validation subsets using random sampling with percentages of 70%, 15% and 15% respectively while keeping a constant "seed" value. This practice ensures results reproducibility but may lead to a model that performs well only with that particular split. To address this, training was also performed with different "seeds" where the samples were shuffled. Results showed that the overall performance of the models remained unvaried with both seeds, highlighting good generalization properties of the MLPs built.
To further assess the model's generalization capabilities, a specific test was conducted using a different approach. The training and validation subsets were created using all flight data except for one particular flight, which was reserved as the test set. This flight was not seen by the model during training, ensuring an evaluation of its performance on unseen data. The evaluation showcased the model's performance on previously unseen flight data, affirming its capacity to generalize to new instances within the overall range of flight data.

8.2. Model time predictions

To have a visual representation of the architecture governing the virtual sensor predictions, Fig 11 comes in help. This structure was replicated for each output parameter considered in this work.

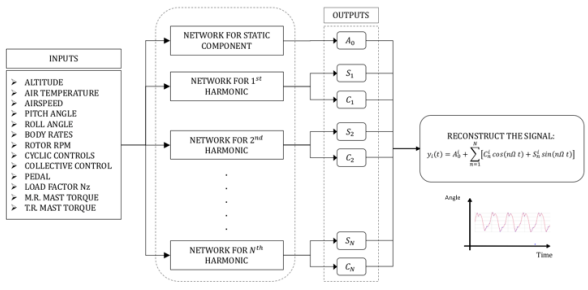
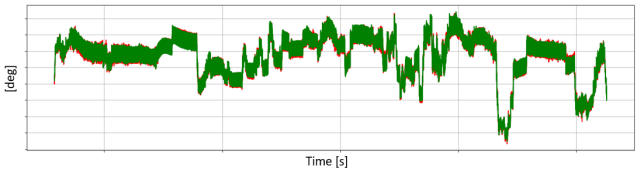
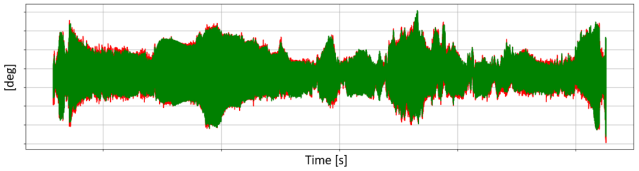


Figure 11: Structure of the MLPs in the generic case of n harmonics

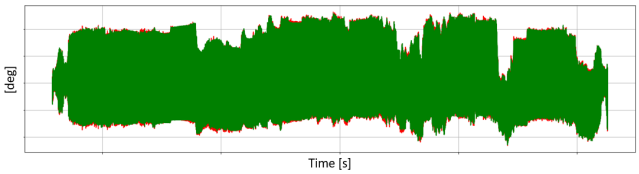
Similarly to what was shown in section 6, here will be presented a comparison between the reconstructed time series obtained from the predictions and the time series reconstructed using the Harmonic Time History (HTH) algorithm on the original data. Visual plots and NRMSE metrics are used to assess the analysis results.



(a) Lag comparison



(b) Flap comparison



(c) Pitch comparison

Figure 12: Time series reconstructed from predictions over time series reconstructed from HTH algorithm, case of a generic flight

In Figure 12 the green line represents the reconstructed time signal from the predictions, while the red line represents the HTH algorithm's reconstruction. Overall, the green line closely overlaps with the red line for all the angles.
Table 2 reports the NRMSE errors calculated for the reconstructed time series in Figure 12. The NRMSE values, rounded to the third decimal, demonstrate the matching between the prediction-HTH time series reconstructions. By comparing the reconstructed time series from the predictions with those obtained using the HTH algorithm, these results confirm a satisfactory accuracy of the MLPs' predictions.

8.3. Damper model simulation

This work implements a damper physical model based on predicted blade angles. Lag, flap and pitch angles from adjacent blades are used to calculate damper rotations along with their corresponding velocities. An iterative solver solves the non-linear equation for damper rotation and rotation speeds are obtained through numerical

differentiation. To meet Simulink ODE's minimum step size requirement, rotation and velocity are resampled at eight times the original rate using an integrated FIR anti-aliasing lowpass filter, with compensated time delay. These quantities are then provided as input to the Simulink model.
Simulink's ODE15 variable stepsize method is chosen as the most suitable solver for the simulation. Other solvers were tested but either didn't converge or had excessively long computation times.
In the final analysis, the developed physical model estimates the force applied to the damper-link component. A comparison is made between the simulated results and the time series data obtained from a strain gauge mounted on the damper link, reported hereafter:

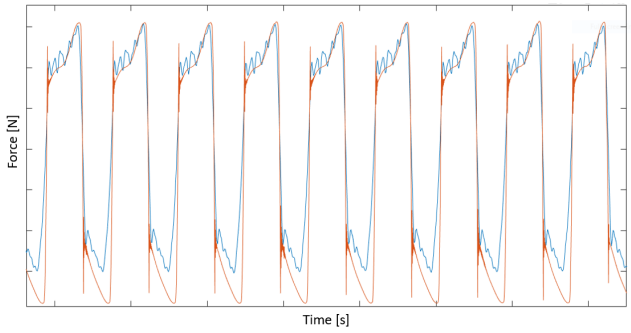


Figure 13: Damper-link axial force Damper model simulation output (input obtained from predictions) against strain gauge corresponding measurement

Figure 13 presents in blue the strain gauge's measured axial force and in orange the damper model's simulation output. The two signals exhibit synchronization, with peaks aligning correctly. However, differences are observed in the valleys. The simulated axial force based on predicted inputs indicates an overestimation. To investigate the source of overestimation, a comparison is made between the axial force obtained from the strain gauge and the simulated axial force using the measured damper rotations as input.
It can be concluded then that the overestimation in Fig 13 was the result of different contributions. The first one was due to an intrinsic damper model overestimation, the second one was due to the differences in the simplified kinematics used to calculate the damper rotations from predictions and the real damper kinematics. The last one could be attributed to the shown differences between the original blade angles and the predicted ones.

9. Conclusions

In conclusion, this research demonstrates the effectiveness of data-driven virtual sensors for accurate predictions of rotor components in helicopters under different flight conditions. The integration of these data-driven approaches with physical modelling proves to be a valuable framework, as virtual sensors provide new data that can be processed by the physical models, leading to a clearer understanding of simulation outputs, particularly in the case of the damper in this study. However, limitations related to the Harmonic Time History (HTH) algorithm and the simplified kinematics approach for damper rotation reconstruction have been identified. Proposed solutions include exploring alternative real-time usable algorithms and employing advanced data-cleaning techniques, as well as developing a more accurate physical model considering the damper's actual kinematics.
This research highlights the importance of advancing prognostics in helicopter rotor components and introduces an innovative methodology centred around virtual sensors. The results emphasize the potential of virtual sensors to significantly enhance the safety and reliability of structure monitoring in various applications. By providing accurate predictions across different flight conditions, virtual sensors enable early detection of potential issues in rotor components, facilitating proactive maintenance and reducing the risk of unexpected failures.
The findings underscore the relevance of this research in promoting improved operational efficiency and ensuring the continued safety of helicopter fleets. By incorporating virtual sensors into existing practices, potential component issues or degradation can be identified, enabling predictive maintenance strategies.

Parameter	Nuber of Neurons	Number of Layers	Blade
Pitch A_0	[50]	[1]	White
Pitch 1^{st} harmonic S & C	[50]	[5]	White
Lag A_0	[50]	[1]	White
Lag 1^{st} harmonic S & C	[50]	[5]	White
Lag 2^{nd} harmonic S & C	[50]	[5]	White
Lag 3^{3d} harmonic S & C	[50]	[5]	White
Flap A_0	[50]	[1]	White
Flap 1^{st} harmonic S & C	[50]	[5]	White
Flap 2^{nd} harmonic S & C	[50]	[5]	White
Pitch A_0	[50]	[1]	Yellow
Pitch 1^{st} harmonic S & C	[50]	[5]	Yellow
Lag A_0	[50]	[1]	Yellow
Lag 1^{st} harmonic S & C	[50]	[5]	Yellow
Lag 2^{nd} harmonic S & C	[50]	[5]	Yellow
Lag 3^{3d} harmonic S & C	[50]	[5]	Yellow
Flap A_0	[50]	[1]	Yellow
Flap 1^{st} harmonic S & C	[50]	[5]	Yellow
Flap 2^{nd} harmonic S & C	[50]	[5]	Yellow

Table 1: MLPs tuning results

Variable	Description	NRMSE error
ξ	lag	0.057
β	flap	0.055
θ	pitch	0.043

Table 2: NRMSE computed over Fig 12 data and rounded to the 3^{rd} decimal

References

[1] Andrew E. Bondoc, Mohsen Tayefeh, and Ahmad Barari. Employing live digital twin in prognostic and health management: Identifying location of the sensors. *IFAC-PapersOnLine*, 55:138–143, 2022.

[2] R.H. Cabell, C.R. Fuller, and W.F. O'Brien. Neural network modelling of oscillatory loads and fatigue damage estimation of helicopter components. *Journal of Sound and Vibration*, 1 1998.

[3] Martina Caliano, Amedeo Buonanno, Antonino Pontecorvo, and Gianluca Sforza. Community energy storage gestione aggregata di sistemi di accumulo dell'energia in power cloud, "comesto". Technical Report 8, ENEA, UNICAL, TEN, 6 2020.

[4] Hazim Dahir, Jeff Luna, Ahmed Khattab, Kaouther Abrougui, and Raj Kumar. *Digital Twin for Healthcare*, chapter 4, pages 73–95. Elsevier, 2023.

[5] M. Dornetti, A. Trezzini, E. Fosco, A. Colombo, G. Facchini, and G. Piccirillo. String potentiometer blade motion measurement system applied to fully articulated inter-blade rotor. In *Conference Programme & Proceedings*, pages 2–5, Southampton, UK, 9 2019. The European Rotorcraft Forum.

[6] Hatem M. Elattar, Hamdy K. Elminir, and A. M. Riad. Prognostics: a literature review. *Springerlink.com*, 6 2016.

[7] Marco Favale, Davide Prederi, and Alberto Angelo Trezzini. Prediction of aw609

rotor loads by means of neural networks. *Vertical Flight Society Conference*, 2019.

[8] Lance A. Flitter, Kelly McCool, and David J. Haas. Rotor system load monitoring using a neural network based approach. *Modeling and Simulation Technologies Conference*, 8 1997.

[9] Caroline Del Cistia Gallimard, Frederic Beroul, Jeremy Jouve, and Konstanca Nikolajevic. Direct load recognition and damage estimation using supervised learning. *Vertical Flight Society Conference*, 2021.

[10] Alberto Graziani, Davide Prederi, Alberto Angelo Trezzini, Pierangelo Masarati, and Marco Favale. Prediction of helicopter rotor loads and fatigue damage evaluation with neural networks. *ERF Journal*, 2022.

[11] David J. Haas, Joel Milano, Lance Flitter, David Taylor, and Model Basin. Prediction of helicopter component loads using neural networks. *American Institute of Aeronautics and Astronautics 34th Structures, Structural Dynamics and Materials Conference*, 4 1993.

[12] Matt Harrigan, Avinash Sarlashkar, Raymond Beale, Jr., Jared Kloda, Mark Kruse, and Dennis Vanill. Rotorcraft digital twin: Exploiting on-board data for enhancing sustainment and operational availability. In Vertical Flight Society's 78th Annual Forum & Technology Display, editor, *Diagnostics, Prognostics, and Health Management*, Ft. Worth TX. USA, 5 2022. Sikorsky, a Lockheed Martin Company.

[13] Chris Harris, Richard Pymar, and Colin Rowat. Joint shapley values: a measure of joint feature importance, 2022.

[14] Jeff Luna, GilAnthony Ungab M. D., Kaouther Abrougui, Hazim Dahir, Ahmed Khattab, and Raj Kumar. *Digital Twin for Healthcare*, chapter 13, pages 263–281. Elsevier, 2023.

[15] Lundberg, Scott M., and Su-In Lee. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 2017.

[16] Weichao Luo, Tianliang Hu, Yingxin Ye, Chengrui Zhang, and Yongli Wei. A hybrid predictive maintenance approach for cnc machine tool driven by digital twin. *Robotics and Computer-integrated Manufacturing*, 65:403–422, October 2020.

[17] Raymon van Dinter, Bedir Tekinerdogan, and Cagatay Catal. Predictive maintenance using digital twins: A systematic literature review. *Information and Software Technology*, 151, 11 2022.

[18] Isabella Verdinelli and Larry Wasserman. Feature importance: A closer look at shapley values and loco, 2023.

[19] Michele Zilletti and Ermanno Fosco. Damper model identification using hybrid physical and machine learning based approach. *47th ERF, Glasgow, UK*, 7-9 September 2021.