Performance Evaluation for Fleet-based and Unit-based Prognostic Methods

Abhinav Saxena[1], Shankar Sankararaman[2], and Kai Goebel[3]

[1,2]*SGT Inc., NASA Ames Research Center, Moffett Field, CA, 94035, USA*
abhinav.saxena@nasa.gov
shankar.sankararaman@nasa.gov

[3] *NASA Ames Research Center, Moffett Field, CA, 94035, USA*
kai.goebel@nasa.gov

ABSTRACT

Within the last decade several new methods for prognostics have been developed and an overall understanding of the various issues involved in predictions for health management has significantly improved. However, it appears that there is still a lack of consensus on how prognostics is defined and what constitutes good performance for prognostics. This paper first differentiates prognostics from other prediction approaches before highlighting key attributes of performance for prediction methods. Then it argues that it is important to understand what factors affect the performance of a prognostic approach. Factors such as the application and end use of a prognostic output, the various methods to make predictions, purpose of performance evaluation, etc. are discussed. This paper presents a comprehensive view of various such aspects that dictate or should dictate what performance evaluation must be as far as prognostics is concerned. It is also discussed what should be used as baseline to assess performance and how to interpret commonly used comparisons of algorithm predictions to observed failure times. The primary goal of this paper is to present some arguments of how these issues can be addressed and to stimulate a discussion about meaningful evaluation of prognostic performance. These discussions are followed by a brief description of prognostics metrics proposed recently, their applicability, and limitations. This paper does not intend to suggest any metrics in particular rather highlights important aspects that must be covered by any performance evaluation method for prognostics.

1. INTRODUCTION

The demand for engineering systems with sophisticated functionality, high safety levels, low environmental footprint, and other requirements is accompanied by increasing cost to build and operate these systems. Besides increased manufacturing cost, it is the mitigation of operational disruptions caused when hardware or software break down that are driving up life-cycle cost and affecting operational availability. The malfunction of just a small part can seriously degrade the utility of a large portion of a complex system – or cause it to seize performing its primary function altogether. To counteract that, operators and manufacturers are increasingly looking towards system health management as a mechanism to actively deal with changing performance characteristics of individual components. This is accomplished by assessing the state of health of the system components, estimating their remaining useful life, and by initiating mitigating action that will either prevent the breakdown, minimize downtime, avoid unscheduled maintenance, or result in similar results that minimize life-cycle cost of the system. At the core of system health management is Prognostics, the method by which remaining useful life of a component (or system) is estimated. The ability to predict future events, conditional on anticipated usage and environmental conditions, is the Achilles heel of System Health Management. It is therefore not surprising that considerable attention has been given to this technology in the last few years. Indeed, the term "prognostics" has been used by various practitioners in any context that has a predictive element, not all of which actually result in estimation of remaining life. In the first part of this paper, the different instantiations of life prediction are reviewed in the context of methods that are based on fleet-level and unit-based life prediction and the term "Prognostics" is clarified.

An indispensable element in maturing prognostics is the ability to measure the performance of a prognostic algorithm. Traditional metrics that are, for example, used for diagnostics do not capture the unique characteristics of prognostics. Since the discipline is still young, new metrics are emerging that each measure specific features of prognostics. The second

part of this paper explores the most important metrics that have emerged. The paper also discusses general considerations when evaluating Prognostics. While assessing and ranking one method over another, it is important to pick metrics that evaluate the same components and do not, for example, penalize one algorithm (but not another) for poor quality of external inputs (such as noisy or missing data, inadequate domain models, etc.). Furthermore, the metrics should consider evaluating various aspects of a prediction that are useful towards decision making, such as time to prediction or confidence in prediction. Finally it is important to consider what the performance is being measured against. In online applications where it may not be possible to know the ground truth, it is difficult to measure accuracy aspects of performance because the failure has not yet happened (and hopefully will not happen when human life or costly assets are at stake) (Engel, Gilmartin, Bongort, & Hess, 2000). However, even in offline cases where ground truth is established through prior experiments, it may not be the plausible course of action to compare the predictions against one outcome (realization) of an, otherwise stochastic, process in light of several sources of uncertainties.

The paper concludes with a discussion of the path ahead for Prognostics. Specifically, the following issues are considered in detail:

- What does prediction performance mean in different application contexts?
- What are different components of algorithms that need to be evaluated and compared in prognostics applications?
- What are various assessment approaches that are currently used and how to interpret the results?
- What are lacking issues that need to be considered?

2. CONSIDERATIONS IN PERFORMANCE EVALUATION

2.1. Attributes of Prediction Performance – Correctness, Timeliness, and Confidence

The essence of a meaningful prediction lies in three key attributes that are important to achieve regardless of the prediction method used. These key attributes are – correctness (accuracy and precision), timeliness, and confidence in a prediction. It should be noted that attributes as defined here are not metrics themselves but a set of properties that define performance of a prediction algorithm. Suitable metrics can be defined to measure and quantify these attributes as discussed in latter sections.

Correctness: By definition performance evaluation refers to the notion of assessing correctness of a system output with respect to its desired specification. Prediction outputs are generally understood to be in the form of probability density functions due to inherent uncertainties involved. Hence the notion of correctness translates into accuracy and precision of the predicted distributions. Accuracy is a measure of deviation of a prediction output from measured, observed, or inferred ground truth. Specifically the prediction accuracy is a quantitative measure of error between the predicted end-of-life and the observed end-of-life of the monitored component/system. Several metrics can be used to define prediction accuracy such as but not limited to those listed in (Saxena, Celaya, Balaban, Goebel, Saha, Saha, & Schwabacher, 2008). Precision on the other hand is a measure of spread of a distribution. By definition (precision = [standard deviation]$^{-1}$) narrower distributions are considered more precise. When estimating a single point, ideal precision would be infinite if accuracy is 100%. However, it must be kept in mind that higher precision (or narrower distribution) is not always better. More than a decade ago Engel et al. (2000) explained the paradox in prognostics - "The more precise the remaining life estimate, the less probability that this estimate will be correct". Furthermore, it was analytically shown by (Sankararaman & Goebel, 2013) that the end-of-life point (or the RUL) is stochastic by nature. Therefore, a prognostics algorithm should estimate a probability distribution function and not just the observed single instance of a failure. However, the ideal value for precision of a predicted distribution would be to match the precision of ground truth distribution. In other words, arbitrarily narrow distributions could lead to risky decisions, just as arbitrarily wide distributions lead to larger ambiguity (or less confidence) in a prediction.

Timeliness: This refers to the time aspects related to availability and usability of predictions. It measures how quickly a prediction algorithm produces its outputs, in comparison to the failure effects that they are mitigating.

Prediction Horizon: The measure of how early, before the actual failure event, a prediction system produces a correct (w.r.t. specifications) prediction of end-of-life to be able to implement an actionable decision and response as part of the health management activity. For prognostics it is measured as Prognostic Horizon or Prognostic Distance at the time a prediction is made (Johnson, Gormley, Kessler, Mott, Patterson-Hine, Reichard, & Scandura Jr, 2011; Saxena, Celaya, Saha, Saha, & Goebel, 2010).

Prediction Response Time*:* The measure of how quickly the prognostic function produces a correct output, from a given set of system measurements. It includes the time it takes for an algorithm to converge to a reasonable performance level.

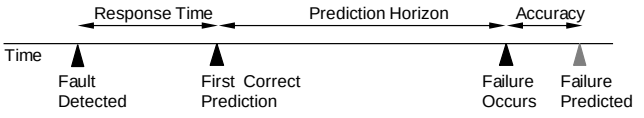


Figure 1. Correctness and timeliness attributes of prediction performance.

Confidence: It is a measure of trust (or conversely, the measure of uncertainty) in a prediction method's output. It is generally viewed in several related but different contexts. In predicting end-of-life for a unit confidence is expressed as probability of failure at any given time computed from a given failure-time distribution. From a decision making point of view is it is expressed through precision of the predicted distribution, i.e. more precise distributions lead to higher confidence and less ambiguity for decision making (Engel et al., 2000). Similarly confidence is also associated to the notion of risk of failure with respect to the time an action is taken. In the broader context of validation confidence is expressed as trust in a prediction method based on stability of predictions over time through sensitivity and robustness measures (Guan, Jha, Liu, Saxena, Celaya, & Goebel, 2010; Johnson et al., 2011). These measures are evaluated with respect to factors that directly affect predictions such as data quality (amount of data, sampling rates, noise levels, etc.), model quality (granularity of models, correctness, adaptability, etc.), accuracy of priors, etc.

The three performance attributes as described above are the most important ones from prognostics point of view. There are several metrics that can be used to assess each of these attributes, however, the important message here is that prognostic performance evaluation must account for all three of these and which specific metrics are used depends on several other factors as discussed in further sections.

2.2. Type of Prediction Method

Within the Health Management (HM) community there are several different interpretations of what is meant by the term prognostics. Although all interpretations involve some type of predictions about system's health the basis for such predictions is very different. This paper acknowledges the significance of all prediction methods but at the same time considers *Prognostics* strictly as condition based prediction methods. It is argued that depending on the type of prediction method and the data used to make these predictions the metrics to evaluate prediction performance should be slightly different. As discussed above in Section 2.1, at its core prediction performance is characterized by three attributes namely, *Correctness, Timeliness,* and *Confidence*, although the specific metrics that measure these could differ from each other in different cases.

A classification of various prediction methods was proposed in (Coble Jamie Baalis, 2010). While the author tried to classify these methods into well-defined categories, there is often a fuzzy boundary where a method may fall into one category or the other. Furthermore, it can be observed that in that classification one method follows naturally from another as one moves from predictions based on information from a fleet towards using information from a single specific unit. A brief definition for each is provided here for readability, but a more detailed description and some examples can be found in (Coble Jamie Baalis, 2010).

Type-I or *Reliability-based Prediction* methods predict component failure time based on statistical models fit to lab testing data or historical failure data. These methods are not considered prognostic methods in a strict sense but are the basis for much of how the assets have been maintained traditionally. Predictions are expressed in terms of Mean-Life metrics such as Mean Time Between Failures (MTBF) and many other variations expressing observed failure rates (Saxena & Roemer, 2013). Theoretically speaking it is possible to make life predictions for a specific unit through models used in these methods and assess correctness based on actual end of life, the performance is likely to be within expectations only when predictions for a large number of similar units is aggregated. Therefore, aggregate error and precision based metrics are generally used. Notions of timeliness do not quite apply here as predictions can be made at any time as they are based on historical data already processed to build models. Furthermore, since these models are static and do not get updated with time, the prediction of end-of-life does not change irrespective of when in time that prediction is made. Therefore there is no notion of performance tracking as in prognostics. Confidence is usually expressed as probability of failure at any given time computed from failure-time distribution. These metrics are generally useful for operators, maintainers, designers, and policy regulators for gauging and optimizing operational performance at the fleet level. The key shortcoming of this approach is that it cannot take into account the effects of operational conditions that have a significant bearing on actual component life.

Type-II or *Damage Accumulation-based Prediction* - These models estimate the lifetime of an average component operating under a given set of usage conditions (stressors). The output of these models is a distribution of failure times due to stochastic nature of operating conditions. These models however do not rely on condition monitoring data to estimate the state of a specific system and the predictions are based on population models of failure of such systems. For performance evaluation, correctness can be measured using any of the accuracy and precision metrics drawing comparisons from the actual ground truth for a specific unit. Although predictions here tend to be more accurate than for Type-I methods, algorithms are best evaluated by aggregating performance from several units as they are still based on population models. However unlike Type-I methods, notions of timeliness become relevant here as predictions must be updated regularly to account for changes due to recent operational conditions. Therefore, most metrics for Type-III methods may be applicable with slight modifications to aggregate performance from several units. Confidence is generally expressed as probability of failure and precision based metrics, although concepts of robustness to data quality may be applicable.

Type-III or *Condition-based Prediction or Prognostics* – Prognostics is the prediction of remaining useful life of a specific component or system based on its usage history inferred from monitoring data and expected future load profile. Prognostics generally utilizes a degradation model that predicts the future states based on inputs about current system state and expected load levels (stressors) on the system. These domain specific models are generally adaptable and can be developed based on physics of failure or can be learned from run-to-failure data through data-driven methods. Since the predictions are made specifically for a given unit correctness is measured for that individual unit and aggregation over multiple units is not required. Due to the notion of runtime adaptation or learning, it is important to track the response time and consequently the prediction horizon every time a prediction is generated. Similarly the concept of online performance measurements is most relevant in these scenarios. Confidence is expressed through expressing uncertainties properly and computing probability of failure within acceptable error bounds.

Type-IV or Data Analytics-based Prediction or Predictive Analytics - Predictive analytics is a term that has surfaced recently and is often being used interchangeably with prognostics in the PHM contexts. While it does involve making predictions based on information gleaned from past usage history data, the nature of predictions itself is not exactly the same as that in prognostics. A key difference being prognostics generates a prediction over a continuous space and therefore provides exact values of RUL over a set of real values in $\mathbb{R}$. Predictive analytics is more suited towards making discretized predictions that may not be a real number but a range over $\mathbb{R}$ or a qualitative set, such as [low, medium, high]. It is different from reliability based prediction in that here the predictions are based on trends observed in a multidimensional space that includes observations from a verity of non-homogenous and often unstructured data such as time sequences of complex operational patterns, sensor data, operator observations, environmental factors, geographical features, etc. just to name a few. Here the key problem to deal with is to mine information from large datasets and identify complex patterns that have been shown to lead towards anomalies of failures through collected history data. The approaches are mostly based on a data-driven (data-mining and machine learning) methods and are employed in situations where modeling the system behavior and its interaction with the external environment including human operators is often too complex to model. Correctness in such cases is measured through metrics used in pattern classification literature such as error rates (false positives and false negatives), Confidence in a prediction is expressed through similarity ranking metrics, or probability of failure occurring.

2.3. Purpose of Performance Evaluation

Relevance of a prediction is truly defined by the purpose it serves towards meeting overall system goals. In one application performance assessment could be used to optimize system operations at run-time, in another it could be used to optimize logistics chain to improve maintenance and repair efficiencies over a longer time horizon. Actions based on predictions range from fully autonomous to human controlled. Therefore, while it is important to measure prediction performance at the algorithmic level to assess technical quality (accuracy, uncertainty handling, performance improvement over time, convergence, etc.), from a practitioner's perspective it is equally important to design metrics that measure effectiveness of predictions towards improving system performance. A classification of metrics was proposed based on their relevance to various PHM stakeholders, which showed that not all metrics are relevant to all practitioners (Goebel Kai, Saxena, Saha, Saha, & Celaya, 2011). Similarly, following hierarchy can be observed in performance metrics depending on the scope of the system within which prediction performance is measured defining the overall goal of performance evaluation.

Table 1. Hierarchy of prediction performance metrics based on scope and function.

System Scope	Goal	Metrics
Core algorithm level (software and logic)	Improve prediction algorithm performance	Algorithm performance metrics assessed during development
Implementation level (software and hardware)	Efficient design of PHM system	Computational performance metrics during system design
System level with prognostics outside the decision loop	Logistics planning	PHM effectiveness metrics at system/fleet level assessed over long periods
System level with prognostics in decision loop	Operational planning	PHM effectiveness metrics at decision control loop level assessed both at long and short terms

2.4. Sources of Errors in Prognostics

Irrespective of the overall approach taken (data-driven, model based or any combination thereof) any prognostic (condition based prediction) method consists of several components each of which must together perform well to achieve good prediction performance. As described in (Roychoudhury, Saxena, Celaya, & Goebel, 2013) a general prognostics method can be thought of being composed of at least four independent elements (data sources, domain models, implementation aspects, and a core prediction algorithm), each of which contributes to the overall

prediction performance. For instance, given a choice of a particular algorithm, the performance will additionally depend on the quality of sensors (location, resolution, sampling rates, signal-to-noise ratio, etc.), method of signal processing (information loss, feature extraction, etc.), quality of degradation model, and the ability to accurately estimate future load profile.

Generally speaking a core prognostic algorithm itself consists of steps like state estimation, state propagation, future load and uncertainty estimation, failure threshold determination, etc. Therefore, a performance evaluation method must be cognizant of which factors are being evaluated so the performance can be attributed to the right elements and not necessarily generalized to the prognostic algorithm. For reference, some examples of core algorithms and corresponding sources of errors are described below.

- Model based filtering algorithm for prediction generally consists of state estimation step followed by state propagation for prediction of RUL. Degradation models are developed based on domain knowledge about the physics of failures. Magnitude of errors in models therefore depend on quality of domain expertise. While state propagation step is the only true predictive element in these algorithm, overall performance is also affected by quality of state estimation and the estimation of future loading on the system.
- Data-driven algorithms that were compared by (Goebel K., Saha, & Saxena, 2008) used a common preprocessing step to eliminate variability due to data preprocessing and uncertainties in state estimation while comparing prediction performance of several regression algorithms. Here the degradation models are learnt from available run-to-failure data and hence errors in models here depend on quality of information available from data and the choice of data models or mappings that describe relationships between sensor observations and system states, and operational conditions and fault growth rates.
- Other pure data-driven approaches used such as in PHM08 challenge used a variety of preprocessing steps. See for instance the methods used by (Coble J. B. & Hines, 2008; Wang & Lee, 2009). These approaches bypass an explicit state-estimation step and make predictions purely based on similarity computations. Here errors depend on choice of variables used for computing similarity, similarity measure itself, and the vector length to compute similarity, for example.

While it is arguable which factors should be included as part of prognostic algorithm and which as external to the algorithm, from a PHM system level viewpoint performance of the following must be evaluated at a minimum (1) correctness of state estimator (2) correctness of assumed future loading, operating, and environmental conditions; and (3) correctness of degradation (or fault propagation) model. A more detailed discussion on this is provided in Section 3.

Furthermore, in an operational context, performance of a prognostic method can only be evaluated through overall effectiveness observed together with the decision making control loop. For example, whether the overall failure rates have gone down due to implementing of a prognostics algorithm, or whether a system was able to optimize its operation to maintain safety and maximize mission goals based on prognostics. It is important to determine which factors should be included in performance assessment, which accordingly guides the choice of specific metrics. For example, from an operational view point one is interested in the performance of overall prognostics and health management (PHM) system, but at the low level the interest lies in identifying which algorithm performs better given the same set of inputs (measurement data quality, domain models, implementation hardware, etc.), which is the focus of this paper.

2.5. Offline vs. Online Performance

Offline performance measurement generally refers to testing prediction ability of an algorithm on a dataset where failure time is precisely known as that event has already taken place. Performance is assessed based on how well a predicted estimate matches the true outcome. This, however, has limited usefulness and does not fully help when an algorithm is implemented on a real system. It is often desirable to track prediction performance to ensure that appropriate and timely decisions can be taken to benefit from advanced warnings from predictions. Therefore, online metrics are designed to track algorithm performance in real-time and predict system's RUL while the actual EOL will not be known until it actually fails or may never be known if a repair action is executed based on predicted impending failure. In the absence of availability of true failure time it becomes challenging to assess how well an algorithm is predicting at runtime and most offline performance evaluation metrics are of little or no use. While, this is still an area of active research some attempts have been made. For instance, two main approaches have been suggested. A short term fixed-*k* step ahead state prediction is generated in addition to RUL predictions. These short term predictions can then be evaluated for correctness with only a *k*-step delay and not having to wait until the failure time. Consistently good values or convergence of the correctness metrics is taken as a measure of confidence in RUL prediction performance. Similarly, other metrics such as stability (less fluctuations from one prediction to the next) of short term predictions can be used to improve confidence and usability in a decision making loop.

3. WHAT SHOULD BE MEASURED?

Several metrics have been developed and currently used for assessing prognostic performance that also account for uncertainties in predictions. It is, however, rarely discussed how distributions of predicted RUL are to be interpreted, what they should be compared to for correctness, or how to actually make such comparisons. While the role of uncertainties in RUL predictions was discussed in (Celaya, Saxena, & Goebel, 2012; Sankararaman & Goebel, 2013) this section sheds some light on the contribution of uncertainties in RUL predictions to unravel the details of what comparisons are mathematically meaningful, and how to correctly interpret various types of comparisons within a performance evaluation task.

Existing methods for performance assessment can be broadly classified as being applicable to two types of situations: (1) where the RUL of a component/system is stochastically predicted using a prognostic algorithm, and the ground truth end-of-life (that is measured after failure) is compared against the algorithm prediction; and (2) where the RUL of a component/system is stochastically predicted using a prognostic algorithm, and this prediction is compared against an ensemble of end-of-life realizations available by running multiple nominally identical components to failure; sometimes, historical run-to-failure data sets are readily available in the literature for this purpose.

While the former requires the comparison of a probability distribution to a point value, the latter requires verifying whether the run-to-failure times are samples of the predicted probability distribution. Sometimes, in the latter case, the different run-to-failure times may be used to construct a probability distribution, and therefore, it is necessary to measure the extent of agreement between the run-to-failure probability distribution and the RUL distribution predicted by the prognostic algorithm. This section explores the scientific philosophy behind these two approaches for performance evaluation, and investigates the interpretation and relevance of such comparison.

To begin with, it is necessary to understand why the prediction of a prognostic algorithm is uncertain. Sankararaman and Goebel (2013) explain that, in condition-based prognostics, all the uncertainty needs to be interpreted subjectively. In other words, the uncertainty is simply reflective of the analyst's knowledge and not related to true randomness. For example, the component/system is at a particular state at any time instant. Since this state cannot be estimated accurately, it is represented using a probability distribution. Similarly, though future loading conditions are expressed using probability distribution(s), only one realization (based on that probability distribution) would actually occur during the course of operation of the component/system. Similarly, the degradation model also predicts how the health deteriorates; though this model may be uncertain, this uncertainty is not related to physical randomness. A prognostic algorithm aims at processing all of these sources of uncertainty (state, loading conditions, and degradation model), and quantifies their combined effect by computing the uncertainty in the RUL. Thus, the uncertainty estimated by the prognostic algorithm is not (and should not be) related to true randomness, and is purely subjective in nature.

This raises the question: What is related to physical randomness? True randomness occurs while running multiple nominally identical components to failure. The material properties of these components exhibit true variability. The initial state of these components exhibits true variability. The loading conditions that these components are subjected to experience true variability. Therefore, the RUL distribution estimated by running multiple components to failure exhibits true variability. It is not really meaningful to compare this probability distribution against the probability distribution predicted by the prognostic algorithm, since the former reflects the presence of true variability (in properties and loading conditions) across multiple nominally identical components/systems, whereas the latter focuses on predicting the RUL of one particular component/system. This implies that comparing the stochastic prediction of a prognostic algorithm to historical run-to-failure data sets does not necessarily help in evaluating the performance of the algorithm, since the sources and interpretation of uncertainty underlying these two statistical distributions are completely different.

In other words, if prognostic algorithms are meant for condition-based RUL assessment, then they should predict the RUL of only the intended component/system, and hence, it is necessary to rely on the ground truth end-of-life of that particular component/system in order to evaluate algorithm performance.

The prediction of a prognostic algorithm depends on four factors:

1. Choice of degradation model and associated uncertainty
2. State estimate and associated uncertainty, at time of prediction
3. Assumed future loading conditions and associated uncertainty
4. Procedure by which the algorithm processes all the above three uncertainties, in order to compute the uncertainty in the RUL.

The first three of these four factors need to be both accurate and precise, in order to achieve the best possible performance, from the perspective of the prognostic algorithm. The fourth factor needs to be mathematically and statistically exact, without making any approximations and/or assumptions regarding the probability distribution type and parameters of the RUL.

Note that, at present, it is not possible to verify whether the first three factors accurate or check whether the predicted

uncertainty in the RUL is truly reflective of the combined effect of the different sources of uncertainty. It is necessary to directly evaluate the prediction of the prognostic algorithm by directly comparing against the ground truth RUL. The rest of this section explores how this goal can be accomplished, by analyzing what quantities can be measured, in order to evaluate prognostic algorithm performance.

3.1. Ideal, Hypothetical Scenario

Consider an engineering component/system and a particular time-instant at which the RUL needs to be predicted using a prognostic algorithm. The algorithm, first, estimates the state, in terms of a probability distribution. Assume that a degradation model is readily available. Further, the uncertainty regarding the future loading conditions is also assumed to be available.

Imagine a hypothetical scenario wherein it is possible to run the same component/system to failure multiple times. From one run to another, the properties of the component/system do not change because the same system is being used, and the initial state is also invariant. However, the loading experienced in each run is different from another run. It is unreasonable to assume that the prognostic algorithm would possess knowledge regarding the statistics of the actual future loading conditions; therefore, the assumed loading statistics may or may not be identical to the actual loading statistics. (This, in fact, is the major challenge in prognostics in comparison with several other disciplines, because future loading conditions need to be anticipated accurately, in order to predict failure.)

It is possible to test whether the observed run-to-failure times are actually realizations of the probability distribution predicted by the algorithm using statistical methods, and such a test will be indicative of the prognostic algorithm performance. Note that the prognostic algorithm is likely to overestimate the uncertainty because (1) while the true state estimate is point-valued, the algorithm only estimates a probability distribution; and (2) the degradation model adds additional uncertainty. However, (1) if these two factors are infinitely accurate and precise; (2) if the algorithm assumes loading conditions that are exactly similar to those observed in reality; and (3) if the algorithm accurately processes the different sources of uncertainty, then the probability distribution predicted by the algorithm will be exactly identical to the probability distribution of the observed run-to-failure times.

Note that this evaluation jointly evaluates all of the aforementioned four factors, i.e., even if one factor were not accurate/correct, this would be reflected as a difference between the probability distributions corresponding to prediction and observation. However, as it can be seen from the description of the scenario, such evaluation is only hypothetical because it is not possible to fail the same component multiple times, while starting from the same time-instant. Therefore, it is necessary to investigate other evaluation measures that are useful in practice.

3.2. Post End-of-Life: Point-Valued Evaluation

As mentioned at the beginning of this section, the most commonly preferred method of evaluation is to wait until the end-of-life is reached, and compare the actual run-to-failure time against the algorithm prediction. The accuracy and precision of the prediction can be estimated easily. However, such comparison is not only unfair, but, sometimes, it may lead to incorrect conclusions.

Unfairness: From the time of prediction until the time of failure, the algorithm assumes some uncertainty regarding the future loading and usage conditions. However, the observed ground truth is reflective of only one loading/usage condition, thereby implying that similar quantities are not compared.

Concluding poor performance for a good algorithm: The aforementioned unfairness can sometimes lead to concluding that a good algorithm is poor. Consider the case where an algorithm is provided future loading conditions that are completely different from the actual loading conditions. The algorithm may process the provided information accurately and compute the RUL. However, this prediction may be completely different from the observed RUL. This difference needs to be attributed only to the incorrectly assumed loading conditions and it is not reasonable to penalize the prognostic algorithm in this context.

Concluding good performance of a poor algorithm: Suppose that the prediction of the algorithm is extremely accurate and precise, with respect to the observed ground truth. Then, it cannot be inferred that the algorithm is performing well. For instance, if the true damage (expressed in terms of the states) had been overestimated, and if the degradation model depicts a slower degradation rate than reality, then, ground-truth-based evaluation may suggest that the algorithm is indeed performing well. It is generally understood that a good prognostic algorithm needs to accurately estimate the state, and if the state estimation is not accurate, then the algorithm needs to be penalized. Clearly in this case, the algorithm is not penalized.

3.3. Post End-of-Life: Informed Evaluation

It is possible to eliminate the effect of not knowing the loading condition in advance, by waiting until failure. The actual loading/usage condition experienced by the component/system can be observed, and the prediction algorithm can be provided with this information. Therefore, the algorithm prediction can be "informed" with the actual loading condition, and the informed-prediction can be computed easily. Note that, at the time of prediction, this information would not be available to the algorithm. Therefore, this procedure is only to evaluate the algorithm performance, after eliminating the effect of unknown future

loading conditions. All the other information provided to the algorithm need to be reflective of the information available to the algorithm at the time of prediction.

Similar to the traditional ground-truth-based evaluation, the informed prediction of the algorithm can be compared against the observed ground truth. Note that the former is uncertain because of uncertainty in the state estimate and the degradation model. The precision and accuracy of the prediction can be computed. It can be easily seen that this evaluation is stricter than the evaluation in Section 3.2, and this performance evaluation needs to be meet requirements. However, whether this evaluation is sufficient, is unclear at present. This is because, just as in Section 3.2, overestimate damage and underestimated degradation rates may compensate each other and lead to higher accuracy and precision.

3.4. Pre End-of-Life Evaluation

While the above described measures of evaluation focus on characterizing the effects of state estimates, future loading conditions, and degradation model, it is also necessary to check whether the algorithm is accurately processing the different sources of uncertainty. This is not related to accurately predicting the RUL, but is directly associated to the mathematical treatment of the various sources of uncertainty.

Some algorithms may average the effect of the different sources of uncertainty on the RUL, and arbitrarily calculate the variance of RUL using approximations and assumptions (Celaya et al., 2012). It is important not to underestimate or overestimate the underlying uncertainty and accurately calculate the probability distribution of RUL. The ideal approach to perform such calculation is the use of Monte Carlo simulation with a large number of samples; though this requires high computational power, this method can be used to check the performance of other algorithms that are suitable for online prediction. In other words, the probability distributions obtained using the specific algorithm and Monte Carlo simulation can be compared and any discrepancy can be quantified, in order to evaluate the performance of the algorithm, from the perspective of integrating the different sources of uncertainty.

3.5. Summary

The search of prognostic performance evaluation measures raises several important questions and concerns. There are four important critical factors that control the performance of prognostic algorithm, and it is not practically possible to individually evaluate the goodness of these factors. While testing the performance against observed ground truth seems to be the most widely used method, it is not only unfair but may lead to incorrect conclusions. The informed-prediction method eliminates the uncertainty regarding the future loading conditions, and quantifies the combined effect of state uncertainty and degradation model uncertainty on the RUL prediction. The fourth factor, i.e., whether all the sources of uncertainty are being processed and integrated accurately, can be verified by comparing the algorithm prediction against rigorous Monte Carlo simulation.

An important challenge is the inability to check whether the loading conditions assumed by the algorithm are reflective of what is expected in reality. Is it reasonable to penalize the algorithm for poor performance? Another issue is the ability to identify whether the adverse effect of two (or more) incorrectly estimated quantities jointly cancel out one another, and deceivingly suggest that the prediction is highly accurate and precise. Further research is necessary to address these issues and improve the state of the art techniques for prognostic performance evaluation.

4. STATE-OF-THE-ART ON PROGNOSTICS METRICS

Several performance metrics were proposed earlier that evaluate key attributes (correctness, timeliness, and confidence) of prognostic performance as described in Section 2.1). Specifically following four metrics were suggested –

Prediction horizon – quantifies how early a prediction algorithm can make reasonable predictions to allow maximum advance warning before an impending failure.

Alpha-lambda accuracy – specifies whether an algorithm's prediction error is within desired accuracy bounds (specified by α) at any given time (specified by λ).

Relative accuracy – quantifies the prediction error normalized by remaining component life at any given time.

Convergence – tracks the rate of improvement in prognostic algorithm's performance as time progresses.

As described in (Saxena et al., 2010) these metrics convey how prognostic performance evolves with time as end-of-life time approaches closer. These metrics also acknowledged that prognostics must account for uncertainties and that any prediction method should include a representation of uncertainties through abstractions such as, for instance, the probability distributions. These metrics are parameterized through several parameters α (accuracy modifier), β (confidence modifier), and λ (time window modifier), which must be derived as specifications for prognostics based on high level requirements. In their latter publications authors described and illustrated through an example how such a flowdown can be carried out to derive numerical specifications for these parameters (Saxena, Roychoudhury, Celaya, Saha, Saha, & Goebel, 2012). There have been other recent efforts that acknowledge the need to evaluate performance under uncertainty. Generally speaking individual efforts are driven by respective application needs, however, it appears that many research articles have been developing metrics without explicitly discussing the

interpretation of the quantities being compared, therefore largely ignoring the issues such as those discussed in Section 3.

In (Leao Bruno P, Gomes, & Yoneyama, 2011; Leao Bruno P. & Yoneyama, 2013) a Probability Integral Transform (PIT) based method is presented that evaluates whether an algorithm processes uncertainties adequately by comparing the statistics of predicted RUL distributions to a ground truth distribution obtained from several run-to-failure datasets. The advantage of this method is in that it allows comparisons of arbitrary (parametric or non-parametric) distribution types obtained from field data or experimentation to address the scenario described in Section 3.1. Since the statistical significance of the analysis depends on the number of run-to-failure test cases available, limits on values can be computed for a desired significance level to assert whether a particular algorithm processes the uncertainty (as observed through several examples) correctly in a statistical sense. Authors also proposed some graphical visualizations to express confidence bounds in such assertions. As a limitation, availability of statistically sufficient ground truth data and validity of aggregating the field data into a single histogram is always questionable for such approaches to work properly. As presented in some of the earlier works from the authors (Saxena et al., 2008; Saxena, Celaya, Saha, Saha, & Goebel, 2009b; Saxena et al., 2010) there has been a general tendency towards computing an aggregate metric score over performance of several units under test. However, in the context of condition based prognostics, where users are concerned with prognostic performance of an algorithm on specific use case, applying aggregation or averaging metrics may not be valid due to effects of different operational and loading conditions on the usage life of units included in a historical dataset.

Next, the various metrics proposed based on PIT do not address the timeliness attributes of performance as discussed in Section 2.1. In fact, unfortunately, it is still very common to find metrics that disregard the timeliness aspect of prognostic performance. In (Sharp, 2013) several averaging metrics are presented that can be considered an improvement over traditional error or variance based metrics, but suffer from same limitations that it is not technically correct to average predictions made at different times. Although, by means of a user defined weighting function this limitation is somewhat alleviated, but choosing an appropriate weighting function is another subjective proposition that makes these metrics non-standardized and difficult to implement. Metrics such as Weighted Error Bias (WEB), Weighted Prediction Spread (WPS), Confidence Interval Coverage (CIC), Confidence Convergence Horizon (CCH), and a weighted sum total of all to create a Total Score Metric (TSM) may not be as simple or intuitive as authors intended them to be.

While most of the above metrics were proposed primarily for offline evaluation of prognostic performance, there have been other works that tackle specific challenges. Much of the recent literature either focuses on incorporating uncertainties or attempts to develop methods for online performance evaluation. Some of the recently published methods are summarized in Table 2. The aspect of online performance evaluation is mostly addressed by assessing performance on short term predictions of the system state (not necessarily the end-of-life). Correctness and consistency of these predictions over time is used to assert confidence in long term RUL predictions, where there cannot be an explicit evaluation of correctness and timeliness in the absence of end-of-life ground truth. Short term correctness is measured through usual accuracy and precision metrics, and consistency is generally measured by variance between successive predictions. There is no denying the fact that these are still conceptual challenges in evaluating prognostic performance and the research community continues to work towards finding a robust solution.

ACKNOWLEDGEMENT

The funding for this work was provided by the NASA System-wide Safety and Assurance Technologies (SSAT) Project.

REFERENCES

Celaya, J., Saxena, A., & Goebel, K. (2012). *Uncertainty Representation and Interpretation in Model-based Prognostics Algorithms Based on Kalman Filter Estimation.* Paper presented at the Annual Conference of the Prognostics and Health Management Society (PHM12), Minneapolis, MN.

Coble, J. B. (2010). *Merging Data Sources to Predict Remaining Useful Life – An Automated Method to Identify Prognostic Parameters.* PhD Dissertation, The University of Tennessee, Knoxville.

Coble, J. B., & Hines, J. W. (2008). *Prognostic Algorithm Categorization with PHM Challenge Application.* Paper presented at the 1st International Conference on Prognostics and Health Management (PHM08), Denver, CO.

Engel, S. J., Gilmartin, B. J., Bongort, K., & Hess, A. (2000). *Prognostics, the Real Issues Involved with Predicting Life Remaining.* Paper presented at the IEEE Aerospace Conference, Big Sky, MT.

Goebel, K., Saha, B., & Saxena, A. (2008). *A Comparison of Three Data-Driven Techniques for Prognostics*. Paper presented at the 62nd Meeting of the Society For Machinery Failure Prevention Technology (MFPT), Virginia Beach, VA.

Goebel, K., Saxena, A., Saha, S., Saha, B., & Celaya, J. (2011). Prognostic Performance Metrics. *Machine Learning and Knowledge Discovery for Engineering Systems Health Management*, 147.

Guan, X., Jha, R., Liu, Y., Saxena, A., Celaya, J., & Goebel, K. (2010). Comparison of Two Probabilistic Fatigue

Damage Assessment Approaches Using Prognostic Performance Metrics. *International Journal of Prognostics and Health Management, 2(1)*(5), 11.

Johnson, S. B., Gormley, T., Kessler, S., Mott, C., Patterson-Hine, A., Reichard, K., & Scandura Jr, P. (2011). *System health management: with aerospace applications*: John Wiley & Sons.

Leao, B. P., Gomes, J. P., & Yoneyama, T. (2011). *Improvements on the offline performance evaluation of fault prognostics methods.* Paper presented at the Aerospace Conference, 2011 IEEE.

Leao, B. P., & Yoneyama, T. (2013). *Performance Metrics in the Perspective of Prognosis Uncertainty*. Paper presented at the Annual Conference of the Prognostics and Health Management Society (PHM13), New Orleans, LA.

Liu, S., & Sun, B. (2012). *A Novel method for online prognostics performance evaluation.* Paper presented at the Prognostics and System Health Management (PHM), 2012 IEEE Conference on.

Olivares, B. E., Muñoz, M. A. C., & Orchard, M. E. (2013). Particle-Filtering-Based Prognosis Framework for Energy Storage Devices With a Statistical Characterization of State-of-Health Regeneration Phenomena. *IEEE Transactions on Instrumentation and Measurement, 62*(2), 13.

Orchard, M. E., Tang, L., Goebel, K., & Vachtsevanos, G. (2009). *A Novel RSPF Approach to Prediction of High-Risk, Low-Probability Failure Events*. Paper presented at the Annual Conference of the Prognostics and Health Management Society (PHM09), San Diego, CA.

Roychoudhury, I., Saxena, A., Celaya, J. R., & Goebel, K. (2013). *Distilling the Verification Process for Prognostics Algorithms.* Paper presented at the Annual Conference of the Prognostics and Health Management Society (PHM13), New Orleans, LA.

Sankararaman, S., & Goebel, K. (2013, October 2013). *Why is the Remaining Useful Life Prediction Uncertain?* Paper presented at the Annual Conference of the Prognostics and Health Management Society, New Orleans, LA.

Saxena, A., Celaya, J., Balaban, E., Goebel, K., Saha, B., Saha, S., & Schwabacher, M. (2008). *Metrics for Evaluating Performance of Prognostics Techniques*. Paper presented at the 1st International Conference on Prognostics and Health Management (PHM08), Denver, CO.

Saxena, A., Celaya, J., Saha, B., Saha, S., & Goebel, K. (2009a). *Evaluating Algorithmic Performance Metrics Tailored for Prognostics*. Paper presented at the IEEE Aerospace Conference, Big Sky, MT.

Saxena, A., Celaya, J., Saha, B., Saha, S., & Goebel, K. (2009b). *On Applying the Prognostics Performance Metrics*. Paper presented at the Annual Conference of the Prognostics and Health Management Society (PHM09) San Diego, CA.

Saxena, A., Celaya, J., Saha, B., Saha, S., & Goebel, K. (2010). Metrics for Offline Evaluation of Prognostic Performance. *International Journal of Prognostics and Health Management, 1*(1), 21.

Saxena, A., & Roemer, M. (2013). *IVHM Assessment Metrics*: SAE International.

Saxena, A., Roychoudhury, I., Celaya, J., Saha, B., Saha, S., & Goebel, K. (2012). *Requirement Flowdown for Prognostics Health Management.* Paper presented at the AIAA Infotech@Aerospace, Garden Grove, CA.

Sharp, M. E. (2013). *Simple Metrics for Evaluating and Conveying Prognostic Model Performance To Users With Varied Backgrounds.* Paper presented at the Annual Conference of the Prognostics and Health Management Society (PHM13), New Orleans, LA.

Vachtsevanos, G., Lewis, F. L., Roemer, M., Hess, A., & Wu, B. (2006). *Intelligent Fault Diagnosis and Prognosis for Engineering Systems* (1st ed.). Hoboken, New Jersey: John Wiley & Sons, Inc.

Wang, T., & Lee, J. (2009). *On Performance Evaluation of Prognostics Algorithms*. Paper presented at the Machinery Failure Prevention Technology, Dayton, OH.

BIOGRAPHIES

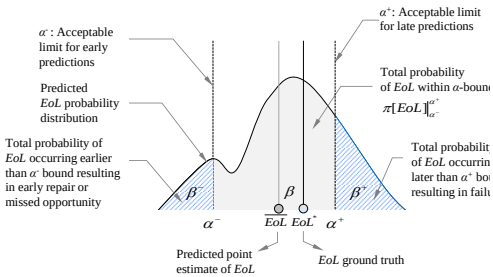
Abhinav Saxena is a Research Scientist with SGT Inc. at the Prognostics Center of Excellence of NASA Ames Research Center, Moffett Field CA. His research focus lies in developing and evaluating prognostic algorithms for engineering systems using soft computing techniques. He has co-authored more than seventy technical papers including several book chapters on topics related to PHM. He is also a member of the SAE's HM-1 committee on Integrated Vehicle Health Management Systems and IEEE working group for standards on prognostics. Dr. Saxena is the editor-in-chief of International Journal of PHM and has led technical program committees in several PHM conferences. He is also a SGT technical fellow for prognostics. He has a PhD in Electrical and Computer Engineering from Georgia Institute of Technology, Atlanta. He earned his B.Tech in 2001 from Indian Institute of Technology (IIT) Delhi, and MS Degree in 2003 from Georgia Tech. He has been a GM manufacturing scholar and is also a member of several professional societies for PHM including PHM Society, SAE, IEEE, AIAA, and ASME

Shankar Sankararaman received his B.S. degree in Civil Engineering from the Indian Institute of Technology, Madras in India in 2007 and later, obtained his Ph.D. in Civil Engineering from Vanderbilt University, Nashville, Tennessee, U.S.A. in 2012. His research focuses on the various aspects of uncertainty quantification, integration, and management in different types of aerospace, mechanical, and civil engineering systems. His research interests include probabilistic methods, risk and reliability analysis, Bayesian networks, system health monitoring, diagnosis and

prognosis, decision-making under uncertainty, treatment of epistemic uncertainty, and multidisciplinary analysis. He is a member of the Non-Deterministic Approaches (NDA) technical committee at the American Institute of Aeronautics, the Probabilistic Methods Technical Committee (PMC) at the American Society of Civil Engineers (ASCE), and the Prognostics and Health Management (PHM) Society. Currently, Shankar is a researcher at NASA Ames Research Center, Moffett Field, CA, where he develops algorithms for uncertainty assessment and management in the context of system health monitoring, prognostics, and decision-making.

Kai Goebel is the Deputy Area Lead for Discovery and Systems Health at NASA Ames where he also directs the Prognostics Center of Excellence. After receiving the Ph.D. from the University of California at Berkeley in 1996, Dr. Goebel worked at General Electric's Corporate Research Center in Niskayuna, NY from 1997 to 2006 as a senior research scientist before joining NASA. He has carried out applied research in the areas of artificial intelligence, soft computing, and information fusion and his interest lies in advancing these techniques for real time monitoring, diagnostics, and prognostics. He holds 17 patents and has published more than 250 papers in the area of systems health management.

Table 2. Recently published metrics in prognostics literature.

Metric	Description	Formula
Online Performance Evaluation		
RUL Online Precision Index (RUL-OPI) **(Orchard, Tang, Goebel, & Vachtsevanos, 2009)**	RUL-OPI quantifies and tracks the precision of predicted RUL distributions by quantifying the length of 95%confidence bounds (CI(i)) normalized by the predicted RUL ($r(i)$) at any given time instant. An algorithm with a high index (close to 1) is preferred, which indicates high precision or narrow confidence bounds.	$\mathrm{I}(\mathrm{i}) = \mathrm{e}^{-\left(\frac{\sup\{\mathrm{CI(i)}\}-\inf\{\mathrm{CI(i)}\}}{\mathrm{r(i)}}\right)}$
Dynamic Standard Deviation (DStd) (Olivares, Muñoz, & Orchard, 2013)	DStd quantifies the stability of predictions within a time window (Δ). Variance between individual predictions made within the time window is computed. The metric is normalized to a range [0,1] using the logistic function φ for easy comparisons.	$\mathrm{DStd} = \varphi\left(\mathrm{Var}\left(\mathrm{E}\left\{\mathrm{EOL} \mid \mathrm{y}_{1:\mathrm{j}}\right\}\right)_{\mathrm{j}\in\Delta}\right)$
Critical-α Performance Measure (Olivares et al., 2013)	Looking from the perspective of actionable decision making, this measure computes the critical percentile (α) of an RUL distribution that would define a Just-In-Time-Point (JITP) for that application. JITP must always occur before actual failure, and hence the value of this metric lies in interval (0,0.5] and should be maximized to avoid unnecessary conservatism in decision making.	$\alpha_{\mathrm{crit}} = \arg\max_{\alpha}\left\{\mathrm{JITP}_{\alpha\%}(\mathrm{k}_{\mathrm{pred}}) \le \mathrm{EOL}\right\};$ $\forall \mathrm{k}_{\mathrm{pred}} \in [1, \mathrm{EOL}]$
Accuracy and Precision over fixed horizon (Liu & Sun, 2012)	The accuracy metric (Ac) computes the probability mass of the predicted RUL within the acceptable α bounds and compares them to actual states realized at the end of the short horizon window. Similarly the precision (Pr) metric compares the spread (based on confidence intervals (CI)) of the predicted (P) probability density function to the true pdf (T) at the end of one horizon window. It is however not clear how the true pdf is obtained for comparison, where one would expect only a point observation from an actual event.	$\mathrm{Ac} = \int_{\alpha^-}^{\alpha^+} \varphi_{\mathrm{p}}(\mathrm{c})\mathrm{dc} \quad \text{or} \quad \sum_{\alpha^-}^{\alpha^+} \varphi(\mathrm{c})$ $\mathrm{Pr} = \begin{cases} 1 - \frac{\mathrm{CI_T} - \mathrm{CI_P}}{\mathrm{CI_T}} & \text{if } \mathrm{CI_T} \ge \mathrm{CI_P} \\ 1 - \frac{\mathrm{CI_P} - \mathrm{CI_T}}{\mathrm{CI_{max}} - \mathrm{CI_T}} & \text{if } \mathrm{CI_T} \le \mathrm{CI_P} \le \mathrm{CI_{max}} \\ 0 & \text{if } \mathrm{CI_P} \ge \mathrm{CI_{max}} \end{cases}$
Metrics Dealing with Uncertainty in Predictions		
β-criterion (Saxena et al., 2010; Saxena et al., 2012)	β-criterion specifies desired level of overlap between predicted RUL PDF and the acceptable error bounds (α^-, α^+) around observed EOL. Further extensions to β-criterion were proposed to bound probabilities of early (β^-) and late (β^+) predictions that are guided by higher level system requirements. These criteria apply to situations described in Section 3.2.	

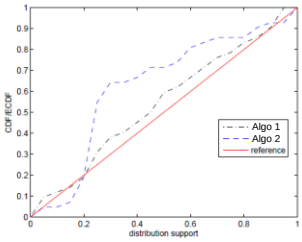
Probability Integral Transform (PIT) and PIT based metrics (Leao Bruno P. & Yoneyama, 2013)	PIT allows to assess how well a predicted distribution match the variability in the actual process. Ground truth RUL values from several run-to-failure datasets are transformed into corresponding PIT values using the cumulative distribution functions for the predicted RULs. Closer the transformed values lie to a uniform distribution $U(0,1)$ better the predicted distribution represents the observed process. To check this resemblance a graphical prognostic performance plot (PPP) was suggested with a quantitative measure prognostic quality index (q). Further, a significance level of the result can be determined based on hypothesis testing. Other such measures are also possible.	PIT: $z_i = F(x_i)$ s.t. $F(x) = \int_{-\infty}^{x} \pi(\xi)d\xi = P(X \leq x)$ and $Z = F(X) \sim U(0,1)$  $q = 1 - \frac{2}{M}\sum_{j=1}^{M} \lvert abs_j - ord_j \rvert$ to quantify deviation from the reference $U(0,1)$

Table 3. Classification of prediction methods and description of metrics typically used for performance evaluation.

Prediction Method	**Prediction Model**	**Applicability**	**Accuracy**	**Timeliness**	**Confidence**
Type I Reliability analysis based predictions	Population-based statistics data from (mostly controlled) experiments or usage history data	Predict mean life of a component. Correctness of predictions is meaningful for a fleet in general, and not for an individual unit	Mean-life metrics such as MTBF, MTBR, etc. can be predicted and then compared to observations from actual field data. These, errors in predictions can be used as a metric of accuracy. Otherwise, if maintenance actions based on these metrics are effective, then any observed change in mean-life estimates can be interpreted as a measure of effectiveness (accuracy, timeliness) of such predictions.		Probability of success metrics such as RxCy specifying x% reliability with y% confidence. E.g. R96C90 is a popular metric in automotive industry
Type II Damage accumulation model based predictions	Unit specific load history data + population based Damage accumulation model	Predict remaining life of an individual unit based on population model	Metrics like alpha-lambda accuracy and relative accuracy quantify correctness of prognostic algorithms (Saxena et al., 2010)	Prediction horizon, and lambda, the time window modifier, based metrics assess timeliness aspects of prognostics	β-criterion (Saxena, Celaya, Saha, Saha, & Goebel, 2009a) assesses confidence in prediction correctness, Robustness (Guan et al., 2010) and sensitivity metrics (Vachtsevanos, Lewis, Roemer, Hess, & Wu, 2006) assess confidence via offline analysis
Type III Condition based predictions - Prognostics	Unit specific degradation model (data-driven or physics based), load history, and condition monitoring data.	Predictions customized for individual unit by learning specific individual behavior			
Type IV Data Analytics based predictions – Predictive analytics	Rich set of data from multiple units in a variety of operating conditions + analytical data model for pattern matching	Predictions for individual unit based on rich operational history data	Classification error rate metrics (such as false positives, false negatives), aggregate error metrics (such as MAPE, MSE, MAD, etc) to evaluate predictions on multiple units.	Timeliness may be expressed by length of history sequence considered for accurate predictions.	Similarity scores between two high dimensional history vectors establish confidence. Similarity metrics such as precision and recall are often employed