Unsupervised Modeling of Progressive Wear in Aircraft Engines for Predictive Maintenance

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

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

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

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

ABSTRACT

Predicting progressive wear in aircraft engines is critical for enabling condition-based maintenance and ensuring operational reliability. A persistent challenge lies in the discrepancy between benchmark datasets and real-world engine data. Although simulated datasets offer controlled and labeled conditions for model development, they often fail to represent the full complexity, noise characteristics, and operational irregularities observed in actual flight environments. This leads to models that perform well in simulation but degrade significantly when applied in practice. To address this limitation, this work introduces a data-driven framework to simulate realistic wear-and-tear effects using high-resolution time-series data collected over sequences of engine missions. The method infers long-term degradation patterns in an unsupervised manner, without relying on explicit wear labels, while accounting for variability introduced by mission conditions.

1. INTRODUCTION

Prognostics and Health Management (PHM) plays a pivotal role in advancing aviation safety, operational efficiency, and cost-effectiveness. By leveraging sensor data and advanced analytical techniques, we are able to detect performance degradation earlier and to predict the remaining useful life (RUL) across diverse operating conditions. This allows us to implement condition-based maintenance strategies and enhance fleet-level planning, significantly reducing unplanned downtime while improving asset readiness and reliability (Xiao et al., 2024).

The exponential growth of data generated by modern industrial systems has accelerated the development of data-driven prognostic modeling techniques. Traditional physics-based models employ systems of differential equations to capture the behavior and interactions of engine subsystems, providing high interpretability and strict adherence to physical laws (Von Krannichfeldt, Orehounig, & Fink, 2024). In contrast, data-driven models leverage statistical and machine learning algorithms to derive relationships between input variables and engine performance metrics captured directly from sensor measurements, eliminating the need for extensive physical parameter calibration. While physics-based models excel in scenarios with limited data and well-understood dynamics, data-driven approaches offer greater adaptability to dynamic operating conditions but are highly dependent on the volume, quality, and diversity of the available data.

One of the persistent hurdles for data-driven prognostics for aircraft engines is the limited availability of run-to-failure data (Saxena, Goebel, Simon, & Eklund, 2008), which are often proprietary and therefore inaccessible to the public. To address this problem, simulated datasets have been developed. NASA, for example, introduced the CMAPSS (Saxena et al., 2008) and N-CMAPSS turbofan engine degradation datasets (Arias Chao, Kulkarni, Goebel, & Fink, 2021), providing researchers with realistic surrogate data to conduct experiments, validate prognostic models, and ensure the reproducibility of research findings. However, a significant challenge remains: a gap exists between simulated datasets and real-world engine data. Although simulated datasets offer controlled and labeled conditions for model development, they often fail to represent the full complexity, noise characteristics, and operational irregularities observed in actual flight environments. This leads to models that perform well

in simulated environments but degrade significantly when applied in real-world applications.

In our prior (Madane, Lacaille, Forest, Azzag, & Lebbah, 2024), we developed a simulator capable of generating multivariate time series representing aircraft engine sensor measurements for a given mission profile. This simulator was trained exclusively on non-degraded engine data, ensuring that the simulated output reflected the expected behavior of a healthy engine. When applied to the N-CMAPSS dataset, we compared the simulator outputs to the corresponding mission data. The observed differences between the two were found to be systematic and constant over time for each sensor channel. This bias was interpreted as a signature of the engine's wear and tear, since the simulator did not incorporate degradation effects. In that setting, estimating the bias and adding it to the simulator output was sufficient to accurately reconstruct the actual mission data.

The present work aims to extend and validate this methodology using real-world flight data. Unlike the N-CMAPSS case, our analysis revealed that the biases between the non-degraded simulator outputs and the real measurements were not constant over time. Consequently, no uniform bias term could be applied across the entire time series for each sensor channel. This variability in the bias distribution suggests that degradation effects in real engines manifest in a more complex and time-dependent manner than in the simulated environment.

To address this challenge, we developed a new approach for constructing a wear indicator and correcting simulator outputs. First, we computed the bias vectors across multiple flight phases as the difference between the real mission data and the simulator predictions. Once these bias vectors were extracted, they were used to train a predictive correction model. This model learns the mapping between the bias patterns and the corresponding deviations from non-degraded behavior. By applying this learned correction to the simulator output, we were able to produce a more accurate reconstruction of the real sensor measurements.

2. RELATED WORKS

2.1. Continuous Engine Operational Data (CEOD)

Continuous Engine Operation Data (CEOD) refers to time-series measurements acquired by on-board sensors during the in-flight operation of aircraft engines. Depending on the sensor type and monitored parameter, these measurements are typically recorded at varying and often high sampling frequencies. To facilitate analysis and data fusion, raw CEOD usually go through a post-processing stage in which all variables are resampled to a standardized frequency, typically 1 Hz. The high temporal frequency and multivariate nature of the CEOD allow a detailed description of engine performance across different flight phases and operating conditions. Each variable corresponds to a specific engine parameter, for example, the Exhaust Gas Temperature (EGT). Due to the extended duration of flight missions, CEOD sequences are very long, with lengths varying proportionally to the flight duration. This poses significant challenges regarding data storage, computational processing, and algorithmic handling for condition monitoring or prognostics applications. Nevertheless, these challenges have not deterred researchers and engineers from leveraging CEOD, whether for developing innovative processing pipelines (Forest, Lacaille, Lebbah, & Azzag, 2018), advanced anomaly detection frameworks, or other analytical and diagnostic methodologies to enhance engine reliability and operational efficiency (Coussirou, Vanaret, Lacaille, & DataLab, 2022; Forest et al., 2020). This study focuses on the dynamic behavior of three engine performance parameters: low-pressure rotor speed (N1), the temperature before the combustion chamber (T), and Exhaust Gas Temperature (EGT).

2.2. AESim: Data-driven Aircraft Engine Simulator framework

AESim (Madane, Forest, Azzag, Lebbah, & Lacaille, 2023) produces the synthetic CEOD representing the expected engine output for a specified flight scenario. Each scenario is parameterized by five key environmental and operational variables that define the mission profile: ambient temperature, altitude, Mach number, Throttle Lever Angle (TLA), and a binary flag indicating engine operating state (ON/OFF). The overall simulation framework is shown in Figure 1.

The simulator works through a multi-stage process that begins with normalizing raw multivariate time series to create standardized inputs, followed by temporal phase partitioning into pre-cruise, cruise, and post-cruise phases to account for varying flight durations and operational conditions. Each phase is further divided into overlapping 300-second intervals to improve computational efficiency while preserving data continuity. These segments are then processed through generative models replicating real engine behavior's statistical and temporal characteristics. The final step reverses normalization and stitches all segments into a coherent, high-fidelity CEOD that can be used for predictive maintenance studies, augmenting datasets for machine learning, and generating realistic CEOD for mission profiles where such data are unavailable.

2.3. Phase-specific generative models

The models consist of a modified version of the Multivariate Time Series Conditional Generative Adversarial Network (MTS-CGAN) architecture, specifically designed to model the complex dynamics of aircraft engine operations (Madane, Dilmi, et al., 2023; Madane, Forest, Azzag, Lebbah, & La-

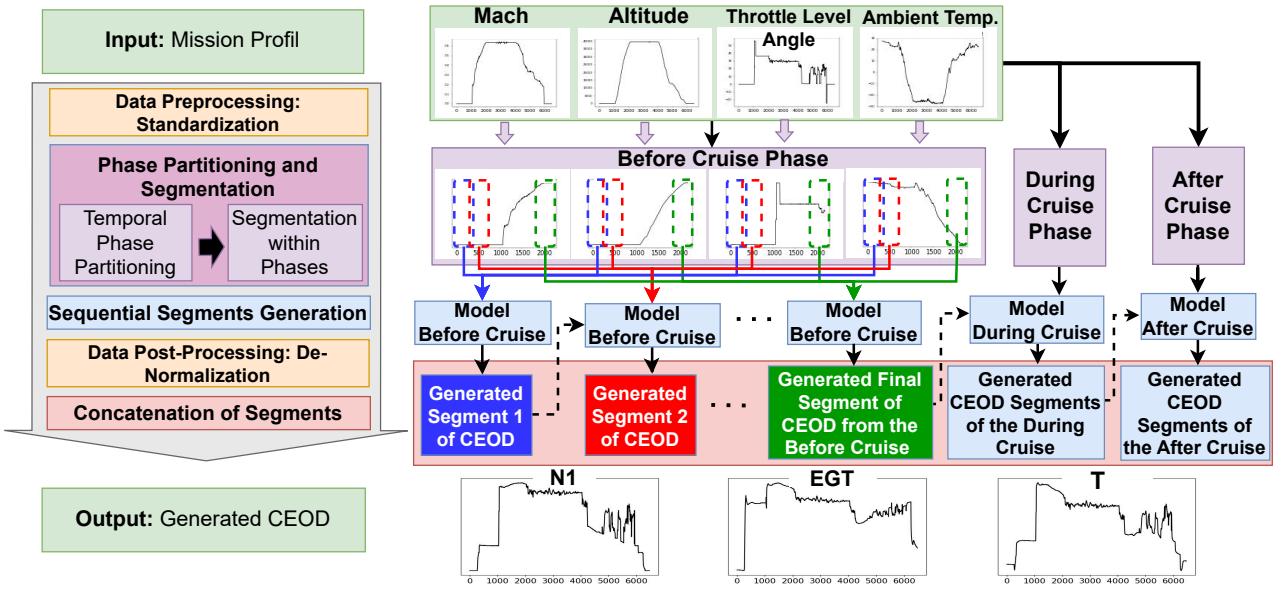


Figure 1. AESim, our proposed data-driven Aircraft Engine Simulator framework. CEOD: Continuous Engine Operational Data. N1: Low-pressure rotor speed. T: temperature before combustion chamber. EGT: Exhaust Gas Temperature.

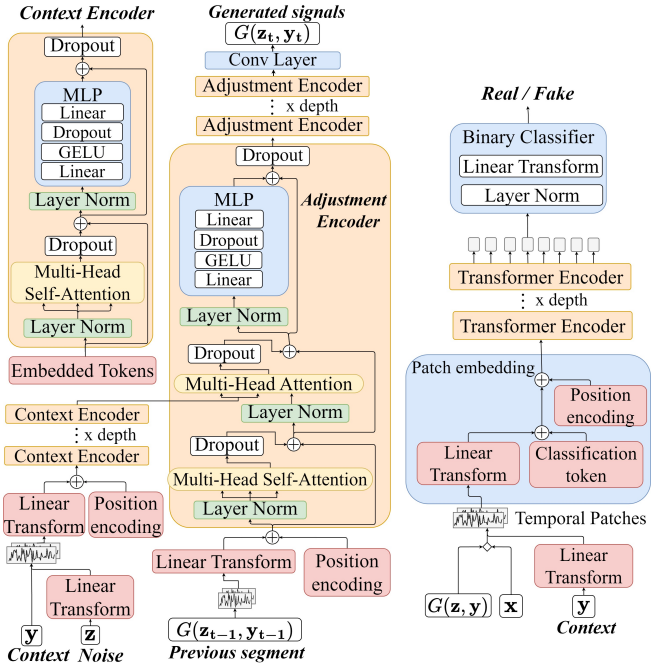
caille, 2024; Madane & Lacaille, 2023). This transformer-based framework consists of a generator and a discriminator, respectively Figure 2a and Figure 2b, with data generation conditioned on the preceding generated time series and the specific flight mission profile segments to preserve temporal continuity. The generator integrates a Context Encoder, which processes a noise vector and mission parameters through transformer encoder blocks with multi-head self-attention to capture contextual dependencies, and an Adjustment Encoder, which refines continuity by combining embeddings from the previous segment with context features. The discriminator classifies whether CEOD inputs are real or generated. We use the Least Squares GAN (LSGAN) loss, with an added custom generator loss that enforces smooth transitions across overlapping segments. Both networks are trained in parallel to optimize L_D and L_G.

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

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

2.4. Wear Modeling

The wear modeling framework addresses the unique wear patterns of individual aircraft engines by comparing their performance to a "clean average engine simulator" S_{clean}



(a) Generator (b) Discriminator

Figure 2. Architecture of the Generator and the Discriminator

trained exclusively on non-degraded fleet data. This baseline simulator predicts optimal engine behavior under given operational and environmental conditions, enabling the calculation of residuals, which in this case were computed as the mean deviations between simulated and actual performance that quantify wear effects. Residual sequences from multiple operational cycles are used to train a forecasting model to predict the next cycle's residual based on the previous ones. The predicted residual is then added to the clean simulator's output, producing an adjusted simulation that reflects the engine's anticipated degraded performance for upcoming missions. This adaptive approach allows near real-time wear tracking and mission-specific simulation adjustments. Given operational inputs X, the simulator predicts

$$Y_{\text{predicted}} = S_{\text{clean}}(X),$$

which is compared to actual performance Y_{actual} to compute the residual

$$R = \text{mean}\big(Y_{\text{actual}} - S_{\text{clean}}(X)\big).$$

Using sequences of residuals $\{R_i\}$ over engine life cycles, a predictive model f estimates the next residual

$$\hat{R}_{t+1} = f(R_t, R_{t-1}, R_{t-2}).$$

The adjusted simulation output incorporating wear is then 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 $S_{\text{clean}}(X)$.

2.5. Residual-Based Wear Incorporation: Synthetic Success vs Real-World Complexity

The proposed wear modeling approach demonstrated outstanding performance when evaluated on the N-CMAPSS dataset. The method accurately learned and reproduced degradation trends, with the residual-based adjustment effectively capturing the deviation between the *clean average engine simulator* S_{clean} and the degraded engine outputs. In this simulated dataset, the residual R exhibited a near-uniform distribution across the time series for each variable, acting effectively as a constant offset as shown in Figure 3. Under these conditions, incorporating degradation through $Y_{\text{adjusted}} = S_{\text{clean}}(X) + \mathbf{1} \cdot \hat{R}_{t+1}$ proved highly effective, producing outputs that accurately align with the degraded engine behavior.

However, a key limitation appeared when applying the same methodology to real-world CEOD. In actual engine data, the degradation-induced residual is not uniformly distributed across the entire time series. Instead, it varies temporally within each parameter, reflecting complex operational and environmental interactions absent in the synthetic N-CMAPSS

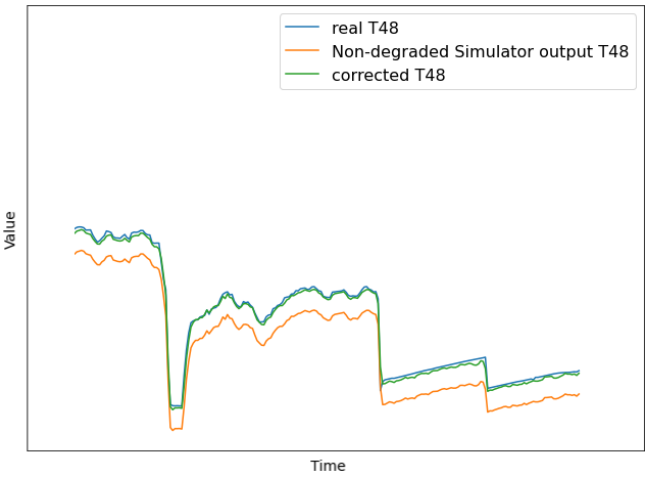


Figure 3. Example visualization of a segment of the T48 parameter for a selected N-CMAPSS flight, illustrating the constant residual between the non-degraded simulator output (T48, orange) and the actual recorded T48 (blue).

setting. In other words, we can't assume that the same degradation pattern applies to all time steps, because the residual patterns vary with flight phase and operational context. This temporal variability means that degradation cannot always be represented as a simple additive offset applied uniformly to all time steps. Consequently, the residual adjustment addition fails to fully capture the time-dependent deviations observed in real operational data. This highlighted the necessity for a more flexible degradation incorporation strategy capable of capturing temporally varying residual patterns throughout the mission profile.

3. PROPOSED APPROACH

We assume that one contributing factor to the temporal variation in observed residuals is the change in engine operating conditions across different segments of the mission profile. To account for this, we treat the data within discretized time frames, such as distinct flight phases, where the operational regime is relatively consistent. So we proceed using a phase-conditioned wear incorporation method. The core idea is to (i) decompose each mission into coherent macro-phases, (ii) within each macro-phase, isolate steady-state operating segments defined as intervals where both the measured inputs and outputs exhibit minimal variation over a sliding window, ensuring the engine operates under quasi–steady conditions where wear manifests as a persistent bias, and (iii) estimate a robust bias through a high quantile of the residuals computed on each micro-phase.

Let $X_t \in R^d$ denote the input features (e.g., flight conditions, control settings) at time t, and let $Y_t \in R^k$ be the corresponding measured engine outputs. The output of a clean (non-degraded) simulator is denoted by $S_{\text{clean}}(X_t)$. The residual is

then defined as:

$$R_t = Y_t - S_{\text{clean}}(X_t),$$

representing the discrepancy between real engine measurements and the non-degraded simulator response.

3.1. Phase Decomposition

We partition each flight into three distinct temporal phases:

$$\text{Mission} = \mathcal{T}_{\text{pre}} \cup \mathcal{T}_{\text{cruise}} \cup \mathcal{T}_{\text{post}},$$

where $\mathcal{T}_{\text{pre}}, \mathcal{T}_{\text{cruise}}, \mathcal{T}_{\text{post}}$ represent the sets of time indices corresponding to pre-cruise (taxi-in, takeoff and climb), cruise, and post-cruise (descent, landing, and taxi-out) phases respectively. Within each phase $\mathcal{T}_p$, $p \in \{\text{pre}, \text{cruise}, \text{post}\}$, we further extract stabilized subphases by applying statistical filters on flight dynamics. Let $\mathcal{T}_p^{\text{stab}} \subseteq \mathcal{T}_p$ denote the set of stabilized time indices in phase p. We then compute the empirical residuals:

$$\mathcal{R}_p = \{R_t : t \in \mathcal{T}_p^{\text{stab}}\}.$$

3.2. Quantile-Based Residuals

To robustly capture degradation signals while mitigating the influence of outliers or sensor noise, we compute the 75th percentile of the residuals for each variable $j = 1, \ldots, k$ within each phase p:

$$\hat{R}_p^{(j)} = \text{Quantile}_{0.75}\left(\{R_t^{(j)}\}_{t \in \mathcal{T}_p^{\text{stab}}}\right).$$

This yields a low-dimensional phase-wise residual vector:

$$\hat{\mathbf{R}}_p = [\hat{R}_p^{(1)}, \hat{R}_p^{(2)}, \ldots, \hat{R}_p^{(k)}]^\top.$$

Collectively, these vectors form a triplet $\{\hat{\mathbf{R}}_{\text{pre}}, \hat{\mathbf{R}}_{\text{cruise}}, \hat{\mathbf{R}}_{\text{post}}\}$, which serves as a compact health signature for the engine at a given flight cycle.

3.3. Wear Tracking and Engine Health Indicators

Over successive flights throughout the engine life cycle, the evolution of each residual quantile vector $\hat{\mathbf{R}}_p^{(i)}$ (for engine i) reveals trends reflective of wear progression. Empirically, we observe that several components of $\hat{\mathbf{R}}_p^{(i)}$ exhibit monotonic or quasi-monotonic decline as engines accumulate cycles, which aligns with expected degradation. These residual quantiles can be used as health indicators, sensitive to subtle deviations not captured by average residuals. Furthermore, the phase-specific nature of the quantiles allows for discriminative tracking of wear that manifests differently across flight regimes.

This strategy improves upon global residual modeling by adapting to the operational regime, capturing subtle degradation effects that occur in specific operational windows, and providing interpretable health indicators easily integrated into PHM applications such as Remaining Useful Life (RUL) estimation and anomaly detection. It's a transition from static, uniform residual adjustment to phase-wise quantile residual modeling, enabling accurate and physically grounded tracking of engine wear in real-world conditions.

3.4. Phase-Wise Predictive Correction Model

Building upon the extracted phase-wise bias vectors $\{\hat{\mathbf{R}}_{\text{pre}}, \hat{\mathbf{R}}_{\text{cruise}}, \hat{\mathbf{R}}_{\text{post}}\}$, we train a corrective model designed to learn the systematic deviation between simulated (non-degraded) outputs and real engine sensor measurements. This model explicitly leverages the residual structure learned in stabilized flight phases to refine the simulator predictions and produce sensor outputs that more faithfully reproduce real-world behavior.

For each flight phase $p \in \{\text{pre}, \text{cruise}, \text{post}\}$, we train a phase-specific predictive model $\mathcal{C}_p$ that maps the residual quantile vector $\hat{\mathbf{R}}_p$ to the corresponding deviation between the simulator output and the actual measured signal:

$$\mathcal{C}_p : \hat{\mathbf{R}}_p \mapsto \Delta Y_p,$$

where $\Delta Y_p = Y_p - S_{\text{clean}}(X_p)$ denotes the observed deviation within phase p.

This deviation is then added to the clean simulator output to yield the corrected prediction:

$$\hat{Y}_p = S_{\text{clean}}(X_p) + \mathcal{C}_p(\hat{\mathbf{R}}_p).$$

The model $\mathcal{C}_p$ is trained by minimizing the mean squared error (L2 loss) between the corrected prediction $\hat{Y}_p$ and the ground truth sensor measurements Y_p.

Each engine's correction model is trained independently, leveraging its own operational history. We split the data into three sets: training, validation, and test. The goal is to capture the underlying structure of wear-induced deviations as a function of the bias pattern $\hat{\mathbf{R}}_p$. This allows us to condition the correction on the estimated health state of the engine.

The architecture consists of a lightweight yet effective neural network. It's a feedforward multilayer perceptron architecture with three hidden layers with progressively decreasing widths: 256, 128, and 64 neurons. , Each hidden layer is followed by a ReLU activation, batch normalization, and dropout (with rate 0.2). The final output layer maps to the predicted deviation $\widehat{\Delta Y}_p$. The corrected output is computed by adding the predicted deviation to the clean simulator signals.

Overall, this approach preserves the physical consistency of the simulator while integrating data-driven corrections in-

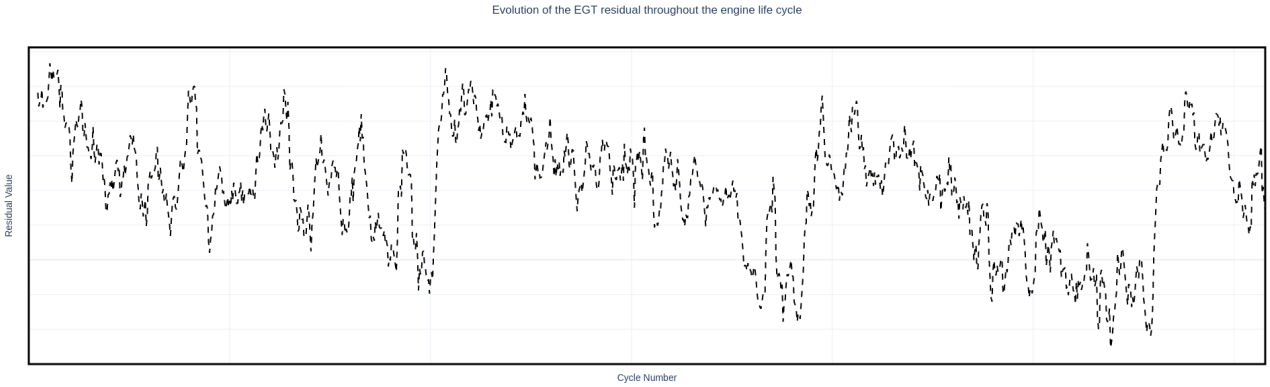


Figure 4. Example of the evolution of the EGT residual throughout an engine life cycle

formed by real-world operational deviations. And by working within each phase independently, the model accounts for the fact that wear can manifest differently depending on the engine's operational regime.

4. RESULTS AND DISCUSSION

Figure 4 shows the temporal evolution of the Exhaust Gas Temperature (EGT) residual throughout the operational life cycle of a single engine. This residual would remain close to zero in a hypothetical scenario with negligible degradation and perfect modeling fidelity, reflecting only stochastic sensor noise and minor environmental discrepancies.

However, the observed trajectory exhibits structured, non-random variation over successive cycles, which strongly suggests the presence of progressive degradation trends. In particular, four distinct wear followed by recovery cycles can be identified: in each, the residual exhibits a gradual decline, indicative of progressive performance loss, followed by an abrupt return to a higher baseline. These sharp upward shifts align temporally with known maintenance interventions. The repetition of this wear and restoration pattern across multiple operational intervals confirms the strong coupling between residual evolution and both the underlying wear mechanisms and the maintenance regime.

The high-frequency oscillations superposed on the long-term trend likely originate from mission-to-mission variability in ambient conditions and power settings, combined with modeling approximations in the clean simulator.

From a PHM perspective, the evolution of the residual provides valuable information about the engine's health state. The overall downward trend can be used as an indicator of wear and an early signal for upcoming performance restoration. Short-term variations in the residual can also be useful for detecting anomalies. These patterns support the use of the proposed phase-wise quantile residual modeling, which

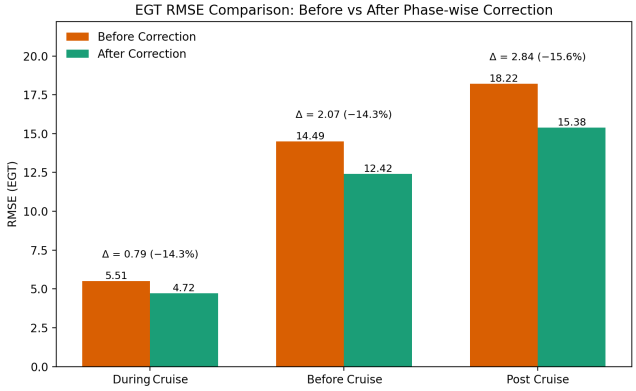


Figure 5. Root Mean Square Error (RMSE) of EGT between real engine measurements and clean simulator outputs, shown for the three major flight phases: before cruise, during cruise, and post cruise. Results are presented before and after application of the proposed phase-wise correction model.

is designed to capture stable, wear-related features from variable data. In addition, the observed recovery events show the importance of using models that can handle non-monotonic wear trends, which often occur in fleet operations due to maintenance interventions.

Engines belonging to the same fleet are referenced to the same clean, non-degraded simulator, which makes the resulting residuals directly comparable across units. This enables an objective assessment of relative wear levels and supports identifying engines exhibiting abnormal deterioration rates. Such ability to compare the engines is highly valuable for fleet-level PHM, where accurate health ranking underpins effective maintenance planning and resource allocation.

Figure 5 presents the RMSE of the EGT between real engine measurements and the outputs of the clean, non-degraded simulator, evaluated for three operational phases: before cruise, during cruise, and post cruise. The evaluations shown

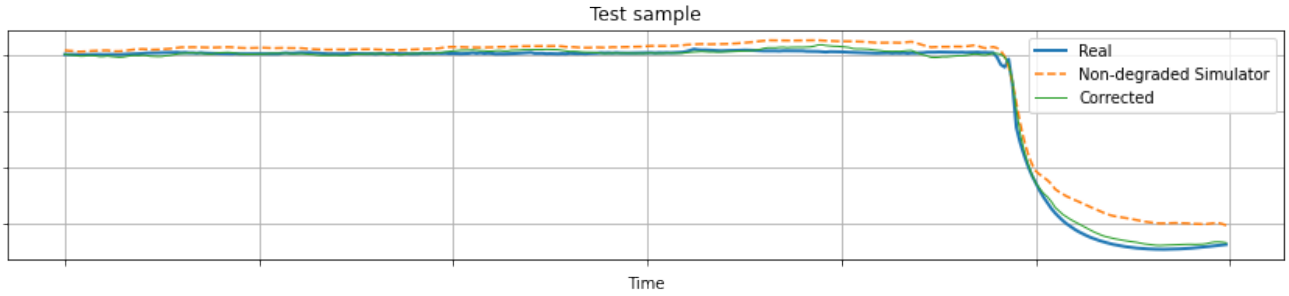


Figure 6. Example visualization of a segment of the EGT parameter from the test subset, illustrating the non-degraded simulator output (orange), the actual recorded measurement (blue), and the corrected output (green).

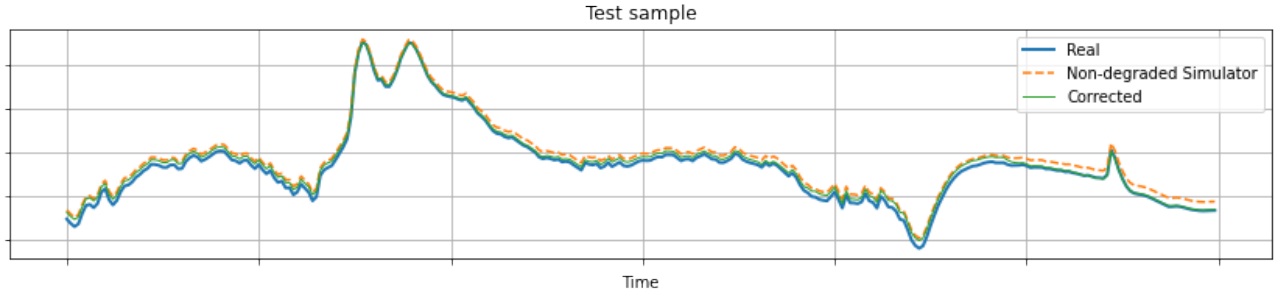


Figure 7. Example visualization of a segment of the EGT parameter from the test subset, illustrating the non-degraded simulator output (orange), the actual recorded measurement (blue), and the corrected output (green).

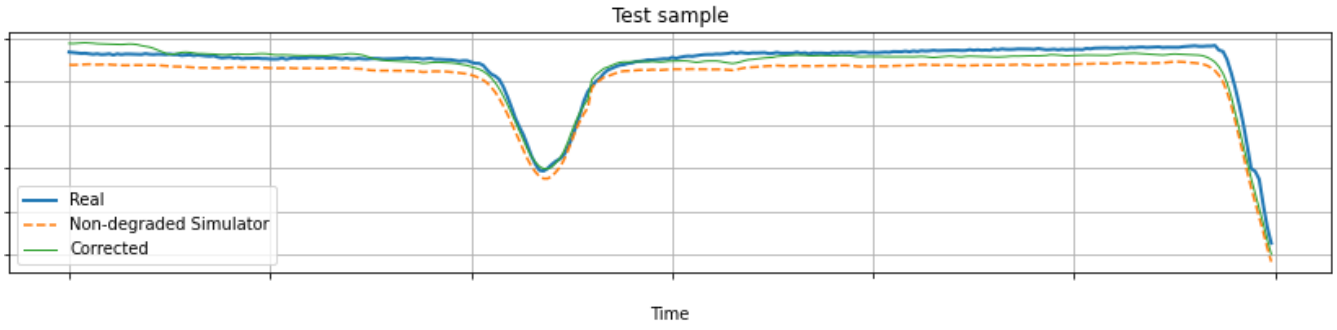


Figure 8. Example visualization of a segment of the EGT parameter from the test subset, illustrating the non-degraded simulator output (orange), the actual recorded measurement (blue), and the corrected output (green).

were conducted on the test subset. For all phases, the proposed phase-wise correction model reduces the RMSE relative to the uncorrected simulator output, indicating improved alignment with the real measured data.

The improvement is most pronounced in the post-cruise phase, where RMSE decreases from 18.22 to 15.38 (15.6%), followed by the during-cruise phase (5.51 to 4.72, 14.3%) and the before-cruise phase (14.49 to 12.42, 14.3%). The cruise phase exhibits lower RMSE because engine operation is relatively stable, with limited quick events. In contrast, the before and post-cruise phases involve frequent throttle changes, resulting in highly dynamic engine behavior that is more difficult for the simulator to replicate precisely.

While the consistent RMSE reduction across phases confirms the model's capability to capture wear-induced deviations, the improvement remains moderate. The residual errors after correction indicate that not all variability is accounted for, which may result from unmodeled operational effects or other environmental influences. Particularly, it may require more complex or phase-specific modeling strategies to achieve further accuracy gains in the pre- and post-cruise segments.

Figures 6-7-8 illustrate example visualizations of EGT segments from the test subset, showing the non-degraded simulator output (orange), the actual recorded measurement (blue), and the corrected output from the proposed model (green). In all cases, the correction model reduces the residual gap between the simulator and measured values. It should be noted that the uncorrected simulator output is already highly accurate, as real-world aircraft engines are subject to regular and effective maintenance, which limits the occurrence of pronounced degradation patterns. Consequently, large discrepancies between simulated and measured EGT are uncommon, yet the proposed correction still achieves measurable improvement.

5. CONCLUSION

This work shows that the methods validated on benchmark datasets of aircraft engines can be extended to real engines by accounting for the phase-dependent, time-varying nature of residuals. By learning how simulator outputs drift from real measurements over different flight phases, we can correct these deviations, track degradation and maintenance cycles, and compare wear levels across a fleet. For operators, that means earlier warning of emerging issues, better maintenance planning, and more informed allocation of resources.

The findings also highlight that real-world engines don't follow simple, steady wear patterns. Future approaches will need to adapt to each phase of flight, blend data-driven learning with physical understanding of engine wear, and account for environmental and operational variability. Closing this gap will be key to delivering scalable, real-time health monitoring that works reliably from research benches to active fleets.

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).

Forest, F., Lacaille, J., Lebbah, M., & Azzag, H. (2018). A generic and scalable pipeline for large-scale analytics of continuous aircraft engine data. In *2018 ieee international conference on big data (big data)* (pp. 1918–1924).

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., Forest, F., Azzag, H., Lebbah, M., & Lacaille, J. (2023). Aesim: A data-driven aircraft engine simulator. In *Proc. 33rd int. joint conf. artif. intell. int. joint conf. artif. intell. org* (pp. 8737–8740).

Madane, A., Forest, F., Azzag, H., Lebbah, M., & Lacaille, J. (2024). One-pass generation of multivariate time series through conditional multivariate modeling. In *2024 international joint conference on neural networks (ijcnn)* (p. 1-9). doi: 10.1109/IJCNN60899.2024.10651016

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

Madane, A., Lacaille, J., Forest, F., Azzag, H., & Lebbah, M. (2024). Assessing aircraft engine wear through simulation techniques. In *Annual conference of the phm society* (Vol. 16).

Saxena, A., Goebel, K., Simon, D., & Eklund, N. (2008). Damage propagation modeling for aircraft engine run-to-failure simulation. In *2008 international conference on prognostics and health management* (pp. 1–9).

Von Krannichfeldt, L., Orehounig, K., & Fink, O. (2024). Combining physics-based and data-driven modeling for building energy systems. *arXiv preprint arXiv:2411.01055*.

Xiao, B., Zhong, J., Bao, X., Chen, L., Bao, J., & Zheng, Y. (2024). Digital twin-driven prognostics and health management for industrial assets. *Scientific reports*, *14*(1), 13443.