Evaluation of Remaining Useful Life Prediction Algorithms in the Absence of Run-to-Failure Ground Truth Data

Indranil Roychoudhury[1], Prasham Sheth[2], and Taoufik Wassar[3]

[1, 3] *SLB, Sunnyvale, California, USA*
iroychoudhury@slb.com, psheth2@slb.com

[2] *SLB, Houston, Texas, USA*
twassar@slb.com

ABSTRACT

Accurate evaluation of Remaining Useful Life (RUL) prediction algorithms is fundamental to the deployment of Prognostics and Health Management solutions. However, for critical industrial assets with extended operational lifespans, run-to-failure ground truth data is typically not available. Preventive maintenance intentionally precludes failure events, creating a fundamental challenge of assessing prognostic accuracy without observing actual end-of-life. This paper presents an algorithm-agnostic framework for the continuous online evaluation of RUL predictions in the absence of run-to-failure data. The innovation is a retrospective methodology that treats the asset's current sensor state as a *pseudo ground truth*, enabling the evaluation of whether past predictions correctly anticipated the trajectory leading to the present condition. The framework includes two evaluation modes: (1) *Measurement-based evaluation* that assesses past sensor forecast accuracy against current observations, and (2) *RUL-based evaluation* that treats the current sensor value as a virtual degradation threshold and evaluates whether past RUL estimates correctly predicted the time to reach the present condition. The RUL-based evaluation adapts the well-established α–λ accuracy framework (Saxena, Celaya, et al., 2008) by replacing the unknown end-of-life with the current time as a pseudo ground truth reference, enabling continuous online assessment without failure observations. Individual prediction verdicts are aggregated using configurable weighting schemes into a single Service-Level Indicator suitable for performance monitoring. Experimental results across several industrial systems demonstrate the framework's generalizability across diverse degradation mechanisms, sensor modalities, and prediction algorithms. The framework requires *only* historical sensor measurements and RUL predictions at different times.

1. INTRODUCTION

Prognostics and Health Management (PHM) has emerged as a critical discipline for ensuring the reliability, safety, and cost-effectiveness of industrial assets (Goebel et al., 2017). At its core, a PHM system aims to predict the Remaining Useful Life (RUL) of a component or system, i.e., the time remaining before it can no longer perform its intended function within acceptable operational bounds. Accurate and timely RUL predictions enable operators to transition from reactive or scheduled maintenance strategies to proactive condition-based maintenance paradigms, thereby reducing unplanned downtime, optimizing spare-parts inventory, and improving overall asset utilization.

Over the past two decades, RUL prediction methodologies have included physics-based models (Lei et al., 2016), data-driven approaches leveraging deep learning architectures (Ma & Mao, 2021; Zhang et al., 2017), and hybrid methods that integrate domain knowledge with statistical learning (Chao et al., 2020). Regardless of the modeling paradigm, a fundamental question persists: *How can one rigorously evaluate the performance of an RUL prediction algorithm?*

The standard approach to evaluating prognostic algorithms relies on the availability of *run-to-failure* data, i.e., complete degradation trajectories from nominal operation through functional failure (Saxena, Celaya, et al., 2008). Well-known benchmark datasets, such as the NASA C-MAPSS turbofan engine degradation dataset (Saxena, Goebel, et al., 2008), provide precisely this type of data and have been widely used to assess prognostic performance via metrics such as Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), scoring functions, and the alpha-lambda (α-λ) accuracy metric (Saxena, Celaya, et al., 2008), (Saxena et al., 2010). These metrics share a common prerequisite: knowledge of the *true* end-of-life (EoL) or, equivalently, the *true* RUL at every prediction instant, since RUL $=$ EoL $-$ Current Time.

In many real-world industrial settings, however, this prerequisite is fundamentally unmet, since these systems may degrade very slowly, particularly when preventive maintenance is performed, resulting in run-to-failure trajectories being scarce or entirely absent. This creates a significant practical challenge: PHM solution providers deploying RUL prediction algorithms on such assets cannot straightforwardly demonstrate the accuracy of their predictions to end users. When a predictive maintenance algorithm forecasts that a membrane will require replacement in eighteen months, neither the provider nor the operator can validate that prediction until the membrane actually degrades to the point of replacement, a state it may never reach if preventive maintenance is performed first. Moreover, for deployed systems, evaluation must be performed continuously in real time, rather than offline.

To address the challenges outlined above, this paper presents a novel, generalizable framework for evaluating RUL prediction algorithms in the absence of run-to-failure ground truth data. The central insight is that, although the true EoL is unknown, the asset's *current* observed sensor state is known with certainty and can serve as a *pseudo ground truth* reference. By "looking back" from the present, we can retrospectively assess whether past predictions correctly anticipated the trajectory leading to the current condition, enabling continuous online evaluation without requiring a failure event.

At each evaluation time t, the evaluation framework retrieves the set of predictions issued at earlier times within a configurable look-back window and assesses each prediction against what is now known at time t. Two complementary evaluation modes are applied:

1. *Measurement-based Evaluation:* At each evaluation time t, the actual sensor measurements are compared to the predictions made by the RUL prediction algorithm for these sensor values at earlier time steps. Predictions that fall within a user-specified error tolerance are deemed acceptable; those that do not are flagged as unacceptable. This mode directly assesses the forecasting accuracy of the underlying predictive model and provides an indicator of model drift, independent of the specific RUL computation methodology.

2. *RUL-based Evaluation:* The current sensor value is treated as a virtual degradation threshold, i.e., RUL $= 0$ at the current time t. For each past prediction time, the framework determines how well the RUL prediction made at that time estimated the duration required to reach the asset's present condition. By deriving a retrospective *pseudo* RUL from the temporal distance between the prediction time and the current evaluation time, predictions can be assessed against this pseudo ground truth using an error tolerance analogous to the α–λ metric (Saxena, Celaya, et al., 2008).

The binary acceptable/unacceptable verdicts from individual past predictions are aggregated into a single performance score using one of five configurable weighting schemes: *simple majority*, *linear*, *nonlinear*, *exponential*, and *custom*. These schemes control the temporal emphasis profile, allowing practitioners to prioritize recent predictions over older ones. The weighted aggregate is reported as a Service-Level Indicator (SLI), a single scalar score summarizing the RUL prediction algorithm's performance over a configurable look-back window. Separate SLIs can be computed for measurement-based and RUL-based evaluation modes. The underlying assumptions for this approach include monotonic degradation, stationarity of degradation dynamics, and measurement reliability.

Experimental validation across four distinct industrial systems spanning different domains (e.g., oil and gas processing, such as coalescer filters, hot-oil heaters, and acid gas separation membranes; and power generation, such as power-unit bushings) demonstrates the framework's generalizability across monotonically increasing and decreasing degradation trends, diverse sensor modalities (pressure, conductance, thermal output, chemical flux), and multiple prediction algorithms. Furthermore, existing evaluation metrics that need ground-truth run-to-failure data can be derived from the proposed framework using the pseudo ground truth.

The remainder of this paper is organized as follows. Section 2 reviews related work on RUL prediction and PHM algorithm evaluation. Section 3 formulates the evaluation problem and presents the proposed methodology, including the measurement-based and RUL-based evaluation modes, the weighting schemes, and the computation of the SLIs. Section 4 describes the experimental setup and presents results from the application of the framework to four industrial systems. Finally, Section 5 summarizes this research and discusses directions for future work.

2. RELATED WORK

This section reviews the literature pertinent to RUL prediction methodologies, prognostics evaluation metrics, and the challenges of deploying PHM solutions in industrial environments where run-to-failure data is absent. The prediction of RUL has been approached from three broad paradigms: *model-based*, such as physics-of-failure models with state estimation techniques such as particle filters and Kalman filters (Huang et al., 2015; Lei et al., 2016; Si et al., 2011), *data-driven*, such as machine learning methods including deep neural networks, LSTMs, and transformers (Chen et al., 2021; Ma & Mao, 2021; Wang et al., 2020; Zhang et al., 2017), and *hybrid* approaches that combine physics-based constraints with statistical learning (Chao et al., 2020). Transfer learning and adaptive methods have also been developed to address domain shift and evolving system dynamics (Cheng et al., 2023; da Costa et al., 2020; Ding et al., 2022; Forgione et al., 2023;

Sheth & Roychoudhury, 2024).

Since the focus of this paper is on the *evaluation* of RUL predictions rather than on prediction algorithm design, we do not review these methods in detail. However, regardless of the algorithm employed, all RUL prediction algorithms ultimately produce forecasts that require rigorous evaluation. This challenge becomes particularly acute in the absence of failure data. The framework proposed in this paper is compatible with any of the above prediction paradigms, requiring only that sensor measurements and prognosis outputs are available.

The evaluation of prognostic algorithms has been extensively researched. However, nearly all existing metrics are predicated on the availability of run-to-failure ground truth data. We review the most prominent approaches here to establish context and highlight the gap addressed by the present work.

Traditional error metrics such as Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) are routinely used to quantify the deviation between predicted and true RUL values (Lei et al., 2018). These metrics compute aggregate statistics over a set of RUL predictions by comparing each predicted RUL value $\widehat{\mathrm{RUL}}(t)$ against the true RUL at time t, which can only be determined if the actual EoL time is known.

The α–λ accuracy metric, introduced by Saxena, Celaya, et al. (2008), is a widely-adopted standard for prognostics evaluation. At prediction time t_p, the predicted RUL is $\widehat{\mathrm{RUL}}(t_p)$ and the true RUL is $\mathrm{RUL}_{\mathrm{true}}(t_p) = t_{\mathrm{EoL}} - t_p$, where t_{EoL} is the known EoL time. A prediction is accurate if it falls within proportional error bounds:

$$(1-\alpha)\mathrm{RUL}_{\mathrm{true}}(t_p) \leq \widehat{\mathrm{RUL}}(t_p) \leq (1+\alpha)\mathrm{RUL}_{\mathrm{true}}(t_p), \tag{1}$$

where $\alpha \in (0,1)$ is the accuracy tolerance (e.g., $\alpha = 0.2$ allows $\pm 20\%$ error). Geometrically, these bounds form a cone in the (time, RUL) plane that narrows as the asset approaches EoL. The parameter λ specifies the fraction of asset life at which evaluation begins, enabling assessment at different degradation stages. Saxena et al. (2010) extended this framework to include timeliness, convergence, and prediction spread, while Goebel et al. (2012) introduced complementary concepts such as the *prognostic horizon* (the earliest time at which predictions become consistently accurate) and *convergence rate* (how prediction error decreases as EoL approaches). Saxena et al. (2010) present relative accuracy and cumulative relative accuracy, which normalize prediction errors by the true RUL at each evaluation point, providing scale-invariant assessments. These are particularly useful for comparing predictions across assets with different total lifetimes.

In prognostics competitions, such as that presented by Saxena and Goebel (2008), the scoring function sometimes penalizes late predictions more heavily than early ones, reflecting the asymmetric cost of over- versus under-estimation of RUL in safety-critical applications. Variants of this asymmetric loss function have been adopted in numerous subsequent studies as a standard benchmark metric (Ramasso & Saxena, 2014).

Work on partial degradation trajectories has examined how to assess prognostic models when only incomplete degradation histories are available. Sikorska, Hodkiewicz, and Ma (2011) surveyed prognostic modeling options across industries and highlighted that censored or truncated trajectories are the norm rather than the exception in practice. However, such approaches still require some portion of the degradation-to-failure trajectory to be observed, which may not be available for preventively maintained assets.

Online validation and forecasting-based evaluation strategies have been proposed for continuously monitoring model performance during deployment. Zio (2022) identified the validation of prognostic models under operational conditions as a key open challenge, noting that short-horizon forecast verification, such as comparing near-term predictions against subsequently observed measurements, provides a practical but limited form of online assessment. Unlike such short-horizon checks, which evaluate only measurement-level accuracy, the proposed framework additionally assesses RUL-level performance through the pseudo ground truth construction.

Uncertainty-aware evaluation methods have gained attention as probabilistic prognostics become more prevalent. Sankararaman and Goebel (2015) emphasized the importance of quantifying and evaluating uncertainty in RUL predictions, proposing metrics that assess the calibration and informativeness of prediction distributions. More broadly, Gneiting, Balabdaoui, and Raftery (2007) established the concepts of calibration and sharpness as fundamental properties of probabilistic forecasts. While the present framework uses deterministic error thresholds rather than probabilistic evaluation criteria, extending the approach to incorporate prediction interval calibration represents a natural future direction (see Section 5).

These related efforts address important aspects of the evaluation challenge. However, to our knowledge, no existing framework provides a unified methodology for continuous, quantitative assessment of both measurement-level and RUL-level prediction performance in the complete absence of failure data. The present work addresses this gap by introducing a retrospective pseudo ground truth methodology that enables online evaluation without requiring any failure observations.

The transition of PHM algorithms from research settings to industrial operations introduces evaluation challenges that the academic literature has only partially addressed (Lee et al., 2014; Vachtsevanos, Lewis, Roemer, Hess, & Wu, 2006). In industrial deployments, PHM solutions are often integrated into Service-Level Agreements (SLAs) between providers and asset operators. SLAs formalize performance commitments,

and SLIs track adherence to them. In software reliability engineering, SLIs and associated Service-Level Objectives (SLOs) are well-established constructs for monitoring system health (Beyer, Jones, Petoff, & Murphy, 2016). However, analogous frameworks for prognostics performance monitoring remain largely absent.

Several practical barriers contribute to this gap. First, the long operational lifetimes of many industrial assets (spanning years or decades) mean that failure events are infrequent, making it impractical to accumulate sufficient run-to-failure data for offline validation (Sikorska et al., 2011). Second, condition-based and preventive maintenance strategies, which are precisely the regimes that benefit most from accurate RUL predictions, intentionally prevent failures from occurring, thereby eliminating the very ground-truth data needed for traditional evaluation (Jardine et al., 2006).

These challenges highlight the need for evaluation methodologies that can assess prognostic performance continuously during deployment, without waiting for a failure event to provide retrospective validation, motivating the look-back framework proposed here.

3. APPROACH

This section presents the complete methodology for evaluating RUL prediction algorithms in the absence of run-to-failure ground truth data. The key insight is that the *current* observed sensor state can serve as a retrospective "pseudo ground truth" reference, enabling continuous online assessment of past predictions without requiring a failure event. The framework is algorithm-agnostic and applies to any RUL prediction method, physics-based, data-driven, or hybrid, as long as historical sensor measurements and prognosis outputs are available.

3.1. Problem Formulation

The evaluation framework operates on two categories of input data: *sensor data* collected from the monitored asset, and *prognosis data* produced by the RUL prediction algorithm under evaluation.

Sensor data: Consider a system instrumented with $X \in \mathbb{N}$ sensors denoted $\mathrm{SN}_1, \mathrm{SN}_2, \ldots, \mathrm{SN}_X$. Measurements are recorded at discrete time steps $T_1, T_2, \ldots, T_N$, where $T_j \in \mathbb{R}_{\geq 0}$ and $T_1 < T_2 < \cdots < T_N$. For sensor SN_i at time T_j, we denote the measured value as $z_i(T_j) \in \mathbb{R}$, representing a scalar physical quantity (e.g., pressure, temperature, etc.). The complete sensor vector at time T_j is $z(T_j) = [z_1(T_j), \ldots, z_X(T_j)]^\top \in \mathbb{R}^X$. An optional health indicator threshold $\theta_{\mathrm{meas}}(T_j) \in \mathbb{R}$ may also be stored at each time step.

Prognosis data: The RUL prediction algorithm produces forecasts at prediction times $T_1^P, T_2^P, \ldots, T_Q^P$, where $T_j^P \in \mathbb{R}_{\geq 0}$ and typically $\{T_1^P, \ldots, T_Q^P\} \subset \{T_1, \ldots, T_N\}$ (predictions are issued at a subset of measurement times). At each prediction time T_q^P, the algorithm generates a forecast of future system behavior. The following quantities are stored for each sensor SN_i:

- The predicted sensor value $\hat{z}_i(t \mid T_q^P) \in \mathbb{R}$ for future time $t > T_q^P$, representing the algorithm's best estimate of what sensor i will measure at time t based on information available up to T_q^P,
- The prediction variance $\sigma_i^2(t \mid T_q^P) \in \mathbb{R}_{\geq 0}$ quantifying forecast uncertainty (optional, model-dependent),
- The predicted RUL value $\widehat{\mathrm{RUL}}(T_q^P) \in \mathbb{R}_{\geq 0}$, representing the estimated time remaining until the system crosses a predefined failure or maintenance threshold,
- The RUL prediction variance $\sigma^2_{\mathrm{RUL}}(T_q^P) \in \mathbb{R}_{\geq 0}$ quantifying prognostic uncertainty (optional, model-dependent),
- The health indicator value $\mathrm{HI}(T_q^P) \in \mathbb{R}$ (optional, algorithm-specific).

The evaluation framework takes the sensor and prognosis data as inputs and produces three outputs at any evaluation time t:

1. A *measurement-based evaluation* verdict assessing the accuracy of past sensor value predictions,
2. An *RUL-based evaluation* verdict assessing the accuracy of past RUL predictions, and
3. A *Service-Level Indicator* (SLI) that summarizes the overall performance of the RUL prediction algorithm over a configurable time horizon.

A central concept in this evaluation framework is the *look-back window*, which defines how far into the past we examine predictions when assessing performance at the current time t. This window can be specified in two equivalent ways:

1. *Count-based:* The number of past prediction data points to examine, denoted N_{lookback}.
2. *Time-based:* A temporal duration to look back over, denoted W_{lookback}.

If the RUL prediction algorithm generates forecasts at every τ time units, these two representations are related by:

$$N_{\mathrm{lookback}} = \frac{W_{\mathrm{lookback}}}{\tau}. \tag{2}$$

The framework uses three independent look-back windows, each serving a distinct evaluation purpose:

- $W_{\mathrm{lookback,meas}}$ (equivalently $N_{\mathrm{lookback,meas}}$): the window for measurement-based evaluation,
- $W_{\mathrm{lookback,RUL}}$ (equivalently $N_{\mathrm{lookback,RUL}}$): the window for RUL-based evaluation,
- $W_{\mathrm{lookback,SLI}}$ (equivalently $N_{\mathrm{lookback,SLI}}$): the window for computing the Service-Level Indicator.

At evaluation time t, let $\mathcal{T}_{\text{meas}}$, $\mathcal{T}_{\text{RUL}}$, and $\mathcal{T}_{\text{SLI}}$ denote the sets of past prediction times that fall within the respective look-back windows. For example,

$$\mathcal{T}_{\text{meas}} = \big\{t' \in \{T_1^P, \ldots, T_Q^P\} \;\big|\; t - W_{\text{lookback,meas}} \le t' \le t\big\}. \tag{3}$$

Separating these three windows provides operational flexibility: for instance, a short measurement look-back window may capture recent forecast quality, while a longer SLI window may provide a more stable long-term performance summary.

3.2. Measurement-based Evaluation

The measurement-based evaluation assesses how accurately the RUL prediction algorithm forecast sensor values at the current time t. Instead of requiring knowledge of the actual time of failure, this evaluation uses the *currently observed sensor value* $z(t)$ as the reference. The evaluation defines symmetric error bounds around this value:

$$\begin{aligned} z^+(t) &= z(t)(1 + \alpha_{\text{meas}}), \\ z^-(t) &= z(t)(1 - \alpha_{\text{meas}}), \end{aligned} \tag{4}$$

where $\alpha_{\text{meas}} \in (0, 1)$ is the user-specified accuracy tolerance (e.g., $\alpha_{\text{meas}} = 0.1$ for $\pm 10\%$ error).

For each past prediction time $t' \in \mathcal{T}_{\text{meas}}$, the prediction is classified as *acceptable* if it falls within the error bounds:

$$\mathbb{1}_{\text{acc}}^{\text{meas}}(t', t) = \begin{cases} 1, & \text{if } z^-(t) \le \hat{z}(t \mid t') \le z^+(t), \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

Equivalently, $e_{\text{meas}}(t', t) = |\hat{z}(t \mid t') - z(t)|$ expresses the prediction error, and the acceptability indicator becomes:

$$\mathbb{1}_{\text{acc}}^{\text{meas}}(t', t) = \begin{cases} 1, & \text{if } e_{\text{meas}}(t', t) \le \alpha_{\text{meas}} \cdot |z(t)| \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

The individual acceptability indicators over the look-back window are aggregated into a single verdict score using a weighted average:

$$\text{Eval}^{\text{meas}}(t) = \sum_{t' \in \mathcal{T}_{\text{meas}}} w_{t'}^{\text{meas}} \cdot \mathbb{1}_{\text{acc}}^{\text{meas}}(t', t), \tag{7}$$

where $\{w_{t'}^{\text{meas}}\}$ are normalized weights with $\sum_{t'} w_{t'}^{\text{meas}} = 1$. The weighting schemes are described in Section 3.4.

The verdict is then converted to a binary label:

$$\text{label}^{\text{meas}}(t) = \begin{cases} \text{“good”}, & \text{if } \text{Eval}^{\text{meas}}(t) \ge 0.5, \\ \text{“bad”}, & \text{otherwise.} \end{cases} \tag{8}$$

Intuitively, if a weighted majority of past predictions for the current sensor value fall within the acceptable tolerance band, the algorithm's measurement prediction capability is deemed satisfactory at time t.

When the system has multiple sensors ($d > 1$), the measurement-based evaluation is performed independently for each sensor SN_i, $i = 1, \ldots, X$. Each sensor produces its own verdict $\text{Eval}_i^{\text{meas}}(t)$. These per-sensor verdicts can be examined individually or combined (e.g., via averaging or by requiring all sensors to pass) depending on the operational context.

Note that the acceptability criterion in Eq. (5) uses a deterministic error threshold and does not explicitly incorporate prediction uncertainty, even when such estimates are available. Extensions are discussed in Sections 4.7 and 5.

3.3. RUL-based Evaluation

The RUL-based evaluation assesses how well the RUL prediction algorithm estimated the time remaining until the system reached its *current* condition. This evaluation mode is a direct adaptation of the classical α–λ accuracy metric (Saxena, Celaya, et al., 2008), with one critical substitution: the unknown true end-of-life time t_{EoL} is replaced by the known current evaluation time t, and the true RUL $t_{\text{EoL}} - t'$ at each past prediction time t' is replaced by the elapsed time $t - t'$. This substitution preserves the geometric and conceptual structure of the α–λ framework, including the acceptance cone that narrows as predictions approach the reference time, while removing the dependency on failure observations. The result is an empirically verifiable evaluation criterion that can be computed continuously during operation.

At the current evaluation time t, let the evaluation threshold be $\theta_{\text{eval}}(t) = \Theta\big(z(t)\big)$, where $\Theta : \mathbb{R}^X \to \mathbb{R}$ is a threshold function that maps the current sensor vector $z(t) \in \mathbb{R}^X$ to a scalar threshold value. The choice of Θ depends on the application domain and sensor characteristics. Common choices include:

- *Arithmetic mean*: $\Theta(z) = \frac{1}{X} \sum_{i=1}^{X} z_i$, suitable when all sensors measure the same physical quantity at different locations or when a balanced view across sensors is desired.
- *Weighted combination*: $\Theta(z) = \sum_{i=1}^{X} \omega_i z_i$ with $\sum_i \omega_i = 1$, where ω_i reflect sensor importance.
- *Single dominant sensor*: $\Theta(z) = z_j$ for a primary health indicator sensor j, appropriate when one sensor directly measures the degradation phenomenon of interest.
- *Domain-specific health indicator*: $\Theta(z) = f(z)$ where f is a physics-informed function (e.g., thermal efficiency, normalized flux, resistance-based indices) that maps raw sensor values to a meaningful degradation metric.

The validity of the virtual threshold approach rests on the following assumption: *if a prognostic algorithm can accurately predict intermediate sensor states and the time required to*

reach them, it is likely to perform similarly well when predicting the actual failure threshold. This assumption is reasonable for systems exhibiting monotonic or quasi-monotonic degradation, where the underlying degradation mechanism remains consistent throughout operation.

The key observation is that, since the system has already reached the state corresponding to $\theta_{\text{eval}}(t)$ at the current time, we know with certainty the time it took to arrive here from any earlier time t'. This gives the *pseudo ground truth* RUL relative to the evaluation threshold $\text{RUL}_{\text{true}}(t', t) = t - t'$. That is, the "true" RUL from the perspective of time t' is simply the elapsed time from t' to t.

The pseudo ground truth construction relies on several key assumptions that practitioners should verify for their specific application:

1. *Monotonic or quasi-monotonic degradation*: The degradation trend (increasing or decreasing) is consistent over the evaluation window. Systems with highly oscillatory or non-monotonic sensor behavior may produce ambiguous threshold-crossing times.
2. *Stationarity of degradation dynamics*: The underlying degradation mechanism does not change abruptly during the evaluation period. Sudden changes in operating conditions, maintenance interventions, or environmental factors that alter the degradation rate may invalidate comparisons between predictions made before and after such changes.
3. *Measurement reliability*: The current sensor value $z(t)$ is an accurate reflection of the asset's true state. Sensor faults, calibration drift, or transient anomalies can corrupt the pseudo ground truth reference.
4. *Sufficient prediction cadence*: The temporal spacing τ between predictions is small relative to the degradation time scale, ensuring that the look-back window captures a representative sample of prediction behavior.

When these assumptions hold, the pseudo ground truth provides a scientifically defensible reference for prognostic evaluation. When they are violated (for instance, if a major maintenance action resets the degradation state mid-evaluation), the framework should be applied separately to each stationary degradation regime, or the window should exclude the discontinuity.

For each past prediction time $t' \in \mathcal{T}_{\text{RUL}}$, we retrieve the RUL predicted at time t' for reaching the virtual threshold $\theta_{\text{eval}}(t)$. Let this be $\widehat{\text{RUL}}(t \mid t')$. We define symmetric error bounds on the pseudo ground truth using $\alpha_{\text{RUL}} \in (0, 1)$ as the RUL accuracy tolerance:

$$\begin{aligned} \text{RUL}^{+}(t', t) &= (t - t')(1 + \alpha_{\text{RUL}}), \\ \text{RUL}^{-}(t', t) &= (t - t')(1 - \alpha_{\text{RUL}}). \end{aligned} \tag{9}$$

The acceptability indicator for each past prediction is:

$$\mathbb{1}_{\text{acc}}^{\text{RUL}}(t', t) = \begin{cases} 1, & \text{if } \widehat{\text{RUL}}(t \mid t') \in [\text{RUL}^{-}(t', t),\ \text{RUL}^{+}(t', t)], \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

The error bounds in Eq. (9) form a cone-shaped acceptance region in the (time, RUL) plane. At the earliest prediction time $t' = t_{\text{start}}$, the cone is widest, with acceptance bounds $(t - t_{\text{start}})(1 \pm \alpha_{\text{RUL}})$. As t' approaches the current evaluation time t, both the pseudo ground truth RUL and the tolerance band shrink linearly toward zero, converging to zero when $t' = t$. A prediction falling inside this cone is deemed acceptable; predictions outside are unacceptable.

This cone geometry is the well-known α–λ accuracy metric from (Saxena, Celaya, et al., 2008), with a critical distinction: the traditional α–λ cone extends from the start of the evaluation period to the true EoL time t_{EoL}, whereas the virtual threshold cone extends from the start of the look-back window to the current evaluation time t. By replacing the *unknown* t_{EoL} with the *known* current time t, the framework constructs an empirically verifiable acceptance criterion without requiring failure observations.

The RUL acceptability indicators are aggregated identically to the measurement-based case:

$$\text{Eval}^{\text{RUL}}(t) = \sum_{t' \in \mathcal{T}_{\text{RUL}}} w_{t'}^{\text{RUL}} \cdot \mathbb{1}_{\text{acc}}^{\text{RUL}}(t', t), \tag{11}$$

with normalized weights $\{w_{t'}^{\text{RUL}}\}$, and the binary label is:

$$\text{label}^{\text{RUL}}(t) = \begin{cases} \text{“good”}, & \text{if } \text{Eval}^{\text{RUL}}(t) \geq 0.5, \\ \text{“bad”}, & \text{otherwise.} \end{cases} \tag{12}$$

As with the measurement-based evaluation, the RUL acceptability criterion uses deterministic error bounds and does not incorporate the RUL prediction variance. Probabilistic extensions are discussed in Sections 4.7 and 5.

3.4. Weighting Schemes

The framework provides five weighting schemes for aggregating individual acceptability indicators into a verdict score. These schemes are used uniformly across the measurement-based evaluation, the RUL-based evaluation, and the SLI computation. The choice of weighting scheme allows practitioners to control how much emphasis is placed on recent versus older predictions.

The rationale for selecting these five schemes is to span the full spectrum of temporal emphasis profiles that arise in practice: from uniform weighting (simple majority) through progressively stronger recency biases (linear, nonlinear, exponential) to fully user-defined profiles (custom). This design reflects a key operational reality: in some deployments, long-term

consistency matters most (favoring uniform weighting), while in others, only recent performance is operationally relevant (favoring exponential weighting). Alternative formulations, such as Bayesian weighting based on posterior model confidence or information-theoretic weighting based on prediction entropy, were considered but excluded from the current framework in favor of schemes that are interpretable, require no additional model-internal information, and require no statistical expertise to configure. The custom weighting scheme provides an extension point for practitioners who wish to encode more complex temporal priorities.

Let $N = |\mathcal{T}|$ denote the number of predictions in the look-back window, and let $t'_1 < t'_2 < \cdots < t'_N$ be the ordered prediction times. Define the evaluation time vector $\mathbf{t}_{\text{eval}} = (t'_1, \ldots, t'_N)$, and let $t_{\text{start}} = t'_1$ and $t_{\text{end}} = t'_N$.

1. *Simple Majority:* All predictions are weighted equally:

$$w_{t'_k} = \frac{1}{N}, \qquad k = 1, \ldots, N. \tag{13}$$

   The verdict reduces to the arithmetic mean of the acceptability indicators, equivalent to a simple majority vote. This scheme treats historical and recent predictions with equal importance.

2. *Custom Weights:* User-provided weights $\tilde{w}_{t'_1}, \ldots, \tilde{w}_{t'_N}$ are normalized:

$$w_{t'_k} = \frac{\tilde{w}_{t'_k}}{\sum_{j=1}^{N} \tilde{w}_{t'_j}}, \qquad k = 1, \ldots, N. \tag{14}$$

   This allows domain experts to encode application-specific priorities.

3. *Linear Weights:* Weights increase linearly with prediction time:

$$w_{t'_k} = \frac{t'_k}{\sum_{j=1}^{N} t'_j}, \qquad k = 1, \ldots, N. \tag{15}$$

4. *Nonlinear Weights:* Weights are inversely proportional to elapsed time:

$$w_{t'_k} = \frac{(t - t'_k + \epsilon)^{-1}}{\sum_{j=1}^{N} (t - t'_j + \epsilon)^{-1}}, \quad k = 1, \ldots, N, \tag{16}$$

   where $\epsilon > 0$ is a small constant (e.g., $\epsilon = 10^{-8}$).

5. *Exponential Weights:* Weights grow exponentially with recency:

$$w_{t'_k} = \frac{\exp(t'_k / \Delta t)}{\sum_{j=1}^{N} \exp(t'_j / \Delta t)}, \quad k = 1, \ldots, N, \tag{17}$$

   where $\Delta t = t_{\text{end}} - t_{\text{start}}$.

For all five schemes, the verdict function has a common form. Given a sequence of binary indicators $\mathbf{a} = (a_1, \ldots, a_N) \in \{0, 1\}^N$, the verdict is:

$$V(\mathbf{a}, \mathbf{w}) = \sum_{k=1}^{N} w_k \cdot a_k, \tag{18}$$

and the performance is deemed "good" if $V \geq 0.5$ and "bad" otherwise.

The choice of weighting scheme should reflect the operational context and the characteristics of the prognostic algorithm being evaluated:

- *Simple Majority:* Use when long-term prediction stability is the primary concern, or when the algorithm is not adaptive and prediction quality is expected to remain constant over time. This scheme is appropriate for initial algorithm validation and for systems with highly stationary degradation dynamics.
- *Linear:* Use for systems with gradually improving prediction quality as more training data becomes available. This scheme provides a moderate bias toward recent performance while still considering the full historical record. Suitable for sequentially-trained models in relatively stable environments.
- *Nonlinear:* Use when recent predictions are substantially more informative than older ones, such as in adaptive algorithms that continuously update model parameters. This scheme emphasizes recent behavior while maintaining some sensitivity to historical performance.
- *Exponential:* Use for highly adaptive algorithms or for systems where operating conditions are changing over time. This scheme provides the most aggressive discounting of historical predictions and is appropriate when only the most recent behavior is operationally relevant (e.g., real-time dashboards, short-horizon decision support).
- *Custom:* Use when domain-specific knowledge suggests a particular temporal weighting profile. For example, weights could be designed to emphasize predictions made during critical operational phases, or to discount periods when sensor quality was known to be degraded.

As a general recommendation for operational deployment, the exponential or nonlinear schemes are generally preferable, as they provide responsive SLIs that reflect current algorithm behavior rather than being dominated by potentially outdated historical performance.

3.5. Service-Level Indicator (SLI)

The Service-Level Indicator provides a single, aggregated performance score that summarizes the overall reliability of the RUL prediction algorithm over a configurable time horizon. While the measurement-based and RUL-based evaluations produce per-timestamp verdicts, the SLI operates at a higher

level by aggregating these verdict labels over the SLI look-back window.

To generate the SLI at each evaluation time s within the operational history, either the measurement-based or the RUL-based evaluation produces a binary label:

$$\ell(s) = \begin{cases} 1, & \text{if } \mathrm{label}^{(\cdot)}(s) = \text{“good”}, \\ 0, & \text{if } \mathrm{label}^{(\cdot)}(s) = \text{“bad”}, \end{cases} \tag{19}$$

where $(\cdot)$ denotes either “meas” or “RUL”, depending on whether the SLI uses measurement-based or RUL-based evaluation. The choice is a user-configurable parameter.

The SLI at the current time t is computed by applying the same weighted aggregation (Section 3.4) to the verdict stream over the SLI look-back window $\mathcal{T}_{\mathrm{SLI}}$:

$$\mathrm{SLI}(t) = \sum_{s \in \mathcal{T}_{\mathrm{SLI}}} w_s^{\mathrm{SLI}} \cdot \ell(s), \tag{20}$$

where $\{w_s^{\mathrm{SLI}}\}$ are normalized weights computed using any of the five weighting schemes described in Section 3.4.

The SLI value lies in $[0, 1]$ and admits a natural interpretation:

$$\mathrm{label}^{\mathrm{SLI}}(t) = \begin{cases} \text{“good”}, & \text{if } \mathrm{SLI}(t) \geq 0.5, \\ \text{“bad”}, & \text{otherwise.} \end{cases} \tag{21}$$

An $\mathrm{SLI}(t)$ close to 1.0 indicates that the RUL prediction algorithm is consistently performing well over the look-back window, while a value close to 0.0 indicates consistently poor performance, and values near 0.5 indicate mixed performance.

The evaluation framework has a natural three-level hierarchical structure. First, each past prediction $\hat{z}(t \mid t')$ or $\widehat{\mathrm{RUL}}(t \mid t')$ is classified as acceptable or unacceptable (Eqs. 5 and 10). Then, the acceptability indicators within a look-back window are aggregated into a single verdict score $\mathrm{Eval}^{(\cdot)}(t)$ and a corresponding binary label (Eqs. 7–8 and Eqs. 11–12). Finally, the binary labels across timestamps are aggregated into the SLI (Eq. 20). This hierarchical structure ensures that the SLI is both interpretable and traceable: a poor SLI score can be decomposed into the specific timestamps and predictions that contributed to the degraded assessment. This traceability is essential for root-cause analysis and for communicating results to stakeholders.

It should be noted that the binarization at each level of the hierarchy entails a deliberate trade-off. By reducing continuous prediction errors to binary acceptable/unacceptable indicators, the framework sacrifices sensitivity to differences in error magnitude: a prediction that barely misses the tolerance bound is treated identically to one that deviates substantially. This design choice prioritizes operational interpretability and SLA-compatible reporting over fine-grained error analysis. When more nuanced assessment is needed, the continuous prediction errors and the raw RUL deviations remain available at the lowest level of the hierarchy and can be examined alongside the binary SLI to provide complementary diagnostic information.

4. EXPERIMENTS

To validate the proposed evaluation framework, we apply it to four distinct industrial systems operating in oil and gas processing and power generation domains. These systems exhibit diverse degradation mechanisms (filter plugging, insulation degradation, thermal efficiency loss, membrane permeability decline), sensor modalities (pressure, conductance, thermal output, chemical flux), and operational time horizons (200–350 days), thereby demonstrating the generality of the framework across different PHM scenarios.

4.1. Experimental Setup

The framework is validated on four industrial systems, summarized in Table 1. Each system is characterized by a gradual degradation process monitored through one or more sensor channels, and none of the datasets contain run-to-failure events as the systems were maintained or replaced before catastrophic failure occurred.

4.1.1. RUL Prediction Algorithms

The proposed evaluation framework is agnostic with respect to the underlying RUL prediction method. The RUL prediction algorithms evaluated on the four systems include Facebook’s Prophet time-series forecasting algorithm (Taylor & Letham, 2018); a kernel-based support vector regression (SVR) method that produces point forecasts without native uncertainty quantification; and a particle filter-based algorithm described in (Roychoudhury et al., 2013). Prophet provides point forecasts and prediction intervals, enabling variance-aware evaluation. When SVR is used, the prediction variance is set to zero. The framework processes the outputs of each algorithm, i.e., the predicted sensor trajectories (mean, upper, lower bounds) and the derived RUL values, identically, regardless of the algorithm that produced them.

4.1.2. Evaluation Parameters

For each case study, the following evaluation parameters are varied to assess framework sensitivity and robustness:

- Error bound on measurements (α_{meas}): The fractional tolerance for measurement predictions, tested at values of 0.10, 0.15, 0.20, and 0.30 (i.e., 10%, 15%, 20%, and 30%).
- Error bound on RUL (α_{RUL}): The fractional tolerance for RUL predictions, tested at values of 0.20, 0.30, and 0.40.
- Evaluation window length (N_{lookback}): The number of past predictions considered in each evaluation, varied from 2 to the total number of available prediction time steps.

Table 1. Summary of industrial systems used for experimental validation.

System	Sensor	Duration (days)	Degradation Direction	Threshold Crossing
Coalescer Filter	Differential Pressure (psi)	∼343	Increasing	Upper
Power Unit Bushing	Conductance (μS)	∼322	Increasing	Upper
Hot Oil Heater	Heat output (BTU/hr)	∼202	Decreasing	Lower
Acid Gas Membrane	CO_2 flux (SCFH)	∼202	Decreasing	Lower

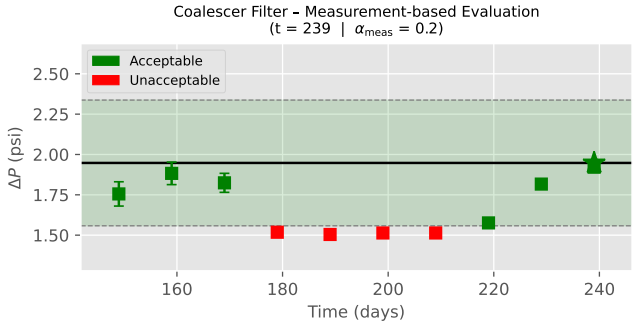

Figure 1. Measurement-based evaluation for the coalescer filter at $t = 239$ days ($\alpha_{\text{meas}} = 0.20$, exponential weighting, $N_{\text{lookback}} = 10$). Green squares indicate past predictions within the $\pm\varepsilon_{\text{meas}}$ tolerance band around the current observed ΔP; red squares fall outside. The star marks the current value. Weighted verdict $= 0.62$ ("good").

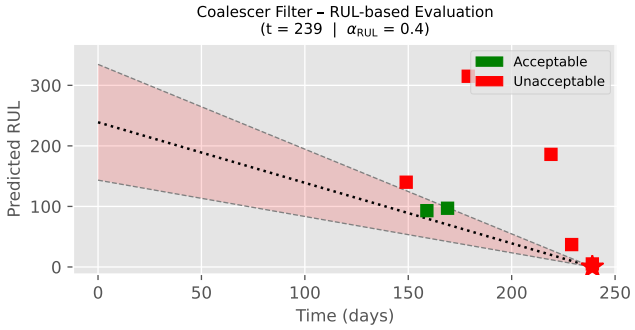

Figure 2. RUL-based evaluation for the coalescer filter at $t = 239$ days ($\alpha_{\text{RUL}} = 0.40$, exponential weighting). The acceptance cone narrows toward the current time. Early predictions overestimate remaining life (red); later predictions converge into the acceptance region (green). Weighted verdict $= 0.14$ ("bad").

- Weighting schemes: All five schemes, simple majority, linear, nonlinear, exponential, and custom, are compared.
- SLI window length ($N_{\text{lookback,SLI}}$): The number of verdict labels aggregated to compute the service-level indicator.
- SLI reference criterion: Whether the SLI is computed from measurement-based or RUL-based verdicts.

Predictions are generated at regular intervals (every 10 days for the filter, bushing, and heater systems; every 20 time steps for the membrane system). At each prediction time, the algorithm produces a forecast of the sensor trajectory extending into the future, from which both measurement predictions and RUL estimates are derived.

4.2. Case Study 1: Coalescer Filter

The coalescer filter is deployed at a gas processing plant and is used to remove liquid droplets from gas streams. The primary degradation mechanism is progressive plugging of the filter element by particulate debris, which leads to a monotonically increasing differential pressure (ΔP) across the filter. The dataset consists of 343 daily ΔP measurements recorded over approximately one year of operation. The threshold crossing direction is upward: the system approaches failure as ΔP rises above an operational limit.

Three RUL prediction algorithms are evaluated on this system. For each algorithm, the forecast data consists of a matrix of predicted ΔP values, where each column corresponds to a prediction issued at a specific day and each row represents a future time step. Some algorithms additionally provide upper and lower confidence bounds, from which prediction variances are derived as $\sigma = (\hat{y}_{\text{upper}} - \hat{y}_{\text{lower}})/4$.

Figure 1 presents the measurement-based evaluation for the coalescer filter at a representative evaluation time. Past predictions that fall within the $\pm\varepsilon_{\text{meas}}$ error band around the current observed ΔP value are marked as acceptable (green), while those falling outside are marked as unacceptable (red). The weighted verdict is computed using the selected weighting scheme.

Figure 2 shows the RUL-based evaluation, where past RUL predictions are plotted against the α–λ cone defined by the pseudo ground truth RUL. The cone narrows as the evaluation time approaches, requiring increasingly precise RUL predictions at later times. Here, early RUL predictions tend to overestimate the remaining life, a common behavior for gradually degrading systems where the degradation rate is not yet fully characterized. As more data becomes available, RUL predictions converge toward the α–λ cone, and the fraction of acceptable predictions increases.

Figure 3 illustrates the time evolution of the SLI for the coalescer filter across the operational period. The SLI provides a single summary score indicating whether the prediction algorithm has been performing acceptably over the look-back window. The coalescer filter exhibits a gradually increasing

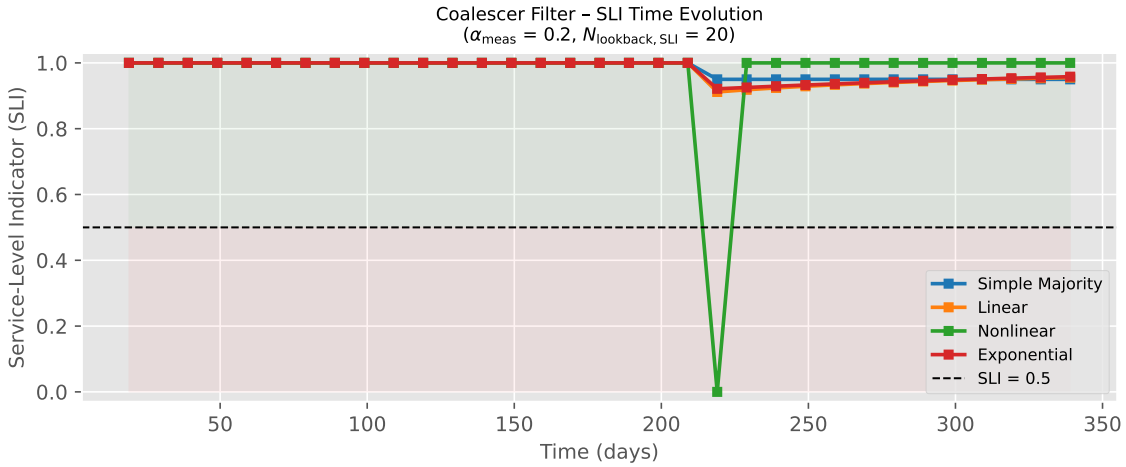

Figure 3. SLI time evolution for the coalescer filter across the full operational period ($\alpha_{\text{meas}} = 0.20$, $N_{\text{lookback,SLI}} = 20$, measurement-based reference). All four non-custom weighting schemes maintain SLI > 0.5 ("good") for most of the period. The nonlinear scheme is most sensitive to the transient performance degradation around day 210–230, while the exponential scheme recovers more quickly.

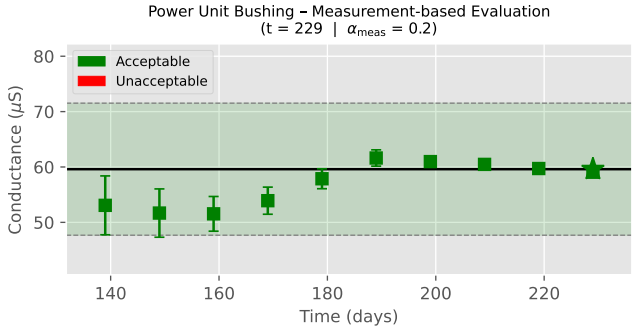

Figure 4. Measurement-based evaluation snapshot for the power unit bushing at $t = 229$ days ($\alpha_{\text{meas}} = 0.20$, exponential weighting). All predictions fall within the tolerance band, yielding a weighted verdict of 1.00 ("good").

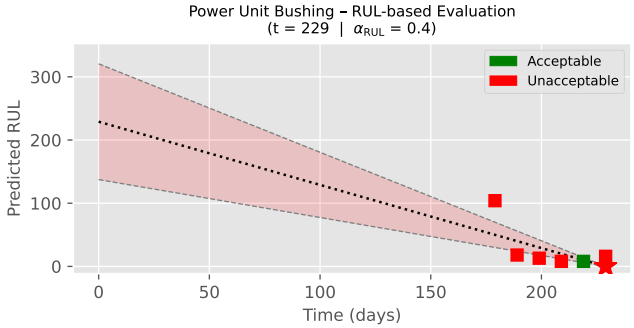

Figure 5. RUL-based α–λ evaluation for the power unit bushing at $t = 229$ days ($\alpha_{\text{RUL}} = 0.40$, exponential weighting). Conductance fluctuations cause dispersion in RUL estimates; the exponential scheme appropriately discounts older, less-informed predictions. Weighted verdict $= 0.14$ ("bad").

degradation trend with moderate noise.

Under an RUL algorithm with a 500-day horizon, the measurement-based evaluation yields acceptable verdicts for $\varepsilon_{\text{meas}} = 0.20$, indicating that short-horizon sensor predictions align well with observed ΔP values. The 2000-day horizon produces wider prediction intervals, which affects the measurement-based evaluation at tighter tolerance settings. SVR, lacking native uncertainty quantification, produces tighter point forecasts that are similarly sensitive to the choice of α_{meas}.

4.3. Case Study 2: Power Unit Bushing

The power unit bushing is a critical component in electrical power systems, providing insulation and mechanical support for high-voltage conductors. Bushing failure is a major failure mode in power units and can lead to catastrophic transformer damage. In this case study, the health of the bushing is monitored through its electrical conductance, computed from voltage and current measurements. The dataset comprises 322 daily conductance measurements collected from a power unit in an oil sands facility. In this system, increasing conductance signals progressive insulation degradation.

Three algorithms are evaluated for this system: Prophet with 500-day and 1500-day forecast horizons, and SVR. The evaluation follows the same methodology as the filter case study. At each evaluation time, the current conductance value serves as the pseudo ground truth reference, and past predictions are assessed for consistency.

Figure 4 shows a representative measurement-based evaluation snapshot for the bushing system. The bushing conductance data exhibits more variability than the filter data, with noticeable short-term fluctuations superimposed on a slowly

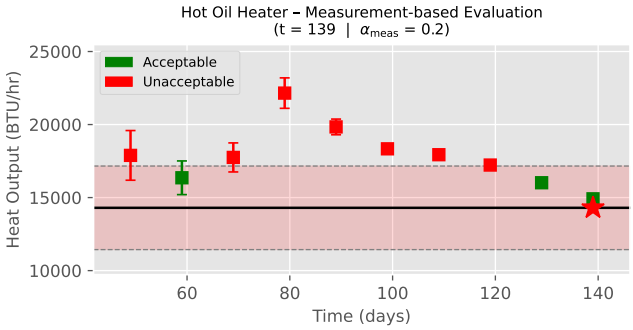

Figure 6. Measurement-based evaluation for the hot oil heater at $t = 139$ days ($\alpha_{\text{meas}} = 0.20$, exponential weighting). Several predictions fall outside the tolerance band at intermediate forecast horizons, yielding a weighted verdict of 0.36 ("bad").

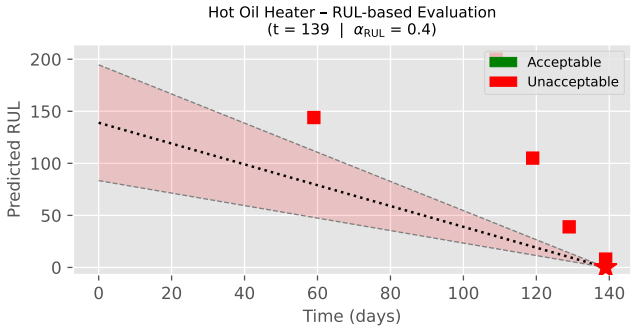

Figure 7. RUL-based evaluation for the hot oil heater at $t = 139$ days ($\alpha_{\text{RUL}} = 0.40$, exponential weighting). All predictions fall outside the acceptance cone, indicating that the algorithm systematically overestimates RUL at this stage of operation. Weighted verdict $= 0.00$ ("bad").

increasing degradation trend. This increased noise presents a more challenging evaluation scenario. With $\varepsilon_{\text{meas}} = 0.20$, the Prophet 500-day model maintains acceptable measurement-based verdicts for the majority of evaluation time steps, while the 1500-day model, which produces broader prediction intervals, yields more conservative forecasts.

Figure 5 presents the corresponding RUL-based α–λ evaluation. The RUL-based evaluation is particularly informative for this system. The conductance fluctuations cause the threshold-crossing-based RUL estimates to vary significantly between consecutive predictions, leading to wider dispersion of RUL predictions within the α–λ cone. The exponential weighting scheme, which emphasizes recent predictions, proves advantageous here, as it appropriately discounts older RUL predictions that were made with less information about the evolving degradation trend.

4.4. Case Study 3: Hot Oil Heater

The hot oil heater is a gas-fired heating system used in oil and gas processing to heat oil for downstream operations. The primary degradation mechanism is a gradual reduction in thermal efficiency, manifested as decreasing heat output over time while fuel consumption remains approximately constant. The dataset consists of 202 daily heat output measurements, along with inlet temperature, outlet temperature, and flow rate. The threshold crossing direction is downward: the system approaches failure as the heat output drops below an operational minimum.

The Prophet 500-day, Prophet 2000-day, and SVR algorithms are evaluated. The heat output serves as the monitored health indicator. Unlike the filter and bushing systems, the degradation trend here is decreasing, which means the evaluation threshold is approached from above.

Figure 6 illustrates the measurement-based evaluation for the heater system, showing how past heat output predictions compare with the currently observed value. The heater system presents a relatively smooth degradation trend with a clear downward slope, making it favorable for time-series forecasting methods. Prophet with both horizon settings produces predictions that fall well within the $\pm\varepsilon_{\text{meas}}$ bounds for moderate tolerance values. SVR also performs competitively on this dataset due to the regularity of the degradation pattern.

Figure 7 presents the RUL-based evaluation in the α–λ cone. At $t = 139$ days, all past RUL predictions fall outside the acceptance cone (verdict = 0.00), indicating that the algorithm systematically overestimates the remaining useful life at this stage of operation. This behavior is consistent with a slowly degrading system where early predictions, made before the degradation trend is well-characterized, tend to project an overly optimistic time-to-threshold.

4.5. Case Study 4: Acid Gas Separation Membrane

The acid gas separation membrane system is used in natural gas processing to separate hydrogen sulfide (H_2S) and carbon dioxide (CO_2) from the feed gas. The membranes degrade over time as their permeability decreases, resulting in declining CO_2 flux. The monitored health indicator is the CO_2 flux, which depends on both the membrane condition and the feed flow rate. For this system, the RUL prediction is performed using a particle filter-based approach.

The particle filter produces predictions with natural uncertainty bounds derived from the particle distribution, which are directly comparable to the Prophet confidence intervals used in the other case studies. The CO_2 flux data exhibits more complex dynamics than the other three systems, as the flux depends on both the membrane degradation state and the (time-varying) feed flow rate. This operational variability introduces additional challenges for both prediction and evaluation. Figure 8 shows a representative snapshot of this evaluation; the framework successfully identifies evaluation time steps where past predictions were consistent with observed flux values

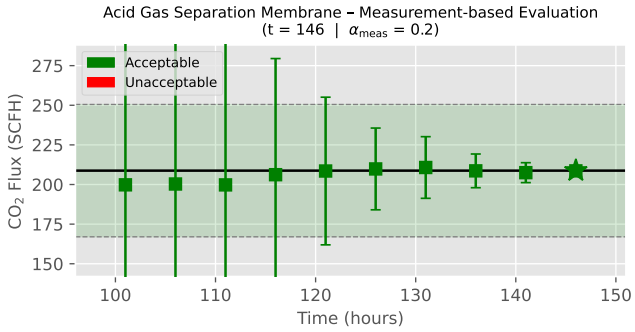

Figure 8. Measurement-based evaluation for the Acid Gas Membrane system at 70% of the operational period ($\alpha_{\text{meas}} = 0.20$, exponential weighting). Green squares indicate predictions within $\pm\alpha_{\text{meas}}$ of the current sensor value; red squares indicate out-of-bounds predictions. Weighted verdict $= 1.00$ ("good").

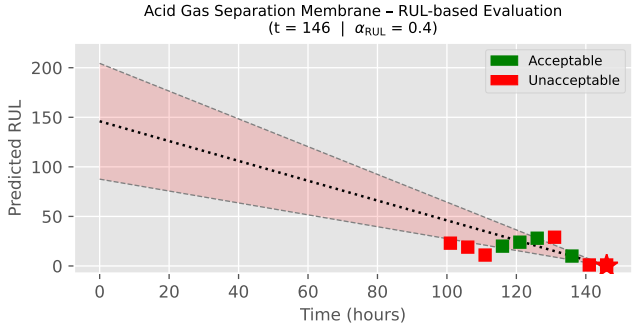

Figure 9. RUL-based α–λ evaluation for the Acid Gas Membrane system ($\alpha_{\text{RUL}} = 0.40$, exponential weighting). Each square represents a past RUL prediction; green indicates predictions within the α–λ cone, red indicates predictions outside the cone. The dashed diagonal shows the ideal prediction trajectory. Weighted verdict $= 0.40$ ("bad").

and distinguishes these from time steps where the algorithm's predictions diverged. Figure 9 presents the corresponding RUL-based α–λ evaluation.

Across multiple membrane units, the evaluation framework reveals heterogeneous prediction quality: some units exhibit smooth degradation trends that are well-captured by the prognostic model, while others show abrupt flux changes (e.g., due to operating condition changes or membrane fouling) that lead to prediction failures. The SLI correctly reflects these differences, providing an actionable indicator for operators, as shown in Figure 10.

4.6. Comparison of Weighting Schemes

One of the central contributions of this framework is the provision of multiple weighting schemes that allow the evaluation to emphasize recent predictions over older ones, or to treat all predictions equally. To assess the practical impact of this design choice, we compare the five weighting schemes across all four systems. Table 2 reports the mean SLI values for all four systems under each weighting scheme, and several observations emerge from this comparison:

1. The simple majority scheme treats all predictions in the look-back window equally. In systems with a stable degradation trend (e.g., the heater), this produces results similar to the recency-weighted schemes. However, in systems where prediction quality improves over time as more data becomes available (e.g., the bushing), the simple majority scheme can be unduly influenced by early, less accurate predictions that remain in the look-back window.
2. The exponential scheme provides the strongest emphasis on recent predictions, followed by the nonlinear (quadratic) and linear schemes. For systems where the most recent predictions are substantially more accurate, as is typical for adaptive or sequentially-updated algorithms, the exponential scheme yields higher SLI values. Conversely, for systems where prediction quality is relatively uniform across the look-back window, all three schemes produce comparable results.
3. Tighter error bounds ($\varepsilon_{\text{meas}} = 0.10$) amplify the differences between weighting schemes, as more predictions are classified as unacceptable and the weighting of the remaining acceptable ones becomes more consequential. At looser bounds ($\varepsilon_{\text{meas}} = 0.30$), most predictions are acceptable, and the choice of weighting scheme has less impact.
4. For operational deployment, the exponential or nonlinear weighting schemes are recommended, as they provide a more responsive SLI that reflects current algorithm performance rather than being dominated by historical prediction quality.

4.7. Discussion

The framework successfully evaluates RUL predictions from three fundamentally different algorithmic families without requiring any modification to the evaluation procedure.

A critical requirement for any evaluation methodology is the ability to distinguish between algorithms (or algorithm configurations) that produce reliable predictions and those that do not. The proposed framework achieves this in several ways:

- For each system, different RUL prediction algorithms (e.g., Prophet vs. SVR) yield different SLI trajectories, enabling direct comparison of algorithm quality on the same data.
- Within a single algorithm's output, the framework identifies periods where predictions are consistent with observations (high SLI) and periods where they diverge (low SLI), providing actionable diagnostics for when an algorithm

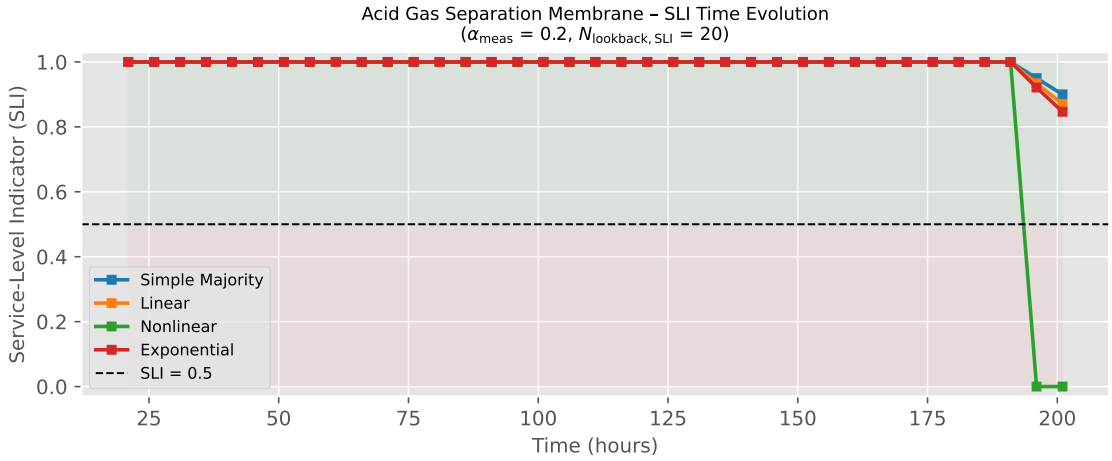

Figure 10. SLI time series for the Acid Gas Membrane system under five weighting schemes (Simple Majority, Linear, Nonlinear, Exponential, Custom). All schemes converge to high SLI values, reflecting consistent prediction accuracy from the physics-informed particle filter.

Table 2. Comparison of SLI values across systems and weighting schemes ($\alpha_{\text{meas}} = 0.20$, $\alpha_{\text{RUL}} = 0.40$, measurement-based SLI reference, $N_{\text{lookback,SLI}} = 20$). Values represent the mean SLI computed over the final 30% of each system's operational period. Membrane values computed using the physics-informed particle filter.

System	Simple Majority	Linear	Nonlinear	Exponential	Custom
Coalescer Filter	0.950	0.945	1.000	0.948	0.948
Power Unit Bushing	1.000	1.000	1.000	1.000	1.000
Hot Oil Heater	0.763	0.670	0.800	0.707	0.600
Acid Gas Membrane	0.986	0.982	0.818	0.979	0.962

may need retraining or recalibration. In operational terms, a sustained SLI excursion below a predefined threshold (e.g., $\text{SLI} < 0.5$ for more than three consecutive evaluation windows) could trigger automated model retraining, escalation to a subject-matter expert, or an alert to the asset operator, depending on the criticality of the system and the terms of the SLA.

- By comparing Prophet at different forecast horizons (e.g., 500 vs. 2000 days for the filter), the framework reveals how forecast horizon length affects prediction reliability within the evaluation window.

The central innovation of this work is the use of the current sensor value as a virtual ground truth reference. However, the virtual ground truth approach has an inherent limitation: it evaluates the algorithm's ability to predict sensor values and times-to-threshold at the *current* condition, which is not necessarily equivalent to predicting the true EoL.

The choice of error bounds (α_{meas} and α_{RUL}) significantly influences the evaluation outcome. Overly tight bounds may classify a well-performing algorithm as unacceptable due to inevitable measurement noise and modeling uncertainty, while excessively loose bounds may fail to identify genuinely poor predictions. The appropriate error bound depends on the application context: safety-critical systems may warrant tighter bounds (e.g., $\alpha = 0.10$), while early-stage algorithm development may benefit from more lenient evaluation criteria (e.g., $\alpha = 0.30$). In practice, bounds should be selected based on: (1) the inherent sensor measurement noise level, (2) the acceptable operational tolerance for the monitored quantity, and (3) the consequences of false positives (incorrectly flagging good predictions as bad) versus false negatives (accepting poor predictions as good).

The effectiveness of the look-back evaluation depends critically on two temporal parameters: the prediction cadence τ (the time spacing between successive RUL predictions) and the look-back window length W_{lookback} (or equivalently, the number of predictions N_{lookback}).

If predictions are issued too infrequently relative to the degradation time scale, the look-back window may not capture sufficient samples to provide statistically meaningful verdicts. Conversely, if predictions are issued very frequently, consecutive predictions may be highly correlated, reducing the effective information content of the look-back window. As a practical guideline, the prediction cadence should be chosen such that W_{lookback} spans at least 10–20 independent prediction events, and the cadence should be small relative to the expected RUL values (e.g., $\tau \ll \widehat{\text{RUL}}$).

Short look-back windows (e.g., $N_{\text{lookback}} = 2$ to 5) provide highly responsive SLIs that react quickly to changes in algorithm performance, but may be sensitive to transient anomalies or outliers. Long look-back windows (e.g., $N_{\text{lookback}} > 50$) provide more stable SLIs but may lag behind actual changes in algorithm quality. The choice depends on the operational use case: for real-time dashboards and rapid anomaly detection, shorter windows are preferable; for long-term trend analysis and SLA reporting, longer windows provide more robust indicators.

A foundational assumption of the proposed framework is that accurate prediction of intermediate system states is indicative of accurate RUL estimation. This hypothesis is well-supported for systems exhibiting the following characteristics: (1) monotonic or quasi-monotonic degradation, where the health indicator trends consistently in one direction over time; (2) stationary degradation dynamics, where the underlying physical mechanism driving degradation does not change qualitatively during the evaluation period; and (3) smooth degradation trajectories, where the sensor signal evolves gradually rather than through abrupt step changes.

The four industrial systems studied in this paper satisfy these conditions to varying degrees: the coalescer filter and hot oil heater exhibit nearly ideal monotonic trends, the bushing shows moderate noise superimposed on a clear trend, and the membrane system introduces operational variability through time-varying flow rates. The framework's successful application across these systems provides empirical support for the hypothesis under representative industrial conditions.

However, the hypothesis may not hold in several important scenarios:

1. In systems with regime changes or nonlinear degradation dynamics, the relationship between intermediate-state prediction accuracy and end-of-life prediction accuracy may weaken. For example, if a system transitions from gradual wear to accelerated degradation near failure, an algorithm that accurately predicts intermediate states may still fail near EoL.
2. Maintenance interventions that partially restore system health create discontinuities in the degradation trajectory, invalidating comparisons between predictions made before and after the intervention.
3. Systems with highly oscillatory or non-monotonic sensor behavior may produce ambiguous virtual thresholds where the current sensor value does not represent a meaningful degradation milestone.

When these conditions are encountered, the evaluation should be segmented into stationary degradation regimes, or the look-back window should be restricted to exclude discontinuities. The framework must be viewed as a necessary but not sufficient condition for prognostic reliability: consistent performance at intermediate states provides strong evidence of prognostic capability, but does not guarantee equivalent accuracy at the true failure threshold.

The pseudo ground truth relies on the assumption that the current sensor value $z(t)$ accurately reflects the asset's true state. In practice, industrial sensor data may be affected by measurement noise, missing values, calibration drift, or transient anomalies, any of which can corrupt the reference and lead to spurious evaluations. Sensor health monitoring and data quality checks should be integrated upstream of the evaluation framework. For the datasets used in this study, standard preprocessing (outlier removal, gap filling) was applied prior to evaluation, but the framework does not prescribe specific data cleaning procedures, as these are application-dependent.

As a practical guideline, the error bounds should be selected based on: (1) the inherent sensor measurement noise level, which sets a lower bound on achievable prediction accuracy; (2) the acceptable operational tolerance for the monitored quantity, reflecting the consequence severity of prediction errors; and (3) the balance between false positives (incorrectly flagging good predictions as bad) and false negatives (accepting poor predictions as good). Adaptive error bound selection, for instance through cross-validation or noise-floor estimation, represents a promising direction for improving out-of-the-box applicability.

Both evaluation modes currently rely on deterministic error thresholds (α_{meas} and α_{RUL}) to classify predictions as acceptable or unacceptable, even though many prediction algorithms provide uncertainty estimates (e.g., prediction intervals from Prophet, particle distributions from particle filters). In the present framework, these uncertainty estimates are used for visualization and diagnostic purposes but are not explicitly incorporated into the acceptability criterion. Extending the evaluation to account for predictive uncertainty, for instance by assessing whether the observed value falls within the algorithm's stated prediction interval or by computing the probability mass of the prediction distribution within the error bounds, would yield a more nuanced assessment that rewards well-calibrated uncertainty estimates. This represents an important direction for future work, as discussed in Section 5.

It should be noted that the present case studies evaluate each sensor channel independently. For the systems studied here, a single dominant health indicator (differential pressure, conductance, heat output, or CO_2 flux) captures the primary degradation phenomenon. However, many industrial assets are instrumented with multiple correlated sensors, and evaluating predictions independently for each channel may miss cases where individual sensor predictions appear acceptable but the joint prediction vector is physically inconsistent. Joint multivariate evaluation, for instance using a Mahalanobis-distance-based acceptability criterion that accounts for cross-sensor correlations, represents a clear extension of the current framework

and is discussed further in Section 5.

The proposed framework addresses a specific gap: continuous online evaluation of RUL predictions without failure data. To our knowledge, no existing framework provides an equivalent capability with both measurement-level and RUL-level assessment modes, configurable weighting schemes, and a unified SLI formulation. The closest related approaches include short-horizon forecast verification (which assesses measurement accuracy but not RUL performance) and offline cross-validation on historical datasets (which requires complete degradation trajectories). The proposed framework complements rather than replaces these approaches: when run-to-failure data is available, traditional metrics such as the standard α–λ accuracy, RMSE, and scoring functions should be used alongside the proposed framework to provide a comprehensive evaluation. The pseudo ground truth approach is specifically designed for the operational scenario where such data is absent. A quantitative head-to-head comparison against alternative no-failure evaluation strategies (e.g., a rolling short-horizon RMSE baseline) would further clarify the incremental value of the proposed framework; such a benchmark is beyond the scope of this paper and is left for future work.

If the RUL prediction algorithm is retrained or its parameters are updated during the evaluation period, the performance characteristics may change discontinuously. The framework evaluates the predictions as they were issued, but practitioners should be aware that historical predictions may reflect an earlier version of the algorithm.

5. CONCLUSIONS

This paper presented a novel framework for evaluating the performance of RUL prediction algorithms in the absence of run-to-failure ground truth data, a critical gap that has hindered the industrial deployment and validation of prognostic solutions for assets with extended operational lifespans. The core contribution is a retrospective look-back methodology that treats the asset's current observed sensor state as a *virtual threshold*, enabling continuous online evaluation without requiring observation of actual EoL.

The framework's measurement-based evaluation quantifies the accuracy of past sensor-value forecasts against currently observed measurements, providing a direct indicator of model drift independent of RUL computation. RUL-based evaluation assesses how well past RUL estimates predicted the time to reach the asset's present condition.

The experimental validation across multiple distinct industrial assets demonstrated the framework's generalizability and practical utility. The SLI provided stable, interpretable indicators of prediction quality even for extended time horizons, and the different weighting schemes offered meaningful flexibility in emphasizing recent versus historical performance depending on operational priorities.

From a practical standpoint, the methodology addresses a critical unmet need in industrial PHM. PHM solution providers can now quantitatively demonstrate prediction reliability to customers using scientifically grounded metrics, rather than qualitative assurances alone. The SLI is directly suitable for integration into SLAs, enabling transparent communication about prognostic performance.

Several limitations should be acknowledged. The pseudo ground truth evaluates consistency at intermediate states rather than absolute prognostic accuracy at true EoL; the validity conditions for this approximation, along with other limitations, are discussed in Section 4.7. The framework currently requires manual tuning of look-back windows and error bounds, evaluates sensors independently, and does not incorporate prediction uncertainty into the acceptability criterion.

A few concrete directions for future work can be envisioned. First, extending the evaluation to uncertainty-aware metrics that leverage predictive variances would yield more nuanced assessments. For instance, computing the probability mass of the prediction distribution that falls within the error bounds, or evaluating the calibration of prediction intervals (i.e., whether stated 95% intervals actually contain the observation 95% of the time), would reward well-calibrated uncertainty estimates and penalize overconfident or underconfident algorithms.

Second, the SLI is currently computed as a deterministic weighted average of binary labels. Developing bootstrap-based or analytical confidence intervals for the SLI itself would enable practitioners to reason about the statistical reliability of the indicator and to distinguish significant changes in algorithm performance from random fluctuation.

Third, the current method assesses sensors independently. Extending it to evaluate predictions across correlated measurements would give a more comprehensive assessment for systems with interdependent indicators. This would address cases where individual sensors seem acceptable, but their joint predictions are physically inconsistent.

The framework requires manual configuration of look-back window sizes, error bounds, and weighting schemes. Automating the parameter selection, for instance, using cross-validation to optimize the discriminative power of the SLI, or using change-point detection algorithms to adaptively adjust window sizes in response to changes in degradation dynamics, would improve out-of-the-box applicability.

Currently, the framework provides diagnostic feedback on prediction quality but does not take corrective action. Integrating the SLI with automated model retraining or recalibration logic, triggering retraining when the SLI falls below a specified threshold for a sustained period, would enable closed-loop prognostic maintenance.

Finally, the current framework assumes quasi-stationary degradation dynamics. Extending the methodology to explicitly account for time-varying operating regimes, for instance, by normalizing predictions with respect to operating condition changes, or by segmenting the evaluation into distinct operational modes, would broaden applicability to complex industrial assets with dynamic duty cycles.

REFERENCES

Beyer, B., Jones, C., Petoff, J., & Murphy, N. R. (2016). *Site Reliability Engineering: How Google Runs Production Systems*. O'Reilly Media.

Chao, M. A., Kulkarni, C. S., Goebel, K. F., & Fink, O. (2020). Fusing physics-based and deep learning models for prognostics. *Reliab. Eng. Syst. Saf.*, *217*, 107961.

Chen, Z., Wu, M., Zhao, R., Guretno, F., Yan, R., & Li, X. (2021). Machine Remaining Useful Life Prediction via an Attention-Based Deep Learning Approach. *IEEE Transactions on Industrial Electronics*, *68*(3), 2521-2531.

Cheng, H., Kong, X., Wang, Q., Ma, H., Yang, S., & Chen, G. (2023). Deep Transfer Learning Based on Dynamic Domain Adaptation for Remaining Useful Life Prediction Under Different Working Conditions. *Journal of Intelligent Manufacturing*, *34*(2), 587-613.

da Costa, P. R. d. O., Akçay, A., Zhang, Y., & Kaymak, U. (2020). Remaining Useful Lifetime Prediction via Deep Domain Adaptation. *Reliability Engineering & System Safety*, *195*, 106682.

Ding, Y., Ding, P., Zhao, X., Cao, Y., & Jia, M. (2022). Transfer Learning for Remaining Useful Life Prediction Across Operating Conditions Based on Multisource Domain Adaptation. *IEEE/ASME Transactions on Mechatronics*, *27*(5), 4143-4152.

Forgione, M., Muni, A., Piga, D., & Gallieri, M. (2023). On the Adaptation of Recurrent Neural Networks for System Identification. *Automatica*, *155*, 111092.

Gneiting, T., Balabdaoui, F., & Raftery, A. E. (2007). Probabilistic Forecasts, Calibration and Sharpness. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *69*(2), 243–268.

Goebel, K., Daigle, M. J., Saxena, A., Roychoudhury, I., Sankararaman, S., & Celaya, J. R. (2017). *Prognostics: The science of making predictions*.

Goebel, K., Saxena, A., Saha, S., Saha, B., & Celaya, J. (2012). *Prognostic Performance Metrics* (M. Pecht & M. Kang, Eds.). CRC Press.

Huang, Z., Xu, Z., Wang, W., & Sun, Y. (2015). Remaining Useful Life Prediction for a Nonlinear Heterogeneous Wiener Process Model With an Adaptive Drift. *IEEE Transactions on Reliability*, *64*(2), 687-700.

Jardine, A. K., Lin, D., & Banjevic, D. (2006). A Review on Machinery Diagnostics and Prognostics Implementing Condition-Based Maintenance. *Mechanical Systems and Signal Processing*, *20*(7), 1483-1510.

Lee, J., Wu, F., Zhao, W., Ghaffari, M., Liao, L., & Siegel, D. (2014). Prognostics and Health Management Design for Rotary Machinery Systems—Reviews, Methodology and Applications. *Mechanical Systems and Signal Processing*, *42*(1–2), 314-334.

Lei, Y., Li, N., Gontarz, S., Lin, J., Radkowski, S., & Dybala, J. (2016). A Model-Based Method for Remaining Useful Life Prediction of Machinery. *IEEE Transactions on Reliability*, *65*(3), 1314-1326.

Lei, Y., Li, N., Guo, L., Li, N., Yan, T., & Lin, J. (2018). Machinery Health Prognostics: A Systematic Review from Data Acquisition to RUL Prediction. *Mechanical Systems and Signal Processing*, *104*, 799-834.

Ma, M., & Mao, Z. (2021). Deep-Convolution-Based LSTM Network for Remaining Useful Life Prediction. *IEEE Transactions on Industrial Informatics*, *17*(3), 1658-1667.

Ramasso, E., & Saxena, A. (2014). Performance Benchmarking and Analysis of Prognostic Methods for CMAPSS Datasets. In *International Journal of Prognostics and Health Management* (Vol. 5, p. 1-15).

Roychoudhury, I., Hafiychuk, V., & Goebel, K. (2013). Model-based diagnosis and prognosis of a water recycling system. In *2013 ieee aerospace conference* (pp. 1–9).

Sankararaman, S., & Goebel, K. (2015). Significance, Interpretation, and Quantification of Uncertainty in Prognostics and Remaining Useful Life Prediction. *Mechanical Systems and Signal Processing*, *52–53*, 228–247.

Saxena, A., Celaya, J., Balaban, E., Goebel, K., Saha, B., Saha, S., & Schwabacher, M. (2008). Metrics for Evaluating Performance of Prognostic Techniques. In *2008 International Conference on Prognostics and Health Management* (p. 1-17). IEEE.

Saxena, A., Celaya, J., Saha, B., Saha, S., & Goebel, K. (2010). Metrics for Offline Evaluation of Prognostic Performance. In (Vol. 1, p. 1-20).

Saxena, A., & Goebel, K. (2008). Turbofan Engine Degradation Simulation Data Set. In *NASA Ames Prognostics Data Repository.* (NASA Ames Research Center, Moffett Field, CA)

Saxena, A., Goebel, K., Simon, D., & Eklund, N. (2008). Damage Propagation Modeling for Aircraft Engine Run-to-Failure Simulation. In *2008 International Conference on Prognostics and Health Management* (p. 1-9). IEEE.

Sheth, P., & Roychoudhury, I. (2024). Robust remaining useful life prediction using jacobian feature regression-based model adaptation. In *PHM Society European Conference* (Vol. 8, pp. 11–11).

Si, X.-S., Hu, C.-H., Chen, M.-Y., & Wang, W. (2011). An Adaptive and Nonlinear Drift-based Wiener Process for Remaining Useful Life Estimation. In *2011 Prognostics and System Health Management Conference* (p. 1-5).

Sikorska, J. Z., Hodkiewicz, M., & Ma, L. (2011). Prognostic Modelling Options for Remaining Useful Life Estimation by Industry. *Mechanical Systems and Signal Processing*, *25*(5), 1803-1836.

Taylor, S. J., & Letham, B. (2018). Forecasting at Scale. *The American Statistician*, *72*(1), 37-45.

Vachtsevanos, G., Lewis, F. L., Roemer, M., Hess, A., & Wu, B. (2006). *Intelligent Fault Diagnosis and Prognosis for Engineering Systems*. John Wiley & Sons.

Wang, Y., Zhao, Y., & Addepalli, S. (2020). Remaining Useful Life Prediction using Deep Learning Approaches: A Review. *Procedia Manufacturing*, *49*, 81-88.

Zhang, Y., Xiong, R., He, H., & Liu, Z. (2017). A LSTM-RNN Method for the Lithuim-ion Battery Remaining Useful Life Prediction. In *2017 Prognostics and System Health Management Conference (PHM-Harbin)* (p. 1-4).

Zio, E. (2022). Prognostics and Health Management (PHM): Where Are We and Where Do We (Need to) Go in Theory and Practice. *Reliability Engineering & System Safety*, *218*, 108119.

BIOGRAPHIES

Indranil Roychoudhury is a Principal AI Scientist at SLB Technology Innovation Center, and his primary area of research is time-series analysis by combining physics-based approaches with machine learning approaches. He holds his Ph.D. and MS in CS from Vanderbilt University and was a Senior Research Scientist at NASA Ames before joining SLB. He is a Fellow of the Prognostics and Health Management Society and a Senior Member of IEEE.

Prasham Sheth is a Senior Data Scientist in the Intelligent Systems Laboratory at the SLB Technology Innovation Center. His research interests include the application of machine learning, deep learning, and hybrid modeling-based approaches to solving complex problems in computer vision and time-series analysis. He holds a Master of Science in Data Science from Columbia University, New York, New York, USA and a Bachelor of Technology degree in Computer Engineering from Nirma University, Ahmedabad, Gujarat, India.

Taoufik Wassar is a Data Scientist at SLB within Process Technologies and Solutions, where he applies advanced analytics to solve complex industrial challenges. With multidisciplinary engineering expertise spanning mechanical, electrical, and process engineering, as well as physics-based and data-driven modeling, he bridges domain knowledge and artificial intelligence to deliver high-impact solutions. Taoufik has recently contributed to oil and gas midstream projects, including gas and water treatment systems, focusing on developing robust health and performance monitoring solutions that enhance reliability, efficiency, and operational insight.