

Operational Prognostic Model Evaluation

Shashvat Prakash¹, Katarina Vuckovic², Sanket Amin³

^{1,2,3}*Raytheon Technologies - Collins Aerospace, Windsor Locks, CT, 06096, USA*

Shashvat.Prakash@collins.com

Katarina.Vuckovic@collins.com

Sanket.Amin@collins.com

ABSTRACT

Prognostic models have gained favor for the promise of avoiding costly unscheduled operational interruptions. The models collect information and make predictions that ultimately support a decision for proactive interventions. Yet, the overall value proposition of these models can become unclear when the prediction errors compound or the observed rate of degradation changes. Any model output is essentially a degradation estimate and can be converted to a remaining life estimate. This life estimate and its associated confidence interval directly influence the decision. Considering both the cost of action and inaction, model behavior and uncertainty can inform when intervention makes the most operational sense. In this manner, an evaluation construct is proposed that can serve both to measure performance and optimize it. With an effective means of evaluating prognostic models, better benchmarks can be established to communicate model effectiveness and appropriately schedule routine service.

1. INTRODUCTION

Unscheduled or unanticipated component failures present serious challenges to the operator. First, there is the prospect of lost revenue from the mechanism itself. Second, in order to accommodate the first, provisions and spares must be stocked to minimize overall disruptions. Third, labor must be held in reserve to affect the overall mitigation plan, and finally, customers may require compensation for lost service. A fifth point may be made regarding the overall market perception of the operator. Any disruption presents a challenge to reputation, and reputation impacts are not quite quantifiable. The combined cost of these challenges are nonetheless formidable, and so there is an interest in developing strategies to avoid unscheduled failures in the first place, whether by early planned removals, condition-based maintenance, or data-driven prognostic models.

Recent trends in compute availability, data storage, and sensor proliferation have expanded the scope for prognostic models and prognostic health management (PHM). Models become feasible when the degradation modes are observable with the available sensors, the data processing infrastructure exists, and a timely mitigation plan is implementable. A model that is implemented for PHM is essentially a degradation estimate.

There is no single approach to prognostic model development, as they can incorporate varying levels of physics-based modelling, data-driven statistical elements, and prior event data. One class of models will simply indicate when failure is imminent. These are similar to the ‘low oil’ indicator light in an automobile. A higher fidelity model will track degradation over time and estimate the remaining useful life (RUL). The automobile analog would be a ‘miles remaining’ indicator on a fuel tank. The fidelity of the model correlates closely to the amount of operational flexibility it allows the end user along with the level of confidence in the estimate. A low oil light indicator provides almost no indication of the actual issue and therefore becomes more urgent than a fuel gauge which can predict e.g.20 miles remaining range during a trip when assuming a near constant fuel consumption rate.

Beyond the forementioned costs associated with unplanned interruptions, regulations may play a role as well. Regulatory bodies have investigated the airworthiness equivalency of reducing inspection frequency when prognostic health monitoring is available (International Maintenance Review Board Policy Board, 2018). Under review is a strategy to use prognostics to increase inspection intervals and apply airworthiness credits for condition-based maintenance (Le, Ghoshal, and Cuevas, 2011). However, the framework for evaluating these models is not standardized. This deficiency will become more acute as more health monitoring methods seek airworthiness credits. There is therefore a need to formalize methods which best evaluate the performance of prognostic models.

Shashvat Prakash et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

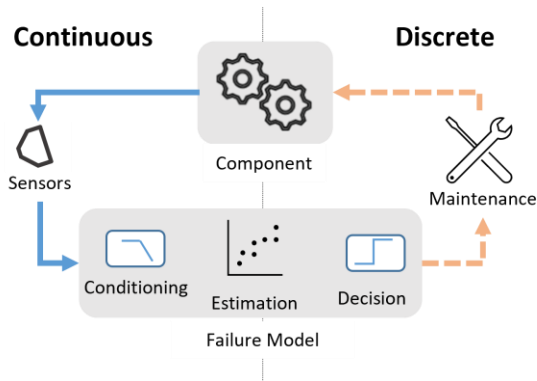


Figure 1. Prognostic analytic diagram. Sensor signals are continuously fed to a failure estimation model, which supports a decision for discrete maintenance.

1.1. Prognostic Models

The role of a prognostic model is to aggregate available time-varying data from the component to ultimately support a proactive intervention decision. The intended result is to provide an operational benefit over allowing an in-service failure event. A depiction of the prognostic model environment is shown in Figure 1. With sensors providing information about the component health, it falls on the system to decipher the data and develop a decision architecture that drives a discrete maintenance. The burden of the model is to fulfill three sub-roles: sensor signal processing (including data fusion), accurate estimation of health condition, and decision support. Relative to the stream of monitoring inputs, the maintenance actions are discrete events. Hence, there is a continuous side of the environment where conditions are constantly being evaluated and a discrete side that involves concrete actions. The model and component bridge the two realms.

Advances in sensors, electronics, and data have enabled an evolution from more traditional reliability-based inspection and maintenance intervals (e.g. Weibull) to some data augmented hybrid. Generally, more information on failure modes and usage profiles can drive a more efficient physics-based maintenance paradigm. However, with the high availability of data and computation, these approaches can incorporate more data-driven heuristics. Further on the spectrum are the purely data-driven approaches, where the outcome is calculated by statistical inference (Luo, Namburu, Pattipati, Qiao, Kawamoto, and Chigusa, 2003, and Fornlof Galar, Syberfeldt, Almgren, Catelani, and Ciani, 2016).

While this allows for decisions outside the realm of human expertise, they are more difficult productionize since the outcomes are less explainable. To overcome the lack of interpretability and transparency of data-driven models, reliable uncertainty quantification (UQ) is required. UQ is critical for decision making in real world scenarios. Yet, many data-driven models with promising RUL estimations provide deterministic solutions that lack the UQ component.

Among these data-driven models, different types of deep learning (DL) models have shown the most promising results in RUL estimation. Most common DL architectures for RUL estimation include Convolutional Neural Networks (CNNs) (Zhu, Chen, Peng, 2018), (Li, ding, and Sun, 2018), (Ren, Sun, and Wang 2018), Recurrent Neural Networks (RNNs) (Zhang, Xiong, Hem and Pecht, 2018), (Song, Li, Peng, and Liu, 2018), (Deng, Zhang, Cheng, Zheng, Jiang, Liu, and Peng, 2019), Transformers (Ding and Jia 2021), Auto-Encoders (AE) (Ren, Sun, Cui, and Zhang, 2018) and Deep Belief Networks (DBN) (Zhang, Lim, Qin, and Tan, 2017).

Attempting to attribute UQ to DL models is an active area of research. Bayesian modeling an effective way to introduce uncertainty to a DL model by treating the previously deterministic weights and random variables. In recent years,

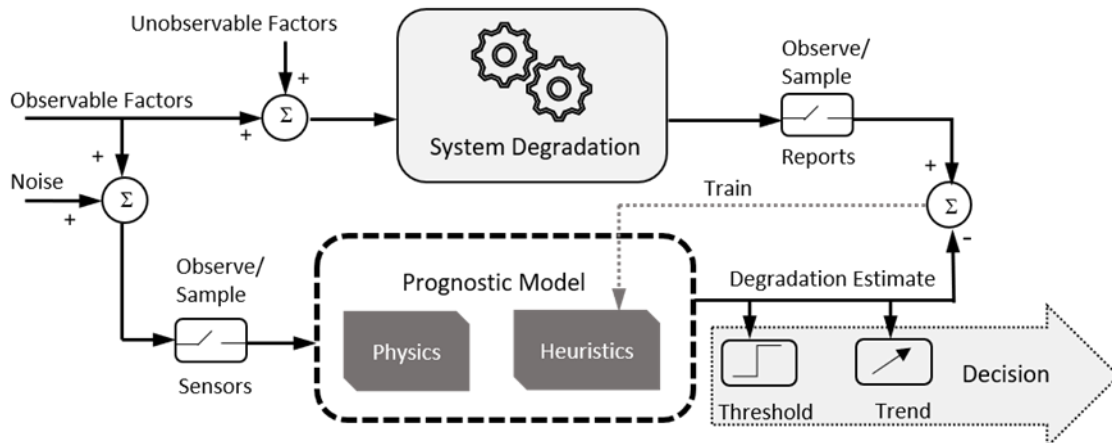


Figure 2. Prognostic model environment and structure with physics and heuristic components. Factors driving degradation are not all observable, and degradation is apparent only when observed at failure. The model must align the available inputs to the degradation trend with some connection to physical fundamentals.

various Bayesian DL (BDL) frameworks have been proposed for addressing the uncertainty of RUL estimations including BDL framework that estimates both the uncertainty of the model (epistemic uncertainty) and uncertainty due to random disturbance (aleatoric uncertainty) (Li, Yang, Lee, Wang, and Rong, 2021), and Bayesian CNNs (BCNNs) (Mazaev, Crevecoeur, and Hoecke, 2021). Additionally, data-driven solutions that provided UQ information based on non-parametric solutions include Gaussian Process Regression (GPR) (Biggio, Wieland, Chao, Kastanis, and Fink, 2021), and Relevance Vector Machine (Zhang, Lim, Qin, and Tan, 2017) approaches.

Combining a physics model with a data-driven one was termed as a ‘physics – data hybrid’ by Sprong, Jiang, and Polinder (2020). Considering the expense involved in preventative maintenance, it makes sense to maintain the failure mode models grounded in physics but in a way that is enhanced by additional data.

Model based and expert systems use, respectively, a model and a set of rules to infer when the degradation level requires attention (Zhang, Li, and Yu, 2006). Models may be trained on a set of failure modes each with its own signature and associated probabilities in a Hidden Markov Model (HMM) (Kwan, Zhang, Xu, and Haynes, 2003). Zhang, Xu, Kwan, Liang, Xie, and Haynes (2005) developed and implemented an approach where principal components of the input signals were mapped to HMM degradation states. Capturing degradation modes in this manner may not always be scalable across multiple components of a complex system. Further, identifying and training against perceived discrete degradation states might not be needed if a simple signature can be processed out from the available data. The anomaly detection methods like HMM and multivariate Gaussian methods (Liu and Chen, 2019) work best when there is some level of cleaned data. In other words, we need to first separate, as best as possible, the signal from the noise in order to produce the best detection outcomes. One hybrid approach is shown schematically in Figure 2. The system inputs and outputs are rarely known to a complete extent. Rather, only the observables inputs are captured at some time interval and those signals are accompanied with noise. Similarly, real time actual output is not traceable, but a sampled version is available. The truncated input and output information can inform a physics model, with some data-based augmentation to compensate for the modeling and sensing deficiencies. The goal is to reduce estimation error while maintaining the intuition. The ultimate utility of a prognostic model is in the level of certainty it delivers to the decision making process. The uncertainty around the expected life represents the overall operational flexibility offered by the model. As the estimate of remaining life diminishes, the horizon to take action is a function of the relative error distribution of the estimate.

1.2. Novel Contributions

This paper will deconstruct the development of a prognostic model in two steps described below:

- (1) Signal Estimation: The problem of maximizing a degradation signal and minimizing noise, building upon previous work in (Prakash and Brzoska, 2021), (Prakash, Brzoska, and Ensberg, 2022). This will involve defining a sensitivity objective, which can be maximized to produce an optimal time-series filter, and used as a proxy for evaluating signal correlation to events.
- (2) Characterizing Error: The amount of uncertainty in predicting RUL determines the effective operational life before a prognostic removal. As less uncertainty provides more operator flexibility, the usefulness of a model may be characterized by this metric. Optimizing for error increases reliability.

Finally, the elements will come together in an implementation framework. The concepts presented here overlap with previous works, but then go on to develop a few key assertions not previously mentioned.

- (1) Methods of optimizing a filter to detect degradation over background noise are explored
- (2) Numerical examples of signal processing and value assessment are provided in detail
- (3) The value in reducing unscheduled expense to a scheduled expense can be directly converted to a unit of operational life; this life saving is the critical ‘break-even’ point

This paper is organized as follows. First, in section 2, we will review the current status quo evaluation, including the binary classifier and RUL evaluation metrics like mean absolute error (MAE), mean absolute percent error (MAPE), and root mean squared error (RMSE). Then, we will propose an alternate methods such as (1) signal sensitivity to the event and (2) the confidence interval for RUL estimate in section 3, including a numerical example where different filter objectives are compared. This will constitute the first part of the evaluation construct, examining how well the signal correlates to the event. Then, we will discuss operational impacts that affect the placement of the threshold. Section 4 will discuss the uncertainty model and how RUL uncertainty affects the overall estimation. Section 5 will delve into the operational impact of the prognostic model, building up from a run-to-failure case and an inspection interval case. Taken together, the correlation of the prediction signal to the discrete event and the level of uncertainty around the remaining life estimate constitute a holistic picture on how to evaluate a prognostic model. Section 6 will discuss the implementation from start to finish, with a numerical example.

2. CURRENT MODEL EVALUATION METHODS

Common assessments of prognostic models have considered the overall goal of maximizing reliability (Fornlöf, 2016) or the trade-off between part availability versus operational efficiency (Pipe, 2008). Such considerations are relevant when the model is already operational and logistic questions remain. This work will consider whether a given model can be tuned to optimize its key performance metrics, and what those metrics should be. Currently, in the literature and in practice, the commonly accepted evaluation constructs are either the binary classifier or the RUL evaluation schemes like MAE and MAPE.

		Reality		
		No Event	Event	
Predicted	Alert +	False Positive FP	True Positive TP	Precision $\frac{TP}{TP+FP}$
	No Alert -	True Negative TN	False Negative FN	
				Recall $\frac{TP}{TP+FN}$

Figure 3. The binary classifier confusion matrix, showing the outcome when the prediction agrees and disagrees with the observed reality. Derivative metrics precision and recall capture high-level performance.

2.1. Binary Classifier

A common assessments of prognostic model performance is the binary classifier [26]. The capability to detect is quantified with ‘recall’ and the reliability of prediction is similarly quantified with ‘precision’. Recall and precision are based on a confusion matrix, where one axis is detection and the other reality (Figure 3). If both reality and detection agree, the model has scored a ‘true positive’ and if the model alerts without the event it is a ‘false positive’ while a ‘false negative’ means the model failed to adequately detect.

$$\begin{aligned} \text{Recall} &= P(\text{Prognostic Alert} \mid \text{Impeding Event}) \\ &= \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \end{aligned} \quad (1)$$

$$\begin{aligned} \text{Precision} &= P(\text{Impeding Event} \mid \text{Prognostic Alert}) \\ &= \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \end{aligned} \quad (2)$$

Recall is then the probability of alerting given the event will occur, while precision is the probability of the event occurring given an alert has been announced.

2.2. Remaining Useful Life Evaluation Metrics

When models are developed for predicting RUL specifically, the error between the predicted and actual life remaining are generally evaluated using a suite of three commonly used metrics (Liu and Chen, 2019). These are 1) the mean absolute error (MAE), 2) the mean absolute percent error (MAPE), and 3) the root mean square error (RMSE). The equations are defined in (3)-(5), respectively, where y_i is the true and y_i^* is the predicted RUL, and n is the total number of data points.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - y_i^*| \quad (3)$$

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - y_i^*}{y_i} \right| * 100\% \quad (4)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - y_i^*)^2}{n}} \quad (5)$$

Each of the three performance metrics provides a different insight into the results. The MAE gives less weight to the outliers which means that the metric is less sensitive to them. This also means that it may not adequately reflect the performance when dealing with large error values. On the other hand, the RMSE is heavily dependent on large errors. Furthermore, both MAE and RMSE are absolute errors specific to the scale of the data. These metrics do not provide any insight into the performance of data at different scales. For such application it is better to use MAPE which is the absolute error normalized over the data. The MAPE generates a metric that can be used to compare results across different scales. However, the MAPE does have drawbacks. True value data points that are equal to zero have to be excluded from the dataset to avoid dividing by zero. Additionally, errors at small values of y_i will have a large bearing on the result (Vuckovic, 2022).

While the above equations are adequate in evaluating the performance of the prediction across all time intervals, this unnecessarily over-weights the predictions at large RUL values. Indeed, the error tolerance of the prediction is much larger when the component is still healthy and the prediction is likewise not indicating a critical condition. The MAPE algorithm is less affected by this as the evaluation is more sensitive at small values of y_i , but the sensitivity increases geometrically whereas practicality, there is an interval around RUL = 0 where the prediction is equally critical to overall performance.

The overall problem of a clear evaluation regime can be subdivided into two parts. First, the issue is whether the model is properly capturing the intended degradation leading up to the event. The second part is the fundamental trade-off between too early or unnecessary maintenance against the probability of an unanticipated failure.

3. FEATURE SENSITIVITY

A prognostic model is only as effective as its ability to detect a set of given failure modes. This is a correlation problem. Evaluating a model only on the discrete outcomes (like true positives or false positives) misses the nuance of whether the underlying degradation is even well observed.

As depicted in Figure 4, the first problem in estimation is that the available data might not be complete. There are observable as well as unobservable system inputs that contribute to the degradation state. Second, even when the degradation is perfectly observable, the derived features will detect not only the degradation, but other confounding noises. Usually, there are several noise sources, each of which operate with a unique frequency signature. In the case of aircraft components, often there is a strong seasonality effect, driven by ambient temperature variation. Additionally, the idiosyncrasies of flight schedules, flight patterns, and daily weather also result in high variability. The goal is then to find the appropriate filter that can best track the real degradation (Figure 4).

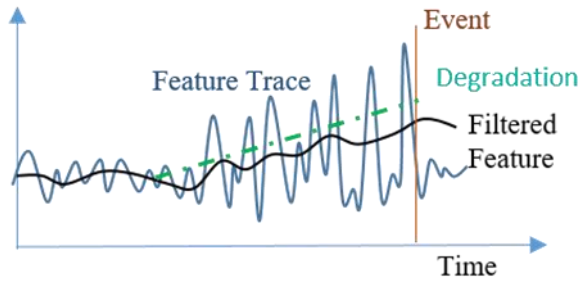


Figure 4. Noise in the feature may prevent it from accurately capturing the escalating level of degradation. A time-series filter can improve the outcome.

3.1. Filter Design

A base feature can be derived many ways, usually reflecting some amount of physical modeling. While a feature is sometimes a directly measured attribute like petal length (Dua and Graf, 2017), in PHM applications a feature can itself be an estimated quantity like an effective age or crack length. However, given limitations on sensing and observation, the computed feature will propagate these inaccuracies. Therefore, a signal processing step is required to improve the overall estimation. A dynamic filter modifies the feature in a manner that amplifies certain spectral content and suppresses the rest. Filters are commonly applied for noise rejection, modeling, estimation, and data fusion. The generic linear discretized filter that produces filtered output Y from input U has the form:

$$Y_k = \sum_i a_i U_{k-i} + \sum_j b_j Y_{k-j} \quad (6)$$

The filtered output at time instance k is Y_k and the output at preceding time samples are $Y_{k-1}, Y_{k-2}, \dots, Y_{k-M}$. Similarly, the input at the current k^{th} instance is U_k and the preceding values are $U_{k-1}, U_{k-2}, \dots, U_{k-N}$. The filter output at the current time instance is therefore a weighted summation of current and previous inputs, and in some cases, previous outputs. The filter coefficients a_i and b_j are chosen to achieve a spectral objective: suppression and amplification or a specified frequency range.

The filter described in equation (6) is a specific case of a linear filter. However, the filter structure need not follow this form. In this paper, a filter is simply any operation that acts along the time dimension of the input and prior outputs to produce an output at the current time instance.

In this case, we wish to evaluate the filter and with this evaluation, drive it to an optimal result. In the linear case above, we can assign coefficients a_i and b_j if we have the right objective function capturing performance. A description of such a performance evaluation criterion follows.

3.2. Lead Time Aggregation

Designing the appropriate filter requires an evaluation construct. Since the main objective is detection, the ideal filter will produce a signal that deviates most from its standard values during time intervals preceding known events and will return to its standard values once events have transpired.

Capturing the filter behavior across all known events requires isolating the filter output for a set lead time interval before each event (Figure 5, top). Each lead interval data point is averaged on a time or cycle basis with all other lead time traces at the same relative distance from the event.

$$X_i = \sum_{j=1}^N F_{ij} / N \quad (7)$$

The lead interval value X at the i^{th} sample before the event is the average of all trace feature values F across N events, at the i^{th} value before each j event. The resulting signal represents typical behavior for the signal ahead of an event.

These aggregated averages X are then standardized using z-score normalization.

$$Z = (X - E(F)) / STD(F) \quad (8)$$

The normalized value Z is the aggregated signal value X subtracted by the original signal mean $E(F)$ divided by the original signal standard deviation $STD(F)$. The z-score normalization has the advantage of allowing comparisons across all signals with different base units. Further, normalization produces a signal in terms of its standard deviation value so that the larger values, either positive or negative, are more anomalous.

There may be events which produce no detectable precursor, as may happen with a false negative. In that case, all filtered

outputs X will be penalized equally, and the event will not play a role in filter selection. Conversely, there may be maxima in a signal that are not associated with an event. These false positives will raise the mean value $E(F)$ and result in a lower normalized Z value.

The aggregated and normalized pre-event trace Z of each filter can be considered at some fixed interval before the event for comparison (Figure 5). In the figure, the normalized and averaged traces of two candidate features are plotted, and the time axis has time of event (t_E), minimum lead time to act before the event (t_0), and the initial time of the lead interval (t_i). Filter 1 in the bottom plot of Figure 5 exhibits maxima both well ahead of t_0 as well as in the t_0 to t_E interval. The filter value between t_0 and t_E is irrelevant since there isn't enough lead time to mitigate the event. However, too much lead time reduces useful life. Filter 1's behavior is less desirable compared to Filter 2, which has a maximum just before t_0 , providing ample lead time ahead of the anticipated event without sacrificing much useful life.

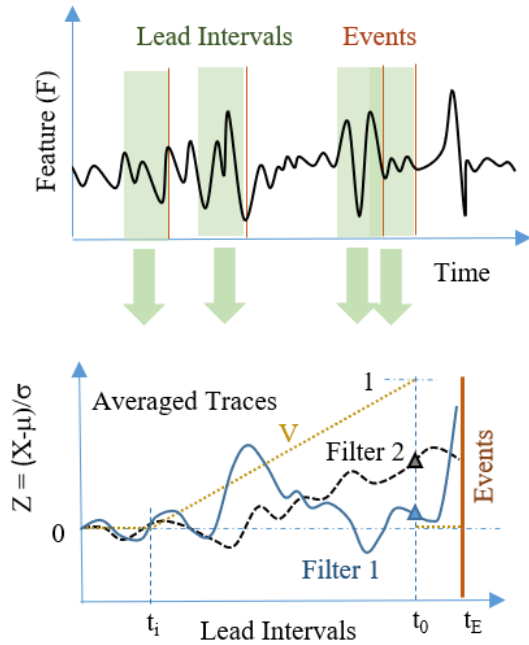


Figure 5. Method of averaging the lead intervals before each event

Table 1. Evaluation Methods for Lead Intervals

<i>Method</i>	<i>Formula</i>	<i>Description</i>
Critical Z	$Z_c = Z(t_0)$	Aggregated normalized value at min lead time t_0
Score S	$\Sigma_i(Z_i V_i) / \Sigma_i(V_i^2)$	Weighted Correlation to value function V
Correlation $CORR$	$\frac{\Sigma Z_i (V_i - E(V))}{STD(Z) * STD(V)}$	Pearson's Correlation to value function V
Recovery R	$Z(t_0) - Z(t_E + \Delta t)$	Ability of feature to return to nominal
Nominal Stability	$STD(Z(t_i - \Delta t : t_i))$	Variance of nominal signal
Monotonicity	$d/dt Z > 0$	Tendency of feature to increase
Estimate Uncertainty	$\frac{STD(F(t_0))}{Z(t_0)}$	Relative error of the anomaly
RUL Error	$\frac{Z_c - Z}{dZ/dt} - RUL$	Remaining useful life prediction error

3.3. Evaluation Methods

The Z value at the critical lead interval, $Z_c=Z(t_0)$, can be a useful gauge of the relative performance of a given filter compared to others. This does not require any arbitrary rules or limits, only the process-defined, requisite minimum lead time.

In certain cases, the lead trace Z values can be better evaluated with a weighting function V (Figure 5) which rises monotonically from 0 at some $t_i < t_0$ up to 1 at t_0 , then returning to 0 for the t_0 to t_E interval. For a given filter, each Z value can be combined with its weight V in a weighted sum:

$$S = \Sigma_i(Z_i V_i) / \Sigma_i(V_i^2) \quad (9)$$

The above equation ascribes a score S to the lead time averaged trace by weighting the i Z values with the weights V_i . High Z values near the event but before the critical actionable time (t_0) will have high V weights and increase the score while the other Z values will have less bearing on the score.

Applying equation (9) as an objective is functionally close to applying Pearson's correlation, shown as 'CORR' in Table 1. There is a benefit to equation (9), however, since maximizing 'CORR' can lead to the trivial result where $STD(Z)$ is minimal or even near zero. In both cases, the result is a correlation of the filtered value against a severity function.

The V function can be customized to suit the filter objectives. Any monotonically increasing function over t_i and t_0 will isolate signal components which show similarly increase

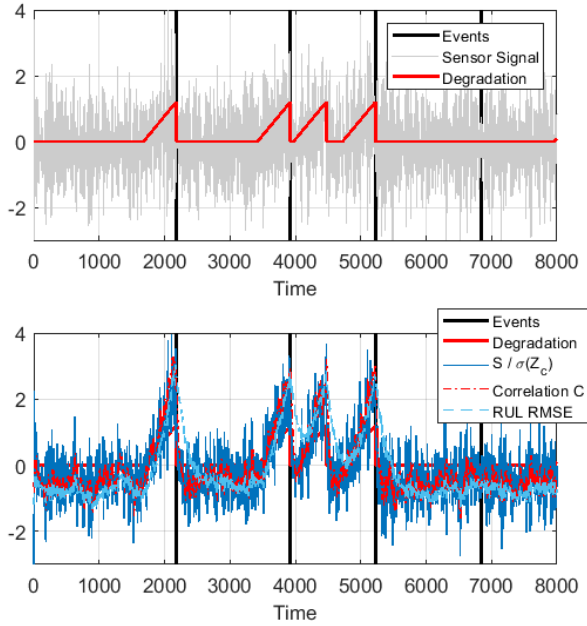


Figure 6. (top) Degradation, in red, grows linearly ahead of each event except the fourth, and an extra signature is inserted ahead of the third event. The sensor signal, in gray, is comprised of this degradation signal and multispectral noise. (bottom) Filtering the sensor signal with filters optimized for different objective functions.

steadily in the interval. A linearly increasing V value, either over cycles or flight hours, best captures the consumption of useful life on a cost basis. A sigmoid or step would acknowledge that any signal anomaly in the interval has comparable value, and loss of useful life is less important.

Note that correlating with a linear V as a proxy for RUL is fundamentally different from measuring RUL prediction error. In the former case, we are aiming to converge on a degradation estimate that has some increasing trend near the event with enough lead time. In the latter case, there is no consideration for a critical lead time or the beginning of the lead time period, rather the entire history of the RUL estimate is compared to the actual RUL.

3.4. Signal Conditioning

Employing optimized time-domain filters is a more meaningful alternative to the traditional precision, recall, and receiver operating characteristic of binary classifiers as well as a less ambiguous version of the RUL evaluation criteria. The latter methods have niche applications which do not translate as well to a continuous signal which can be filtered and correlated.

In this framework, the evaluation methods in Table 1 are both a measure of signal correlation to a discrete event and the mechanism for obtaining the optimal signal. If V is chosen to

reflect the value of remaining useful life, the score S and correlation $CORR$ reflect monetary benefit of the detector.

Conceptually, the filter tuning method is a way to model missing physical elements in a catch-all filter, with the ultimate objective of producing a feature anomaly near failure events but with sufficient lead time. This idea is shown in Figure 2. True degradation is driven by both known and unknown sources. Detection of the degradation is not perfect because confounding factors and sensing limitations respectively introduce noise and limit observability. The resulting signals are arranged into features using the known degradation mechanisms so that the features are physically explainable. The heuristics acknowledge the imperfections in the feature and attempts to compensate for them in order to estimate the degradation level.

3.5. Numerical Results

Three different objective functions were evaluated against simulated degradation data. In order to synthesize data, we consider a component which has a lifespan distributed normally with mean 1250 cycles and standard deviation 250 cycles. Over the course of 10000 cycles, there are 7 discrete failure events.

A degradation signature is modeled as a linearly increasing signal in the 500 cycles leading up to the event, and zero at all other points. This represents a form of physical process where the degradation is evident only in the final stage of life and progresses at a constant rate until failure. In many practical scenarios, the degradation signal may not always be present before failure. To simulate this case, the degradation element has been removed for failure number 4. This is a false negative example. Furthermore, the interval between failure numbers 2 and 3 has an additional degradation progression that does not immediately precede an event. This represents a case where the event record is missing, or a different action (aside from the labeled event) resolved the degraded state.

A noisy indicator will contain traces of the component degradation and noise from various sources. For this example, the noise is a set of 8 sinusoidal signals with random amplitudes up to .65 and frequencies spanning .01 to .03 Hz. Then, uniform random noise is added with zero mean and amplitude 2.5. The resulting noisy degradation signal is shown in the top plot of Figure 6. This represents a raw sensor signal.

This raw signal now contains degradation signatures prior to each event but one, and an additional degradation signature has been inserted between removals 2 and 3. Where these signatures exist, they are almost indistinguishable against the noise. A plot of lead time traces prior to each of the seven events is shown, mean-normalized, in Figure 7. The individual profiles are the signal value subtracted by the mean and then divided by the standard deviation, in the 500 cycles

immediately before each event. Even though Profile 4 has no degradation component, it is similar to all the other profiles which do contain the linear degradation.

Next, a set of optimal filters were developed in the form of equation (6). Note that we could have also used a nonlinear time series approach like a deep neural net or Gaussian process regression (GPR). The coefficients a_i and b_i were chosen to maximize different objective functions listed below:

1. $S/STD(X(t_0))$

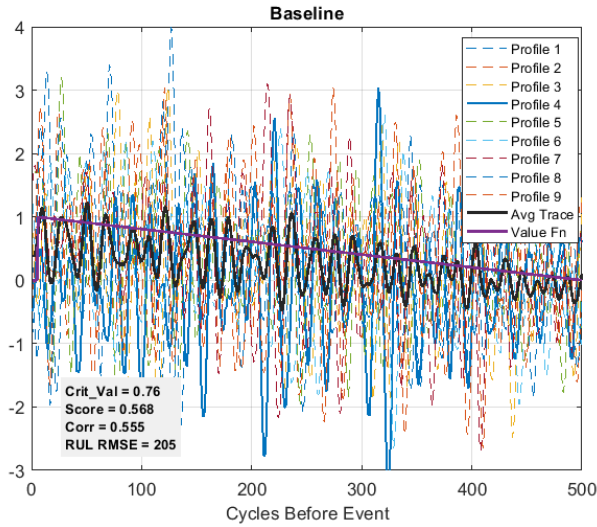


Figure 7. Behavior of the raw sensor signal (profiles) ahead of each event. Note that profile 4 had no underlying degradation, but it is indistinct from the other profiles.

2. Correlation ($CORR$)
3. RUL RMSE

These three filters with their associated performance metrics, compared against the no filter case is shown in Table 2. The time series results are shown in the bottom plot of Figure 6. Profiles of the resulting filtered signal leading up to each event are shown in figures 8-10.

Considering only the skipped degradation profile, i.e. profile 4, it is clear that each of the three filters effectively separates this case from the others, where degradation is present.

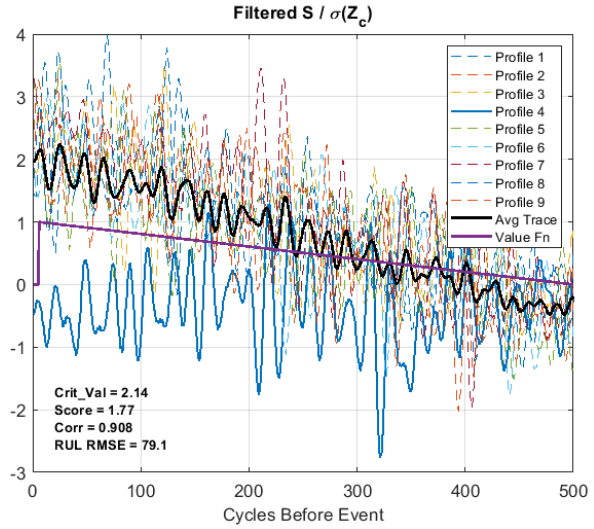


Figure 8. Behavior of the filtered signal (profiles) ahead of each event, using a filter optimized to the objective function $S/STD(X(t_0))$. Note that profile 4 lacks a signature and is distinct from the other profiles.

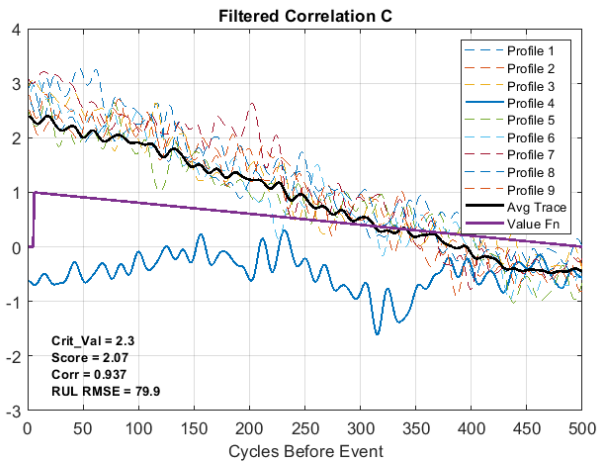


Figure 9. Behavior of the filtered signal (profiles) ahead of each event, using a filter optimized to the objective function based on correlation $CORR$. Note that profile 4 lacks a signature and is distinct from the other profiles.

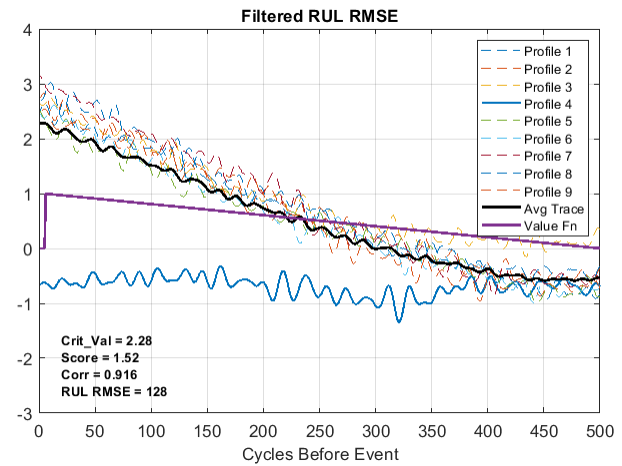


Figure 10. Behavior of the filtered signal (profiles) ahead of each event, using a filter optimized to the objective function RUL RMSE. Note that profile 4 lacks a signature and is distinct from the other profiles.

Table 2. Result of different filter objectives on performance metrics

	NO FILTER	S $STD(X_c)$	$CORR$	RUL RMSE
Z_c	0.76	2.14	2.3	2.28
$STD(X_c)$	1.16	1.04	1.18	1.11
S	0.568	1.77	2.07	1.52
$CORR$	0.555	0.908	0.937	0.916
RUL MAE	161	63.9	65.5	100
RUL MAPE	291	52	52.9	50.5
RUL RMSE	205	79.1	79.9	128

Beyond that behavior, each filter has its own nuances. Note that the best RUL RMSE is achieved not by minimizing this quantity, but rather going after either a high score S or a high correlation $CORR$. Also, the recovery behavior post-event seems to be slower for the filter that is tuned for RUL RMSE, even though it appears to have the lowest noise. By eliminating more high frequency content, this filter produces a smoother trace which is also slower to respond to an abrupt change in condition post-event.

4. RUL UNCERTAINTY

The degree of correlation to RUL (as shown with S in equation (9)) is more intuitively applicable to prognostic models than comparing the error against RUL. When S is maximized, the optimal filter transforms the base signal F such that the aggregated and normalized Z is most correlated with RUL. Using S as an evaluation metric avoids the pitfall of a growing error near low RUL like the MAE and determines whether there is a trend, unlike the MAE. Nonetheless, there is the impetus to translate this correlation to a RUL since that is the more applicable metric. The degree to which the prognostic model can estimate RUL belies its fundamental usefulness.

Having a low degree of confidence in the RUL estimate means that operationally, the mitigation must be more urgent, perhaps even rising to the level of the unscheduled event itself. Hence, there is a direct relationship between model operational performance and its RUL uncertainty.

4.1. Uncertainty Estimation

If we start with the scenario where the underlying degradation signal has been filtered such that the correlation to RUL (via an objective like S or C , with a linearly increasing V), the resulting signal will be as close to linear as possible with RUL. We can consider a conversion from the averaged lead interval trace Z to RUL. One technique is described below.

$$RUL_{est} = (Z_c - Z)/(dZ/dt) \quad (10)$$

Or, alternatively,

$$RUL_{est} = -\tilde{Z}/\tilde{Z}' \quad (11)$$

Here, the estimated remaining useful life RUL_{est} is the difference between the normalized anomaly (Z) at its critical value Z_c (the value of Z considered effectively failed) and its current value Z . We define the margin parameter $\tilde{Z} = Z_c - Z$. \tilde{Z}' is the time derivative of \tilde{Z} and is equal to $-dZ/dt$. The time parameter t can be expressed in the relevant ‘life units’ of operation time or operation cycles.

The determination of Z_c is not trivial, but the ambiguity can be factored in symbolically. The variability for the anomaly Z and its critical value Z_c can be expressed as a combined variability on \tilde{Z} . This variability is denoted $\Delta\tilde{Z}$. If we consider the uncertainty around the estimated RUL, we get the following relationship

$$\Delta RUL_{est} = (\Delta\tilde{Z}' \cdot \tilde{Z} - \tilde{Z}' \cdot \Delta\tilde{Z})/Z'^2 \quad (12)$$

The uncertainty of the RUL estimate is governed by not just the uncertainty of the anomaly indicator \tilde{Z} , but also its rate of change \tilde{Z}' .

Generally, there is a limit where the uncertainty on the RUL estimate ΔRUL_{est} approaches a sufficiently large percentage of the RUL itself. At this point, any buffer has been expended and mitigation will have to be prioritized. We can consider the uncertainty normalized by the value itself as a relative uncertainty.

$$\frac{\Delta RUL_{est}}{RUL_{est}} = \frac{\Delta\tilde{Z}}{\tilde{Z}} - \frac{\Delta\tilde{Z}'}{\tilde{Z}'} = \frac{\Delta\tilde{Z}}{\tilde{Z}} + \frac{\Delta\tilde{Z}'}{\tilde{Z}'} \quad (13)$$

The analysis shows that the relative uncertainty in RUL is a combination of the relative uncertainty in the anomaly margin estimate \tilde{Z} and the relative uncertainty of the slope of the anomaly estimate \tilde{Z}' (Note that $\tilde{Z}' = -Z'$)

At large RUL when \tilde{Z} is also large, there can also be a large overall uncertainty ΔRUL . This is captured by the first term in (12). Intuitively, this means the uncertainty for predicting the future grows with how far away that future is. But the mere virtue of a high RUL does not mean that the relative uncertainty ($\Delta RUL/RUL$) is high, so long as the rate of degradation can be ascertained (Equation (13)).

Determining the rate of change of anomaly growth and the error associated with it has its own challenges. In spectral terms, a derivative operator ($j\omega$ in Fourier Transforms)

increases linearly with frequency so high frequency noise will be amplified. In a general sense, more data drives a better estimate so there must be some ongoing degradation before the slope and its uncertainty can be ascertained. This represents another tradeoff between accurately determining the relative uncertainty of RUL and giving up some RUL to do so.

Secondly, the rate of RUL consumption can change due to a variety of factors and these changes may not be immediately detected. This is a fundamental tradeoff of signal processing. Either a lot of time is consumed in forming an accurate estimate or there must be a reckoning with high speed but inaccurate calculations.

4.2. Operational Implications

The threshold for prognostic mitigation, considering only the RUL estimate, can be expressed in terms of the relative uncertainty from Equation (13). Action should be planned when the relative uncertainty is above some fractional value, and considering that positive error on RUL is operationally palatable while negative error is not. Further, the planned action should take into consideration the required scheduling lead time for corrective mitigation. We can express the mean end of operational life as follows

$$T_{pr} = MTBUR - K * \Delta RUL_{est,eol} - \Delta t_{sd} \quad (14)$$

Here, the average component life (prognostic end of life T_{pr}) with prognostic estimation of RUL is reduced from the nominal mean time between removal (MTBUR) by subtracting the appropriate estimation confidence interval with scale factor K and the time interval required to schedule the intervention, Δt_{sd} . Note that the RUL uncertainty at this end of operation condition $\Delta RUL_{est,eol}$ is from Equation (12) evaluated at T_{pr} . Since \tilde{Z} diminishes to small values near end of life, and $\tilde{\Delta Z} \cong \Delta Z$, $\Delta RUL_{est,eol}$ becomes:

$$\Delta RUL_{est,eol} \approx \frac{\Delta Z}{Z'}|_{t=T_{pr}} \quad (15)$$

In this manner, the uncertainty around the anomaly estimate, $\Delta \tilde{Z}$ scaled by the inverse of the rate of change of the anomaly growth Z' determines a key term in the overall operational life reduction. This is an important consideration when weighing different strategies for maintenance cost management.

5. STRATEGY ANALYSIS

The operator can pursue a few different strategies depending on resources and data availability. The overall strategy that best suits a given scenario can be identified by evaluating the relative costs. The goal would be to best characterize the advantages and disadvantages of each strategy in a common set of unit costs such that the trade-offs can be modeled. Fundamentally, maintenance strategies will prefer a lower cost of scheduled removal, C_{sr} , over the cost of unscheduled

removal, C_{ur} provided that there isn't too much additional cost in terms of inspection time and labor and lost operational life.

5.1. Run to Failure Case

In order to construct the overall cost model, the case of no interventions may be considered first. The expected cost per unit life ($J = C/T$) is simply the following:

$$J_{rtf} = \frac{C_{ur}}{MTBUR} \quad (16)$$

In this 'run to failure' scenario the cost per unit life of run to failure, J_{rtf} , is the unscheduled removal cost C_{ur} divided by the mean time between unscheduled removal ($MTBUR$).

5.2. Inspection Interval Case

When operational risks of failure are somewhat higher, there is occasionally an inspection routine implemented to monitor and manage component health. Further, when there is a feasibility for in-situ repairs, the overall life may even be extended.

Over the course of component life, the component may be removed at final inspection, when the inspection is effective, or at its end of life when the inspection is ineffective. The total number of inspections over the course of component life can be described as follows:

$$N_{ins} = \frac{MTBUR + \Delta T_{rep_gain} - 0.5 * \Delta T_{ins} * P_{ins}}{\Delta T_{ins}} \quad (17)$$

The number of inspections N_{ins} is equal to the total component life divided by the inspection interval ΔT_{ins} . The total life is increased when there are beneficial repairs by ΔT_{rep_gain} and reduced when an inspection with effectiveness probability P_{ins} can detect the issue within an inspection period before in-service failure. For this analysis, we will consider only the case where there is no life extension from repairs, i.e. removals are the only option. Further, we will assume the fractional loss of life with an inspection interval is relatively small compared to the overall expected life, the $MTBUR$. This simplifies equation (17) to $N_{ins} = MTBUR/\Delta T_{ins}$.

The total cost of operation for a component undergoing regular inspections must take into account the cost of the inspections themselves. The payoff is a reduction in unscheduled repair costs when the inspection can successfully locate the impending failure in advance. The cost per unit life of this strategy is given as follows:

$$J_{ins} = \frac{C_{ins}N_{ins} + C_{sr}P_{ins} + C_{ur}(1 - P_{ins})}{MTBUR} \quad (18)$$

The overall maintenance cost has the summation of the inspection costs C_{ins} , the scheduled removal costs C_{sr} when the inspection was effective, multiplied by the aggregate probability that the failure would be detected by the inspection, P_{ins} . The last term captures the unscheduled removal costs C_{ur} when the inspection fails to catch the

failure mode, with a probability of $1 - P_{ins}$. This overall maintenance cost is divided by expected component life, which was simplified to the mean time between unscheduled removals, MTBUR.

Applying the simplification $N_{ins} \approx MTBUR/\Delta T_{ins}$ to (18), and incorporating (16) yields the following.

$$J_{ins} = J_{rtf} + \frac{C_{ins}}{\Delta T_{ins}} + \frac{P_{ins}(C_{ur} - C_{sr})}{MTBUR} \quad (19)$$

When the costs of inspection C_{ins} or scheduled removal C_{sr} are low relative to the unscheduled removal cost C_{ur} , this scenario can provide some cost savings over the run to failure case J_{rtf} . Further improving this scenario's advantage would mean increasing the inspection interval ΔT_{ins} or improving the effectiveness of the inspection regime in detecting impending failures P_{ins} .

5.3. Prognostic RUL Estimation Case

With the proper framework, the case of estimated RUL may now be examined for the relative cost advantage against the run to failure and the inspection cases. Generally, prognostic applications require some overhead infrastructure like sensors, and data handling, storage, and compute resources.

These costs will be excluded for the analysis as we consider an overall advantage without these contributors. Like the inspection scenario, the prognostic has some probability of observing the incipient failure with enough lead time to avoid the unscheduled event. Unlike the case of inspections, however, the operational life lost with an analytic is not limited to the inspection time interval, but grows as a function of RUL uncertainty, as already shown in Equation (14). The lost life from estimation error presents a concrete loss of value in terms of lost operational time. The expected life of a component with prognostic health monitoring through an analytic is as follows:

$$T_{pr,eff} = MTBUR * (1 - Re) + T_{pr} * Re \quad (20)$$

Essentially, we can expect to see the effective component life $T_{pr,eff}$ reduced when the analytic can detect the failure mode. This is the concept of recall Re from Equation (1). When there is no detection, the component life is unaltered.

The overall expected time lost while including the scenario of imperfect recall may now be calculated. This is simply a manipulation of Equation (20).

$$\Delta T_{pr,eff} = MTBUR - T_{pr,eff} = Re * \Delta T_{pr} \quad (21)$$

Here, ΔT_{preol} Here, $\Delta T_{pr} = MTBUR - T_{pr}$. The overall maintenance cost per unit life, considering the savings when the detection is successful and a penalty for lost operational time is expressed below:

$$J_{pr} = \frac{C_{sr}Re + C_{ur}(1 - Re) + \gamma \Delta T_{pr,eff}}{MTBUR - \Delta T_{pr,eff}} \quad (22)$$

This equation acknowledges that there is a conversion from operational time lost to cost impact. The scale factor γ converts the lost time to a cost. One conversion may take $\gamma = C_{comp}/MTBUR$, where C_{comp} is the price of a new component. This would mean that the price of a component reflects its expected service life so lost operational time is as valuable monetarily as a fractional price of a new component.

To compare the prognostic maintenance scenario more directly to the run to failure case, Equation (22) can be manipulated assuming that the lost operational life ΔT_{pr}^* is small relative to the overall component life, so $MTBUR - \Delta T_{pr}^* \approx MTBUR$. With this simplification, the cost per unit life of prognostic maintenance becomes the following:

$$J_{pr} = J_{rtf} + \frac{Re}{MTBUR} (\gamma \Delta T_{pr} - (C_{ur} - C_{sr})) \quad (23)$$

The advantage of a prognostic analytic is improved when there is a large difference between the unscheduled removal cost C_{ur} and the scheduled removal cost C_{sr} , when there are a large fraction of failures detected early enough to avoid the unscheduled event (recall Re), and when the loss of life ΔT_{pr} is relatively small when scaled by the time to cost conversion rate γ . The last two criteria are in opposition; detection must be early enough to allow for avoiding the event, yet close enough to the actual failure so that operational time is maximized.

Substituting the life lost relationships from the previous section, as outlined in (14) and (15), the overall advantage of a prognostic can be expressed in terms of the RUL uncertainty ΔRUL and relative anomaly uncertainty $\Delta Z/Z'$. Equation (23) becomes:

$$J_{pr} = J_{rtf} + \frac{Re}{MTBUR} (\gamma (K \frac{\Delta Z}{Z'} |t_{pr} + \Delta t_{sd}) - (C_{ur} - C_{sr})) \quad (24)$$

It is evident that the additive costs associated with a high anomaly estimation error ΔZ can counteract any value gained from avoiding the unscheduled costs $C_{ur} - C_{sr}$. Key in this analysis is the value of K , which scales the uncertainty into a confidence interval around the RUL estimation, and γ , the conversion from lost time to lost value.

From equation (24), there is a clear point where the prognostic analytic adds value and where it subtracts value compared to the nominal run to failure case. Generally, in order to show a positive cost outcome, the following must hold:

$$K \frac{\Delta Z}{Z'} \leq \frac{(C_{ur} - C_{sr})}{\gamma} - \Delta t_{sd} \quad (25)$$

When evaluated near the end of life, equation (25) can determine whether the component should be removed proactively or not. The scale factor K represents a confidence interval providing some threshold that the part will still be functional upon removal. Ultimately, as the ratio $\Delta Z/Z'$ varies near the expected prognostic life T_{pr} , the decision of whether

to take action on the indicator will depend on the value of that ratio.

6. IMPLEMENTATION

Considering all the relationships established so far, a process may be implemented to evaluate a prognostic model. This would involve two main steps. First, a feature must be composed using either a data-driven or physics based approach. This indicator will be subject to limitations around data availability and sensor noise. Accordingly, this feature may be processed in a manner that best reflects the underlying degradation. Finally, operational cost considerations and signal uncertainties can be factored together for the best performing model.

6.1. Signal Processing

It is evident that the health indicator should reliably trend toward anomalous values as end of life approaches. In the most fundamental sense, the indicator should be anomalous at a critical stand off period before the event, and this should be proven with historical failure data. Time series filters can be tuned to reduce noise and amplify the degradation signature, as was demonstrated. The most effective methods involve maximizing the time correlation of the indicator to the remaining useful life, or RUL, either via a weighted score S or direct correlation CORR from Table 1.

6.2. Operational Value Assessment

Once the sensitivity to degradation has been optimized, operational costs may be considered, as described in section 5. Begin first by establishing a generalized removal criterion, considering equations (13) and (14).

$$\frac{\Delta RUL_{est}}{RUL_{est,crit}} = \frac{\Delta \tilde{Z}}{\tilde{Z}} + \frac{\Delta Z'}{Z'} = \frac{1}{K} \quad (26)$$

Here, $\tilde{Z} = Z_c - Z$, this is the difference or margin between the prognostic indicator value Z and its critical value Z_c . Z' and $\Delta Z'$ are the slope and the uncertainty of the slope, respectively. Fundamentally, the component should be removed when the error margin of the RUL (or ΔRUL) is a large enough fraction of the estimated RUL.

Near the beginning of life, \tilde{Z} is large and the second term dominates the inequality. If there is enough error in determining the slope, $\Delta Z'$, or if the slope is sufficiently small compared to the uncertainty, this could lead to an early removal, or a reduced reliance on the indicator Z .

Towards the end of life, as \tilde{Z} shrinks, the first term dominates and other considerations become relevant. We may now apply the approximation from equation (15) and the relationship from equation (27).

$$RUL_{est,crit} = \frac{\tilde{Z}}{Z'} \approx \frac{(C_{ur} - C_{sr})}{\gamma} - \Delta t_{sd} \quad (27)$$

According to the equation above, the critical time to remove a component depends on the relative cost savings between unscheduled and scheduled removal ($C_{ur} - C_{sr}$), converted from cost to time with conversion factor γ , and then reduced by a standard response time Δt_{sd} . This critical RUL value depends on the ratio between the margin of the estimator to its critical failure value and the slope of the estimator. As either quantity changes, the critical value to remove the component changes as well.

The critical RUL from equation (27) represents a 'break even' point where any loss of operational RUL is balanced by the savings from converting the unscheduled to a scheduled removal. If the estimate uncertainty ΔZ is small relative to its rate of change Z' , the component may remain operational while the inequality in equation (25) remains true.

6.3. Numerical Results

An example was generated using simulated operational data similar to the one examined in section 3. The assumed values are shown in Table 3.

Table 3. Initial values for example calculations

Parameter	Value
Z'	.01 score/cycle
ΔZ	0.8 score
K	2
Δt_{sd}	10 cycles
γ	\$5 / cycle
Re	0.75
$MTBUR$	1350 cycles
$C_{ur}-C_{sr}$	\$5000

Applying the relationship in equations (23-25), we see that the maximum allowable useful operational time lost due to the prognostic is computed as follows:

$$\Delta T_{pr,max} = K \frac{\Delta Z}{Z'} + \Delta t_{sd} = 170 \text{ cycles} \quad (28)$$

This represents a 'break even' where any further operation beyond 170 cycles presents positive value. Converting this to cost using conversion γ yields \$850. This is the minimum value gain between unscheduled and scheduled cost to make this detector viable, or:

$$\gamma \Delta T_{pr,max} = \$850 \leq C_{ur} - C_{sr} \quad (29)$$

Note that neither the recall Re nor the nominal life of the component $MTBUR$ has any impact on the above assessment. If we now suppose that $C_{ur}-C_{sr} = \$5000$, we can compute the overall cost advantage against the run-to-failure case using equation (24):

$$J_{pr} = J_{rtf} + \frac{.75}{1350 \text{ cycles}} (\$850 - \$5000) \quad (30)$$

The savings compared to a run-to-failure scenario are about \$2.305 per cycle, or \$2720 per unit if it is removed, on

average, at the critical time period of $MTBUR-\Delta T_{pr,max}=1180$ cycles. This will be true when the RUL determination is at 170 cycles.

Note that while there is value in letting the component run beyond the RUL=170 cycles point, the risks due to the estimation error exceeds the acceptable bounds dictated by the confidence interval parameter K .

7. CONCLUSIONS

The paper has investigated an evaluation method for a prognostic analytic, comparing it to both a run-to-failure case and an inspection interval case. While the analytic may be based on an understanding of physical degradation, unknown effects, confounding factors, and signal limitations will present estimation challenges. The level of signal correlation to an expected degradation progression is one type of evaluation metric, but ultimately the uncertainty around the RUL estimation determines the operational benefit.

The approaches presented here address shortcomings evident in alternate evaluation methods like the binary classifier and the various RUL estimation errors. The binary classifier cannot handle the time varying nature of signals without introducing ambiguity and strict errors to RUL throughout the life neglect that RUL is most critical near end of life.

Instead, this paper evaluates behavior in the lead intervals ahead of events and considers key behavior patterns like whether the signal is anomalous (critical Z from Table 1) or whether the signal correlates with the RUL (S score and CORR correlation from Table 1). Ultimately, the key marker for applied benefit is the uncertainty of the RUL estimate, ΔRUL . The level of uncertainty can be used to define a confidence interval. The uncertainty at the point of decision crucially determines one aspect of the expected loss of operational life as a result of estimate uncertainty.

While there is significant cost saving from avoiding unscheduled removal events, this savings is possible only when failure modes are detectable in advance. Accordingly, savings are tempered by lost value in unrealized operational time. This time increases with higher estimation uncertainty, thereby closing the conceptual loop on evaluating the prognostic model estimate.

Understanding and modeling all aspects of degradation estimation and operational cost savings reveals a better picture of the overall benefit and impact of an analytic. These relationships can then be applied to optimize the right model parameters or converge to a more optimal model design.

NOMENCLATURE

CNN	convolutional neural networks
$(B)DL$	(Baysean) deep learning
FN	false negative
FP	false positive

GPR	Gaussian process regression
HMM	hidden Markov model
MAE	mean absolute error (to RUL)
$MAPE$	mean absolute percentage error (to RUL)
$MTBUR$	mean time between unscheduled removal
PHM	prognostics and health management
RMSE	root mean squared error (to RUL)
RUL	remaining useful life
TN	true negative
TP	true positive
UQ	uncertainty quantification

C_p	pressure coefficient
F_x	X component of the resulting force
F_y	Y component of the resulting force
m	mass
dt	time step
T	temperature
P	pressure
f, g	generic functions
h	height
I	current
V	voltage
α	dummy variable

REFERENCES

- Biggio, L., Wieland, A., Chao, M. A., Kastanis, I., and Fink, O., (2021) Uncertainty-Aware Prognosis via Deep Gaussian Process. *IEEE Access*, vol. 9, pp. 123517-123527, 2021, doi: 10.1109/ACCESS.2021.3110049.
- Deng, K., Zhang, X., Cheng, Y., Zheng, Z., Jiang, F., Liu, W. and Peng, J., (2019). A remaining useful life prediction method with automatic feature extraction for aircraft engines. *Proceedings of 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*. IEEE, 2019, pp. 686–692.
- Ding, Y., and Jia, M., (2021). A convolutional transformer architecture for remaining useful life estimation. *Global Reliability and Prognostics and Health Management (PHM-Nanjing)*, pp. 1–7
- Dua, D., and Graff, C., (2017). UCI machine learning repository. 2017. [Online]. Available: <http://archive.ics.uci.edu/ml>
- Fawcett T. (2006). An Introduction to ROC Analysis, *Pattern Recognition Letters*. 27 (8): 861–874.
- Fornlöf, V., Galar, D., Syberfeldt, A. and Almgren, T. (2016). Maintenance, prognostics and diagnostic approaches for aircraft engines, 2016 *IEEE Metrology for Aerospace (MetroAeroSpace)*, pp. 403-407, doi: 10.1109/MetroAeroSpace.2016.7573249

- International Maintenance Review Board Policy Board, Aircraft Health Monitoring (AHM) Integration in MSG 3, Issue Paper 180. (2018) MSG-3: Operator/Manufacturer scheduled Maintenance Development, Vol. 1- Fixed Wing Aircraft Rev. 2018.1, Air Transport Association of America, <https://www.easa.europa.eu/document-library/imrbpb-issue-papers>.
- Le, D.D., Ghoshal, A., and Cuevas, E. (2011). Condition-based maintenance plus and maintenance credit validation. Army Research Laboratory, Dept. of Defense, Aberdeen Proving Ground, MD 21005. <https://apps.dtic.mil/sti/pdfs/ADA589574.pdf>
- Li, G., Yang, L., Lee, C.-G., Wang, X., and Rong, M., (2021). A Bayesian deep learning RUL framework integrating epistemic and aleatoric uncertainties. *IEEE Transactions on Industrial Electronics*, vol. 68, no. 9, pp. 8829–8841.
- Li, X., Ding, Q., and Sun, J.-Q., (2018) “Remaining useful life estimation in prognostics using deep convolution neural networks,” *Reliability Engineering System Safety*, vol. 172, pp. 1–11.
- Liu, J., & Chen, Z. (2019). Remaining useful life prediction of lithium-ion batteries based on health indicator and gaussian process regression model. *IEEE Access*, 7, 39474-39484. doi: 10.1109/ACCESS.2019.2905740
- Luo, J. H., Namburu, M., Pattipati, K., Liu, Q., Kawamoto, M., and Chigusa, S., (2003). Model-based prognostic techniques [maintenance applications], *Proc. AUTOTESTCON 2003. IEEE Systems Readiness Technology Conference.*, pp. 330-340, (22-25 Sept.) doi: 10.1109/AUTEST.2003.1243596.
- Mazaev, G., Crevecoeur, G., and Hoecke, S. V., (2021). Bayesian convolutional neural networks for remaining useful life prognostics of solenoid valves with uncertainty estimations. *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 8418–8428.
- Pipe K. (2008). Practical prognostics for condition based maintenance, *International Conference on Prognostics and Health Management*.
- Prakash, S., & Brzoska, A. (2021). Evaluating and Optimizing Analytic Signals. *Annual Conference of the PHM Society*, Virtual, 13(1). November 11-13, doi: 10.36001/phmconf.2021.v13i1.2968
- Prakash, S., Brzoska A., and Ensberg, J. (2022). A Framework for Evaluating PHM Models. *Proc. IEEE Aerospace Sciences Conference*
- Ren, L., Sun, Y., Wang, H., and Zhang, L., (2018) Prediction of bearing remaining useful life with deep convolution neural network. *IEEE Access*, vol. 6, pp. 13 041–13 049.
- Song, Y., Li, L., Peng, Y., and Liu, D. (2018) Lithium-ion battery remaining useful life prediction based on GRU-RNN. *Proceedings of 12th International Conference on Reliability, Maintainability, and Safety (ICRMS)*, pp. 317–322.
- Sprong, J. P., Jiang, X., & Polinder, H. (2019). A Deployment of Prognostics to Optimize Aircraft Maintenance: A Literature Review. *Annual Conference of the PHM Society*, 11(1). doi: 10.36001/phmconf.2019.v11i1.776.
- Vuckovic, K., and Prakash, S. (2022) Remaining useful life prediction using gaussian process regression model. *Annual Conference of the PHM Society*, vol. 14, no. 1.
- Zhang, C., Lim, P., Qin, A. K., and Tan, K. C., (2017). Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics. *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2306–2318, 2017.
- Zhang, X., Xu, R., Kwan, C., Liang, S. Y., Xie, Q., and Haynes, L. (2005). An integrated approach to bearing fault diagnostics and prognostics. *Proceedings of the 2005 American Control Conference*. IEEE, pp. 2750–2755.
- Zhang, Y., Xiong, R., He, H., and Pecht, M. G., (2018). Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries. *IEEE Transactions on Vehicular Technology*, vol. 67, no. 7, pp. 5695–5705.
- Zhu, J., Chen, N., and Peng, W. (2018). Estimation of bearing remaining useful life based on multiscale convolutional neural network. *IEEE Transactions on Industrial Electronics*, vol. 66, no. 4, pp. 3208–3216.

BIOGRAPHY



Shashvat Prakash received a B.S. in Mechanical Engineering from University of Illinois, Urbana, an M.S. in Mechanical Engineering from Carnegie Mellon University, and a Ph.D. in Mechanical Engineering from the Georgia Institute of Technology. He has worked on satellite attitude and orbit control at NASA Goddard, combustion and control of aviation turbines at General Electric, and prognostics and health management (PHM) of aircraft components at Raytheon Technologies-Collins Aerospace



Katarina Vuckovic received her B.S. in Aerospace Engineering (2017), B.S. in Electrical Engineering (2017), and M.S. in Electrical Engineering (2019) from Florida Institute of Technology. She is currently pursuing her Ph.D. studies in Electrical Engineering at the University of Central Florida. She has been with Collins Aerospace for five years working as a systems engineer on wireless communication systems, aircraft automation applications, and prognostics and health management of aircraft components.



Sanket Amin received B.S. and M.S. in Mechanical Engineering from The Pennsylvania State University. His 15+ year career is mostly in the area of CBM and PHM and spans from aircraft engine monitoring to various aircraft systems. He has contributed papers in this area related to fault detection of

bearing and gear failures, fuzzy logic reasoning, and big data processing for PHM. He has collaborated with various companies and organizations in this area including Impact Technologies, NASA Ames, GE, Pratt and Whitney, NAVAIR, AFRL. He currently leads a group of data scientists and systems engineers in developing predictive analytics for commercial aircraft.