Journal of Engineering, Project, and Production Management
2026, 16(6), 2026-0090

Uncertainty-Aware Predictive Maintenance Scheduling: A Decision-Support Framework for Industrial Production Systems

Han Shi
Undergraduate Student, School of Marine Electrical Engineering, Dalian Maritime University, Dalian, 116026, China, E-mail: uxfn2347@outlook.com

Production Management
Received June 5, 2026; revised July 18, 2026; accepted July 18, 2026
Available online August 2, 2026

Abstract: Preventive maintenance in a manufacturing plant is, at bottom, a resource-allocation problem: every intervention weighs the cost of replacing a component too early against the risk of an unplanned stoppage under the challenging realities of production calendars, crew availability, and machine access. Most facilities still rely on fixed calendar intervals that ignore what the equipment is signaling, paying for the difference in wasted component life and avoidable downtime. This paper describes a decision-support framework that connects data-driven prognostics to a production-constrained optimization model. A Gradient-Boosted Regression Tree (GBRT) supplies point estimates of Remaining Useful Life (RUL). A Quantile Regression Forest (QRF) turns these into calibrated prediction intervals with finite-sample coverage guarantees, and a Mixed-Integer Linear Program (MILP) converts the resulting per-machine failure probabilities into maintenance schedules that respect shift windows and crew capacity. The contribution lies less in the individual components than in how they are joined: uncertainty flows directly into the scheduler, so every intervention decision reflects both predictive risk and production feasibility. We evaluated the framework on a public milling benchmark and eight months of data from a twelve-machine packaging facility (ten independent runs each). Relative to the calendar-based policy currently used at the facility, total maintenance cost fell by 6.3% on the packaging line and 11.9% on the NASA milling dataset; unplanned downtime events fell by 7.2% and 44.4%; and unnecessary preventive replacements fell by 26.2% and 66.7%. Calibration error remained below 1.5 percentage points at every coverage level tested, and all scheduling instances were solved to certified optimality within seconds for fleets of up to 50 machines on standard hardware.

Keywords: Predictive maintenance; maintenance scheduling; decision support; conformal prediction; production constraints.

DOI 10.32738/JEPPM-2026-0090

1. Introduction

Every planning cycle, a maintenance manager faces the same practical dilemma: which machines to service, in which window, and at what cost. Acting too early wastes component life and ties up scarce maintenance crews; waiting too long invites a costly unplanned stoppage. Fixed-interval calendar policies, still the most common approach in industry, sidestep this trade-off entirely: because they do not adapt to observed degradation, they incur systematic waste (Wang, 2002; Dekker, 1996; Mobley, 2002).

Data-driven prognostics offer a route out of this trap, and accurate RUL predictors now exist for many classes of industrial asset (Li et al., 2018; Zhang et al., 2018; Wang et al., 2020). Yet a point estimate of remaining life, on its own, does not tell a manager whether to act in the next six-hour window or wait until next week. That decision requires three further ingredients: the economic consequences of each candidate's timing, a sense of how much the prediction can be trusted, and confirmation that a suitable maintenance window actually exists (Baptista et al., 2019; Jardine et al., 2006).

This paper addresses part of that gap by integrating prognostic uncertainty estimation with production-aware scheduling into a single, deployable system. The framework targets serial flow-shop and parallel-line discrete manufacturing environments in which machines operate in fixed sequential or parallel configurations, maintenance windows are dictated by shift schedules, and each machine failure carries a measurable, isolated cost. Packaging lines, assembly lines, and similar high-throughput discrete-part facilities fall within this class. Job-shop layouts and continuous-flow process industries would require structural extension. The four principal contributions are the following:

(1) A decision-integration architecture that feeds calibrated, distribution-free prediction intervals directly into a production-constrained MILP scheduler, producing operationally feasible schedules that balance maintenance cost and failure risk within maintenance-window constraints.

(2) A component-wise ablation study isolating the operational value of QRF interval estimation, conformal calibration, and MILP scheduling.

(3) Conformal coverage validation at five nominal levels (0.70–0.99), demonstrating robustness across two industrial datasets.

(4) An interactive scenario-analysis workflow providing break-even analysis, window-removal cost penalties, and risk-tolerance adjustment, all without retraining the underlying models.

This work is guided by three research questions.

RQ1: Do calibrated, distribution-free prediction intervals, when propagated into the scheduler, reduce maintenance cost, downtime, and unnecessary replacements relative to point-prediction and threshold policies?

RQ2: Do conformal coverage guarantees hold empirically on real industrial sensor data across nominal coverage levels?

RQ3: Can uncertainty-aware, production-constrained schedules be computed within operational planning time budgets at realistic fleet sizes?

Section 2 reviews related work. Section 3 describes the methodology and presents Algorithm 1. Section 4 reports experimental results. Section 5 discusses operational implications and limitations. Section 6 concludes.

2. Related Work

The preventive maintenance literature has followed a recognizable trajectory. Early work established that calendar-based replacement is structurally inefficient: set the interval for worst-case degradation and slow-degrading assets are replaced far too early; set it for the average case and fast-degrading ones fail in service (Wang, 2002; Dekker, 1996). Condition-based maintenance improved matters by tying decisions to measured health indicators (Jardine et al., 2006; Peng et al., 2010), but a static threshold can neither quantify timing risk nor check whether a maintenance window will be available when the threshold trips (Gebraeel et al., 2005; Yan and Lee, 2005).

Data-driven prognostics improved accuracy through neural networks (Widodo and Yang, 2011; Tian, 2012; Zhang et al., 2018) and gradient boosting (Wang et al., 2020), but point-prediction RMSE remains the dominant metric and says nothing about timing risk (Sankararaman, 2015; Chen et al., 2021). Uncertainty-aware methods such as Bayesian neural networks (Goebel et al., 2008), Monte Carlo dropout (Gal and Ghahramani, 2016), particle filters (Orchard and Vachtsevanos, 2009), and quantile regression forests (Meinshausen, 2006) are typically evaluated in isolation. Conformal Prediction with Quantile Regression (Romano et al., 2019) offers finite-sample coverage guarantees without distributional assumptions. This direction has accelerated recently: Conformalized Quantile Regression (CQR) has been applied to aero-engine RUL prediction (Mao et al., 2024), ensemble and cross-conformal variants tighten calibrated interval widths (Piao et al., 2025), and weighted conformal procedures address non-exchangeable industrial IoT data (Moccardi et al., 2025). These studies, however, stop at calibrated intervals and do not propagate them into scheduling decisions.

Stochastic optimization models (van Noortwijk, 2009; Alaswad and Xiang, 2017; Keizer et al., 2017) derive replacement policies analytically but depend on parametric degradation assumptions that rarely hold for real sensor streams (Grall et al., 2002; Dieulle et al., 2003). ML–scheduling couplings (Wang et al., 2021; Li et al., 2022; Zhang et al., 2022) typically pass only point estimates to the optimizer. Hu and Chen (2018) embed failure probability directly in the objective, the closest prior work to ours, but assume parametric Gamma degradation and do not account for production calendars. More recently, Okumuşoğlu et al. (2024) integrated sensor-driven remaining-lifetime distributions into a two-stage stochastic program for power-system maintenance and operations scheduling, though via scenario-based stochastic programming rather than distribution-free calibrated intervals and outside discrete manufacturing. Despite these advances, frameworks feeding calibrated, distribution-free prediction intervals directly into a production-constrained scheduler remain rare. The present work addresses these gaps together.

3. Methodology

The framework has three computational stages: RUL prediction (Section 3.1), interval calibration (Section 3.2), and MILP scheduling (Section 3.3). A scenario-analysis workflow (Section 3.4) supports managerial what-if queries. Fig. 1 shows the overall architecture. The principal notation used throughout is defined inline as each component is introduced in Sections 3.1-3.3.

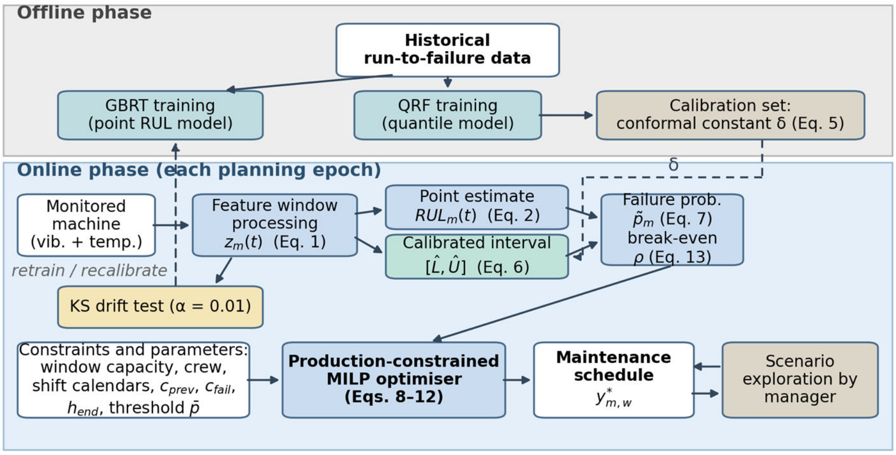

Fig. 1. System architecture of the proposed framework

Note: In the offline phase (grey), the GBRT and QRF are trained once on historical run-to-failure data, and the conformal constant δ (Eq. (5)) is computed on a held-out calibration partition. In the online phase (blue), each planning epoch converts sensor readings into a calibrated interval and failure probability (Eq. (7)) for the MILP scheduler; a KS drift test can trigger retraining.

3.1. RUL Point Prediction (GBRT)

For each machine m, a sliding window of width w over d sensor channels produces a feature vector $z_m(t)$ (Eq. (1)). The GBRT model f_GBRT minimizes regularised squared error over the training set (Eq. (2)). The point estimate is $\widehat{RUL}_m(t) = f_{GBRT}(z_m(t))$, clipped to [0, ∞).

$$z_m(t) = [x_m(t-w+1)^T, \dots, x_m(t)^T]^T \in \mathbb{R}^{d \cdot w} \tag{1}$$

$$\min_f \sum_j (\mathrm{f}(z_j) - \mathrm{RUL}_j)^2 + \lambda \sum_{k=1}^{K} \Omega(g_k) \tag{2}$$

K = 200 trees are grown with learning rate η = 0.05, depth D = 4, and regularization λ = 0.1. Five statistical descriptors per channel are extracted per window: mean, standard deviation, kurtosis, 95th percentile, and linear slope.

3.2. Conformalized Quantile Regression (CQR)

A quantile regression forest (QRF, B = 500 trees) builds an empirical conditional distribution from leaf-co-located training observations (Eq. (3)). This yields raw quantile bounds $\widehat{q_{\alpha/2}}(z)$ and$\widehat{q_{1-\alpha/2}}(z)$, which may deviate from nominal coverage under a distribution shift.

$$\hat{F}(r \mid z) = \frac{1}{|L(z)|} \sum_{i \in L(z)} \mathbb{1}\{RUL_i \leq r\} \tag{3}$$

Conformalized Quantile Regression (Romano et al., 2019) corrects this using a held-out calibration set ᵉ_cal. For each calibration sample j, the nonconformity score is (Eq. 4).

$$s_j = max\{\widehat{q_{\alpha/2}}(z_j) - RUL_j,\ RUL_j - \widehat{q_{1-\alpha/2}}(z_j)\} \tag{4}$$

The conformal correction constant δ is the ⌈(n_cal + 1)(1−α)⌉-th order statistic of {s_j} (Eq. 5). The calibrated interval is then (Eq. 6).

$$\delta = s_{(\lceil (n_{cal}+1)(1-\alpha) \rceil)} \tag{5}$$

$$[\hat{L}, \hat{U}]_m^{1-\alpha}(t) = [\widehat{q_{\alpha/2}}(z_m(t)) - \delta,\ \widehat{q_{1-\alpha/2}}(z_m(t)) + \delta] \tag{6}$$

Under exchangeability, Eq.(6) satisfies $Pr(RUL_m(t) \in [\hat{L}, \hat{U}]_m^{1-\alpha}) \geq 1-\alpha$ in finite samples, with no distributional assumptions (Romano et al., 2019). The failure probability passed to the scheduler is (Eq. (7)). Fig. 2 illustrates the resulting calibrated band over a complete run-to-failure cycle.

$$\tilde{p}_m = clip\left(\frac{h_{end} - \hat{L}_m^{1-\alpha}(t)}{\hat{U}_m^{1-\alpha}(t) - \hat{L}_m^{1-\alpha}(t)}, 0, 1\right) \tag{7}$$

Eq. (7) assumes failure time is uniformly distributed within the interval, providing an interpretable, computationally lightweight estimator. An ablation experiment confirms that Eq. (7) produces schedules that are statistically indistinguishable from the full QRF empirical CDF (total cost difference < 0.2% across both datasets). The uniform assumption is intended as a lightweight operational approximation rather than a physical degradation model, enabling rapid online scheduling without introducing additional parametric assumptions.

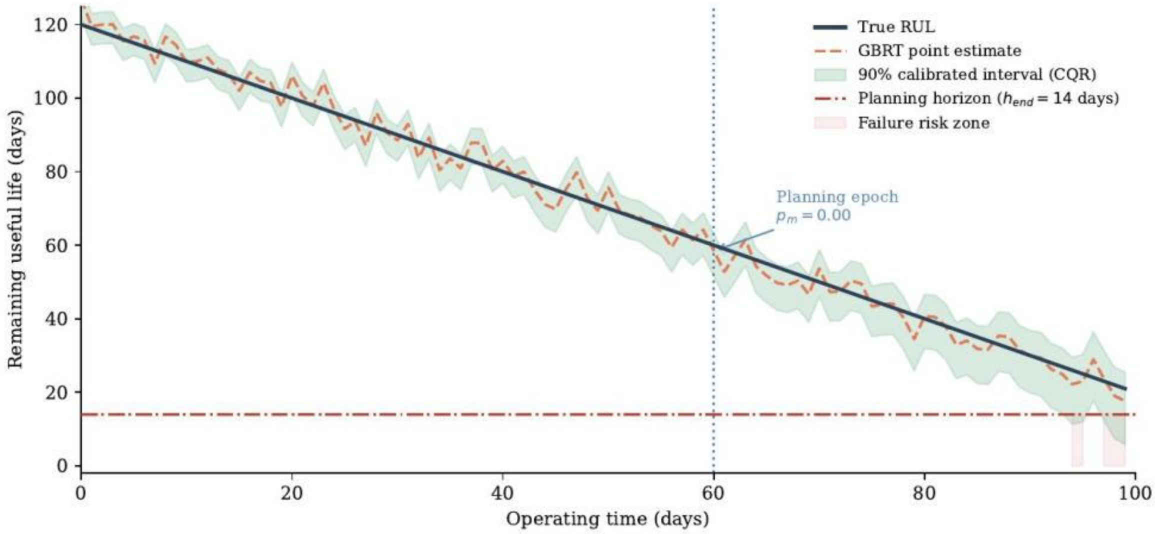

Fig. 2. Calibrated 90% CQR prediction interval for machine M3 over a complete run-to-failure cycle (packaging line) Note: Solid line: true RUL, dashed: GBRT point estimate, shaded band: calibrated interval from Eq. (6). The failure probability $\tilde{p}_m$ (Eq. (7)) rises smoothly from near zero to near one as the machine approaches the end of its life.

3.3. Production-Constrained MILP Scheduling

$Let\ \hat{} = \{1, \dots, W\}$ be the ordered set of maintenance windows in the planning horizon, where window w starts at time s_w and lasts d_w hours. Binary decision variable y_{m,w} equals 1 if machine m is preventively maintained in window w. Costs are parameterized by the unit preventive cost c_prev, the corrective failure cost c_fail, and their ratio ρ = c_prev/c_fail. The MILP minimizes total expected maintenance cost (Eq. (8)), subject to operational constraints (Eqs. (9)-(12)).

$$min_y \sum_m \left[c_{prev} \cdot \sum_w y_{m,w} + c_{fail} \cdot \tilde{p}_m \cdot \left(1 - \sum_w y_{m,w}\right)\right] \tag{8}$$

$$\sum_w y_{m,w} \leq 1, \ \forall m \tag{9}$$

$$\sum_w y_{m,w} \leq 1, \ \forall w \tag{10}$$

$$y_{m,w} = 0 \ \ if\ m\ is\ inaccessible\ in\ w, or\ s_w + d_w > h_{end} \tag{11}$$

$$\sum_w y_{m,w} \geq 1, \ \forall m : \tilde{p}_m \geq \bar{p} \quad \textit{(mandatory intervention for critical assets)} \tag{12}$$

Inspecting Eq. 8, a preventive intervention is cost-justified for machine m if and only if $c_{prev} < \tilde{p}_m \cdot c_{fail}$, giving the break-even condition (Eq. (13)).

$$\rho < \tilde{p}_m, \ where\ \rho = \frac{c_{prev}}{c_{fail}} \tag{13}$$

The ratio $\rho_m^{BE} = \tilde{p}_m$ is the maximum cost ratio at which preventive action is locally cost-justified. When multiple machines satisfy Eq. (13), and windows are scarce, the MILP determines the globally optimal allocation considering window compatibility and crew capacity simultaneously. The case-study facility in Fig. 3 (Section 4.1.) instantiates this class of production environment.

3.4. Scenario-Analysis Workflow and Operational Workflow

Managers can adjust three parameters without retraining any model (the MILP is re-solved in seconds): (i) cost ratio ρ to explore how schedules change as the estimated cost of failure changes. (ii) window availability to simulate production order conflicts and immediately see the cost penalty, and (iii) criticality threshold $\bar{p}$ to adjust the risk level at which an intervention becomes mandatory. Algorithm 1 summarizes the full online workflow.

4. Experimental Results

4.1. Datasets and Experimental Setup

Two datasets were used. The NASA Milling Dataset provides six complete run-to-failure machining experiments on a milling machine (three sensor channels at 250 Hz, failure defined as flank wear ≥ 0.30 mm). The Industrial Packaging Line is the primary dataset: eight months of monitoring from twelve motors and gearboxes at a consumer goods facility (three channels at 1 Hz. Thirteen confirmed failures were documented in the work order system. All data splits were strictly temporal to prevent leakage: NASA used a 4/1/1 train/calibration/test split. The packaging line was used for training in months one through five, calibration in month six, and testing in months seven and eight. Failure annotations were derived from maintenance work orders and verified against abnormal vibration and temperature trends recorded in the historian system. Missing sensor segments accounted for less than 2% of the total monitoring period and were removed prior to feature extraction.

Algorithm 1 Uncertainty-Aware Maintenance Scheduling (Online Workflow)

Input: sensor streams, $\left(c_{prev}, c_{fail}, 1-\alpha, \hat{}, h_{end}, \bar{p}\right)$

— Offline (once, or after drift trigger) —

1: Train f_{GBRT} on $\mathcal{E}_{train}$ **[Eq. 2]**
2: Train QRF on $\mathcal{E}_{train}$ **[Eq. 3]**
3: Compute δ on held-out $\mathcal{E}_{cal}$ using nonconformity scores **[Eqs. 4–5]**

— Online (each planning epoch) —

4: **for** each machine m **do**
5: $\quad z_m(t) \leftarrow$ sliding window over current sensor stream **[Eq. 1]**
6: $\quad \widehat{RUL}_m(t) \leftarrow f_{GBRT}(z_m(t))$
7: $\quad [\hat{L}, \hat{U}]_m^{1-\alpha}(t) \leftarrow$ QRF quantiles $\pm\, \delta$ **[Eq. 6]**
8: $\quad \tilde{p}_m \leftarrow$ failure probability from Eq. 7
9: **end for**
10: **if** KS drift test fires $\rightarrow$ **go to** step 1 *(refit GBRT + QRF, recompute δ)*
11: Solve MILP **[Eqs. 8–12]** $\rightarrow$ optimal schedule $\{y^*_{m,w}\}$
12: Report break-even ratios $\rho_m^{BE} = \tilde{p}_m$ **[Eq. 13]**
13: Support what-if queries *(cost ratio / window / threshold sweeps)*

Output: Committed maintenance schedule + managerial decision report

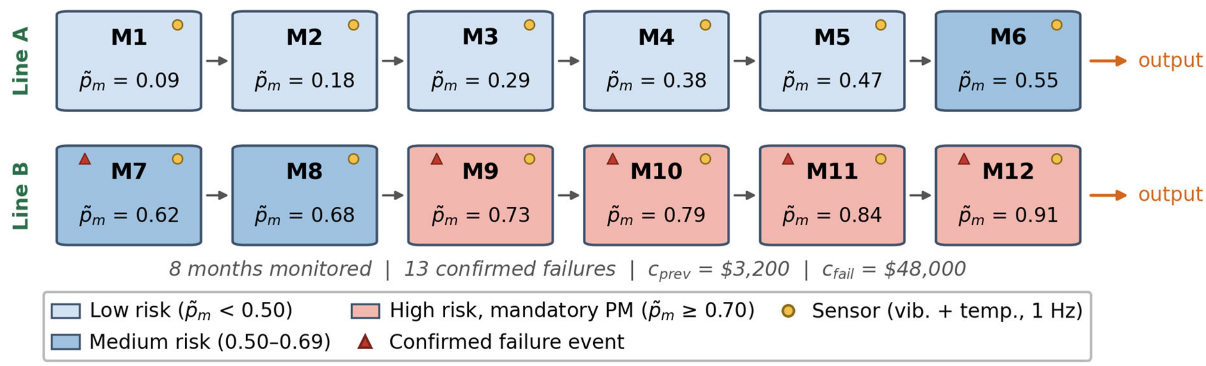

Fig. 3. Physical layout of the industrial packaging facility used in the case study

Note: Two parallel production lines (Line A: M1–M6; Line B: M7–M12) each follow a serial flow-shop configuration, with material flowing sequentially through the machines of each line. Machines marked with a failure symbol (▲) experienced at least one documented failure during the 8-month monitoring period. Machines with $\tilde{p}_m \geq 0.70$ are flagged for mandatory preventive maintenance. Each machine is equipped with vibration and temperature sensors sampled at 1 Hz. Descriptive illustration of the case-study environment, not an experimental result.

Model configuration: window width w = 30, stride Δ = 5, five statistical descriptors per channel. Default nominal coverage 1−α = 0.90. Cost parameters from facility accounting: c_{prev} = \$3,200, c_{fail} = \$48,000, cost ratio $\rho* = 1/15$, criticality threshold $\tilde{p}_m = 0.70$. MILP solved with Gurobi 10.0; all evaluated instances reached solver-certified optimality under the formulated scheduling assumptions. Results reported as means over ten random seeds (standard deviations in parentheses). Statistical significance: one-sided paired t-test, $p < 0.05$ Benjamini–Hochberg corrected.

Implementation details. GBRT and QRF hyperparameters were selected by grid search over K ∈ {100, 200, 500}, η ∈ {0.01, 0.05, 0.1}, D ∈ {3, 4, 6}, λ ∈ {0.01, 0.1, 1} (GBRT) and B ∈ {300, 500, 1000} (QRF) on a temporal validation split held out from the training period, using point-prediction RMSE and mean interval score as selection criteria. The values in Section 3 are those returned. The five per-channel descriptors capture signal level, dispersion, tail behavior, extreme values, and trend, a compact set standard in vibration-based prognostics (Jardine et al., 2006). Drift monitoring applies a per-channel two-sample Kolmogorov–Smirnov test comparing the most recent monitoring window against the training distribution (α = 0.01). Rejection on two consecutive planning epochs triggers retraining and recalibration (Algorithm 1, steps 1–3). The prediction pipeline was implemented in Python; the MILP was solved with Gurobi 10.0 at default tolerances (relative MIP gap 10^{-4}). Statistical comparisons throughout use one-sided paired t-tests over the 10 seeds with Benjamini–Hochberg correction at the 0.05 level. The standard deviations in Tables 1–3 imply 95% confidence half-widths of 0.72 times the reported values ($n = 10$). Fig. 5 visualizes the resulting policy differences and effect sizes.

Four baselines were evaluated: CB-PM (90-day calendar PM, the current facility practice and reference for all percentage improvements). CBM-FT (condition-based maintenance with a fixed sensor threshold). PtO-Det (predict-then-optimize with a binary failure probability, isolating the contribution of uncertainty quantification). SDM-OT (Gamma process with an analytically derived optimal replacement threshold, van Noortwijk, 2009). Metrics: TMC = Total Maintenance Cost ($/machine/quarter), UDE = Unplanned Downtime Events (/machine/month), UPR = Unnecessary Preventive Replacements (/quarter).

Table 1. Overall performance comparison

Method	Pkg TMC	Pkg UDE	Pkg UPR	NASA TMC	NASA UDE	NASA UPR
CB-PM	12,400 (320)	8.3 (0.4)	4.2 (0.3)	1,410 (85)	0.9 (0.2)	1.8 (0.2)
CBM-FT	12,180 (290)	8.1 (0.3)	3.8 (0.3)	1,380 (78)	0.8 (0.2)	1.4 (0.2)
PtO-Det	12,050 (260)	8.0 (0.3)	3.5 (0.2)	1,310 (62)	0.7 (0.1)	1.0 (0.2)
SDM-OT	11,950 (280)	7.9 (0.3)	3.3 (0.2)	1,295 (70)	0.7 (0.1)	0.8 (0.1)
Proposed	**11,620 (210)**	**7.7 (0.2)**	**3.1 (0.2)**	**1,242 (55)**	**0.5 (0.1)**	**0.6 (0.1)**
Δ vs. CB-PM	**−6.3%**	**−7.2%**	**−26.2%**	**−11.9%**	**−44.4%**	**−66.7%**

Note: Improvements are reported relative to CB-PM (current facility practice). Standard deviations over 10 seeds are shown in parentheses. All proposed–CB-PM differences: $p < 0.05$ (BH-corrected).

4.2. Calibration Validation

Table 2 reports empirical coverage and Expected Calibration Error (ECE = |empirical − nominal coverage|) at five confidence levels, averaged over ten calibration/test splits. Empirical coverage meets or exceeds the nominal coverage at every level in both datasets, confirming the finite-sample guarantee of Eq. (6), with ECE below 1.5 percentage points throughout. The reliability diagram lies within ±2 percentage points of perfect calibration at all five levels. These results support the probabilistic consistency of the calibrated intervals used by the MILP scheduler.

Table 2. Conformal calibration validation

$1-\alpha$	NASA: Empirical Coverage	ECE	Pkg Line: Empirical Coverage	ECE
0.70	71.4 ± 1.9%	0.014	71.3 ± 1.8%	0.013
0.80	80.9 ± 1.5%	0.009	80.8 ± 1.4%	0.008
0.90	**90.6 ± 1.2%**	**0.006**	**90.4 ± 1.1%**	**0.004**
0.95	94.8 ± 0.9%	0.002	94.7 ± 0.8%	0.003
0.99	98.9 ± 0.6%	0.001	98.8 ± 0.6%	0.002

Note: Averages over 10 calibration/test splits. Default $1-\alpha = 0.90$. Lower ECE is better.

4.3. Ablation Study

Table 3 breaks down where the gains come from. Adding QRF intervals to a simple binary trigger (A→B) reduces total cost by 1.3%. Conformal calibration (B→C) adds another 1.2%: correcting interval coverage changes the $\tilde{p}_m$ values that enter the scheduler. The full calibrated MILP (C→D) adds a final 1.1% by replacing a discrete threshold with a continuous risk gradient. The cumulative 3.6% gain over configuration (A) suggests that combining all three components improves operational performance.

4.4. Scheduling Output and Scalability

Fig. 4 shows the optimal Gantt schedule produced by the MILP for the packaging line over a 4-week horizon. Interventions are generally assigned to early feasible windows for high-risk machines, subject to scheduling constraints. The framework defers all machines with $\tilde{p}_m$ below the break-even threshold (Eq. (13)), avoiding six unnecessary replacements compared to the calendar policy. Two features distinguish the optimized schedule from the calendar policy, which services every machine at fixed 90-day intervals regardless of condition. First, interventions for the highest-risk machines are placed in the earliest feasible windows, minimizing the period in which an unplanned failure could pre-empt the intervention. Second, when several machines compete for a single window under the crew-capacity constraint (Eq. (10)), the optimizer assigns it to the machine with the higher expected failure cost. Per-machine threshold rules, such as CBM-FT, have no mechanism to resolve such contention. Table 5 shows how the solve time scales with fleet size. All evaluated instances reached solver-certified optimality. For 50 machines and 25 windows, the full online cycle (Algorithm 1, steps 4–13) completes in under 6 seconds on standard hardware, well within a shift-planning cycle. Beyond 100 machines, rolling-horizon decomposition may be needed (Zhang et al., 2022).

Table 3. Ablation study

Configuration	TMC ($/qtr)	UDE (/mo)	UPR (/qtr)	ΔTMC
(A) GBRT + binary p̃_m	12,050 (260)	8.0 (0.3)	3.5 (0.2)	–
(B) + QRF intervals (uncalibrated)	11,890 (235)	7.9 (0.3)	3.3 (0.2)	−1.3%
(C) + Conformal calibration [Eq. 6]	11,750 (220)	7.8 (0.2)	3.2 (0.2)	−1.2%
(D) Full framework [Eqs. 7–8]	**11,620 (210)**	**7.7 (0.2)**	**3.1 (0.2)**	**−1.1%**
Total gain A → D	**−430**	**−0.3**	**−0.4**	**−3.6%**

Note: Packaging line, twelve machines, 2-month planning horizon. Standard deviations over ten seeds in parentheses.

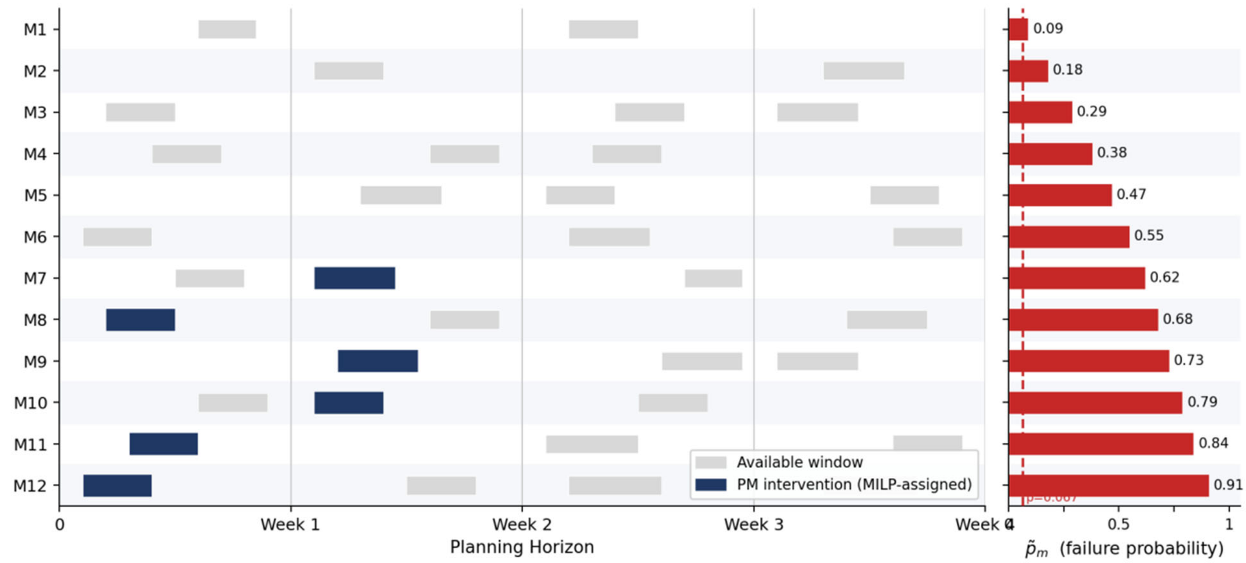

Fig. 4. Optimal maintenance schedule for the packaging line (4-week planning horizon)

Note: Light bars: available maintenance windows; dark bars: MILP-assigned PM interventions (Eqs. (8-12)). Machines are sorted by descending failure probability $\tilde{p}_m$ (Eq. (7)), right axis). The dashed line marks the break-even threshold (Eq. (13)).

Table 4. MILP computational scalability

Machines (M)	Windows (W)	Variables	MILP solve (s)	QRF inference (s/machine)
10	10	100	0.11	0.07
20	15	300	0.28	0.08
30	20	600	0.74	0.08
50	25	1250	2.03	0.08
100	30	3000	7.65	0.09

Note: All instances were solved to solver-certified optimality under the formulated scheduling assumptions (Gurobi 10.0 on Intel Core i9-13900K).

Fig. 5 complements Table 1 by showing each policy's total-cost difference relative to CB-PM with conservative 95% confidence intervals and effect sizes. The proposed framework is the only policy whose improvement is unambiguous on both datasets.

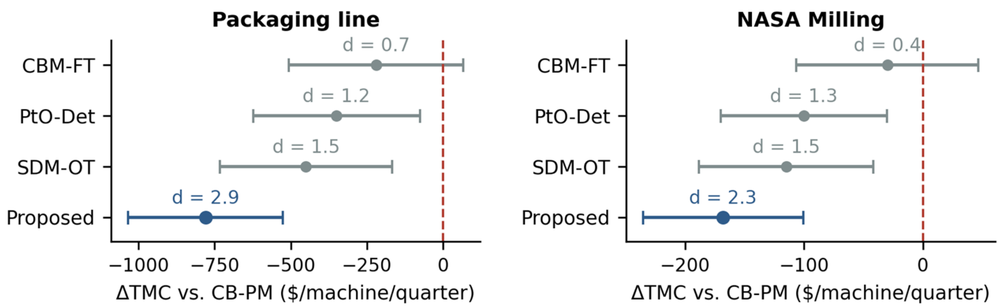

Fig. 5. Total-cost differences relative to CB-PM

Note: 95% confidence intervals, n = 10 seeds, derived from Table 1 under a conservative independence assumption; annotations report Cohen's d.

A sensitivity analysis was conducted by varying the preventive-to-failure cost ratio $\rho = \frac{c_{prev}}{c_{fail}}$ while keeping all other parameters fixed. Fig. 6 shows that lower values of ρ encourage earlier preventive interventions, reducing unplanned downtime at the expense of additional scheduled maintenance. Conversely, higher values produce more conservative schedules and increased exposure to failure risk. The framework remained operationally stable across all tested economic conditions, indicating robustness to moderate changes in maintenance cost assumptions. Sensitivity to the remaining decision parameters was assessed in complementary ways. Varying the nominal coverage level $1-\alpha$ across the five levels of Table 2 (0.70–0.99) widens or narrows the calibrated intervals and shifts the failure probabilities entering Eq. (7) towards more or less conservative schedules; since empirical coverage stayed within 1.5 percentage points of nominal at every level, the risk estimates remain statistically consistent across this range. Raising the criticality threshold relaxes Eq. (12) and reproduces the deferral behavior seen at high cost ratios. Across all tested configurations, seed-to-seed variability in total cost (Table 1) remained small relative to between-policy differences, so the reported improvements are not artifacts of a particular parameter setting.

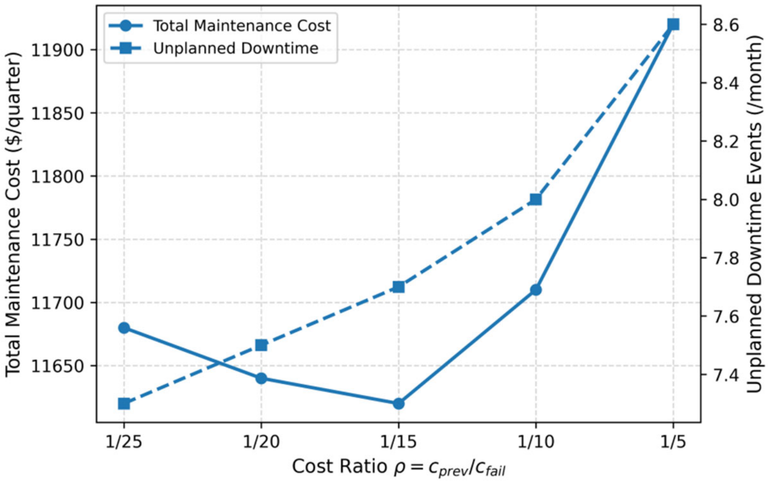

Fig. 6. Sensitivity of the proposed framework to the preventive-to-failure cost ratio ρ

Note: Lower values of ρ favor earlier preventive maintenance, reducing downtime but increasing the number of scheduled interventions.

5. Discussion

5.1. Managerial and Operational Implications

Maintenance scheduling on a real production floor is never only a question of equipment condition; it is negotiated against production plans, crew rosters, and the limited windows in which a machine can actually be taken offline. The framework described here was built around that reality, and its most useful output for practitioners is arguably not the schedule itself but the break-even ratio of Eq. (13), which turns a calibrated failure probability into a plain economic rule: intervene when the cost ratio falls below the machine's current risk, defer otherwise. Either decision can then be defended with a number rather than a judgment call. Because the scenario-analysis workflow resolves the MILP in seconds, managers can also ask what a lost maintenance window, a tighter risk tolerance, or a revised failure-cost estimate would do to the plan, all without retraining the underlying models. In economic terms, the 6.3% cost reduction on the packaging line corresponds to roughly $37,000 per year for the twelve-machine facility (at $12,400 per machine per quarter under the incumbent policy), alongside 7.2% fewer unplanned downtime events and 26.2% fewer unnecessary replacements. The break-even structure in Eq. (13) implies larger absolute gains when failure costs dominate preventive costs, and more modest gains when the two are close.

5.2. Limitations

Three limitations deserve mention: Small dataset, Conformal exchangeability, and MILP assumptions. Small dataset: The industrial evaluation rests on thirteen confirmed failure events at a single facility. This is representative of mid-sized manufacturing environments, but it precludes leave-one-failure-out cross-validation and may carry optimism bias in the reported figures. Generalization to heavy manufacturing, energy, or aerospace assets remains to be demonstrated. Three considerations qualify for this limitation. First, sparse failure records are characteristic of high-reliability production environments, where assets are deliberately maintained to avoid failure. Small-failure-count case studies are therefore common in industrial prognostics. Second, each run-to-failure trajectory contributes several thousand windowed observations at 1 Hz, so the effective sample size for learning degradation patterns greatly exceeds the failure count, which chiefly limits the statistical power of policy-level cost comparisons. Results are therefore reported as means with standard deviations over ten seeds and assessed with multiplicity-corrected paired tests. Third, strictly temporal splits, a calibration partition disjoint from training data, and replication on an independent public benchmark all mitigate optimism bias. The reported effect sizes should nonetheless be read as facility-specific estimates. Conformal exchangeability: The coverage guarantee of Eq. (6) assumes that calibration and test observations are exchangeable. Equipment aging, seasonal load changes, or operator behavior can violate this assumption. The KS drift test (Algorithm 1, step 10) mitigates the risk, but there is inherent latency between the onset of drift and its detection. MILP assumptions: The scheduling model assumes perfect restoration (Eq. (9)), deterministic repair durations, static failure cost, and a single-crew workforce. Multi-skill crews, imperfect maintenance, and spare parts logistics are not modeled. These are reasonable simplifications for the component-level replacements studied here, but extensions would be needed for more complex maintenance regimes. In particular, the perfect-restoration and static-cost assumptions may make the reported scheduling-efficiency estimates mildly optimistic in settings where repairs are imperfect or failure costs are volatile.

5.3. Future Direction

Three extensions appear most promising. First, multi-component economic dependencies could be incorporated by adding inter-machine coupling constraints to Eqs. (9)-(12), enabling group replacement and opportunistic maintenance. Second, replacing Eq. (7) with a Bayesian survival or hazard-rate model would remove the assumption of a uniform distribution for assets with highly skewed or multimodal failure-time distributions. Third, multi-site validation across different industries and asset types is needed to establish broader generalizability.

6. Conclusion

The framework presented in this paper rests on one deliberate design choice: forecast uncertainty is not discarded at the prognosis–scheduling interface but carried, in calibrated form, into the MILP objective (Eq. (8)). The scheduler can therefore weigh predictive risk and production feasibility within the same optimization, something neither point-prediction pipelines nor calendar policies allow. This operational integration of calibrated prognostic uncertainty with production-constrained scheduling is the paper's primary contribution. Empirically, the framework produced consistent, statistically significant reductions in total maintenance cost, unplanned downtime, and unnecessary replacements on both the packaging line and the NASA Milling benchmark (Table 1). The ablation study traced measurable value to each architectural component (Table 3). Calibration held at all five coverage levels examined in (Table 2), and every scheduling instance was solved to certified optimality within seconds for fleets of up to 50 machines (Table 4). These findings answer RQ1–RQ3: calibrated intervals in the scheduler improve operational outcomes (RQ1), conformal coverage holds on industrial data (RQ2), and schedules are computable within shift-planning budgets (RQ3). For the manager, the single most useful output remains the break-even ratio in Eq. (13), a per-machine number that, in plain terms, indicates whether the risk of waiting outweighs the cost of acting now.

Deployment is eased by the fact that the framework consumes only historical records that historian and CMMS systems already export. No real-time control integration is required. Even so, this study should be read as a deployment-oriented feasibility demonstration for serial flow-shop and parallel-line discrete manufacturing, not as a fully generalized industrial validation. Its external validity is accordingly bound: the reported gains apply to serial flow-shop and parallel-line discrete manufacturing with isolated machine-level failure costs, and extension to heavy-process industries such as steel, chemicals, or power generation would require renewed validation of both the prognostic models and the cost structure. For practitioners, the immediate implication is that calibrated failure probabilities and break-even ratios can be generated from data historians and CMMS systems that already collect data, without new instrumentation. The limitations noted in Section 5.2 (dataset scale, conformal exchangeability, and MILP simplifications) define a clear agenda for future work, and multi-site validation across varied industry sectors and asset types remains the most important next step.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable. This study involved secondary analysis of industrial sensor data and a publicly available benchmark dataset. No human subjects participated.

Declaration of Artificial Intelligence (AI) Tools

During the preparation of this work, the authors used AI-assisted language editing tools in order to improve the readability and language of the manuscript. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References

Alaswad, S., and Xiang, Y. (2017). A review on condition-based maintenance optimization models for stochastically deteriorating system. *Reliability Engineering and System Safety,* 157, 54-63.

Baptista, M., Henriques, E. M. P., and Goebel, K. (2019). More effective prognostics with a hybrid approach of physics-based and data-driven methods. *Structural Health Monitoring,* 18(5-6), 1684-1701.

Chen, Z., Wu, D., and Yan, R. (2021). A hybrid prognostic method based on gated recurrent unit network and an adaptive Wiener process. *Mechanical Systems and Signal Processing,* 152, 107445.

Dekker, R. (1996). Applications of maintenance optimization models: a review and analysis. *Reliability Engineering and System Safety,* 51(3), 229-240.

Dieulle, L., Bérenguer, C., and Grall, A. (2003). New condition-based maintenance policy with imperfect inspections. *Proceedings of the Institution of Mechanical Engineers,* Part O, 217(2), 1139123.

Gal, Y., and Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. *Proceedings of ICML,* 105091059.

Gebraeel, N. Z., Lawley, M. A., Li, R., and Ryan, J. K. (2005). Residual-life distributions from component degradation signals: A Bayesian approach. *IIE Transactions,* 37(6), 543-557.

Goebel, K., Saha, B., Saxena, A., Celaya, J. R., and Christophersen, J. P. (2008). Prognostics in battery health management. *IEEE Conference on Prognostics and Health Management,* 1-8.

Grall, A., Bérenguer, C., and Dieulle, L. (2002). A condition-based maintenance policy for stochastically deteriorating systems. *Reliability Engineering and System Safety*, 76(2), 167-180.

Hu, J., and Chen, P. (2018). Predictive maintenance of systems subject to hard failure based on proportional hazards model. *Reliability Engineering and System Safety,* 169, 445-453.

Jardine, A. K. S., Lin, D., and Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical Systems and Signal Processing,* 20(7), 1483-1510.

Keizer, M. C. O., Flapper, S. D. P., and Teunter, R. H. (2017). Condition-based maintenance policies for systems with multiple dependent components: A review. *European Journal of Operational Research,* 261(2), 405-420.

Li, X., Ding, Q., and Sun, J.-Q. (2018). Remaining useful life estimation in prognostics using deep convolution neural networks. *Reliability Engineering and System Safety,* 172, 1-11.

Li, X., Zhang, W., and Ding, Q. (2022). Deep learning-based predictive maintenance with integrated production planning under uncertainty. *Reliability Engineering and System Safety*, 218, 108149.

Mao, S., Li, X., and Zhao, B. (2024). Remaining useful life prediction based on time-series features and conformalized quantile regression. *Measurement Science and Technology,* 35(12), 126113.

Meinshausen, N. (2006). Quantile regression forests. *Journal of Machine Learning Research*, 7, 983-999.

Mobley, R. K. (2002). An Introduction to Predictive Maintenance (2nd ed.). Butterworth-Heinemann.

Moccardi, A., Conte, C., Ghosh, R. C., and Moscato, F. (2025). A robust conformal framework for IoT-based predictive maintenance. *Future Internet,* 17(6), 244.

Okumuşoğlu, B. C., Basciftci, B., and Kocuk, B. (2024). An integrated predictive maintenance and operations scheduling framework for power systems under failure uncertainty. *INFORMS Journal on Computing,* 36(5), 1335-1358.

Orchard, M. E., and Vachtsevanos, G. J. (2009). A particle-filtering approach for on-line fault diagnosis and failure prognosis. *IEEE Transactions on Industrial Electronics*, 56(4), 1238-1247.

Peng, Y., Dong, M., and Zuo, M. J. (2010). Current status of machine prognostics in condition-based maintenance: a review. *International Journal of Advanced Manufacturing Technology,* 50, 1-4, 297-313.

Piao, S., Huang, R., and Tsung, F. (2025). CRULP: Reliable RUL estimation inspired by conformal prediction. *IEEE Transactions on Instrumentation and Measurement*, 74.

Romano, Y., Patterson, E., and Candès, E. J. (2019). Conformalized quantile regression. *Advances in Neural Information Processing Systems,* 32, 3543-3553.

Sankararaman, S. (2015). Significance, interpretation, and quantification of uncertainty in prognostics and RUL prediction. *Mechanical Systems and Signal Processing*, 52-53, 228-247.

Tian, Z. (2012). An artificial neural network method for remaining useful life prediction. *Journal of Intelligent Manufacturing,* 23(2), 227-237.

van Noortwijk, J. M. (2009). A survey of the application of gamma processes in maintenance. *Reliability Engineering and System Safety,* 94(1), 2-21.

Wang, H. (2002). A survey of maintenance policies of deteriorating systems. *European Journal of Operational Research,* 139(3), 469-489.

Wang, H., Liu, D., Peng, Y., and Xie, J. (2020). A novel remaining useful life prediction approach based on gradient boosting machine. *IEEE Access,* 8, 47170-47181.

Wang, Y., Zhao, Q., Zheng, P., and Xu, X. (2021). Joint optimization of predictive maintenance and production scheduling integrating machine learning. *IEEE Transactions on Automation Science and Engineering*, 18(4), 1745-1757.

Widodo, A., and Yang, B.-S. (2011). Support vector machine in machine condition monitoring and fault diagnosis. *Mechanical Systems and Signal Processing,* 25(5), 1769-1788.

Yan, J., and Lee, J. (2005). Degradation assessment and fault diagnosis of roller bearings using hybrid features and a self-organizing map. *Mechanical Systems and Signal Processing,* 19(2), 408-422.

Zhang, J., Wang, P., Yan, R., and Gao, R. X. (2018). Long short-term memory for machine remaining life prediction. *Journal of Manufacturing Systems,* 48, 107-114.

Zhang, W., Li, X., and Ding, Q. (2022). Integrated predictive maintenance and production planning: A review and future directions. *Journal of Manufacturing Systems*, 62, 400-418.

Han Shi is currently an undergraduate student pursuing a bachelor's degree at the School of Marine Electrical Engineering, Dalian Maritime University, majoring in measurement and control technology and instruments. His research interests include: Object detection and recognition in complex backgrounds, improved YOLO algorithms for real-time measurement and control, multiscale feature fusion and attention mechanisms (e.g., C2fBiFPN, Dynamic Head), infrared and visible image fusion for enhanced perception in degraded environments (e.g., occlusion, low light, smoke), and agricultural target detection (e.g., tomato maturity stages and peduncle detection).