Assessing Aircraft Engine Wear through Simulation Techniques

Abdellah Madane[1], Jérôme Lacaille[2], Florent Forest[3], Hanane Azzag[4], and Mustapha Lebbah[5]

[1,2] *DataLab, Safran Aircraft Engines, 77550 Moissy-Cramayel , France*
abdellah.madane@safrangroup.com
jerome.lacaille@safrangroup.com

[3] *IMOS Lab, EPFL, Lausanne, Switzerland*
florent.forest@epfl.ch

[4] *LIPN, UMR CNRS 7030, Sorbonne Paris Nord University, 93430 Villetaneuse, France*
azzag@univ-paris13.fr

[5] *David Lab, UVSQ, Paris-Saclay University, 78035 Versailles, France*
mustapha.lebbah@uvsq.fr

ABSTRACT

In the field of aeronautical engineering, understanding and simulating aircraft engine performance is critical, especially for improving operational safety, efficiency, and sustainability. At Safran Aircraft Engines, we were able to demonstrate the effectiveness of using time series collected from the engines after each flight to build a digital twin that provides a dynamic virtual model able to mirror the real engine's state by using a transformer-based conditional generative adversarial network. This virtual representation allows for advanced simulations under diverse operational scenarios like flight conditions and controls, aiding in understanding the impact of different factors on engine health. It is, therefore, possible for us to provide virtual flights performed by our engines in their actual state of wear. This research paper presents a machine learning model that effectively simulates and monitors the state of aircraft engines in real-time, enabling us to track the evolution of the engines' health over their life cycle. The model's adaptability to incorporate new data ensures its applicability throughout the engine's lifespan, marking a step forward in proactive aeronautic maintenance and potentially enhancing engine longevity through timely diagnostics and interventions.

1. INTRODUCTION

In the realm of aircraft engine systems, strategically placed sensors within the engines play an integral role by capturing essential operational data throughout flight cycles. This data is vital for implementing Prognostics and Health Management (PHM) systems (Lacaille & Langhendries, 2022; Forest et al., 2020). Notably, as demonstrated in recent studies (Langhendries & Lacaille, 2022), such frameworks can benefit from machine learning models like recurrent neural networks, which utilize temporal data to predict engine degradation patterns accurately.

The evolution of data acquisition technologies has facilitated the continuous collection of Engine Operational Data (CEOD) during flights. This comprehensive dataset includes a variety of sensor outputs and computational analyses performed by onboard systems, with data processing occurring post-flight. Leveraging this continuous data flow enhances the development of algorithms that surpass the traditional models based on snapshot data. This continuous monitoring is particularly beneficial for improving anomaly detection techniques, as detailed in (Coussirou, Vanaret, Lacaille, & DataLab, 2022). By harnessing state-of-the-art computational techniques, such as machine learning and big data analytics, engineers and researchers are now able to process and interpret vast amounts of operational data in real-time. This capability not only enhances the accuracy of predictive maintenance models but also facilitates a more proactive approach to engine management. The granularity of CEOD allows for a detailed understanding of engine performance under various conditions, thus aiding in the optimization of engine efficiency and reducing unscheduled maintenance.

The following research focuses on two main objectives. The first is the development of a data-driven simulation framework for aircraft engines that utilizes CEOD to replicate the

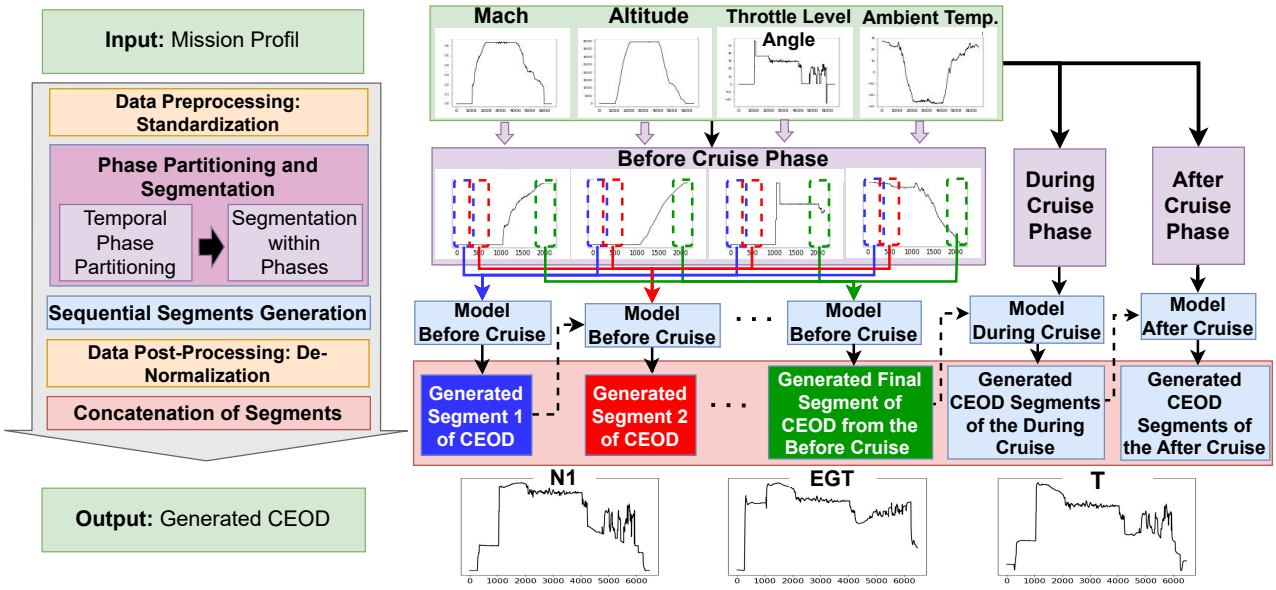


Figure 1. AESim, our proposed data-driven Aircraft Engine Simulator framework. CEOD: Continuous Engine Operational Data. N1: Low-pressure rotor speed. T: temperature before combustion chamber. EGT: Exhaust Gas Temperature.

complex dynamic behavior observed in real engine operations. This framework supports simulations across a range of flight conditions and engine management settings, providing deep insights into the factors influencing engine health and performance. The second objective examines the practical application of this simulator in modeling and understanding degradation processes, which are critical to maintaining engine reliability and performance.

This novel application serves as a comprehensive algorithmic platform for not only simulating the operational dynamics of aircraft engines but also for continuous health monitoring. It integrates seamlessly with existing diagnostic systems, enhancing their predictive capabilities and extending engine life through informed maintenance decisions. The versatility of the platform extends beyond aviation, promising significant value for scientific research and practical applications in sectors such as maritime, energy, and automotive industries, where similar engine technologies are employed. This cross-industry applicability ensures a broader impact, facilitating advancements in engine technology and health management practices globally.

2. DATA-DRIVEN SIMULATOR

2.1. Framework

Generating CEOD within this simulator framework starts with normalizing raw multivariate time series data, establishing a standardized input for consistent data. The data then undergoes temporal phase partitioning to navigate the challenges presented by varying flight durations and operational conditions. This technique divides the flight data into pre-cruise, during-cruise, and post-cruise phases, each tailored to specific operational contexts. Figure 2 shows these three flight phases. Further refinement is achieved by segmenting the data into intervals of 300 seconds, each overlapping by 20 seconds. This segmentation boosts computational efficiency and maintains the continuity and integrity of the data across lengthy sequences. Then, each phase is processed through generative models designed to simulate the statistical and temporal properties of actual engine behavior, ensuring the coherence of the simulated data with real-world scenarios. The culmination of this process involves reversing the initial normalization—thereby restoring the data to its original scale—and stitching together all segments to form a unified and detailed CEOD. This simulated dataset is now primed for comprehensive analysis, as a crucial tool for enhancing predictive maintenance strategie, and augmenting CEOD datasets used to train other machine learning models. Additionally, for profile missions lacking related CEOD, this simulation capability allows us to generate the missing data.

2.2. Continuous Engine Operational Data (CEOD)

Our simulation framework is constructed using a CEOD dataset sourced from engines across a designated fleet. This dataset contains measurements from various onboard sensors, capturing diverse operational metrics. To ensure consistency in the data, all sensor readings are subsampled to a standard frequency of 1 Hz. Our research primarily focuses on the dynamics of critical engine parameters, namely, the low-pressure rotor speed (N1), the temperature before the

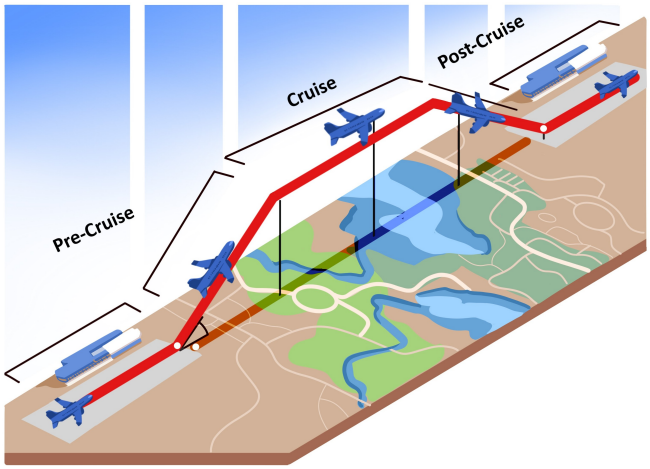


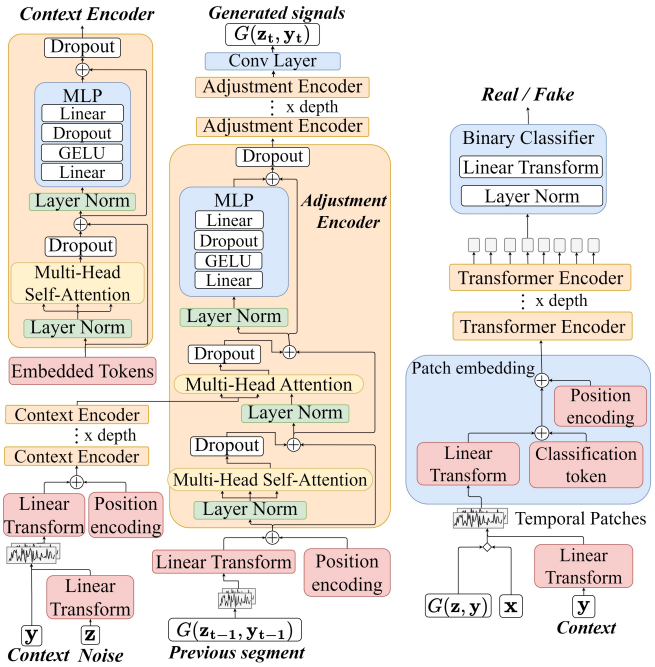
Figure 2. Flight Phases

combustion chamber (T), and the Exhaust Gas Temperature (EGT). These parameters are pivotal in understanding engine behavior under operational stresses. The simulator emulates the engine's response to specified flight conditions. This includes five key environmental and operational variables that define the mission profile of a flight: ambient temperature, altitude, Mach number, Throttle Lever Angle (TLA), and a boolean indicating the engine's status (ON/OFF). The framework is illustrated in Figure 1. Once trained, the simulator can forecast engine behavior under hypothetical scenarios not covered in the training data. This approach allows us to understand and anticipate how changes in operational conditions affect engine performance, which in turn provides valuable insights into engine design and operation.

2.3. Phase-specific generative models

Our models adopt the architecture of Multivariate Time Series Conditional Generative Adversarial Networks (MTS-CGAN) (Madane et al., 2023; Madane & Lacaille, 2023), engineered for modeling the complex dynamics of aircraft engine operations. This transformer-based CGAN architecture is adept at producing context-sensitive multivariate time series, vital for accurate simulations. The structure comprises two core elements: a generator (G) and a discriminator (D), where data generation is strategically conditioned on both the immediate past data segment and the specific flight mission profile, ensuring that each data segment naturally progresses from its predecessor and upholds the essential temporal linkages.

The generator, showcased in Figure 3a, consists of: the *Context Encoder* that takes a noise vector and the specifics of the flight mission profile as input and runs them through a series of transformer encoder blocks. Utilizing multi-head self-attention, this encoder extracts and synthesizes the intricate interdependencies from the contextual data, essential for the next data generation stage. The second component is the *Adjustment Encoder*, which plays a pivotal role in maintaining both continuity and contextual accuracy throughout the generated data segments. It features a two-layer architecture: the initial layer is a multi-head self-attention mechanism that encodes embeddings from the preceding data window, effectively extracting essential contextual attributes. Following this, a subsequent multi-head attention layer refines this process, using the self-attention outputs as the query while integrating the key and value from the outputs of the Context Encoder. This setup ensures that each segment not only reflects but also builds upon the data from prior segments in conjunction with the encoded parameters of the current flight mission profile.



(a) Generator (b) Discriminator

Figure 3. Architecture of the Generator and the Discriminator

On the other side, the discriminator (illustrated in Figure 3b) is trained to evaluate whether the input CEOD is genuine or artificially generated, a crucial step for validating the authenticity of the simulation outputs.

For optimization, we use the Least Squares GAN (LSGAN) loss, augmented with an innovative loss term explicitly designed for the generator. This term is crucial for ensuring fluid transitions between overlapping segments, a feature critical for maintaining the integrity of the simulated data streams. Training both the discriminator and generator is executed in parallel, focusing on minimizing their respective loss functions L_D and L_G, thus enhancing the overall effectiveness and accuracy of the model simulations.

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

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

3. DEGRADATION MODELING

Developing models to simulate the wear and tear on aircraft engines involves addressing the inherent complexity of each engine's unique degradation path. This path is shaped by diverse factors, including the nature of the operational missions, pilot handling techniques, and the engines' maintenance records. The challenge in creating accurate models lies in the distinct wear state unique to each engine. Initially, we introduced the aircraft engine simulator, which we refer to as the "average engine simulator". This simulation tool is trained on a comprehensive dataset encompassing a whole fleet of aircraft engines, thereby providing a generalized representation of an average engine's behavior within this collective.

To precisely model the wear-specific behavior of an individual engine, it is essential to characterize its unique degradation profile and adjust the simulator's output to reflect the specific engine's state of decay accurately. Our approach involves training the simulator only on data of engines in a non-degraded state to create what we call a "clean average engine simulator". This model serves to simulate the fundamental behavior of an engine without any signs of wear, thus providing a benchmark against which to measure actual engine wear. This process establishes a baseline for understanding an engine's performance under an optimal, undegraded state. Following the establishment of the baseline model, we continuously track the operational data from each engine over its entire life cycle, comparing this real-time data with the baseline predictions from our simulator. The discrepancies identified, referred to as residuals, effectively quantify the deviations between the expected performance of a non-degraded engine and the actual performance observed in engines experiencing wear through their operational cycles.

Leveraging our comprehensive fleet data, we have compiled an auxiliary dataset comprised of these residuals across different engine cycles. We used this data to train a forecasting model that uses the residuals from the previous three cycles to predict the upcoming residual. In other words, it is designed to analyze the data from an engine's last three operational missions to predict the next deviation. Upon accurately forecasting this residual, we adjust the output of our simulator. This ensures that the simulation considers the state of wear of the engine so that it mirrors the actual behavior expected in the next cycle more closely. This method allows for a highly tailored simulation that dynamically adapts to the wear patterns observed in individual engines.

To elaborate, let us define S_{clean} as our simulation model that has been trained exclusively on non-degraded engine data. In this context, $S_{\text{clean}}(X) = Y_{\text{predicted}}$ symbolizes the forecasted behavior of an engine assuming no wear and tear, where X represents the input parameters such as engine operational settings, environmental conditions, and other relevant operational data.

For engines that display specific signs of degradation, we compute the scalar residual R as $R = mean(Y_{\text{actual}} - S_{\text{clean}}(X))$, wherein Y_{actual} is the actual performance data collected from the engine during its operations. This residual quantifies the difference between a non-degraded engine's expected performance and the degraded engine's observed performance under the same operational conditions.

Utilizing the series of residuals $\{R_i\}$ collected across various operational cycles, we have developed a predictive model f. This model is trained to predict the next residual R_{t+1} based on a sequence of the last three residuals, i.e., $\hat{R}_{t+1} = f(R_t, R_{t-1}, R_{t-2})$. This approach allows us to adaptively forecast the degree of deviation in engine performance due to degradation in near real-time.

Once the residual $\hat{R}_{t+1}$ is forecasted, we adjust the simulator's output to accurately reflect the anticipated behavior of the engine in its subsequent operational cycle. The adjusted simulation output, Y_{adjusted}, is thus computed as $Y_{\text{adjusted}} = S_{\text{clean}}(X) + \mathbf{1} \cdot \hat{R}_{t+1}$, where $\mathbf{1}$ is a vector of ones of the same length as $S_{\text{clean}}(X)$. This adjusted output incorporates the predicted wear effects and provides a more accurate representation of the engine's performance under specific upcoming operational conditions based on the mission profile and historical wear data.

We develop a Long Short-Term Memory (LSTM) model to forecast the Mean Squared Error (MSE) between the output of the "clean average engine simulator" and the real engine data. The architecture of this model is composed of two LSTM layers followed by two dense layers. Specifically, the first LSTM layer has 100 units, employs ReLU activation, and is configured to return sequences, which are then passed to the second LSTM layer of 50 units. Afterward, a dense layer with 20 ReLU units processes the output from the LSTM layers, leading to a final dense layer with a single unit that produces the prediction. The model uses an Adam optimizer with a learning rate of 0.005 and is compiled to minimize mean squared error.

4. EXPERIMENTS AND RESULTS

4.1. Dataset

In order to conduct degradation experiments, our study utilizes the New Commercial Modular AeroPropulsion System Simulation (N-CMAPSS) dataset, specifically DS01 (Arias Chao, Kulkarni, Goebel, & Fink, 2021). This dataset simulates the operational wear across a small fleet of large turbofan engines, providing detailed run-to-failure trajectories. It comprises data from 10 engines, each averaging approximately 90 flights throughout its lifecycle. For experimental purposes, the dataset is, by default, divided into a

Table 1. Measurement Parameters

Symbol	Description	Units
Wf	Fuel flow	pps
Nf	Physical fan speed	rpm
Nc	Physical core speed	rpm
T24	Total temperature at LPC outlet	°R
T30	Total temperature at HPC outlet	°R
T48	Total temperature at HPT outlet	°R
T50	Total temperature at LPT outlet	°R
P15	Total pressure in bypass-duct	psia
P2	Total pressure at fan inlet	psia
P21	Total pressure at fan outlet	psia
P24	Total pressure at LPC outlet	psia
Ps30	Static pressure at HPC outlet	psia
P40	Total pressure at burner outlet	psia
P50	Total pressure at LPT outlet	psia

training set and a test set, consisting of 6 and 4 engines, respectively. The data consists of two types of parameters: scenario descriptors and measurements. Scenario descriptors encompass four parameters integral to the operational profile of the flight, namely altitude, flight Mach number, throttle-resolve angle, and the total temperature at the fan inlet. These parameters are critical for simulating the environmental and operational conditions encountered during flight missions. On the other hand, the measurements component of the dataset includes a set of 14 parameters, detailed in Table 1, which provide quantifiable metrics essential for assessing engine performance and degradation.

4.2. Experiments

4.2.1. Analyzing Model Accuracy in Capturing Progressive Engine Wear

We began our experiments by training our "clean average engine simulator", denoted as S_{clean}, using training data from engines in their non-degraded state. This determination is made using the boolean "health state" parameter provided in the dataset, which indicates the cycles during which the engine's state is degraded. As for the CEOD case, the simulator receives the four scenario descriptors as inputs and simulates the 14 measurements as outputs. Once training is complete, we go through the engines individually and simulate all the missions they have undergone. Thereafter, we compute the residuals between the simulated measurement outputs and the actual data from these engines. Thus, we compile a series of residuals for each engine that describe its lifecycle. These residuals form the dataset we used to train the model that forecasts the next residual based on the last three residuals. In the DS01 dataset, there is one failure mode; we noted that it predominantly affected two parameters, T48 and T50. Therefore, we specifically focused on the residuals of these two parameters.

After training the model, we test our approach on the test dataset. The process involves the following steps: For a given engine, we sequentially simulate the behavior of the average non-degraded engine of the fleet for each mission it has undertaken. We then predict the residual for each cycle based on the last three residuals and adjust the simulator's output accordingly. To evaluate our method, we compare, for each cycle, the Mean Squared Error (MSE) or Root Mean Squared Error (RMSE) between the outputs of the non-degraded simulator and the actual data, with the MSE or RMSE between the corrected outputs and the actual engine data. This comparison provides insight into the effectiveness of our residual correction in simulating and predicting engine behavior more accurately.

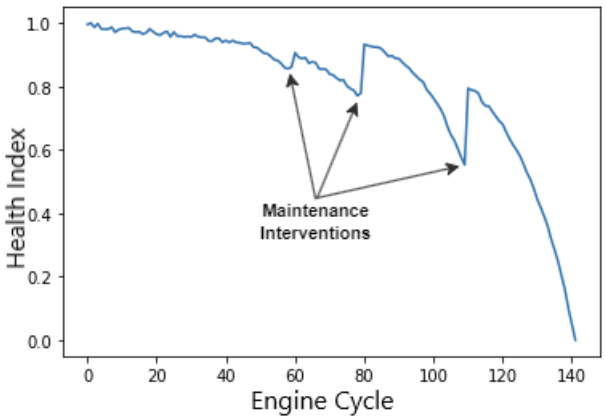


Figure 4. Health index trajectory of an engine lifecycle with maintenance interventions

4.2.2. Assessing Model Capability in Detecting Maintenance Interventions and Their Effects

For this experiment, we conducted simulations of engine lifecycles with maintenance events. We created engine life cycle data that included maintenance interventions, as such data are not available in the N-CMAPSS dataset. The method consists of substituting the flights following each maintenance with flights from another, less degraded engine. This approach mimics the reality where, post-maintenance, an engine is considered to be in a near-new condition. These simulated flights were used solely for testing; the training dataset remained unchanged. Figure 4 illustrates the health index trajectory of an engine lifecycle, incorporating maintenance interventions. Each peak on the graph signifies a return to a healthier state, characterized by reduced degradation following maintenance. Similarly, to evaluate our method, we compare, for each cycle, the Mean Squared Error (MSE) or Root Mean Squared Error (RMSE) between the outputs of the non-degraded simulator and the actual data, with the MSE or RMSE between the corrected outputs and the actual engine data. This comparison provides insight into the effectiveness of our residual correction in accurately simulating and predicting engine behavior, particularly in reflecting maintenance interventions' impact on engine health.

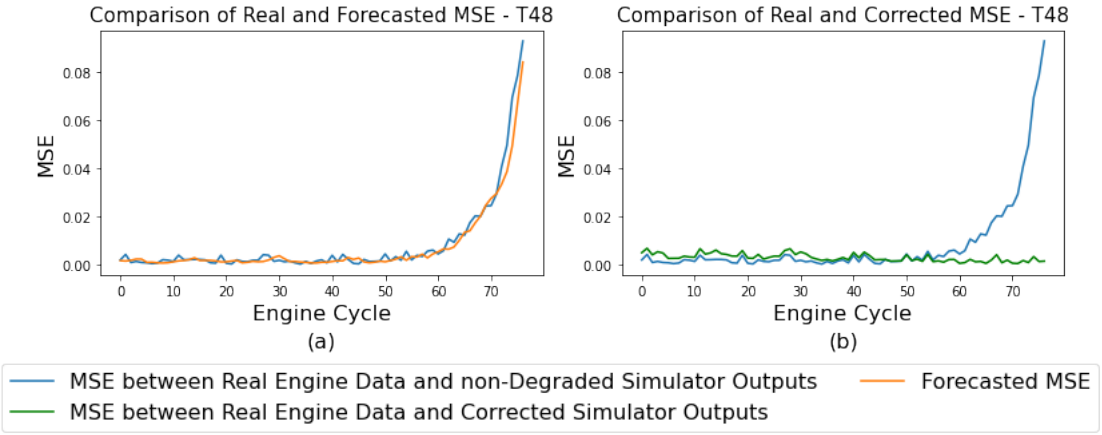


Figure 5. Comparison of MSE between corrected simulator and Real Engine Data with Actual MSE between non-degraded Simulator and Real Engine Data - T48 parameter

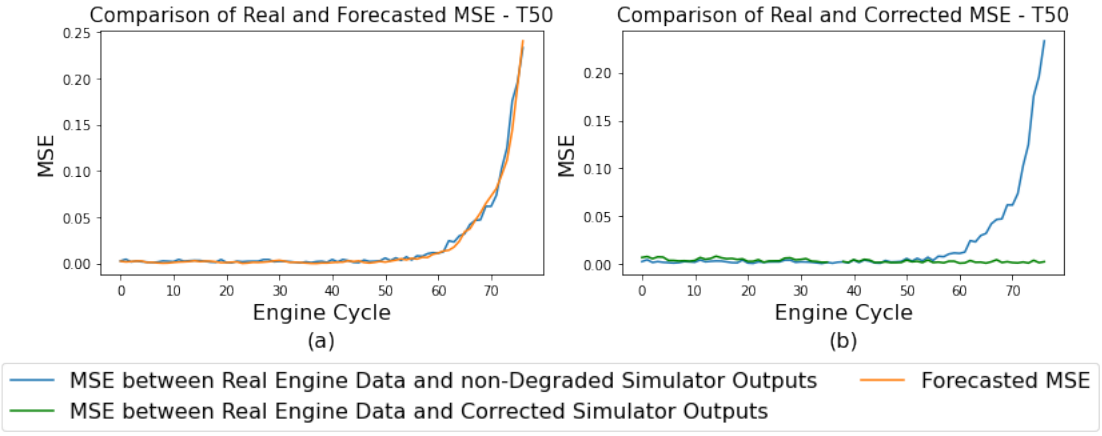


Figure 6. Comparison of MSE between corrected simulator and Real Engine Data with Actual MSE between non-degraded Simulator and Real Engine Data - T50 parameter

4.3. Results

The findings showcased in this section pertain to an engine randomly chosen from the test dataset. Results for the other engines in the test dataset are provided in the appendix.

Figure 5a and 6a present a detailed comparison between the forecasted Mean Squared Error (MSE) and the actual MSE, derived from the data produced by the non-degraded simulator and the actual engine measurements throughout an engine's lifecycle within a test dataset. The forecasting model's precision is underscored by its ability to accurately predict forthcoming MSE values (residuals) based on the preceding three cycles. The blue line in the graph demonstrates a notable increase in MSE as the engine undergoes progressive degradation, highlighting the disparity between the simulator's output for a non-degraded engine and the real engine measurements.

In Figure 5b and 6b, we observe a comparison of MSE values between corrected simulator outputs and real engine data, plotted against the MSE derived from comparing the non-degraded simulator outputs with the same engine real data. The corrections to the simulator outputs are achieved by integrating the forecasted MSE bias, leading to a noticeable improvement in the accuracy of the simulation. The corrected outputs consistently show a lower MSE, which does not increase as the engine's degradation progresses. This consistency suggests that the modifications applied effectively align the simulator's outputs with the actual degradation patterns of the engine, enhancing the predictive accuracy of the simulator across the engine's operational lifespan.

Figures 7a and 8a illustrate the comparison between the predicted and actual residuals. A notable correspondence is observed, particularly in the reduction of actual residuals post-maintenance, which aligns with the expectation that aircraft are less degraded following maintenance. This decrease is also evident in the predicted residuals, demonstrating the model's proficiency in accurately forecasting the effects of maintenance and incorporating them effectively into the ad-

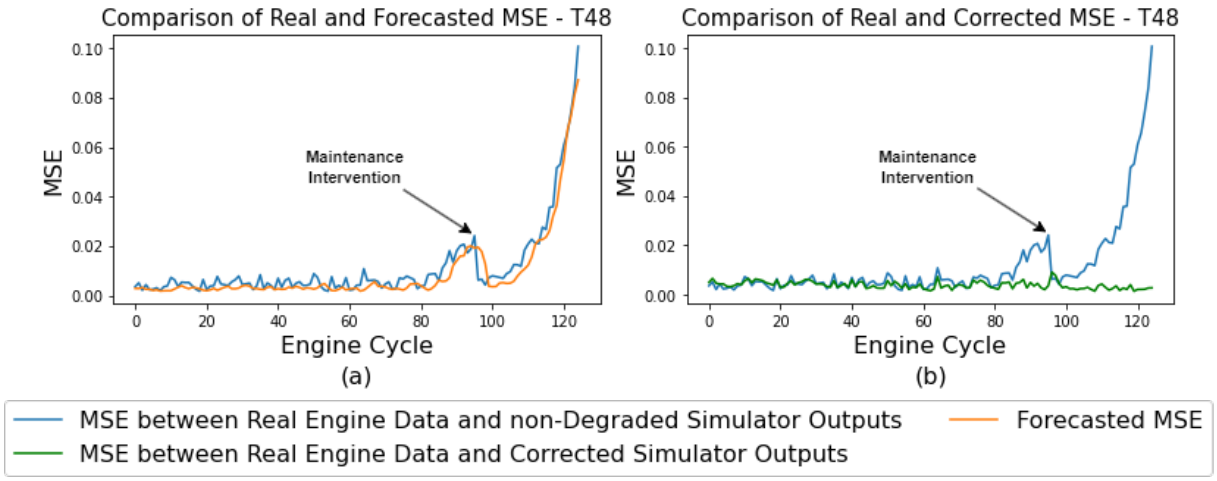


Figure 7. Comparison of MSE between corrected simulator and Real Engine Data with Actual MSE between non-degraded Simulator and Real Engine Data on engine lifecycle with maintenance events - T48 parameter

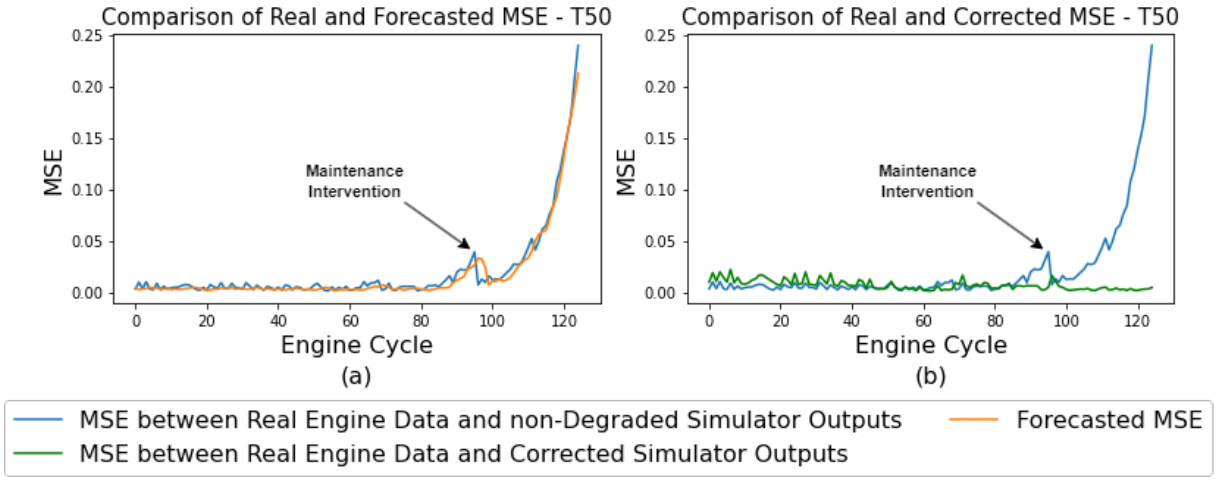


Figure 8. Comparison of MSE between corrected simulator and Real Engine Data with Actual MSE between non-degraded Simulator and Real Engine Data on engine lifecycle with maintenance events - T50 parameter

justments of the data derived from the simulator. It is noteworthy that during the first two maintenance intervals, the engines were not significantly degraded, and the non-degraded simulator was able to emulate a behavior that was comparably accurate, as indicated by the low MSE. However, prior to the third maintenance, an increase in the MSE between the simulator outputs and the actual data was observed. This increase was due to the more significant degradation, making the discrepancy between the outputs of the non-degraded simulator and the actual data more pronounced (evidenced by the peak observed before it decreases post-maintenance).

On Figures 7b and 8b, the objective is to ensure that the residual of the corrected simulator data remains very low throughout the engine cycle. This is confirmed here, as the MSE remains low and stable, even in instances of engine degradation and following maintenance.

Figures 9 and 10 present visualizations of the corrected outputs for T48 and T50 parameters, respectively, in comparison with the outputs from the non-degraded simulator and the actual data collected from the physical engine. These figures demonstrate that our methodology effectively produces outputs closely resembling the actual engine signals, effectively capturing the true behavior of the engine. It is important to note that this data corresponds to the 87th flight in the lifecycle of the engine, indicating that it is nearing the end of its service life and is in a significantly degraded state.

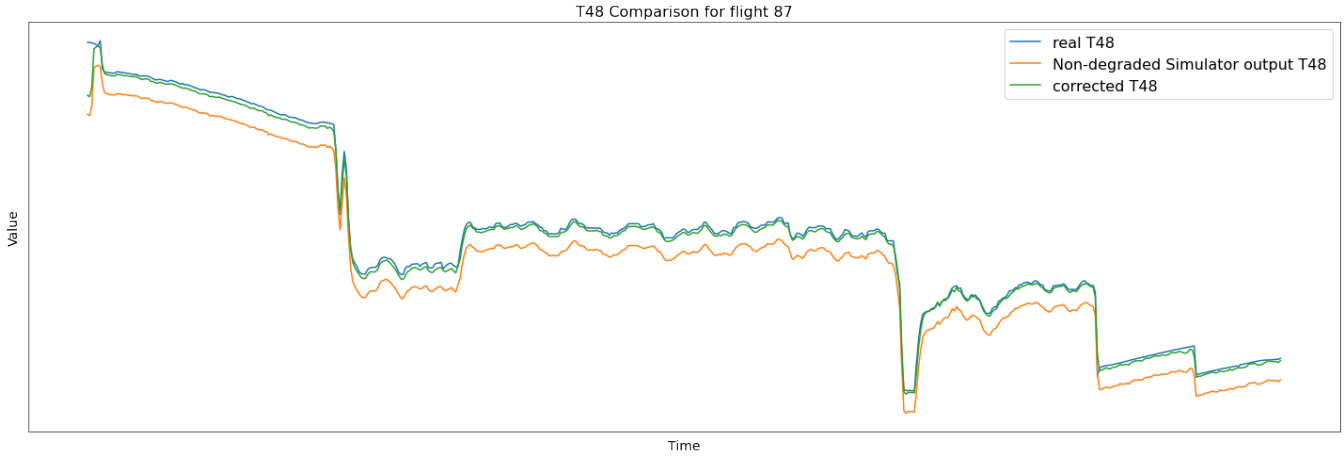


Figure 9. Vizualisation of T48 parameter for flight n°87

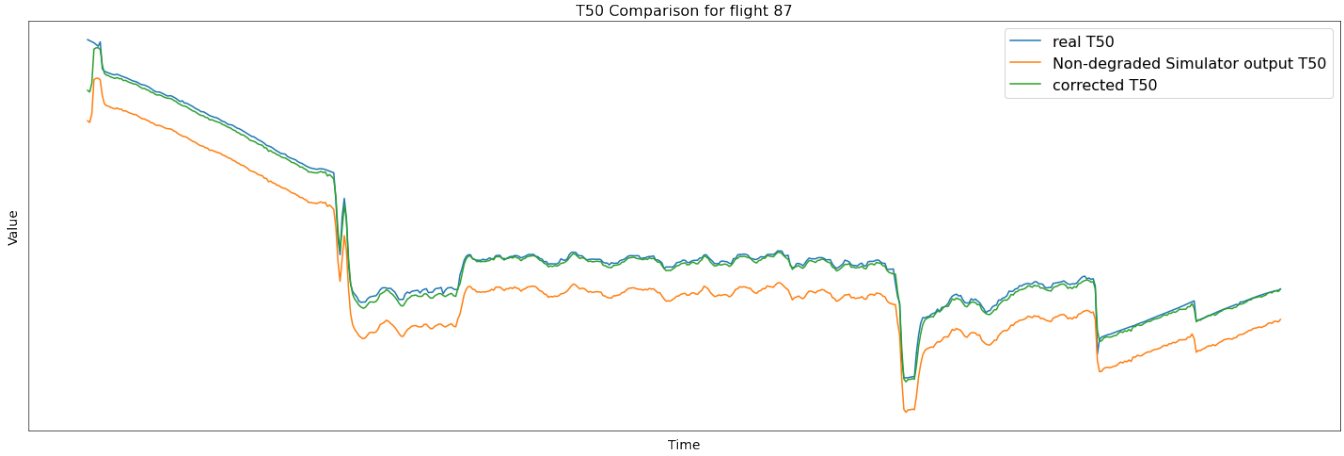


Figure 10. Vizualisation of T50 parameter for flight n°87

5. CONCLUSION AND PERSPECTIVES

In this research, we have proposed a novel method for simulating aircraft engine degradation using an advanced simulator framework. By leveraging simulated data that accurately emulates real-world engine behavior, our approach enables proactive failure prediction and performance optimization, ultimately enhancing the longevity of aircraft engines. Moving forward, we aim to enhance these degradation models to more closely mirror the intricate and diverse operating conditions encountered in actual flight scenarios. Such improvements are expected to yield more accurate predictions of engine degradation and provide a more comprehensive insight into engine performance across different operational environments. This progression will not only improve reliability but also increase the efficiency of maintenance schedules, contributing to safer and more cost-effective aeronautical operations.

REFERENCES

Arias Chao, M., Kulkarni, C., Goebel, K., & Fink, O. (2021). Aircraft engine run-to-failure dataset under real flight conditions for prognostics and diagnostics. *Data*, *6*(1), 5.

Coussirou, J., Vanaret, T., Lacaille, J., & DataLab, S. A. E. (2022). 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.*

Forest, F., Cochard, Q., Noyer, C., Joncour, M., Lacaille, J., Lebbah, M., & Azzag, H. (2020). Large-scale vibration monitoring of aircraft engines from operational data using self-organized models. In *annual conference of the phm society* (Vol. 12, pp. 11–11).

Lacaille, J., & Langhendries, R. (2022). Corrosion risk estimation and cause analysis on turbofan engine. In *Annual conference of the phm society* (Vol. 14).

Langhendries, R., & Lacaille, J. (2022). Turbofan exhaust

gas temperature forecasting and performance monitoring with a neural network model. In *European conference on safety and reliability (esrel).*

Madane, A., Dilmi, M.-D., Forest, F., Azzag, H., Lebbah, M., & Lacaille, J. (2023). Transformer-based conditional generative adversarial network for multivariate time series generation. In *International workshop on temporal analytics@pakdd 2023.* doi: https://pakdd2023.org/wp-content/uploads/2023/05/pakdd23$_w$1$_p$2.*pdf*

Madane, A., & Lacaille, J. (2023). Simulation of the behaviour of engines in their current state of wear. In *Proceedings of the international conference on condition monitoring and asset management* (Vol. 2023, p. 1-11). The British Institute of Non-Destructive Testing. doi: 10.1784/cm2023.5d2

APPENDIX

In this appendix, we present the results for additional engines from the test dataset and also visualize the simulated measurements taken at both the beginning and the end of an engine's lifecycle within the test dataset.

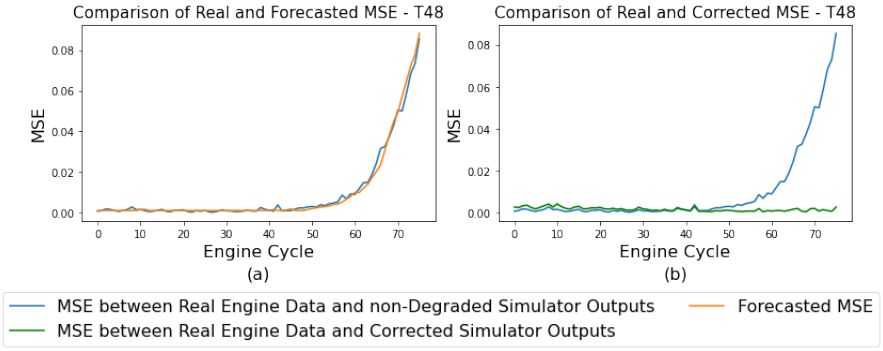


Figure 11. Comparison of MSE between corrected simulator and Real Engine Data with Actual MSE between non-degraded Simulator and Real Engine Data - T48 parameter - Engine n°1 from test dataset

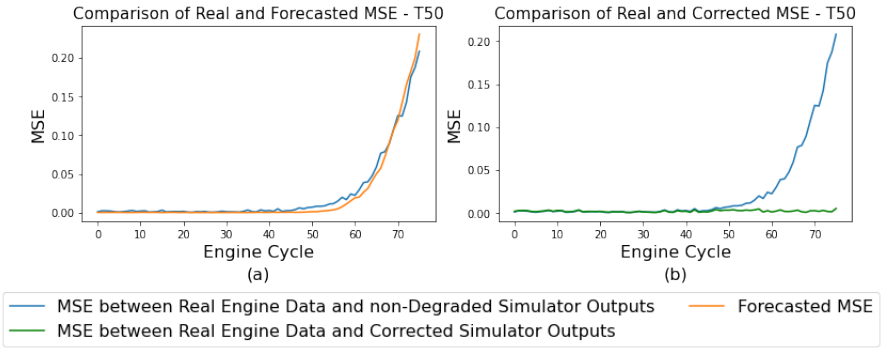


Figure 12. Comparison of MSE between corrected simulator and Real Engine Data with Actual MSE between non-degraded Simulator and Real Engine Data - T50 parameter - Engine n°1 from test dataset

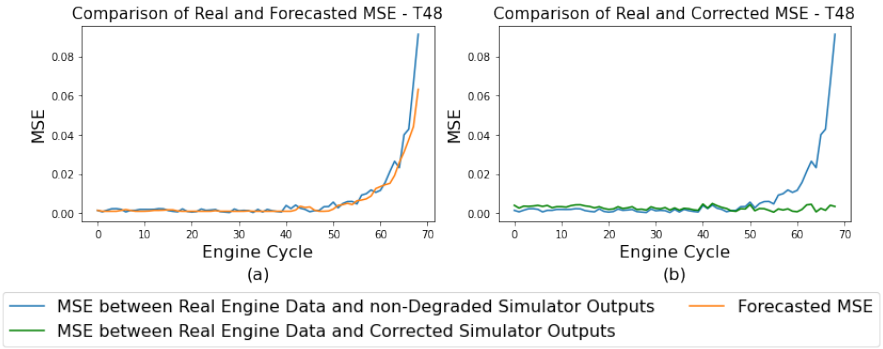


Figure 13. Comparison of MSE between corrected simulator and Real Engine Data with Actual MSE between non-degraded Simulator and Real Engine Data - T48 parameter - Engine n°2 from test dataset

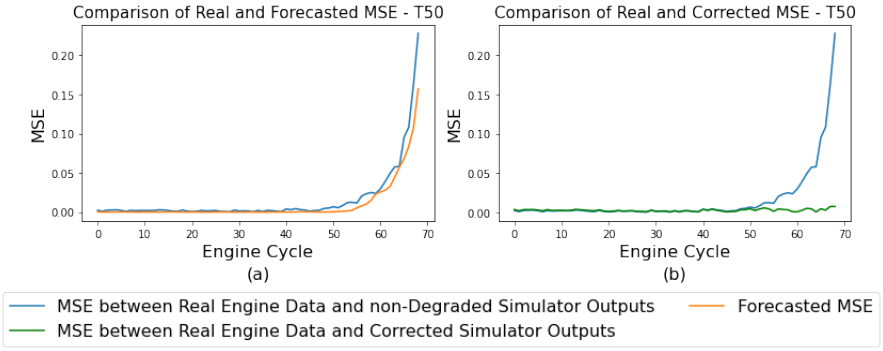


Figure 14. Comparison of MSE between corrected simulator and Real Engine Data with Actual MSE between non-degraded Simulator and Real Engine Data - T50 parameter - Engine n°2 from test dataset

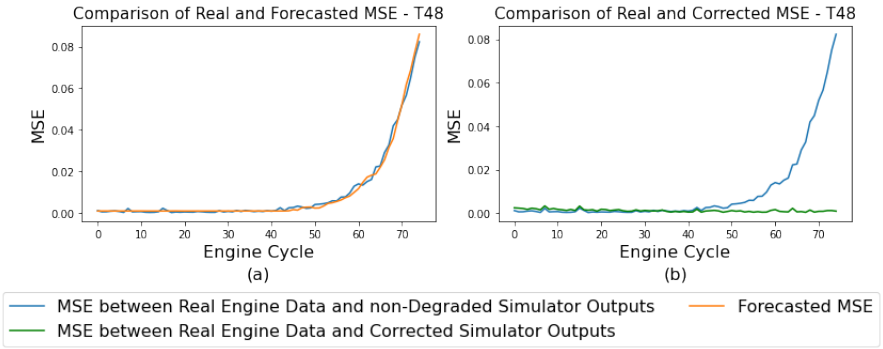


Figure 15. Comparison of MSE between corrected simulator and Real Engine Data with Actual MSE between non-degraded Simulator and Real Engine Data - T48 parameter - Engine n°3 from test dataset

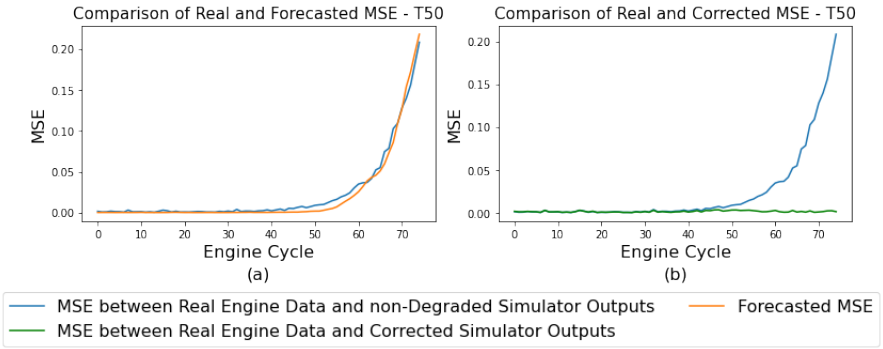


Figure 16. Comparison of MSE between corrected simulator and Real Engine Data with Actual MSE between non-degraded Simulator and Real Engine Data - T50 parameter - Engine n°3 from test dataset

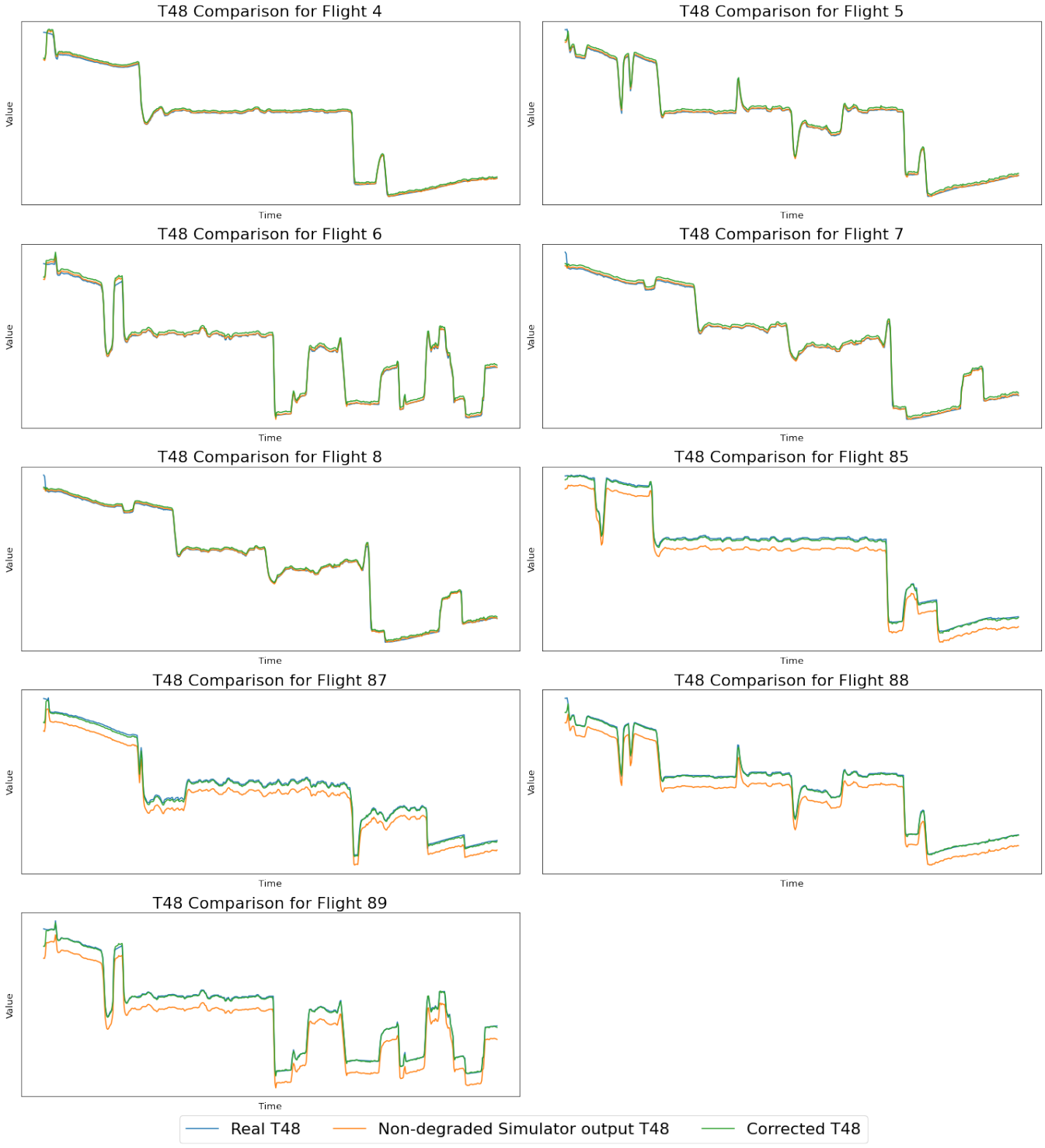


Figure 17. Visualization of T48 parameter for selected flights from the beginning and end of an engine cycle in the test dataset

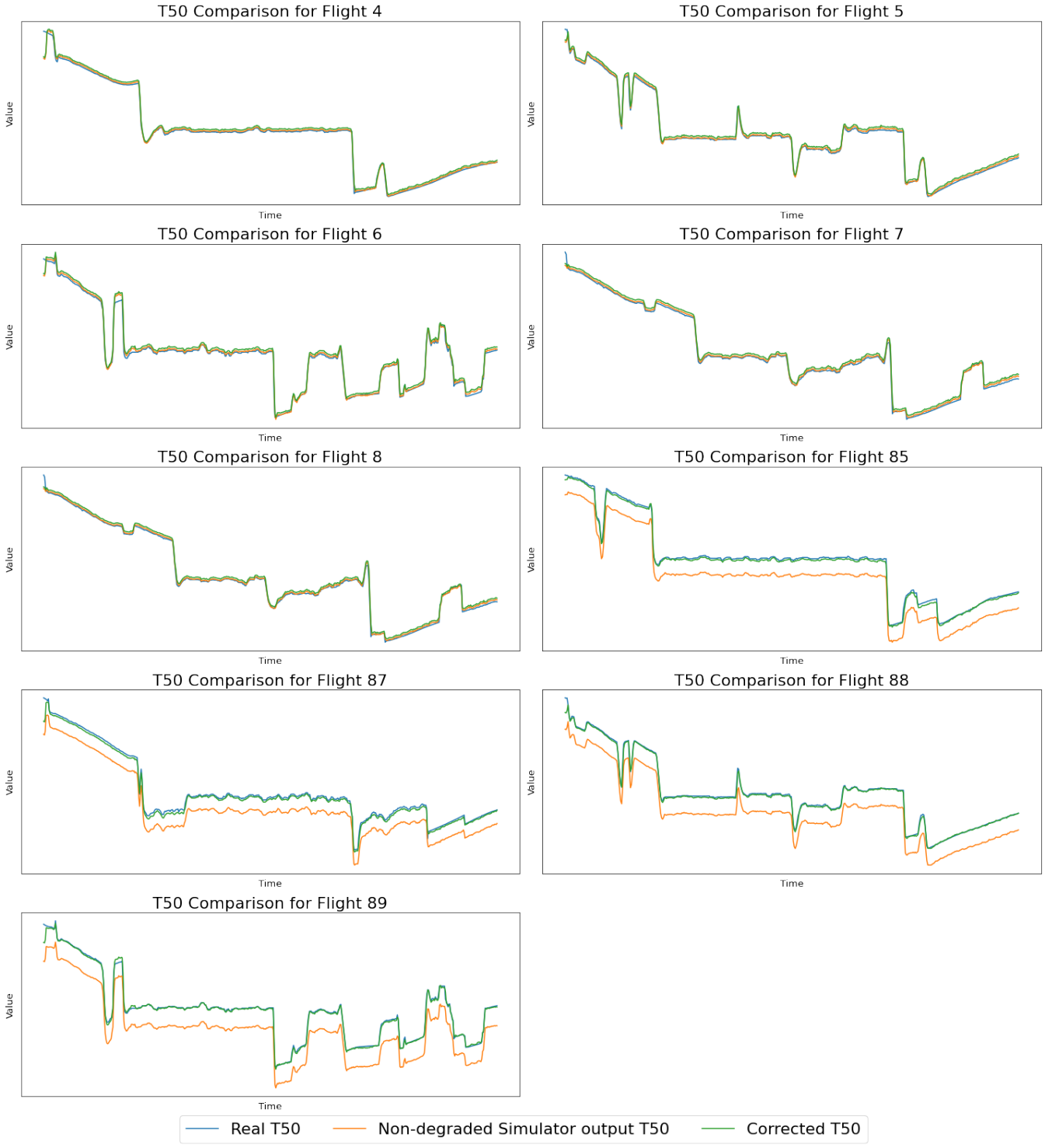


Figure 18. Visualization of T50 parameter for selected flights from the beginning and end of an engine cycle in the test dataset