ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2.
Available from http://www.i6doc.com/en/.

# From Data to Simulation: Capturing Aircraft Engine Degradation Dynamics

Abdellah Madane[1,2], Florent Forest[3], Hanane Azzag[4],
Mustapha Lebbah[1] and Jérôme Lacaille[2]

1- Paris-Saclay University - DAVID Lab, UVSQ, Versailles, France

2- Safran Airraft Engines - Moissy-Cramayel, France

3- EPFL - IMOS Lab, Lausanne, Switzerland

4- Sorbonne Paris Nord University - LIPN, UMR CNRS 7030, Villetaneuse, France

**Abstract**. The analysis and simulation of aircraft engine behavior have garnered significant attention in the aeronautical industry, primarily due to its implications for performance, maintenance, safety, and sustainability. Our work successfully showcases the efficacy of utilizing time series data collected from our aircraft engines to construct a digital twin capable of dynamically emulating their real-time behavior. We then introduce a new methodology to model the physical engine's degradation and meticulously monitor its evolution over time. By continuously analyzing the simulated data against real-world performance measurements, our approach offers valuable insights into the engine's long-term behavior and health trajectory.

## 1 Introduction

In aircraft engine systems, sensors embedded within the engines collect critical operational data during flight cycles, which is crucial for Prognostics and Health Management (PHM) frameworks. For instance, in this study [1], the authors introduce a recurrent neural network trained on temporal snapshot data to derive a state vector that indicates engine degradation. Recent advancements have led to the acquisition of continuous CEOD (Continuous Engine Operational Data) during flight operations, providing a more comprehensive dataset. CEOD comprises multiple sensor readings and computational outputs obtained by onboard systems and subsequently processed post-flight. Utilizing this continuous data stream shows potential for refining algorithms to achieve greater precision and efficiency, overcoming the constraints associated with using snapshot data exclusively. Notably, it has been instrumental in anomaly detection methodologies [2]. Our research work addresses two primary objectives. First, it presents a methodology for constructing a data-driven simulator for aircraft engines utilizing CEOD. This simulator emulates the intricate dynamical behavior of real aircraft engines, enabling sophisticated simulations under diverse operational conditions, including varied flight regimes and engine controls. Such simulations offer valuable insights into the diverse factors influencing engine health. Secondly, it demonstrates the utility of this simulator in modeling the degradation processes observed in physical engines. The proposed application represents a versatile algorithmic framework capable of simulating aircraft engines and monitoring their

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

health status. It holds considerable potential for deployment across scientific and industrial sectors within and beyond the aviation domain.

## 2 Data-driven Simulator

### 2.1 Dataset

We use continuous engine data to train our simulator. They are sourced from aircraft engines of the same fleet, notably the Continuous Engine Operational Data (CEOD) in our context. Onboard sensors capture a multitude of measurements at varying intervals, all of which we standardize to a uniform frequency of 1 Hz. Throughout our case study, our focus lies on three key parameters: low-pressure rotor speed (N1), temperature before the combustion chamber (T), and Exhaust Gas Temperature (EGT). We simulate these parameters based on five external conditions, as illustrated in Figure 1, which represent the flight mission profile and encompass variables such as ambient temperature, altitude, Mach number, Throttle Lever Angle (TLA), and a boolean variable indicating the boolean engine's operational status (ON/OFF).

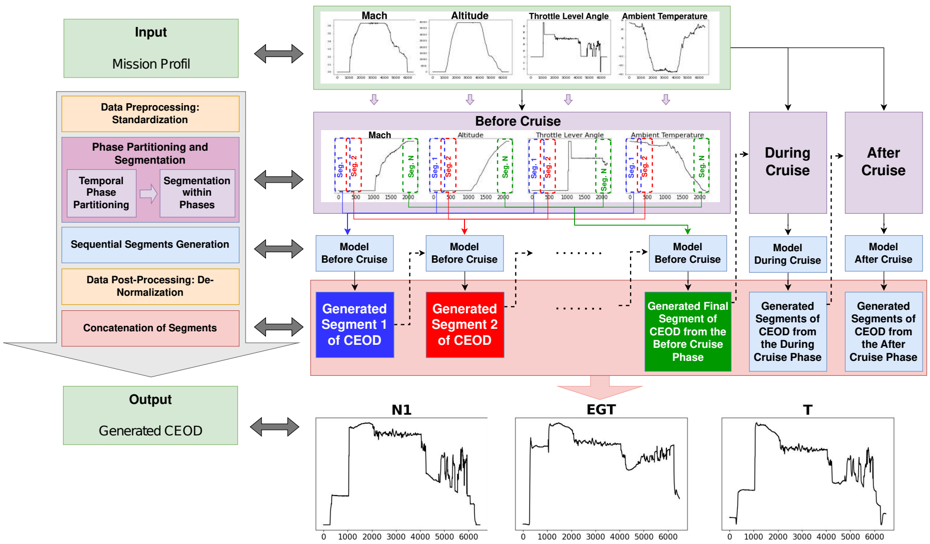



Fig. 1: Data-driven Aircraft Engine Simulator framework. CEOD: Continuous Engine Operational Data. N1: Low-pressure rotor speed. T: temperature before combustion chamber. EGT: Exhaust Gas Temperature.

### 2.2 Methodology

This simulator framework generates CEOD using a structured input-to-output process. Starting with a standardized input where raw multivariate time series

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

data are normalized, the workflow employs temporal phase partitioning to handle the inherent complexities of varying flight lengths and operational states by dividing data into phases (pre-, during, and post-cruise). Within these phases, data is segmented into 300-second intervals with an overlap of 20 seconds, enhancing manageability, computational feasibility, and continuity for long sequences. Phase-specific generative models then simulate data that retains real engine behavior's statistical and temporal characteristics, ensuring that each segment is coherent with adjacent ones. The final steps include de-normalizing to revert data to original scales and concatenating all segments to yield a comprehensive CEOD representation ready for analysis.

### 2.3 Phase-specific generative models

These models are based on a unified model architecture, the Multivariate Time Series Conditional Generative Adversarial Nets (MTS-CGAN) [3, 4], adapted to simulate aircraft engine behaviors. This transformer-based CGAN generates context-dependent multivariate time series data. It incorporates a generator (G) and a discriminator (D). We condition data generation on the specific mission profile and the data segment immediately preceding the current one. This dual conditioning ensures that each generated data segment transitions smoothly into the next, preserving the temporal correlations. The architecture, therefore, supports seamless, coherent data flow across segments.

The conditional generator (G) (Fig 2a) consists of two primary components.

Context Encoder: This first component processes a noise vector and the flight profile mission through multiple blocks of a transformer encoder, utilizing multi-head self-attention to extract contextual interdependencies.

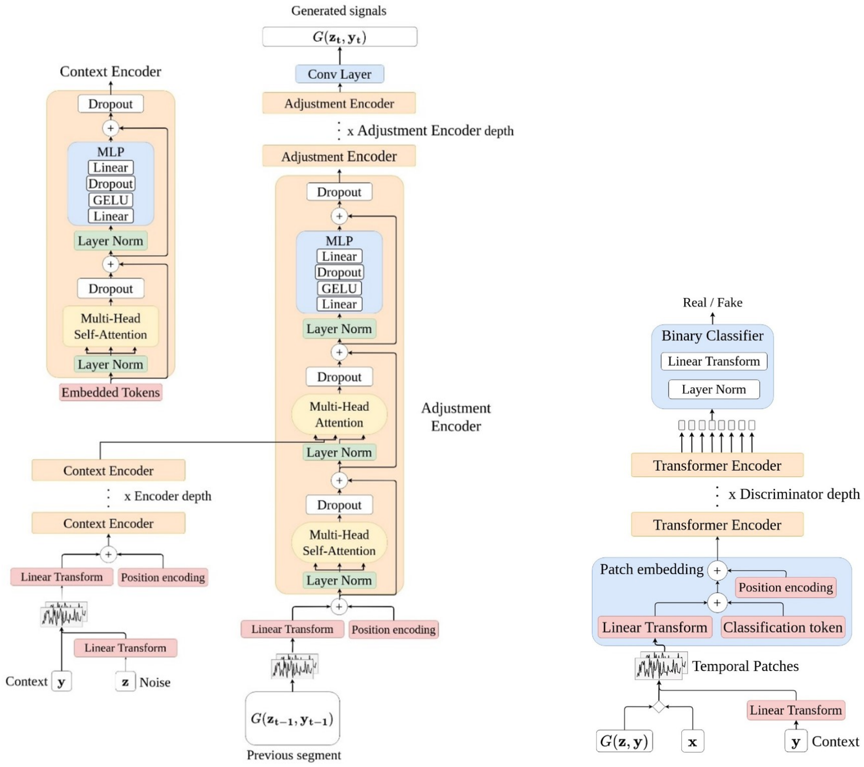
Adjustment Encoder: The second component ensures continuity and contextual relevance across the generated segments. It is comprised of two distinct layers: a multi-head self-attention layer that processes embeddings from the previously generated window to extract contextual features, followed by a separate multi-head attention layer where the query is derived from the self-attention output and the key and value from the first component's output. It incorporates data from the previously generated window with the encoded flight profile mission.

The conditional discriminator (D) (Fig 2b) learns to classify whether its input CEOD is real or generated.

We employ the Least Squares GAN (LSGAN) loss for optimization, with an additional loss term for the generator that ensures smooth transitions between overlapping segments. The discriminator and generator are trained in parallel, respectively minimizing the loss functions $L_D$ and $L_G$.

$$L_D = \frac{1}{2}\mathbb{E}_{\mathbf{x},\mathbf{y}\sim p_{\text{data}}}\left[(D(x,y)-1)^2\right] + \frac{1}{2}\mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}}\left[(D(G(z,y),y))^2\right] \tag{1}$$

$$L_G = \frac{1}{2}\mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}}\left[(D(G(z_t,y_t),y_t)-1)^2\right] + \|G_{1:20}\,(z_t,y_t) - G_{\text{end-19:end}}\,(z_{t-1},y_{t-1})\|_2 \tag{2}$$




(a) Generator (b) Discriminator

Fig. 2: Architecture of the Generator and the Discriminator.

## 3 Degradation Modeling

Modeling the degradation of aircraft engines is a complex task. Each engine has its own degradation trajectory based on the missions it has undergone and how pilots have operated it, among other variables. It is challenging to model because the wear state of each engine is unique and depends on multiple factors. The simulation tool we have introduced thus far is an "average engine simulator", as it was trained on a fleet of aircraft engines. To simulate the accurate behavior of an engine while considering its wear state, one must be able to extract this degradation and adjust the simulator's final output to reflect the specific engine's degradation state. Our approach involves training the simulator on non-degraded data, thereby obtaining what we call a "clean average engine simulator". This means a simulator that predicts the behavior of an engine exhibiting no degradation. Then, for missions completed over an engine's lifecycle, we compare the actual data collected from the specific physical engine to the simulated data. This results in a residual, which represents the discrepancy

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

between the behavior of the non-degraded fleet's average engine and that of the engine considered, which inevitably deviates from a non-degraded state after each cycle, depending on its wear condition and maintenance history. From our engine fleet, we have created a second dataset comprising residuals for various engine cycles. We then develop a second model to forecast the next residual based on the residuals from the last seven engine cycles. Regardless of the engine, the model forecasts the next residual as the output by inputting the last seven flights conducted by that engine. Once we have the forecasted residual, we can simulate the actual behavior of a specific engine in its next cycle, provided we have the mission profile. The current approach involves simulating non-degraded outputs with our clean average engine simulator, then adjusting them based on the forecasted residual, thereby incorporating this bias, which models the deviation of that specific engine's behavior from the non-degraded fleet's average behavior. Define $S_{\text{clean}}$ as the clean average engine simulator trained on non-degraded data, where $S_{\text{clean}}(X) = Y_{\text{predicted}}$ simulates the behavior of an engine with no degradation. For an engine with specific degradation, calculate the residual $R$ by $R = Y_{\text{actual}} - S_{\text{clean}}(X)$, where $Y_{\text{actual}}$ is the data from the physical engine. Using residuals $\{R_i\}$ from various engine cycles, we train a model $f$ to predict the next residual $R_{t+1}$ from the last seven residuals: $R_{t+1} = f(R_t, R_{t-1}, \ldots, R_{t-6})$. The adjusted simulation of the engine's behavior is $Y_{\text{adjusted}} = S_{\text{clean}}(X) + R_{t+1}$.

We design an LSTM model to forecast the Mean Squared Error (MSE) between the "clean average engine simulator" output and the real engine data.

## 4 Experiments and Results

For degradation experiments, we tested on the N-CMAPSS dataset [5], which consists of turbofan engine degradation simulations. This dataset simulates the lifecycle of engines, with degradation occurring over time until their end of life. We trained on a training set of 6 engines and tested on a dataset of 4 engines. Figure 3a illustrates a comparison between the forecasted Mean Squared Error (MSE) and the actual MSE calculated from the outputs of a non-degraded simulator and real engine data throughout an engine lifecycle in a test dataset. The predictive capability of the forecasting model is demonstrated by its accuracy in estimating the next MSE based on the MSE values from the previous seven cycles. Also, the blue line indicates that as the engine degrades, there is an increase in the Mean Squared Error (MSE) between the non-degraded simulator outputs and the actual measurements from the real engine data. In Figure 3b, the MSE comparison between the simulator outputs, which were corrected using the forecasted MSE, and the real engine data is presented alongside the actual MSE calculated from the same engine's non-degraded simulator outputs and real engine data. The correction applied to the simulator outputs involved the addition of the bias identified in the forecasted MSE. It is observed that the bias-corrected simulator outputs consistently exhibit a relatively low MSE, which does not escalate as the engine continues to degrade. This indicates that the adjustments made to the simulator outputs successfully capture the degradation

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

characteristics specific to the engine.

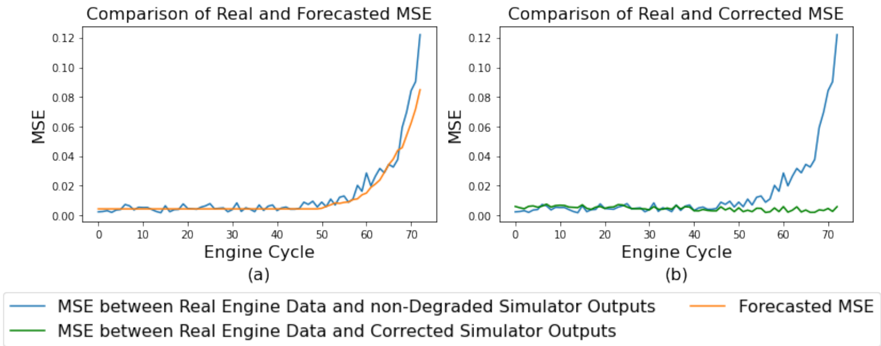



Fig. 3: Comparison of MSE between corrected simulator and Real Engine Data with Actual MSE between non-degraded Simulator and Real Engine Data

## 5 Conclusion

In this paper, we introduced an innovative approach to modeling aircraft engine degradation through a sophisticated simulator. Using simulated data to mimic real engine behavior allows us to anticipate failures and optimize engine performance, thereby extending the operational lifespan of aircraft engines. Looking forward, we aim to refine our degradation models to reflect more realistic and complex operating conditions. This advancement could lead to more precise degradation predictions and a deeper understanding of engine behavior under varied operational scenarios.

## References

[1] Raphaël Langhendries and Jérôme Lacaille. Turbofan exhaust gas temperature forecasting and performance monitoring with a neural network model. In *European Conference on Safety and Reliability (ESREL)*, 2022.

[2] Jean Coussirou, Thomas Vanaret, Jérôme Lacaille, and Safran Aircraft Engines DataLab. Anomaly detections on the oil system of a turbofan engine by a neural autoencoder. In *30th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN*, 2022.

[3] Abdellah Madane, Mohamed-Djallel Dilmi, Florent Forest, Hanane Azzag, Mustapha Lebbah, and Jerome Lacaille. Transformer-based conditional generative adversarial network for multivariate time series generation. In *International Workshop on Temporal Analytics@PAKDD 2023*, 2023.

[4] Abdellah Madane and Jérôme Lacaille. Simulation of the behaviour of engines in their current state of wear. In *Proceedings of the International Conference on Condition Monitoring and Asset Management*, volume 2023, pages 1–11. The British Institute of Non-Destructive Testing, 2023.

[5] Manuel Arias Chao, Chetan Kulkarni, Kai Goebel, and Olga Fink. Aircraft engine run-to-failure dataset under real flight conditions for prognostics and diagnostics. *Data*, 6(1):5, 2021.