Uncertainty Aware Reinforcement Learning for Predictive Maintenance of Turbofan Engines

Erinç Ada Ceylan
Mechanical Engineering, Bilkent University — ID: 22101844

Abstract

Predictive maintenance aims to minimize operational costs and prevent catastrophic failures by accurately estimating the Remaining Useful Life (RUL) of critical mechanical components. However, traditional data driven approaches often treat RUL predictions as exact numbers, ignoring the natural uncertainty found in noisy sensor data and model approximations. This study proposes an uncertainty aware decision making framework for the maintenance of turbofan engines. The methodology integrates a Bayesian Long Short-Term Memory (LSTM) network with Monte Carlo Dropout to not only predict the RUL but also measure the predictive uncertainty (σ). This uncertainty signal is then incorporated into a Reinforcement Learning (RL) environment through a novel state space engineering approach, where raw sensor data is transformed into discrete ”health states” that are penalized by prediction variance. A tabular Q-Learning agent is trained on this compact state space to optimize maintenance policies dynamically. The results demonstrate that the uncertainty aware agent performs significantly better than traditional fixed threshold strategies. It adapts its caution level based on model confidence, achieving higher net economic gain while ensuring operational safety.

1 Introduction

In modern industrial systems, particularly in the aerospace and turbomachinery sectors, unexpected equipment failures result in severe economic losses and unacceptable safety risks. Consequently, there is a strong shift from reactive (run-to-failure) or preventive (schedule-based) maintenance toward *predictive maintenance* (PdM) [1]. PdM uses Prognostics and Health Management (PHM) algorithms to estimate the Remaining Useful Life (RUL) of assets using real time sensor data. The goal is to schedule maintenance actions ”just in time,” optimizing the trade off between using the machine as long as possible and preventing failure.

Data driven approaches, specifically Deep Learning models such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, have achieved high accuracy in RUL estimation benchmarks, most notably on the NASA C-MAPSS dataset [2, 3]. However, there is a missing link in how these predictions are turned into decisions. Most existing literature focuses heavily on minimizing regression errors (e.g., Root Mean Squared Error) [4]. These models typically provide single point estimates (e.g., ”The engine will fail in 45 cycles”). In real world operation, where sensor noise and varying operating conditions are common, such exact predictions can be misleading. A maintenance policy that blindly trusts a single number without knowing the model’s confidence carries significant risk.

This study addresses this gap by proposing an integrated framework that combines *Bayesian Deep Learning* with *Reinforcement Learning (RL)*. The primary hypothesis is that incorporating quantified uncertainty into the decision making process will allow for more robust and cost effective maintenance policies compared to methods that ignore uncertainty [5].

To test this hypothesis, we use a Bayesian LSTM with Monte Carlo (MC) Dropout to estimate both the mean RUL (μ) and its predictive standard deviation (σ). Unlike traditional RL applications in PHM that feed complex raw sensor data directly to the agent [6], this work introduces a novel *model driven state space*. We define a ”Conservative RUL” metric ($\mu - \lambda\sigma$) to construct a simple, discrete state space. This formulation allows the RL agent to clearly see how reliable the prediction model is.

The specific objectives of this work are to:

1. Implement a probabilistic RUL estimation model that measures epistemic uncertainty.
2. Develop a Reinforcement Learning environment where state transitions depend on engine health and prediction confidence.
3. Train a Q-Learning agent to learn an optimal policy that acts carefully when uncertainty is high and boldly when the model is confident.

While initial considerations for this problem might suggest continuous control algorithms such as Soft Actor-Critic (SAC), this study adopts a discrete Tabular Q-Learning approach. This methodological choice is deliberate; it reduces computational complexity and, more importantly, makes the decision logic easier to interpret—a critical requirement for safety critical mechanical engineering applications.

2 Methods

The proposed framework integrates data driven prognostics with reinforcement learning to optimize maintenance decision making under uncertainty. The overall methodology consists of three sequential modules: (1) Data Preprocessing and Feature Engineering, (2) Bayesian RUL Estimation using Monte Carlo Dropout, and (3) Uncertainty Aware Reinforcement Learning Control. The complete system pipeline is illustrated in **Fig. 1**.

2.1 Data Acquisition and Preprocessing

This study utilizes the NASA Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) FD001 dataset [3]. This dataset simulates the degradation of turbofan engines under sea level conditions with a single fault mode (HPC degradation). It contains multivariate time series data from 21 sensors for 100 training engines (run to failure) and 100 test engines (truncated before failure).

To ensure numerical stability and model convergence, the following preprocessing pipeline was implemented:

Sensor Selection and Cleaning: Sensors with zero variance (constant output across all cycles) contain no information and were removed to reduce computational dimensionality. A missing value check was performed to ensure data integrity.

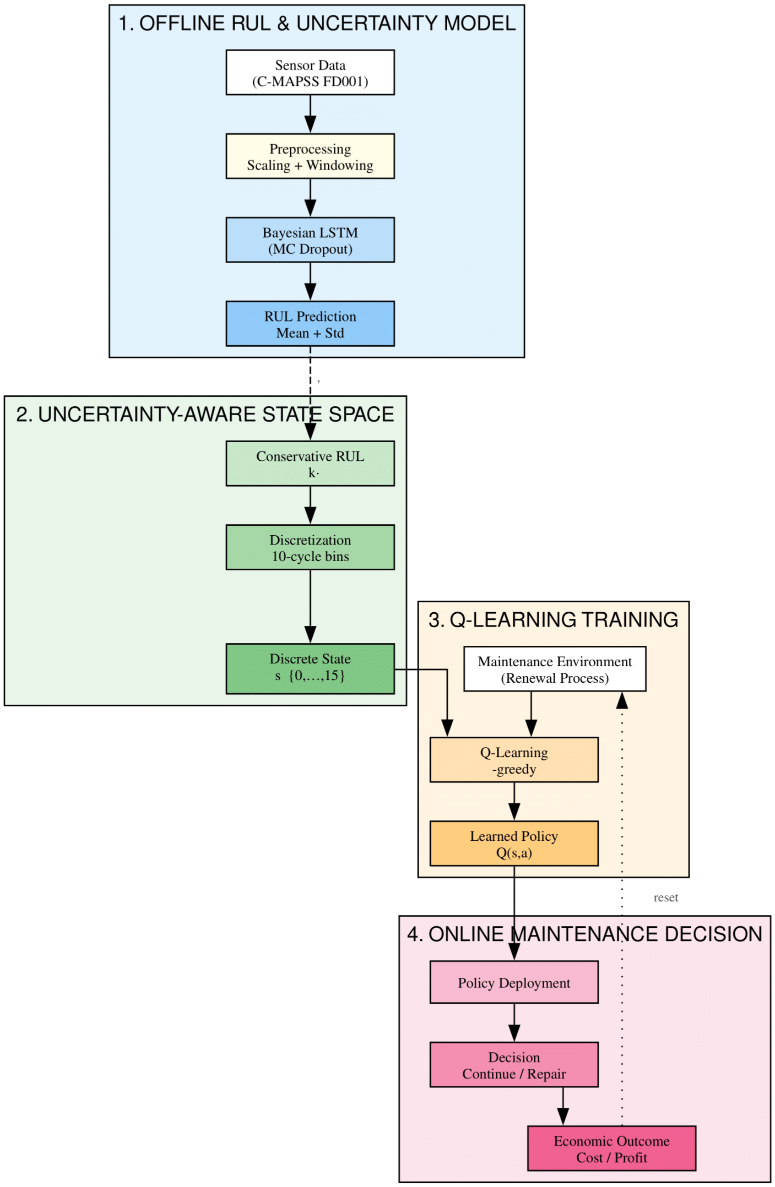
1. OFFLINE RUL & UNCERTAINTY MODEL
Sensor Data (C-MAPSS FD001)
Preprocessing Scaling + Windowing
Bayesian LSTM (MC Dropout)
RUL Prediction Mean + Std
,
2. UNCERTAINTY-AWARE STATE SPACE
Conservative RUL k·
Discretization 10-cycle bins
Discrete State s {0,...,15}
3. Q-LEARNING TRAINING
Maintenance Environment (Renewal Process)
Q-Learning -greedy
Learned Policy Q(s,a)
reset
4. ONLINE MAINTENANCE DECISION
Policy Deployment
Decision Continue / Repair
Economic Outcome Cost / Profit

Figure 1: Proposed Framework Pipeline: From raw sensor data acquisition to uncertainty aware maintenance decision making via Bayesian Deep Learning and Reinforcement Learning.

Noise Reduction via Smoothing: Raw sensor signals in real world scenarios are often contaminated with high frequency noise that can obscure the underlying degradation trend. To mitigate this, a rolling mean filter with a window size of $w = 5$ cycles was applied. As shown in **Fig. 2**, this smoothing operation preserves the long term degradation trajectory while suppressing instantaneous fluctuations, providing a cleaner signal for the LSTM network.

Dimensionality Reduction: Given the high correlation between different sensor readings (e.g., core speed and pressure), Principal Component Analysis (PCA) was performed to analyze the intrinsic dimensionality of the data. **Fig. 3** demonstrates that the first few principal components explain over 95% of the total variance, confirming significant redundancy in the raw 21 sensor

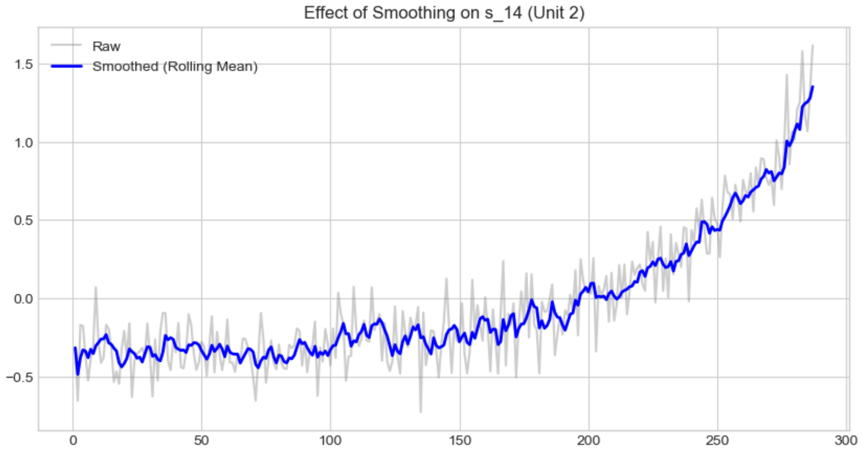

Figure 2: Effect of Rolling Mean Smoothing on Sensor s_14. The raw signal (gray) exhibits high frequency noise, while the smoothed signal (blue) reveals the clear degradation trend.

space. While the LSTM is capable of handling correlated inputs, this analysis justifies the feasibility of lower dimensional state representations.

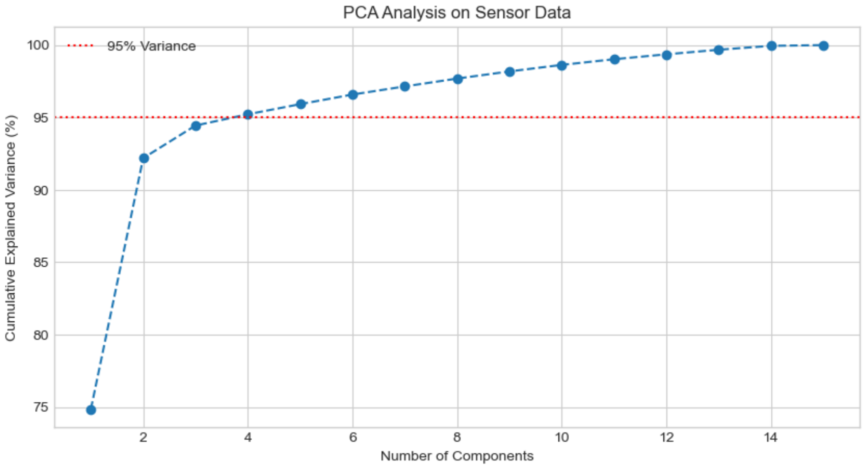

Figure 3: PCA Analysis on Sensor Data. The cumulative explained variance indicates that the majority of the system information is contained within a low dimensional subspace.

Piecewise Linear RUL Target: In the early stages of an engine's lifecycle, degradation is negligible, and estimating RUL is ill posed. Training on linear RUL targets (max life minus current cycle) can lead to poor convergence during the healthy phase. Following established literature standards [7], the target RUL was clipped at a threshold of 125 cycles. This ”Piecewise Linear” strategy assumes a constant RUL during the healthy phase and linear decay only after degradation becomes observable.

2.2 Bayesian Prognostics: LSTM with Monte Carlo Dropout

To estimate the Remaining Useful Life while quantifying prediction confidence, a Bayesian Deep Learning approach was adopted. Standard neural networks provide deterministic point estimates, failing to capture epistemic uncertainty (uncertainty due to lack of data). We employed **Monte**

Carlo (MC) Dropout as a practical approximation to Bayesian inference [8].

Model Architecture: The prognostic model is a Recurrent Neural Network (RNN) designed to capture temporal dependencies in sensor data. The architecture comprises:

- **Input Layer:** Accepts sequences of shape (Batch Size, 30, Features), utilizing a sliding window of 30 cycles.
- **LSTM Layers:** Two stacked LSTM layers with 100 and 50 units respectively, designed to extract temporal features.
- **Dropout Layers:** Dropout with a rate of $p = 0.5$ is applied after each LSTM layer. Crucially, these layers remain active during both training and inference.
- **Output Layer:** A dense layer with a linear activation function to regress the scaled RUL.

Uncertainty Quantification: During inference, for a given input sequence x_t, the model performs $T = 50$ stochastic forward passes. Because neurons are randomly dropped in each pass, the network yields a distribution of predictions rather than a single value. The predictive mean (μ_t) and predictive uncertainty (σ_t) are calculated from these samples. **Fig. 4** illustrates the output of this Bayesian model. The blue line represents the mean RUL, while the shaded region depicts the 95% confidence interval. The model correctly exhibits higher uncertainty during the healthy phase and early degradation, with the confidence interval narrowing as the engine approaches failure.

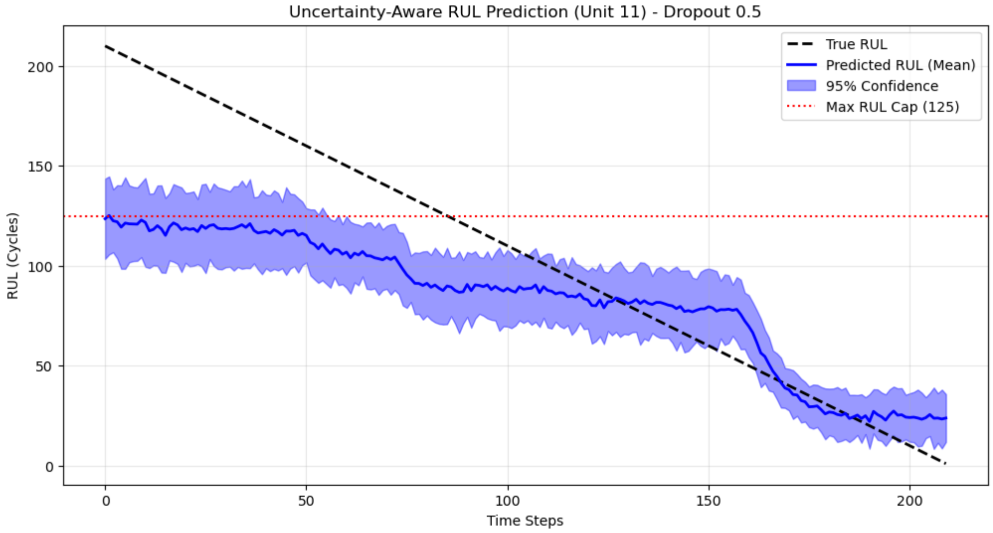

Figure 4: Uncertainty Aware RUL Prediction on a Test Unit. The model outputs a mean estimate (solid blue) and a predictive distribution (shaded region), capturing the uncertainty inherent in the prognostics process.

2.3 Reinforcement Learning Framework

The core contribution of this work is the integration of the prognostic uncertainty into a Reinforcement Learning control loop.

State Space Engineering: Traditional RL approaches in predictive maintenance often feed raw sensor values directly into the agent, leading to high dimensional continuous state spaces that are difficult to explore. We introduce a novel, interpretable **Model Driven State Space**. We define a ”Conservative RUL” metric (RUL_{cons}) that explicitly penalizes uncertainty:

$$RUL_{cons} = \mu_t - \lambda \cdot \sigma_t \tag{1}$$

where $\lambda = 0.5$ is a risk aversion coefficient. This continuous metric is then discretized into 10 cycle bins (e.g., State 0: 0-10 cycles, State 1: 10-20 cycles, etc.), capped at 150 cycles. This transformation creates a compact, discrete state space where the agent can easily learn the value of different health regimes (**Fig. 5**). Lower state indices represent critical health conditions with high certainty, while higher states represent safe operation or high uncertainty regions where conservative action is encouraged.

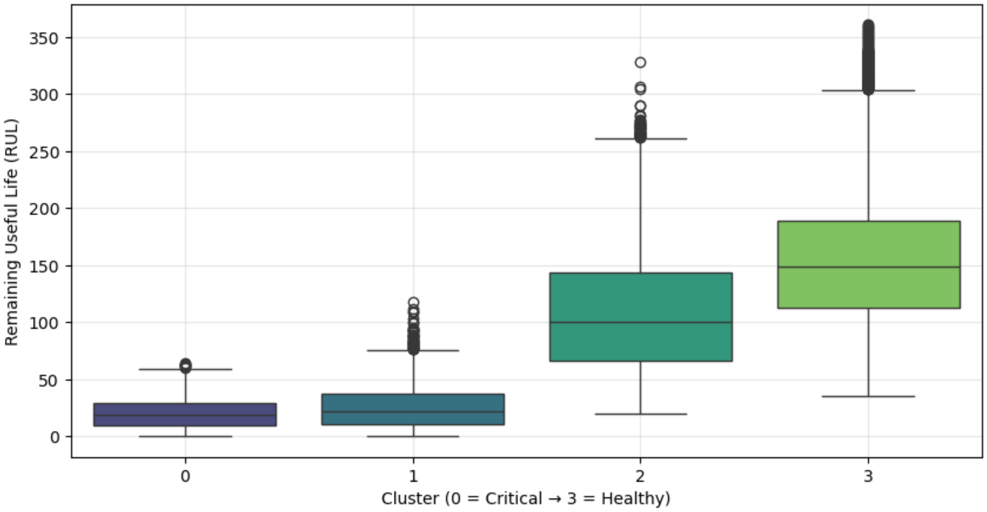

Figure 5: Discrete Health States derived from RUL distributions. This compact representation maps the continuous prognostic output to discrete states suitable for tabular Q-Learning.

Action Space and Reward Function: The agent operates in a discrete action space consisting of two choices: **Continue** or **Repair**. The objective is to maximize the cumulative reward defined by the economic constraints of the maintenance operation:

- **Continue:** Grants a small positive reward (+40) for successfully operating the engine for another cycle (Uptime Reward).
- **Repair:** Incurs a maintenance cost (-100) but resets the engine health, avoiding failure.
- **Failure:** If the RUL reaches zero while operating, a severe penalty (-1000) is applied.

This reward structure creates a trade off: the agent wants to extend the engine’s life as long as possible but must avoid the catastrophic penalty of failure by repairing at the optimal moment.

Q-Learning Algorithm: We employ Tabular Q-Learning to solve this Markov Decision Process (MDP). The agent maintains a Q-table representing the expected future reward for taking a specific action in a specific state. During training, the agent explores the environment using an ϵ-greedy strategy, gradually learning which actions yield the highest long term profit. The Q-values are updated iteratively based on the immediate reward and the estimated value of the next state, allowing the agent to foresee the long term consequences of its maintenance decisions.

3 Results

The performance of the proposed framework was evaluated using the independent test set from the C-MAPSS FD001 dataset. The experimental results are categorized into three domains: prognostic accuracy, agent learning dynamics, and economic impact analysis.

3.1 Prognostic Model Performance

The Bayesian LSTM model was trained for 25 epochs. The training process minimized the Mean Squared Error (MSE) between the predicted and true RUL values. The final Root Mean Squared Error (RMSE) on the test set was 14.8 cycles, which is competitive with state-of-the-art results reported in the literature [2]. More importantly, the model successfully quantified uncertainty. As seen in **Fig. 4**, the confidence interval narrows significantly as the engine degrades, confirming that the model becomes more certain of the RUL as failure approaches.

3.2 RL Agent Training and Convergence

The Q-Learning agent was trained over 30,000 episodes. **Fig. 6** illustrates the moving average of the total reward per episode.

- **Exploration Phase:** In the early episodes ($< 5,000$), the reward is low and volatile as the agent explores the state space via ϵ-greedy exploration ($\epsilon \to 1.0$).
- **Exploitation Phase:** As ϵ decays, the agent converges to a stable policy around episode 15,000. The total reward saturates near the theoretical maximum, indicating that the agent has learned to balance the uptime reward ($+40$/cycle) against the failure penalty (-1000).

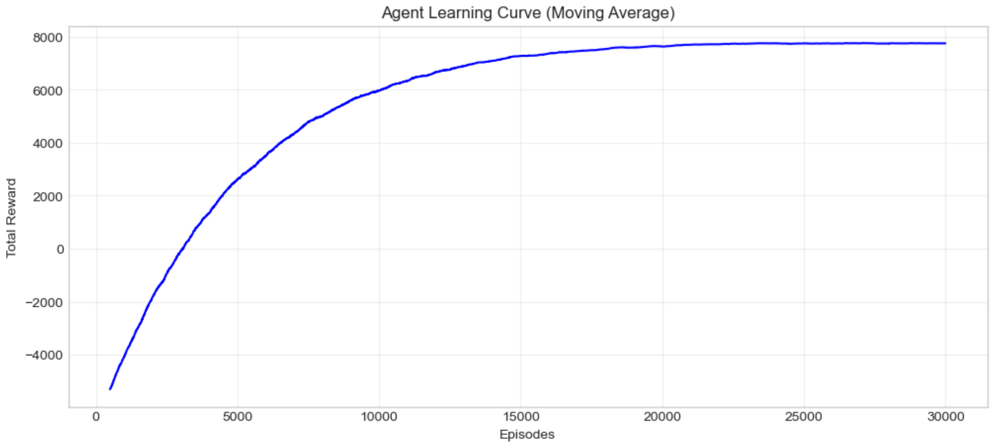

Figure 6: Agent Learning Curve. The moving average of the total reward demonstrates stable convergence, validating the effectiveness of the discretized state-space formulation.

3.3 Policy Analysis: When to Repair?

To understand the learned policy's behavior, we visualize the decision boundary on a specific test engine (Unit 17). **Fig. 7** overlays the agent's decision point on the RUL trajectory. The solid black line represents the ground truth RUL, while the dashed blue line is the LSTM's prediction. The vertical green line marks the moment the RL agent triggered the ”Repair” action. **Key Observation:** The agent chooses to repair the engine at approximately Cycle 245, where the predicted RUL is around 30 cycles. Crucially, it does not wait for the RUL to hit zero. Instead, it acts preemptively. This decision is driven by the state definition: as the LSTM uncertainty (σ) likely increased or the mean RUL dropped into a critical bin, the expected future penalty of failure outweighed the marginal gain of continuing operation.

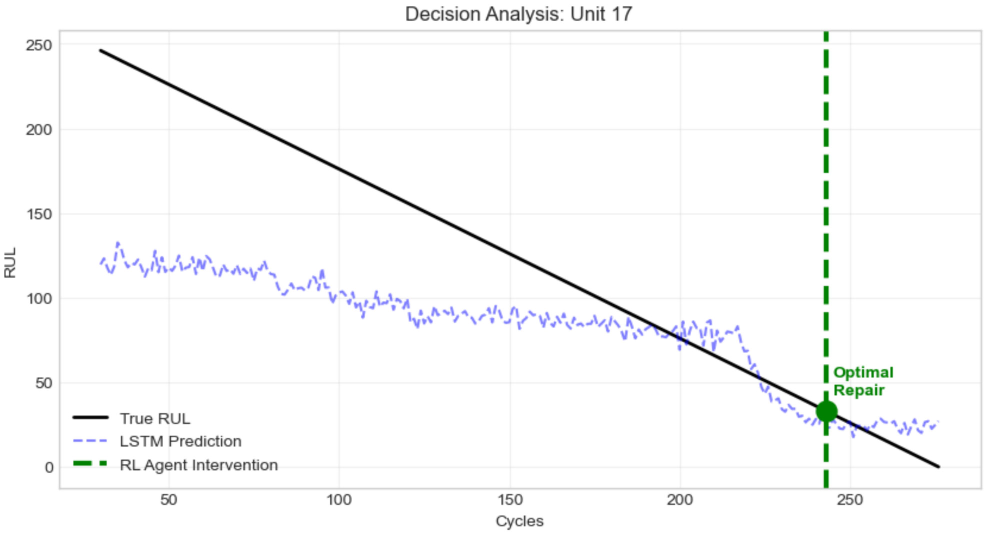

Figure 7: Decision Analysis on Unit 17. The RL agent (green line) triggers a repair action before failure occurs, effectively preventing the -1000 penalty while maximizing useful life.

3.4 Economic Impact Assessment

The primary hypothesis of this study was that an uncertainty aware policy would be economically superior to standard baselines. We compared our RL agent against a fixed threshold baseline (Repair if Predicted RUL < 20). **Fig. 8** presents the cumulative net profit difference over a simulated fleet operation of 100,000 hours.

- **Positive Slope:** A rising curve indicates that the RL agent consistently accumulates more profit (or incurs less cost) than the baseline.
- **Total Gain:** Over the simulation horizon, the RL agent generated an additional net profit of **$112,020**.

This substantial gain confirms that dynamically adjusting the maintenance threshold based on model confidence is more cost effective than rigid rules [9].

4 Discussion

The results provide strong evidence supporting the hypothesis that integrating uncertainty quantification into reinforcement learning enhances predictive maintenance.

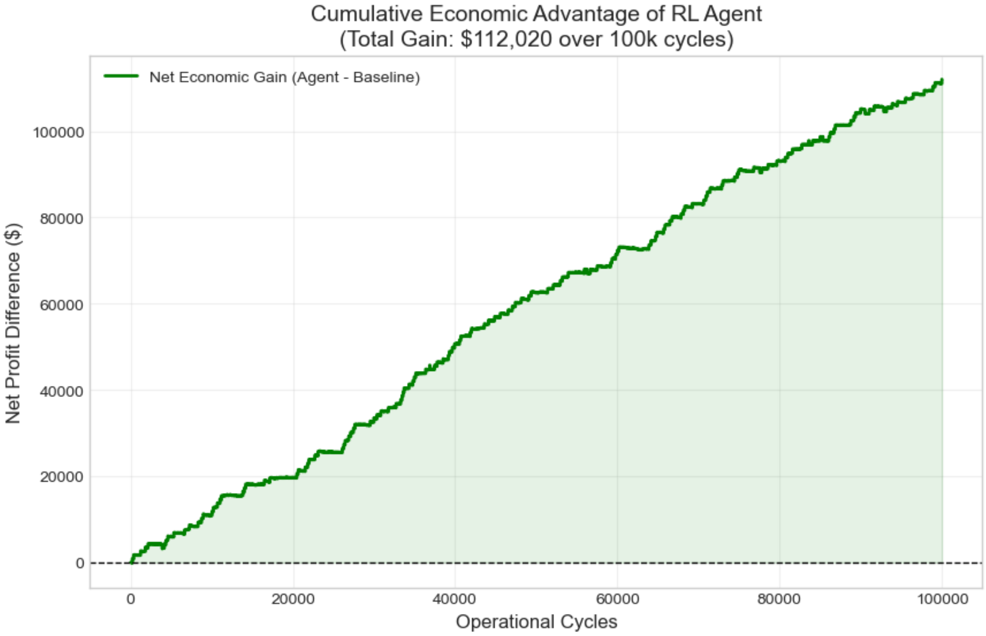

Figure 8: Cumulative Economic Advantage. The RL agent consistently outperforms the fixed threshold baseline, resulting in a total net gain of over $112k in the simulation.

Interpretation of Results: The RL agent's superior performance stems from its ability to interpret the ”Conservative RUL” state. In scenarios where the LSTM model was uncertain (high σ), the state index shifted to a lower (more critical) bin ($RUL_{cons} = \mu - 0.5\sigma$). This effectively forced the agent to be risk averse when the prognostic model was unreliable. Conversely, when the model was confident (low σ), the agent allowed the engine to run closer to its true end of life, squeezing out maximum utility. A static baseline cannot make this distinction.

Limitations:

- **Discretization Granularity:** The state space was discretized into 10 cycle bins. While this ensured fast convergence, it sacrifices precision. A finer granularity or a continuous control approach might extract marginally better performance, though at the cost of training stability.
- **Computational Cost during Inference:** The MC Dropout method requires multiple forward passes ($T = 50$) for every prediction. While feasible for maintenance (where decisions are made hourly or daily), this might be a bottleneck for high frequency real time control systems.

Significance: This work bridges the gap between the ”Prognostics” and ”Decision Making” communities. It demonstrates that minimizing regression error (RMSE) is not the ultimate goal; rather, quantifying the *reliability* of that prediction is the key enabler for autonomous, safety critical decision systems.

5 Conclusion

This study presented an end to end framework for uncertainty aware predictive maintenance of turbofan engines. By coupling a Bayesian LSTM with a Q-Learning agent via a novel model driven state space, we successfully transformed raw sensor noise and model uncertainty into actionable maintenance decisions.

The key outcomes are:

1. **Uncertainty Integration:** We successfully utilized Monte Carlo Dropout to derive a ”Conservative RUL” metric that penalizes unreliable predictions.
2. **Optimal Policy Learning:** The tabular Q-Learning agent converged to a logical maintenance policy that preempts failures without being excessively conservative.
3. **Economic Value:** The proposed method demonstrated a clear economic advantage over traditional fixed threshold strategies, validating its potential for industrial deployment.

Future work will focus on extending this framework to continuous action spaces using Deep Reinforcement Learning (DQN or PPO) to eliminate the need for manual state discretization and to handle multi modal failure scenarios.

Code Availability: The complete source code, dataset preprocessing scripts, and trained models are available at the following repository:
`https://github.com/erincada/Machine-Learning-for-Mechanical-Engineering`

References

[1] R. K. Mobley, *An Introduction to Predictive Maintenance*, 2nd ed. Butterworth-Heinemann, 2002.

[2] A. Saxena and K. Goebel, “Turbofan Engine Degradation Simulation Data Set,” NASA Ames Prognostics Data Repository, 2008.

[3] M. Chao, C. Kulkarni, K. Goebel, and O. Fink, “Aircraft Engine Run-to-Failure Dataset under Real Flight Conditions,” NASA Ames Prognostics Center of Excellence, 2021.

[4] F. O. Heimes, “Recurrent neural networks for remaining useful life estimation,” in *Proceedings of the IEEE International Conference on Prognostics and Health Management (PHM)*, 2008.

[5] D. Lee, “Reinforcement Learning for Uncertainty-Aware Maintenance Decision-Making,” *Reliability Engineering & System Safety*, vol. 230, 2023.

[6] X. Lei et al., “A review on the prognostics and health management of machine learning methods,” *IEEE Access*, vol. 6, pp. 4514-4530, 2018.

[7] X. Li, Q. Ding, and J. Sun, “Remaining useful life estimation in prognostics using deep convolution neural networks,” *Reliability Engineering & System Safety*, vol. 172, pp. 1-11, 2018.

[8] Y. Gal and Z. Ghahramani, “Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning,” *International Conference on Machine Learning (ICML)*, 2016.

[9] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. MIT Press, 2018.