Proceedings of the Annual General Donald R. Keith Memorial Conference
West Point, New York, USA
April 28, 2022
A Regional Conference of the Society for Industrial and Systems Engineering

Design of a Predictive Maintenance System for Navy Jet Engines

Abdulrahman Albawardi, Reed Lawrence, Wyatt Mingus, Mohanned Yasin, and Kuo-Chu Chang

Department of Systems Engineering and Operations Research
George Mason University
Fairfax, VA 22030

Corresponding author's Email: rlawren4@gmu.edu

Author Note: Authors are senior undergraduate Systems Engineering students at GMU. Dr. K.C. Chang is the faculty advisor and sponsor of the associated capstone senior design project.

Acknowledgements: The authors would like to acknowledge GMU PhD student, Zhengyang Fan, who developed the data-driven model and mixed integer program formulation used in this design as part of research for the GMU SEOR Department.

Abstract: Navy objectives for their aircraft fleet availability have become increasingly difficult to meet given the growing complexity of aircraft maintenance and operations. Currently, conventional preventive maintenance practices struggle to maintain availability levels due to fixed scheduling of engine repairs. In this paper, a condition-based predictive maintenance system for improving upon jet engine maintenance planning is examined. The system works by analyzing collected engine sensor measurements using a data-driven model to establish the remaining useful life (RUL) of each aircraft in a fleet. Paired with operations-based input parameters, these RUL values are processed by a mixed integer programming model to optimize the scheduling of engine maintenance and assignment of aircraft for deployment. In modeling both preventive and the proposed predictive methods, experimental results find a decrease in the number of repairs required to sustain aircraft reliability with predictive maintenance. These changes have shown that a predictive maintenance system can increase fleet availability while also decreasing total maintenance costs.

Keywords: Predictive Maintenance, Condition Based Maintenance, C-MAPSS, LSTM, Scheduling Optimization

1. Introduction

1.1 Aircraft Maintenance

Aircraft activities in mission-based operations cycle between 3 categories: available, in-flight, and under maintenance. While in operation, an aircraft engine's moving parts will degrade, and with time will require repair. Overall, aircraft maintenance types are composed of pre/post flight checks, regularly scheduled line maintenance which traditionally occurs every 24 to 48 hours, and heavy checks with complete engine overhaul that are scheduled months and years in advance.

The probability that an aircraft can successfully complete a mission is referred to as reliability. Reliability is calculated at a given time using equation 1, where λ in equation 1 is the failure rate of a given aircraft component, as calculated in equation 2. Maintenance is required to uphold these standards and downtime is required to complete repairs. Maintenance actions are divided into scheduled and unscheduled maintenance, referred to as preventive and corrective maintenance, respectively. A significant performance measure for aircraft maintenance for flight operations is aircraft availability, calculated using equation 3. Availability is calculated over a fleet of given aircraft and historically the peak availability for the Navy has been roughly 80 percent within the 21st century, but has fallen below 60 percent in recent years (Congressional Budget Office, 2022).

$$R(t) = e^\wedge(-\lambda t) \quad (1)$$

$$\lambda = (number\ of\ failures)/(total\ operating\ time) \quad (2)$$

$$1 - (Time\ grounded/under\ maintenance)/(Cumulative\ potential\ operating\ time) \quad (3)$$

Line maintenance has been chosen for optimization, as the nature of its high frequency provides the opportunity to significantly reduce the amount of time an aircraft spends in maintenance. In studying turbofan engines as used by Navy jets in their flight operations, this provides 5 separate engine components which must be maintained every 24 to 48 hours. These integral components are the fan, low- and high-pressure compressors (LPC and HPC), and low- and high-pressure turbines (LPT and HPT).

1.2 Condition-Based Predictive Maintenance

Technological advancements in sensors have created the opportunity for continuous monitoring of system health. In the case of aircraft engines, sensors collect readings of temperature, pressure, speed, and angle. Collected sensor measurements capture the performance of components where fault signatures can be identified through measurements over time. By applying machine learning, this sensor data can provide a window into the future of each engine component's performance, providing the opportunity for maintenance planning support. This form of maintenance planning and scheduling is known as condition-based predictive maintenance (CBPM). The key characteristic of determining an engine's future performance is its calculated remaining useful life (RUL). By calculating the predicted RUL of an engine, failure thresholds can be determined and serve as inputs for scheduling maintenance. In comparison to preventive maintenance, which utilizes expected failure rate to correct potential faults, CBPM allows for exact timings based on when maintenance is required. Therefore, by reducing the amount of routine maintenance, CBPM reduces downtime and labor costs compared to preventive maintenance, even with considerations for potential unscheduled repairs (Fan, Chang, Ji, & Chen, 2021).

For a CBPM system to be developed, a working model must be designed to serve as the basis for aircraft prognostics testing. A working model can be formed by employing a combination of a physics-based system model that generates data for utilization with a data-driven model. In order to facilitate CBPM, run-to-failure data is needed to train the machine learning algorithms comprising the data-driven model. However, there is a scarcity in real-world run-to-failure data due to the large operational costs involved with its collection (Saxena et al., 2008).

An alternative to gathering real-world sensor data is the implementation of physics-based system models, or digital twins, as a method of generating simulated run-to-failure data for testing with data-driven models using machine learning algorithms. One digital twin used in this nature is the NASA program titled The Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) which is a nonlinear, dynamic, component-level model of a commercial, high-bypass, dual-spool turbofan engine (Saxena et al., 2008; Chao et al., 2021; Liu et al. 2012). C-MAPSS allows for a variation of health parameters and operational condition inputs and produces an output of corresponding simulated sensor measurements. The C-MAPSS output sensor data can be generated as run-to-failure data for simulated engines, and this generated engine data acts as an input for the data-driven model. The data-driven model itself is a neural network employing machine learning algorithms. This model is trained with the simulated run-to-failure data from the system model, and tested with additional simulated data that is truncated to verify the neural network's RUL estimation accuracy. Once the accuracy of the RULs have been verified, this trained machine learning algorithm can form the basis of a CBPM system by determining the optimal times for line maintenance actions and improving fleet availability.

2. Methods

2.1 Turbofan Engine Run-to-Failure Data Generation

Due to the lack of access to real-world run-to-failure sensor measurements for turbofan engines, simulated run-to-failure datasets were generated using the C-MAPSS model. C-MAPSS accepts inputs of varying turbofan engine component health parameters and aircraft operating conditions. To model the behavior of degradation values which serve as inputs into C-MAPSS for generating data, the degradation equation used by Saxena et al. in the 2008 IEEE PHM C-MAPSS Dataset was modified by applying the normal and abnormal degradation equations utilized by Chao et al. in the 2021 N-CMAPSS Dataset. The final resulting equation takes the form of equations 4 through 7 (Saxena, Goebel, Simon, & Eklund, 2008). Equations 4 and 5 model normal degradation, which is the natural linear wear and tear of an engine over its utilization. In these equations, $a_n(t)$ is the slope, d_0 is initial degradation between [0.0001, 0.009], and ε is process noise added to the generated health parameters values to simulate pre/post flight maintenance checks which can slightly improve or degrade the health of an engine component. At time t_s a fault signature is introduced into the degradation model, taking the form of abnormal degradation. Equations 6 and 7 model abnormal degradation, which is the exponential degradation of a component once a fault has evolved in the engine. In these equations, a and b are degradation trajectory parameters, between [0.001, 0.003] and [0.4, 0.6], respectively. Equations 4 and 5 include the sum of the normal degradation as well as continued process noise. Degraded health parameter values are generated with equations 6 and 7 until wear thresholds are surpassed and the engine is determined to be failed. Wear thresholds are determined by the user, and represent the allowable degradation of C-MAPSS virtual sensors exhaust gas temperature (EGT), HPC, LPC, and fan stall margins.

$$e\_n\,(t) \;=\; a\_n\,(t) \;+\; d\_e0 + \varepsilon\_e \quad \forall\, t \;\leq t\_s \tag{4}$$

$$f\_n\,(t) \;=\; a\_n\,(t) \;+\; d\_f0 + \varepsilon\_f \quad \forall\, t \;\leq t\_s \tag{5}$$

$$e\_a\,(t\_s) \;=\; 1 \;-\; e\text{^}(a\_e\; t\text{^}(b\_e\,)\,) + \; e\_n\,(t\_s) + \varepsilon\_e \tag{6}$$

$$f\_a\,(t\_s) \;=\; 1 \;-\; e\text{^}(a\_f\; t\text{^}(b\_f\,)\,) + \; f\_n\,(t\_s) + \varepsilon\_f \tag{7}$$

With a modified MATLAB script, C-MAPSs was used to assign a randomly generated value to d_0 , a, and b for a turbofan engine component, while also randomly assigning t_s a value for the introduction of a fault signature. Once this was done, the time parameter t was increased until wear thresholds were exceeded and the output engine sensor measurements with each iteration of the degradation values were collected. For each single and double component failure mode this process was repeated 200 times; generating 100 run-to-failure engines, and 100 testing engines truncated at random points throughout their failure, with the RUL of testing engines collected separately.

2.2 RUL Prediction and Fault Classification

In order to determine the lifespan of an engine and the types of failures that will occur, a long short-term memory (LSTM) neural network was implemented for the data-driven model. The implemented LSTM was developed by George Mason University Systems Engineering and Operations Research (GMU SEOR) department PhD student Zhengyang Fan as part of research for the SEOR Department. The LSTM architecture uses long- and short-term memory states regulated by input, output, and forget gates within each cell of the neural network. Input data is split into 30 consecutive sequences with a learning rate of 0.0015, cell dimension of 145, dropout frequency of 0.5, batch size of 50 samples, and validation split of 0.05%. The data-driven model is trained using the C-MAPSS generated 100 run-to-failure engines which were generated using a process developed separately from Fan, and the LSTM neural network was tested using the 100 generated truncated engines. To improve the model's ability to capture the behavior of abnormal degradation and predict when an engine will fail due to a component's fault, input training sensor data was truncated using the observed time when the abnormal degradation generally occurs in the dataset. This process isolates the abnormal degradation of the run-to-failure data for the model to train on.

2.3 Maintenance and Deployment Planning

A mixed integer program was used to optimize the planning of when aircraft engines were grounded for maintenance and deployed on missions based on the predicted RUL of each engine's component across a given fleet. In addition to his work on the LSTM code, Zhengyang Fan developed the objective function and corresponding constraints which were used in the development of the optimization program, and a brief overview of these constraints will briefly be given. The primary objective function maximizes aircraft availability over the total maintenance costs over a given planning horizon, where the number of aircraft required for deployment M is multiplied by the number of aircraft not currently scheduled or under maintenance over a given time period (1- z), and the cost of maintenance actions by failure mode F is multiplied by the number of maintenance actions performed over the time period ($c * x$). From Fan et. al (2021), constraint 2 enforces demands for aircraft to be deployed on a mission, constraint 3 enforces that aircraft can only perform one mission at a time, constraints 4-8 model the behavior of RUL, constraints 9-13 enforce that aircraft scheduled for maintenance are unavailable until maintenance is complete, constraints 14-16 enforce that aircraft with remaining time in maintenance are scheduled accordingly, constraints 17-19 model the relationship between decision variables related to the scheduling of maintenance by technician trade, component, and aircraft, and constraints 20-21 enforce maintenance station capacity and workforce capacity, respectively (Fan et al., 2021).

A Python script was written which implements Gurobi Optimizer to model Fan's objective function and constraints, and design the script to accept the required input parameters, RUL values, and scheduled missions (by datetime). Additionally, a random failure simulator was written in Python which accepts the Gurobi output and simulates the probability of engine components suffering a random failure after completing a mission with respect to cumulative operating time. In the event of a random failure occurring, the Gurobi model is rerun with the new maintenance requirements. This process is repeated until the full planning horizon is scheduled for mission deployment and necessary maintenance actions.

ISBN: 97819384962-2-6

3. Mission Requirements

Successfully applying the methods of the predictive maintenance system as described above requires certain characteristics and performance metrics to be met. The following high-level requirements, derived from identified stakeholder needs, determine what results must be met by the system for it to pass testing criteria:

MR.1 The CBPM system shall generate simulated run-to-failure data for each single and double failure mode of a turbofan engine given its components: Fan, LPC, HPC, HPT, and LPT.
MR.2 The CBPM system shall estimate the RUL for a given aircraft with a Type II error less than 10% for 100 RUL estimations under 50 operational flights.
MR.3 The CBPM system shall schedule the next maintenance action for a given aircraft by datetime and type of maintenance required by component level.
MR.4 The CBPM system shall identify the next maintenance actions for a given fleet of aircraft (timing and maintenance performed) such that the fleet level availability is at least 80% over a 72 hour period.

Mission requirement 1 ensures that all primary forms of turbofan engine degradation are accounted for in degradation modeling and generated run-to-failure datasets. Therefore, by recreating the degradation process across the majority of failure modes the implemented machine learning model can be accurately trained. Accordingly, mission requirement 2 ensures that the machine learning output of predicted RUL values results in maintenance actions that are not performed too early ahead of failure and not after a failure has already occurred. The format of how these maintenance actions are represented in the predictive maintenance system is determined by mission requirement 3, and mission requirement 4 determines the minimum fleet availability over a 72-hour period.

4. Results

4.1 Performance Gap Analysis

Analysis of the performance gap between preventive and predictive maintenance methods was conducted by simulating the optimization model with fixed parameters. Both simulated maintenance methods used the following input parameters: planning horizon of 7 days (168 hours), fleet size of 5 single turbofan engine aircraft, 10 maintenance technicians composed of two trades (airframe & powerplant, A&P; and inspection authorized, IA), and two maintenance stations. For these simulations, missions were generated such that there are two missions a day per aircraft between 6:00 and 18:00 hours. Each mission has an associated type with corresponding priority. For example, each mission type (training, surveillance, or escort) has an associated duration.

Turbofan component maintenance requires different amounts of technicians to repair, however each repair regardless of component is simulated to require 2 hours to complete. For the preventive simulation, initial RUL for each aircraft engine component is known at the beginning of the planning horizon. For the preventive simulation, generated C-MAPSS run-to-failure datasets were analyzed by component failure mode to establish the average number of operating hours at time of failure. Accordingly, preventive maintenance intervals were randomly assigned at 10% margins of the average failure hour for each aircraft engine component in the simulated fleet.

To better understand the comparison between preventive and predictive methods, a Monte Carlo experiment was conducted with the resulting aircraft availability and total maintenance costs collected with each run of the simulation and paired random failure simulation. After 1000 simulations of the model, average availability with predictive methods were recorded at 92% and preventive methods were recorded at 75% for a 17% increase in aircraft availability by implementing predictive maintenance. Additionally, average total maintenance costs were simulated to decrease by 45%. While costs vary depending on the operation and available resources, this change in maintenance costs is equivalent to a decrease of 10 repairs across the fleet over the 7-day time period. An important feature of this comparison is that preventive method simulations begin to cancel lower priority missions (such as training) in order to keep aircraft maintained and able to be deployed for higher priority missions. Furthermore, running the simulation with increasing durations for component repairs (i.e., 2 hours, 4 hours, 6 hours, etc.) saw the preventive method's availability begin to decrease rapidly. Figure 1 shows how predictive methods were simulated to sustain availability levels as repair times increased while preventive methods struggle to support random failures and therefore must ground aircraft for extended periods of time until maintenance stations are free.

ISBN: 97819384962-2-6

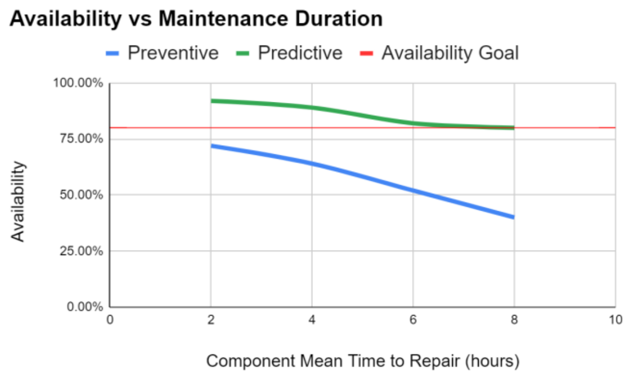

Figure 1. Aircraft Availability vs. Maintenance Duration for Preventive and Predictive Maintenance Methods

4.2 RUL Prediction and Fault Classification Testing

Verification testing of RUL prediction for the C-MAPSS generated run-to-failure datasets measured the root-mean-square error (RMSE) of the predicted RUL output of the implemented data-driven model. In literature, an RMSE value less than 10% of the maximum known true RUL was considered acceptable (Zhang et al. 2018, Wu et al. 2018). Applying this threshold to the testing of the generated datasets and running the associated training and testing data by component failure mode into the LSTM data-driven model, output RMSE values were collected and can be seen in Table 1, although only single failure mode data testing results are included to compensate for spacing.

Based on these tests, RUL prediction results scored acceptable RMSE values of under 10% of the maximum known test RUL for each of the datasets, except for flight class 1 and 2 fan datasets (Note: double failure mode datasets also scored similarly, with only LPC/fan datasets RMSE values above 10% of the acceptable threshold). This can be attributed to the fact that the fan component degrades the slowest compared to the other four turbofan components, and therefore it is common for data to not include significant signs of abnormal degradation in engines that do not see extensive utilization. A proposed solution to the issue of Fan component RUL having too high of a prediction would be a categorical labeling of the component which designates if abnormal degradation is recognized in the data or not.

Further testing of the run-to-failure datasets with the data-driven model serves to verify the accuracy of the LSTM fault classification program. Figure 2 shows a sample output confusion matrix of the classification program, demonstrating that single failure mode classification tests result in accuracy greater than 90% for each turbofan component. Classification tests on double failure modes have proven to correctly identify the predominant (first to fail) component failure mode as the single failure mode classification tests, but do not identify the secondary failure mode. This behavior of the classification program requires real-world implementation of the predictive maintenance system to process engine sensor data frequently, so as to successfully keep fault classifications relevant. This resulted in the need for aircraft sensor data to be collected at frequent intervals such as every post-flight check or every 24 hours.

Table 1. RMSE of Predicted RUL Values for Single Failure Mode Datasets

RMSE of Predicted RUL for Single Failure Mode Data			
	Flight Class 1	**Flight Class 2**	**Flight Class 3**
HPC	14.62	11.16	8.64
HPT	11.17	9.66	8.35
LPC	18.27	11.77	14.69
LPT	12.92	11.16	8.13
Fan	41.67	20.63	17.26

```
[[ 99   0   1   0   0]
 [  0 100   0   0   0]
 [  8   0  92   0   0]
 [  0   0   1  99   0]
 [  0   0   0   0 100]]


Fault Test      Fan %    LPT %    LPC %    HPC %    HPT %
------------  -------  -------  -------  -------  -------
Fan                99        0        1        0        0
LPT                 0      100        0        0        0
LPC                 8        0       92        0        0
HPC                 0        0        1       99        0
HPT                 0        0        0        0      100
```

Figure 2. Sample Confusion Matrix Results for LSTM Fault Classification

4.3 Maintenance and Deployment Planning Optimization Testing

Validation testing of the predictive maintenance system's Python Gurobi optimization program tests that varying parameters interface correctly throughout the program and that resulting availability is acceptable across different time frames. For testing, simulations with varying input parameters, as listed in Methods, were run and maintenance actions were observed to follow accordingly. This includes simulations where available maintenance technicians supported at most two, three, and four maintenance actions at a time, and simulations involving one and two available maintenance stations for conducting repairs.

For testing of availability given different time frames, a Monte Carlo experiment was conducted in which 1000, 72-hour periods were simulated with a fleet of 5 aircraft similar to the Performance Gap Analysis. The resulting simulations returned availability levels above 80% for each simulation, thus demonstrating the successful performance of the predictive maintenance planning. However, given that these simulation experiments are conducted with small fleets of aircraft, further testing is required to examine how applicable these results are with large-scale fleet operations. The number one obstacle in tests involving larger fleets is the processing speed of the optimization solver and rerunning the optimization in the event of simulated random failures. Given classification results, processing of real-world aircraft sensor data would need to be frequent, and therefore would require processing speeds that would not impact the predictive systems ability to provide timely maintenance and deployment planning.

5. Conclusion

The overall goal of the predictive maintenance scheduling optimization model is to reduce aircraft downtime, decrease maintenance costs, and improve fleet availability. With the completion and testing of the proposed predictive maintenance scheduling process outlined within this paper, the successful deployment of this product would likely save millions of dollars a year for military flight command while also greatly increasing the availability of a given fleet, thus meeting the goals defined at the start of this project. These changes will allow for the better allocation of resources while also enabling stronger responses to changing threats as more aircraft will be available to respond. At the moment, all predicted results are based on the simulated data provided from C-MAPSS and simulated mission scheduling systems. To verify that the model will function properly in a real-world setting, more testing will need to be done with sensor readings of physical aircraft engines and a trial run must be done using a small group of planes in a low-risk environment. With the additional verification and validation of the predictive maintenance method in a real-world environment, the product will ideally be implemented into larger fleets to better improve the readiness of all aircraft within the military. Additionally, the work conducted here for aircraft engines will likely be tested on other mechanical systems that allow for sensors to determine when they will need maintenance and repairs as well. There is a strong possibility that these methods can be applied beyond only aircraft and can be used to improve maintenance processes for a wide variety of mechanical systems.

6. References

Arias Chao, 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–. https://doi.org/10.3390/data6010005

Chan, Litt, J. S., Frederick, D. K., DeCastro, J. A., & Liu, Y. (2012). *User's Guide for the Commercial Modular Aero-Propulsion System Simulation (C-MAPSS): Version 2*.

Congressional Budget Office. (2022, January 5). *Availability and use of aircraft in the Air Force and Navy*. Congressional Budget Office. Retrieved February 19, 2022, from https://www.cbo.gov/publication/57713

Saxena, Goebel, K., Simon, D., & Eklund, N. (2008). Damage propagation modeling for aircraft engine run-to-failure simulation. *2008 International Conference on Prognostics and Health Management*, 1–9. https://doi.org/10.1109/PHM.2008.4711414

Wu, Yuan, M., Dong, S., Lin, L., & Liu, Y. (2018). Remaining useful life estimation of engineered systems using vanilla LSTM neural networks. *Neurocomputing (Amsterdam)*, *275*, 167–179. https://doi.org/10.1016/j.neucom.2017.05.063

Zhang, Wang, P., Yan, R., & Gao, R. X. (2018). Long short-term memory for machine remaining life prediction. *Journal of Manufacturing Systems*, *48*, 78–86. https://doi.org/10.1016/j.jmsy.2018.05.011

Zhengyang Fan, KC Chang, Ran Ji, and Genshe Chen, (2021) "Flight and Maintenance Planning for Military Aircraft Fleet: A Data-Driven Approach,".