Proactive Aircraft Engine Removal Planning with Dynamic Bayesian Networks

Bharath Pidaparthi[†], Ryan Jacobs, Sayan Ghosh, Sandipp Krishnan Ravi, Ahmad W Amer, Lele Luan, R Murali Krishnan, Feng Zhang, Victor Perez, Liping Wang

General Electric Aerospace Research
1 Research Circle, Niskayuna, New York, 12309, USA

ABSTRACT

Aircraft engine removal for maintenance is an expensive ordeal, and planning for it while balancing fleet stability objectives is a complex multi-faceted challenge. This is further compounded by uncertainties associated with usage or just-in-time maintenance approaches that are becoming prevalent. Engine removal decisions rely on accurate estimation of damage growth or remaining useful life of critical components and a framework for aggregating these component-level estimates (and their uncertainties) into an engine-level removal forecasting model. An approach to this planning challenge is to develop probabilistic prognostic digital twins tailored to engine-specific operations and calibrate/update them with inspection data from the field. To this end, this work outlines a framework involving: 1) building component-level probabilistic models capable of forecasting damage growth or remaining useful life, 2) aggregating the outputs of these component-level models into a system-level view using a Dynamic Bayesian Network (DBN), and 3) updating the states of the DBN with inspection information as and when they become available.

Keywords: Prognostics, Probabilistic Digital Twins, Dynamic Bayesian Networks, Cumulative Damage Modeling

1. INTRODUCTION:

Aircraft engines are highly sophisticated systems, comprising a multitude of interdependent components, each vulnerable to various damage and failure modes. This inherent complexity makes accurate prognostics challenging. To mitigate these risks and ensure the highest standards of operational safety, regulatory agencies impose stringent requirements for periodic inspections and maintenance. These regulatory standards encompass a wide spectrum of checks and procedures, from routine Bore-scope Inspections (BSI) to comprehensive engine overhauls. Often, these procedures necessitate the removal of engines from the aircraft, leading to significant downtime and potential revenue loss.

The financial and logistical impacts of unexpected maintenance events can be especially severe. These events not only result in significant costs but also create major challenges in re-booking passengers and managing schedules, particularly when spare engines or aircraft are not immediately available. Additionally, unplanned maintenance is frequently hindered by long lead times for replacement parts and limited availability of certified Maintenance, Repair, and Overhaul (MRO) shop slots. These multifaceted challenges highlight the critical importance of efficient engine removal planning, which is essential to strike a balance between maintaining operational safety and ensuring engine availability.

Fleet-level engine removal statistics/models, based on historical data, are inadequate for making engine-specific decisions. These models, by nature of their construction, fail to capture differences in damage states from one engine to another – attributed to individual engine-level variations in manufacturing, material properties, operational profiles, and load factors. Building a prognostic digital twin, therefore, custom-tuned/calibrated for each individual engine is essential for effective removal decision-making. These digital twins, as outlined in (Li, Mahadevan, Ling, Wang, & Choze, 2017; Li, Mahadevan, Ling, Choze, & Wang, 2017), must be formulated to accomplish the following:

- Incorporate various sources of uncertainty from hardware manufacturing variations to loading/operational differences.
- Integrate heterogeneous information – including operational data, laboratory data, physics-based/empirical models, expert opinions, and more.
- Capable of updating the uncertainty in model parameters (to reduce discrepancies between the digital twin and the

[†]Lead & corresponding Author: bharath.pidaparthi@ge.com

actual hardware) and tracking the time-dependent system states using measurement data, i.e., diagnosis.

- Predict/forecast the progression of damage states in the absence of available data, i.e., prognosis.

In this work, Dynamic Bayesian Networks (DBNs) are chosen to construct such digital twins, an idea previously explored in the context of aircraft structural health monitoring (Li, Mahadevan, Ling, Choze, & Wang, 2017; Li, Mahadevan, Ling, Wang, & Choze, 2017). DBNs have shown their versatility in handling various data types and can incorporate existing domain knowledge, all while providing a method for reasoning under uncertainty. These attributes of DBNs make them an ideal modeling approach to meet the above listed requirements for constructing prognostic digital twins.

This paper aims to present a broad overview of the structure necessary for developing such probabilistic prognostic digital twins using DBNs, without going into the intricacies of each element in the framework. For a more comprehensive review of some of the modeling elements mentioned in this framework, readers are referred to (Thelen et al., 2022, 2023; Bhaduri et al., 2024; Luan, Jacobs, Ghosh, & Wang, 2023; Ghosh et al., 2020). The remainder of this paper is organized as follows: The subsequent section offers a brief introduction to Dynamic Bayesian Networks (DBNs) from the perspective of systems structure. The following section delves into component-level damage modeling, with a particular focus on a selection of available modeling techniques for constructing these damage models. Thereafter, inference within DBNs is discussed, accompanied by an explanation of the state *update* concept. Finally, a demonstrative example of a DBN model for engine removal planning is illustrated.

2. Dynamic Bayesian Networks (DBN)

Aircraft engines have multiple interacting components, each contributing to the overall system performance. The failure behavior (i.e., the damage evolution) of each component can be described through variables, states, and the structural relations that describe the conditional dependencies between these entities. The presence of multiple components or failure modes, each with its own set of variables and states affecting the system's functionality, can be difficult to model. Moreover, the limited predictability of the damage models, which will be discussed in the subsequent section, makes it difficult to accurately forecast the system's state at any given moment. This uncertainty can lead to sub-optimal maintenance decisions, such as removing an engine from service too early or too late, resulting in increased maintenance costs, reduced operational efficiency, and increased risk of failure.

Probabilistic graphical models, or Bayesian networks (BNs), are a powerful tool for modeling complex systems. They can represent the relationships between variables within a system and capture beliefs about the sources of uncertainties inherent to the models. In Bayesian networks, system behavior is depicted as nodes in a graph, with edges representing the relationships between these nodes. The nodes denote the system's variables or states, while the edges indicate the conditional dependencies between them. Two classes of conditional dependencies are common in these system models,

$$\begin{aligned} X_{t+\delta t} &= f(X_t, \Omega_t), \\ Y_t &= g(Z_t, \Phi_t), \end{aligned} \tag{1}$$

where X_t are states of the system, Y_t and Z_t are variables that capture local conditions at time t, Ω_t and Φ_t are external factors influencing system behavior, and $f(\cdot)$ and $g(\cdot)$ are the functional form of the conditional dependencies between the system variables. Specifically, $g(\cdot)$ represents structural relations between random variables of a behavioral model at time t, and $f(\cdot)$ represents the Markovian dynamics that describe the evolution of system states from t to $t + \delta t$. Functions that govern the behavior of state variables are conditionally dependent on the value of the system states at the previous time step. Since these state-governing functions (also called transition functions) describe the time-based evolution of system behavior, the probabilistic graphical models used to model such behavior are also called dynamic Bayesian networks (DBNs). DBNs, as such, can be seen as a sequence of static BNs (slices) taken in chronological order with edges connecting nodes across adjacent time-slices (representing temporal dependencies).

This study utilizes GE-DBN, a Dynamic Bayesian Network tool developed by GE Aerospace Research as part of the U.S. Air Force's P2IAT program. Over the recent years, this versatile DBN tool has been refined to address engine prognostics and removal planning needs. The inputs and formalism required for engine prognostics have been structured into modules within the DBN framework as illustrated in the accompanying Figure 1.

3. Component damage modeling:

Forecasting when a component would deteriorate (due to faults, fatigue, or wear) to a threshold level deemed unsafe for its intended use is crucial for prognostics. The moment when the component reaches this failure threshold is known as the *end of useful life* (EOL), and the duration left until this EOL is reached is referred to as the *remaining useful life* (RUL). Both of these measures are characteristics of the damage progression/growth, and hence, constructing models capable of forecasting damage growth in components is fundamental for developing any prognostic tool.

Progression of damage in mechanical components is an accumulative process. Models designed to capture this irreversible build-up of damage through discrete time increments are known as cumulative damage models (CDMs). These models represent the damage at time t as (VanStone, Gooden, &

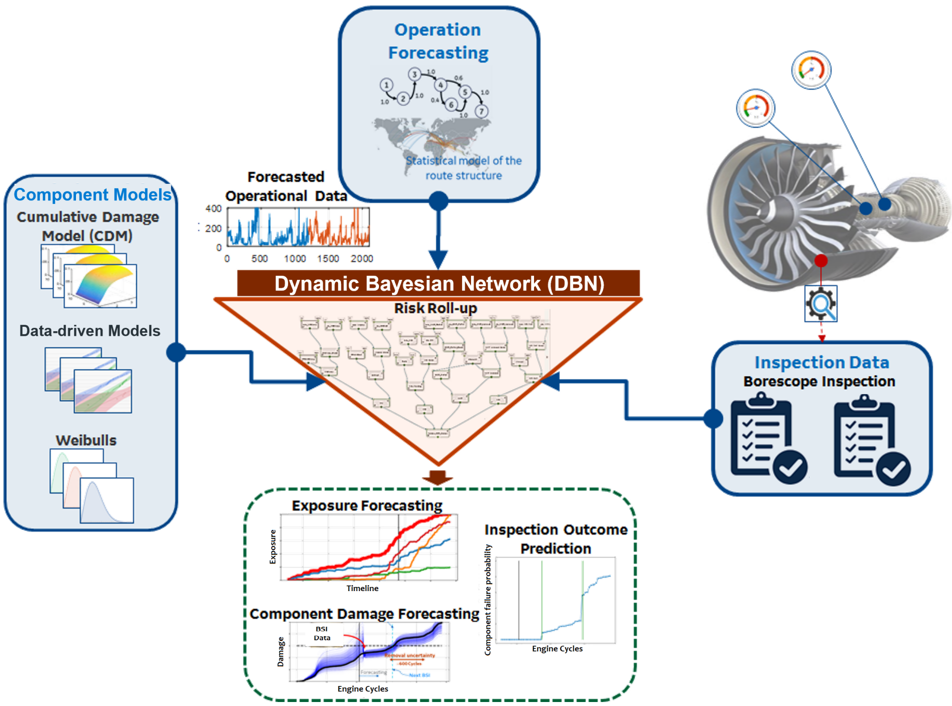

Figure 1. Engine prognostic digital twin framework for removal planning. Operation forecasting module: This module employs statistical models, which are based on transitional probability matrices, to predict flight route structures from historical behavior. These predicted routes are subsequently utilized to estimate engine sensor parameters, either through bootstrapping or via calibrated engine performance models. Component damage modeling module: refer to Sec 3. Inspection data ingestion module: refer to Sec 4

Krueger, 1988; Nascimento & Viana, 2020; Saxena, Goebel, Simon, & Eklund, 2008):

$$\begin{aligned} D_t &= D_{t-1} + \Delta D_t \\ \Delta D_t &= \phi(D_{t-1}, X_t) \end{aligned} \tag{2}$$

where D_{t-1} is the damage level at time $t-1$, and ΔD_t is the damage increment at time t, which is a function of both D_{t-1} and inputs X_t. The inputs X_t usually express time-dependent loading and boundary conditions (e.g., pressures, temperatures, torques, mechanical and thermal stresses, etc.) that the component experiences.

Traditional methods for building these damage models have been deterministic, implying that the components operate under the assumption of per-defined design service conditions with fixed material strength. These models, in practice, are paired with safety factors, derived from experience, to forecast the component's life/durability. Such models, however, fail to account for variations in manufacturing conditions and operating profiles - leading to discrepancies between the predicted and observed life of the components. Therefore, this work proposes construction of probabilistic models capable of capturing these variations through random parameters (with either predefined or calibrated distributions). Probabilistic outputs from such models can not only facilitate effective risk-informed decision-making but can also be seamlessly integrated as random variable nodes within the DBN.

The subsequent subsections provide a concise discussion of two popular modeling strategies used for developing these component-level damage progression models. These models, once trained on engine operational (i.e., sensor) data and observed characteristics of component failures like crack length or area of coating loss, should be capable of predicting component damage progression as a function of engine operational data.

3.1. Data-driven modeling:

Data-driven methodologies have become indispensable for accelerating various modern engineering workflows, particularly in design analysis, optimization and component/system reliability analysis (Pidaparthi, Li, & Missoum, 2022; Ravi, Dong, & Wei, 2022; Ghosh et al., 2020; Pidaparthi & Missoum, 2023; Pidaparthi, Missoum, Xu, & Li, 2023; Ravi et al., 2023; Pidaparthi & Missoum, 2019). The flexibility of these methods are making them an attractive choice for rapidly constructing (low-fidelity) models capable of predicting component-level damage. These models, using engine operational data as input, not only forecast failures but also provide corresponding damage curves when trained on sparse and noisy inspection observations. The set of suitable models for this task encompasses linear regression, Gaussian process regression, probabilistic deep neural networks, random forests, and other advanced techniques. Often, these models are developed with inputs derived from statistical transformations of engine operational data accumulated over operational history to yield outputs that correspond to observed damage.

In recent times, developments in time series modeling have facilitated their application to component damage modeling (Zhao, Wang, Yan, & Mao, 2016; Amer & Kopsaftopoulos, 2019; Wu, Wu, Chen, Li, & Yan, 2021; Choe, Kim, & Kim, 2021). Time series models, such as recurrent neural networks (RNNs) and long short-term memory (LSTM) networks, can be trained using inspection observations from a fleet of engines. A key distinction from previous models is the direct ingestion of time-domain operational data at each time slice. This approach allows for richer information in the modeling framework, resulting in accurate models, albeit with potential challenges related to over-parameterization and overfitting. The trained model can then be deployed to predict fight-by-flight damage evolution.

The predictive capabilities of these models can also be utilized within Explainable AI frameworks to understand the reasons/ mechanisms behind failures. This is particularly important when dealing with black-box models – thorough investigations are essential before making downstream decisions. These investigations can occur at the fleet or engine level through global and local sensitivity analysis. Global sensitivity analysis (such as Sobol Analysis) allows independent evaluation of the overall model intuition. At a more local level, sensitivity analysis (using techniques like SHAP or LIME) on input features provides insights for individual predictions, aiding investigative prognosis regarding failure causes or high damage.

3.2. Hybrid Physics Modeling:

While data-driven approaches are widely used in damage modeling, they face challenges in forecasting damage over extended time horizons. This issue necessitates explicit incorporation of physical principles or constraints into the damage models to maintain realistic trends in damage progression, such as monotonicity. One approach to this is to develop and calibrate analytical physics-based cumulative damage models (CDM). These CDMs are specific to the damage mode (such as oxidation, crack growth, fatigue) and can be probabilistically calibrated using Markov Chain Monte Carlo sampling or the Kennedy O'Hagan formulation (Kennedy & O'Hagan, 2001). Since the physics of the failure mode is integrated into the structure of these models, it is anticipated that their forecasting performance will exceed that of purely data-driven models.

An alternative to expertly crafted CDMs is the development of CDMs via sparse regression algorithms such as compressed sensing, variational relevance vector machines (Zhang, Jacobs, Ghosh, Kulkarni, & Wang, 2022; Tsilifis et al., 2023; Tipping, 2001). Though the fundamental mechanism remains regressions, the cumulative nature of these models to predict ΔD_i (as defined in Eqn 2) ensures monotonicity of the damage evolution, and inherent sparsity in the equation guarantees generalizability. These features incorporate soft forecast-ability to these models. In this context, GE Aerospace Research has developed a hybrid physics modeling tool known as the Physics-Informed Research Assistant and Theory Extractor (PIRATE). PIRATE is designed to discover physical laws in the form of human-readable equations from observed data (Luan et al., 2023). PIRATE offers two data-driven methods for equation discovery: compressed sensing (Zhang et al., 2022) and symbolic regression (Luan et al., 2023). In compressed sensing, an equation is constructed by defining a class of basis functions, and the coefficients of the linear combination of these basis terms are automatically computed via sparse regression. The symbolic regression method, on the other hand, is built on the foundation of genetic programming. It distills the solution by evolving the equation individuals created from basic mathematical operators, variables, and ephemeral constants. Beyond fitting to the observed data, PIRATE also allows the discovered equation to be constrained by prior knowledge. Table 1 lists examples of the constraints that can be enforced during this equation discovery process.

Table 1. Examples of physics-motivated constraints for PIRATE (Luan et al., 2024)

Constraint	Mathematical formulation
Output bound	$f(\mathbf{x}) \leq 0$ or $f(\mathbf{x}) \geq 0$
Monotonic non-increasing	$\frac{\partial f(\mathbf{x})}{\partial x_i} \leq 0$
Monotonic non-decreasing	$\frac{\partial f(\mathbf{x})}{\partial x_i} \geq 0$
Concavity	$\frac{\partial^2 f(\mathbf{x})}{\partial^2 x_i} \leq 0$
Convexity	$\frac{\partial^2 f(\mathbf{x})}{\partial^2 x_i} \geq 0$
Even symmetry	$f(x_i) = f(-x_i)$
Odd symmetry	$f(x_i) = -f(x_i)$

4. INFERENCE & STATE UPDATING:

DBNs can be paired with a variety of inference algorithms - from exact approaches like Forward-Backward and Island Algorithm to approximate algorithms such as Particle Filtering (PF) and Kalman Filters. The choice of the algorithm depends on the DBN structure, variable types (discrete and/or continuous), linearity assumptions, and the desired accuracy-complexity trade-off. Exact methods are computationally expensive but provide precise results, while approximate methods are more efficient and flexible but may introduce approximation errors. This work, GE-DBN in particular, employs Sampling Importance Resampling (SIR) implementation of PF as the inference algorithm owing to its flexibility to handle: both discrete and continuous variables, and linear/non-linear functional relationships in the network.

When an inspection exposes the damage/distress state of a component/sub-system represented by a node in the Bayesian network, the inspection data is used to *update* the network through Bayesian inference. This inference process reduces the uncertainty associated with the hidden state variables and calibration parameters in the network, allowing the network to better match the observed evidence. This *update* mechanism improves the network (and the prognostics digital twin it represents) over time with more field experience/inspections. Note that, in practice however, the observations may become available asynchronously (i.e., inspections of different components in the system could occur at different timestamps). Hence, the minimum time-step required for evolving the system dynamics must be carefully chosen.

While *updating*, the current parameters and system state estimates are adjusted to match the data using Bayesian inference. In PF, each particle (from the priors) is assigned a weight based on the comparison of the predicted and true observation (i.e., the likelihood). The ensemble of these weighted samples forms the updated (i.e., posterior) distribution. An issue often encountered in filtering approaches is that the weight disparity can lead to weight collapse. This can be mitigated by including a resampling step before the weights become too uneven. GE-DBN uses a SIR approach to this end where a new pool of equally weighted particles is generated to match the updated distribution. These distributions are then used to continue the dynamic simulation, or optionally, restart the entire simulation for a more accurate system state history. Details of this implementation can be found in (Bartram & Mahadevan, 2013; Asher, Ling, & Wang, 2018).

5. Demonstrative example:

In this section, a simplified, yet representative, engine prognostics model is showcased to demonstrate the DBN-based digital twin framework for engine removal planning. The datasets/models used in this example are from actual engines but are sanitized/re-scaled to eliminate any descriptors.

The problem considers three distinct failure modes, each associated with a different engine component (A, B, C). These failure modes are assumed to be independent, a reasonable assumption in many cases. For example, the growth of cracks on a High-Pressure Turbine (HPT) blade is typically affected by factors like the blade's temperature conditions and material properties. Conversely, coking of fuel nozzles, which is a distinct damage mode, is influenced by factors specific to the fuel nozzle, such as the quality and flow rates of the fuel. These two modes exert minimal influence on each other and can be considered to evolve independently. The decision to remove an engine is critically influenced by the condition of these components. In specific, the engine must be removed if any one of the components — A, B, or C exceeds its permissible damage tolerance.

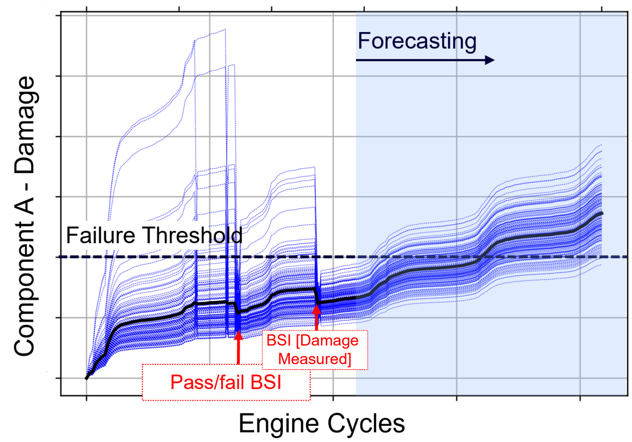

Figure 2. Probabilistic forecast of damage progression for component-A: Each blue progression curve here is a prediction sample from component-A damage model, and the spread of these curves signifies the uncertainty in its prediction. The dashed horizontal line marks the failure threshold, and the ratio of sample predictions that cross this threshold at any given cycle (i.e., flight index) provides an estimate of the component's failure probability. Note that this figure illustrates the prediction for a particular component from a single engine [i.e., obtained by feeding the sensor data of an engine into the damage model of component-A].

The construction of the prognostic digital twin begins with the development of individual damage progression models (of the form Eqn 2) for all three components. These models are trained/calibrated on datasets of sensor data and damage measurements (from a fleet of fielded engines) to predict flight-by-flight damage evolution of the components (given engine sensor data as inputs). Serviceability limits for these components are represented by defining failure thresholds on these damage progression curves (i.e., when the damage level crosses the threshold, the component is deemed to have failed or reached its serviceability limit). Since these component-level models are probabilistic, their outputs can be sampled and compared against the thresholds to estimate their probability of failure as shown in Figure 2.

Once the component-level damage models are developed, they are structured as nodes within the DBN with failure/safe labels becoming their output states determined through thresholds (see Figure 3). The stochastic parameters $\boldsymbol{\theta}$ associated with these damage models are also represented as nodes in the network (with pre-calibrated distributions). When an inspection is performed to observe the damage condition of a component, these findings can be used to *update* the network as discussed in Sec 4. This accomplishes two things: it shifts the prediction from the damage model to match the inspection outcome, and it reduces the uncertainty in the model prediction. This update mechanism customizes/re-calibrates

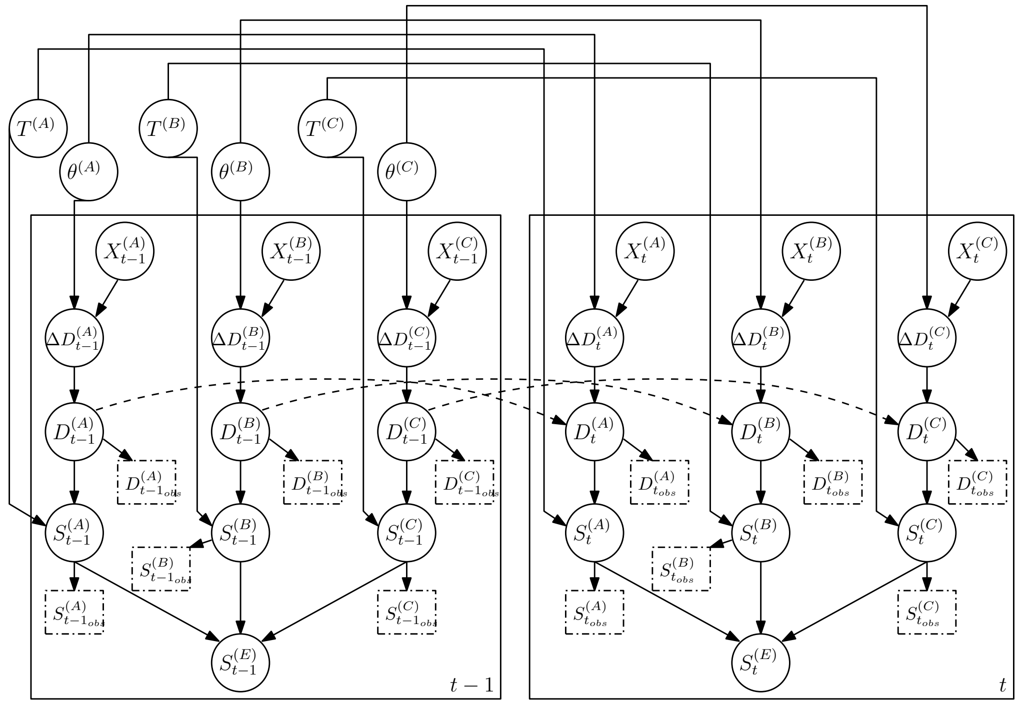

Figure 3. DBN for aggregating component-level damage models to estimate engine failure probability. Here, the network is visualized for two time slices. Nodes are marked with a subscript to denote the time index and a superscript to identify the component they are associated with. Nodes encased in dashed boxes are observed data, and may only be available at specific time indices (from inspections). Additionally, it is not always necessary that the inspection data be available for all components at the same time. The nodes marked S represent the Fail/Safe state of the components by comparing the component damage prediction D with failure-threshold T.

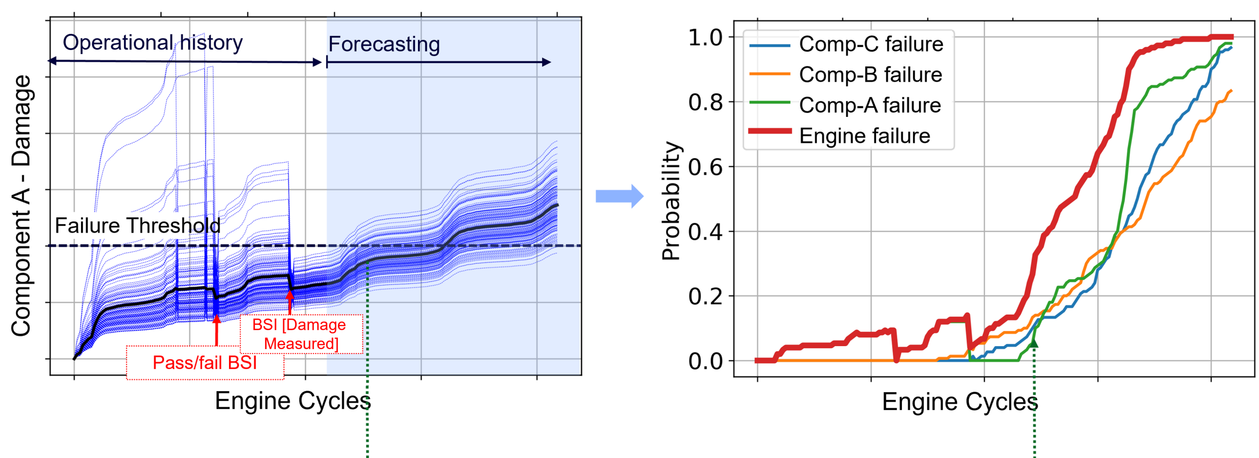

Figure 4. Engine Failure Risk Estimation. This illustration demonstrates aggregation of failure probabilities from three component-level damage models using DBN. The left figure exclusively shows the damage development of component-A. However, analogous curves for the other two component models are also computed here and are shown in the right figure. The bold red curve on the right figure represents the estimated engine failure probability, derived from aggregating all three component failure probability curves.

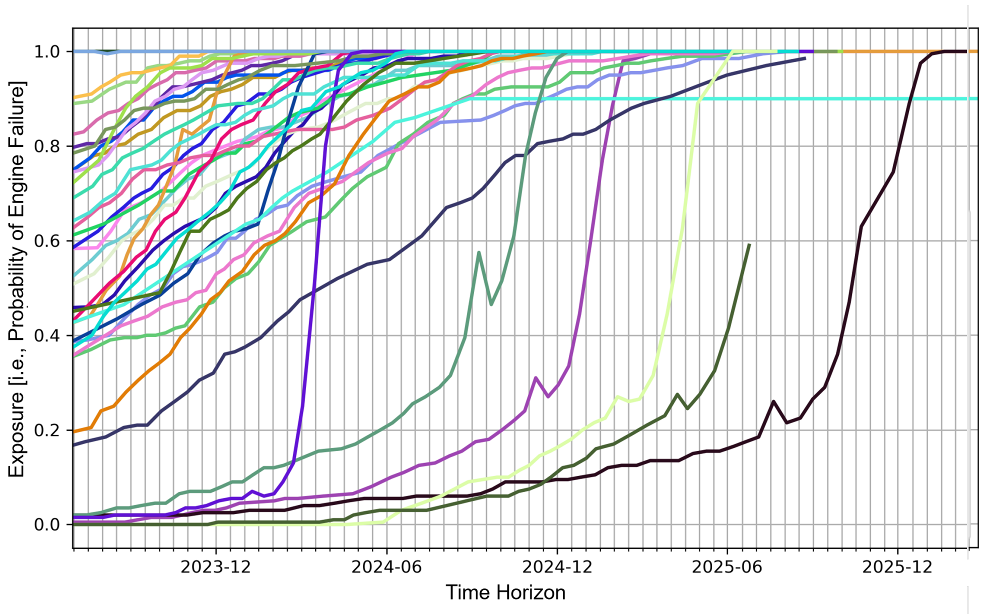

Figure 5. Engine Exposure Across the Fleet: Each curve in this figure corresponds to the failure probability for a distinct engine serial number. The higher the probability, the greater the risk of failure.

the component damage model (pre-trained on a dataset with fleet-wide observations) to a specific part serial number. Inspections generally can yield two types of outcomes: a binary pass/fail result or a quantities measurement of the failure characteristics, such as the crank length. Both of these outcome types can be incorporated into the update mechanism as shown in Figure 2. This can be done either by conditioning on the node that represents the damage, if the damage value is accessible, or by directly conditioning on the node that represents the component's failure state.

The failure probabilities derived from the component-level damage models can be aggregated into a system-level failure estimate. This aggregation is achieved by defining edges, or conditional dependencies, between the nodes representing the states of the individual components and the nodes representing the overall state of the engine, as illustrated in Figure 3. The problem assumes that engine failure occurs if any of the components fail. Consequently, the Dynamic Bayesian Network (DBN) aggregation results in engine failure probabilities, as depicted in Figure 4. This figure shows the failure probability curves for a single engine over time.

This methodology can be extended to a fleet of engines, as demonstrated in Figure 5. Each curve in this figure represents the failure probability, or exposure, for a specific engine serial number over various time horizons. This fleet-wide perspective allows for the ranking of engines based on their exposure at any given date. By identifying engines with high exposure before they reach critical failure probabilities, decisions can be made to withdraw these engines from operation preemptively.

The fleet-wide exposure view enables maintenance planners to align engine withdrawals with the availability of MRO slots. This alignment ensures that engines are removed from service in a timely manner, minimizing operational disruptions and optimizing the use of MRO resources. Additionally, this approach allows for proactive management of the fleet, ensuring that engines with the highest risk of failure are prioritized for maintenance, thereby enhancing overall fleet reliability and safety.

Airline operators can leverage fleet-wide exposure charts in various ways to align with their organizational goals. In addition to scheduling preventive maintenance activities by servicing engines with higher exposure, understanding fleet maturity with the exposure charts allows operators to predict the demand for spare engines, ensuring availability of an adequate number of spares to minimize aircraft downtime. Analyzing these exposure plots also helps in forecasting maintenance costs and allocating budgets effectively, ensuring funds are available when needed. Additionally, exposure plots can inform strategic decisions on fleet expansion by identifying trends in engine operational availability. Finally, optimal engine maintenance enhances fuel efficiency and reduces emissions, supporting environmental regulations and sustainability goals.

6. Conclusions:

This work proposes Dynamic Bayesian Networks (DBN) to construct probabilistic prognostics models for aircraft engine removal planning. The graph-like structure of DBNs has been demonstrated to be particularly effective for modeling failures at the system level amidst uncertainties. Moreover, the study introduces the notion of *updating*, which enables the network to be iteratively enhanced over time to accurately reflect the actual state of the hardware system damage, using asynchronously available inspection data. The practicality of this framework is demonstrated through a representative digital twin model for engine prognostics.

REFERENCES

Amer, A., & Kopsaftopoulos, F. (2019). Probabilistic damage quantification via the integration of non-parametric time-series and gaussian process regression models. *Structural Health Monitoring 2019*.

Asher, I., Ling, Y., & Wang, L. (2018). Improving sir with constrained resampling for dynamic bayesian network applications. In *2018 aiaa non-deterministic approaches conference* (p. 1406).

Bartram, G., & Mahadevan, S. (2013). Dynamic bayesian networks for prognosis. In *Annual conference of the phm society* (Vol. 5).

Bhaduri, A., Ravi, S. K., Pidaparthi, B. K., Jacobs, R., Pandita, P., Zhang, F., ... others (2024). A probabilistic framework for uncertainty quantification in the presence of aleatory and epistemic inputs: Application to sailplane payload delivery. In *Aiaa scitech 2024 forum* (p. 0943).

Choe, D.-E., Kim, H.-C., & Kim, M.-H. (2021). Sequence-based modeling of deep learning with lstm and gru networks for structural damage detection of floating offshore wind turbine blades. *Renewable Energy*, *174*, 218–235.

Ghosh, S., Pandita, P., Atkinson, S., Subber, W., Zhang, Y., Kumar, N. C., ... Wang, L. (2020). Advances in bayesian probabilistic modeling for industrial applications. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering*, *6*(3), 030904.

Kennedy, M. C., & O'Hagan, A. (2001). Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *63*(3), 425–464.

Li, C., Mahadevan, S., Ling, Y., Choze, S., & Wang, L. (2017). Dynamic bayesian network for aircraft wing health monitoring digital twin. *Aiaa Journal*, *55*(3), 930–941.

Li, C., Mahadevan, S., Ling, Y., Wang, L., & Choze, S. (2017). A dynamic bayesian network approach for digital twin. In *19th aiaa non-deterministic approaches conference* (p. 1566).

Luan, L., Jacobs, R., Ghosh, S., & Wang, L. (2023). Physics discovery of engineering applications with constrained optimization and genetic programming. In *Turbo expo: Power for land, sea, and air* (Vol. 87066, p. V11BT25A005).

Luan, L., Jacobs, R., Ghosh, S., & Wang, L. (2024). Physics-informed research assistant for theory extraction (pirate) for missing physics discovery. In *Aiaa scitech 2024 forum* (p. 0171).

Nascimento, R. G., & Viana, F. A. (2020). Cumulative damage modeling with recurrent neural networks. *AIAA Journal*, *58*(12), 5459–5471.

Pidaparthi, B., Li, P., & Missoum, S. (2022). Entropy-based optimization for heat transfer enhancement in tubes with helical fins. *Journal of Heat Transfer*, *144*(1), 012001.

Pidaparthi, B., & Missoum, S. (2019). Stochastic optimization of nonlinear energy sinks for the mitigation of limit cycle oscillations. *AIAA journal*, *57*(5), 2134–2144.

Pidaparthi, B., & Missoum, S. (2023). A multi-fidelity approach for reliability assessment based on the probability of classification inconsistency. *Journal of Computing and Information Science in Engineering*, *23*(1), 011008.

Pidaparthi, B., Missoum, S., Xu, B., & Li, P. (2023). Helical fins for concentrated solar receivers: Design optimization and entropy analysis. *Journal of Energy Resources Technology*, *145*(12), 121706.

Ravi, S. K., Dong, P., & Wei, Z. (2022). Data-driven modeling of multiaxial fatigue in frequency domain. *Marine Structures*, *84*, 103201.

Ravi, S. K., Pandita, P., Ghosh, S., Bhaduri, A., Andreoli, V., & Wang, L. (2023). Probabilistic transfer learning through ensemble probabilistic deep neural network. In *Aiaa scitech 2023 forum* (p. 1479).

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).

Thelen, A., Zhang, X., Fink, O., Lu, Y., Ghosh, S., Youn, B. D., ... Hu, Z. (2022). A comprehensive review of digital twin—part 1: modeling and twinning enabling technologies. *Structural and Multidisciplinary Optimization*, *65*(12), 354.

Thelen, A., Zhang, X., Fink, O., Lu, Y., Ghosh, S., Youn, B. D., ... Hu, Z. (2023). A comprehensive review of digital twin—part 2: roles of uncertainty quantification and optimization, a battery digital twin, and perspectives. *Structural and multidisciplinary optimization*, *66*(1), 1.

Tipping, M. E. (2001). Sparse bayesian learning and the relevance vector machine. *Journal of machine learning research*, *1*(Jun), 211–244.

Tsilifis, P., Ravi, S. K., Zhang, F., Pandita, P., Bhaduri, A., Kulkarni, A., & Wang, L. (2023). Towards sparse and interpretable classification for damage assessment of jet engine components. In *Turbo expo: Power for land, sea, and air* (Vol. 87066, p. V11BT25A002).

VanStone, R., Gooden, O., & Krueger, D. (1988). *Advanced cumulative damage modeling*. Materials Laboratory, Air Force Wright Aeronautical Laboratories, Air Force . . . .

Wu, J.-Y., Wu, M., Chen, Z., Li, X.-L., & Yan, R. (2021). Degradation-aware remaining useful life prediction with lstm autoencoder. *IEEE Transactions on Instrumentation and Measurement*, *70*, 1–10.

Zhang, S., Jacobs, R., Ghosh, S., Kulkarni, A., & Wang, L. (2022). Automated data-driven physics discovery of turbine component damage. In *Turbo expo: Power for land, sea, and air* (Vol. 86076, p. V08BT25A008).

Zhao, R., Wang, J., Yan, R., & Mao, K. (2016). Machine health monitoring with lstm networks. In *2016 10th international conference on sensing technology (icst)* (pp. 1–6).