International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(2) 2026 pp. 2121-2132
https://doi.org/10.70917/ijcisim-2026-5486
Article

Study on Hybrid XG Boost–LSTM Predictor with a Digital-Twin Monte Carlo Virtual Check for Risk-Aware Scheduling in MSME CNC Manufacturing cluster

K Parameswari[1], Dr S Veni[2], Dr. V.S.Thangarasu[3*]

[1]Research Scholar, Department of Computer Science, Karpagam Academy of Higher Education, Coimbatore, India
Email: kpriyamraj6481@gmail.com
Orchid ID: 0009-0002-3955-8602
[2]Professor, Department of Computer Science, Karpagam Academy of Higher Education, Coimbatore, India
Email: venics@kahedu.edu.in
Orcid ID:0000-0002-2999-8463
[3*]Professor & Dean Academics, Department of Mechanical Engineering, Sree Sakthi Engineering College, Karamadai, Coimbatore, India
Email: vstme05@gmail.com
Orcid ID: 0000-0002-3499-5195

Abstract: Smart scheduler proposed in this research is using Risk aware scheduling methods for micro Small and medium enterprises involved in precision components manufacturing processes using Computerised Numerical Control (CNC) machines for a group of operators in an industrial cluster is challenging because of its non-regular availability and also based on different parametric requirements. The normal rule-based scheduling approaches fails in solving such issues for machine condition, tool wear, queue congestion, and deadline-related uncertainty. The study shows most hybrid machine-learning models cannot predict process risk, and do not reflect in ever evolving real-time state of the physical manufacturing system. To address this limitation, this study proposes a two-level decision-fusion framework that integrates a hybrid XG Boost–LSTM risk predictor with a Digital Twin-based Monte Carlo virtual verification mechanism. The hybrid model generates a predictive delay-risk estimate, while the Digital Twin simulates candidate job execution under stochastic tool-wear, machine-availability, and queue-congestion conditions to obtain an independent risk estimate. These estimates are combined using an interpretable weighted fusion model, which is incorporated into a priority-aware machine-selection and scheduling strategy. The proposed framework was evaluated using a synthetic six-machine MSME-style precision-CNC manufacturing environment through independent replications, multi-scenario experiments, alternative fusion models, and Monte Carlo convergence analysis. The results show that fused scheduling with improved pooled on-time delivery to 74.0%, compared with 72.3% for the hybrid predictor alone and 70.3% for availability-based scheduling. The improvement over predictor-only scheduling was statistically significant compared to previous research achievements. Furthermore, the predictive and Digital Twin risk estimates exhibited near-zero correlation indicates complementary information. Linear fusion also matched or outperformed Bayesian and logistic-regression fusion alternatives while maintaining low computational latency, with worst-case scheduling latency below 50 ms for six candidate machines. The findings demonstrate the feasibility of combining predictive intelligence with Digital Twin-based virtual verification for real-time, risk-aware scheduling in MSME manufacturing Key words: Digital twin, cyber-physical systems, hybrid machine learning, decision-level fusion, risk-aware scheduling, Monte Carlo simulation, Industry 5.0, small and medium enterprises.

Keywords: Digital twin, cyber-physical systems, hybrid machine learning, decision-level fusion, risk-aware scheduling, Monte Carlo simulation, Industry 5.0, small and medium enterprises.

1. INTRODUCTION

Most industrial clusters are operating with high end CNC (Computer Numerical Control) machines for precision component manufacturing and are have to share machines for optimized utilization across job orders, and that means an uncertainty prevails in the market either the machines are underloaded and at times overloaded, the different parameters like tool, wear, dimensional changes, queues build up, delivery deadlines get tight. Most scheduling in this setting is still rule-based machine availability, cost, historical trust none of which sees delay or failure risk coming before it hits production. There has been progress on the prediction side. Hybrid predictors that pair gradient-boosted feature selection with sequential learning reach strong offline accuracy on process-risk prediction; a hybrid XGBoost–LSTM model validated based on the publications of Business Process Intelligence Challenge 2011 conference dataset reached 0.92 accuracy, beating standalone baselines ($p < 0.05$) [1]. But these models output one prediction and stop there. Nothing checks that prediction against what the machine is actually doing right now, and the two can drift apart. Digital-twin architectures solve a related but different problem — they keep a synchronized virtual copy of the asset for real-time simulation of candidate actions [2]–[4] — yet they are rarely paired with an independently trained predictor that could act as a second, verifying signal.

This paper closes that gap. R_t is a digital-twin Monte Carlo virtual check that simulates job execution under live wear and queue-congestion state, and it fuses with an existing predictor's output R_h as $D_{final} = \beta \cdot R_h + (1-\beta) \cdot R_t$. We built the architecture as a discrete-event scheduling simulator and evaluated it on a synthetic MSME (Micro Small and Medium Enterprises) style precision CNC units' scenario, reporting effect size, confidence intervals, and statistical power at every step. Three contributions follow: a Monte Carlo virtual-check formulation that turns out to be empirically uncorrelated with an existing predictor's output, pointing to complementary rather than duplicated information; a two-level fusion architecture and scheduling algorithm tested against a naive baseline and two non-linear fusion alternatives; and a statistically confirmed, weight-specific improvement in on-time delivery, checked across independent replications, a multi-scenario sweep, and a Monte Carlo convergence study.

2. RELATED WORKS

Gradient-boosted models handle structured feature selection well but were not built for sequence; recurrent architectures do the opposite, capturing temporal dependencies while selecting structured features only weakly. Put the two together and the combination beats either one alone — that is the basis of the predictor extended here [1]. The catch is that these models are validated offline, against static datasets, with no mechanism to check a prediction against live physical state. That is the gap this work targets. Digital-twin architectures take a different approach: they maintain a synchronized virtual replica of a physical asset for real-time simulation and decision support [2]–[4], and have recently found use in production scheduling [10] and multi-sensor predictive maintenance [9]. Kobayashi and Alam [20] pair a digital twin with explainable-AI techniques for remaining-useful-life estimation. Ferraro et al. [21] compare SHAP and LIME explainability for hard-disk-drive predictive maintenance and conclude SHAP works better. Self-adaptive middleware built on autonomic-computing principles [5], [6] contributes the feedback-control-loop pattern this work's fusion stage follows. What none of this literature does is pair a digital twin with an independently trained predictive model as a second risk signal. Combining imperfectly correlated estimates, on the other hand, is well established — entropy-weighted TOPSIS [8], weighted averaging [11], [12], Bayesian fusion. Geragersian et al. [22] fuse an LSTM predictor with Bayesian uncertainty quantification for navigation sensor fusion, which is structurally close to the R_h/R_t combination proposed here, just in a different domain. This work uses linear weighted fusion because it is interpretable, and tests it directly against Bayesian and logistic-regression alternatives in Section VI rather than assuming it wins by default. Deep reinforcement learning is another thread: it has been applied to dynamic and flexible job-shop scheduling, learning dispatching policies from simulated or historical environments [13]–[16], increasingly with graph neural network state representations [23]. Those methods optimize a policy end-to-end instead of fusing independently computed risk estimates, which makes them a longer-term architectural alternative rather than a direct competitor to the fusion approach evaluated here. Industry 5.0, meanwhile, emphasizes human-centric, resilient automation deployment [17], with particular attention to the resource constraints SMEs and MSMEs face when adopting digital tools [18], [19]. A systematic review of business process automation in SMEs [7] finds that adoption failure usually comes down to organizational and infrastructural barriers, not a lack of available technology — a point that shapes the feasibility discussion in Section VII.

Work	Prediction	Twin/Simulation	Fusion	Domain
[2]	–	DT reference model	–	General manufacturing
[3]	–	CPS-based workshop twin	–	Simulated smart workshop
[10]	–	Cognitive DT scheduler	Twin-internal	Multi-objective scheduling
[8]	–	–	Decision-level (TOPSIS)	Multi-source manufacturing data
[9]	Multi-sensor tool-wear model	–	Sensor-level	Predictive maintenance
[13],[14]	Deep RL dispatching policy	–	–	Dynamic job-shop scheduling
[20]	–	XAI-augmented DT	Explainability, not fusion	Remaining-useful-life
[22]	Bayesian-LSTM	–	Bayesian (non-linear)	Navigation sensor fusion
This work	Hybrid XGBoost–LSTM [1]	Monte Carlo virtual check	Decision-level, linear, tuned β	MSME precision-CNC scheduling

TABLE 1. Comparison of relative to related digital-twin, fusion, and scheduling literature.

Predictive models such as [1] give a risk estimate but no live verification mechanism. Digital-twin and CPS architectures [2]–[4] give real-time simulation but no predictive signal of their own upstream. Nothing in the surveyed work combines a validated hybrid predictor with an independent digital-twin simulation layer, fuses the two at decision level, and evaluates that fusion with effect size, confidence intervals, and power analysis on an MSME-relevant scenario. This paper fills that gap and reports whether, and by how much, such fusion actually improves scheduling outcomes.

3. Proposed Method

A candidate assignment is a (job, machine, time) tuple. R_h(j,m) is the hybrid predictor's delay-risk probability, computed from machine m's current structured features. R_t(j,m,t) is different: it is the fraction of N Monte Carlo trials in which simulating job j's execution on machine m — under sampled sensor-level wear uncertainty and queue-congestion propagation from jobs already committed on m — results in completion after j's due date. Together they give a fused decision score, D_final(j,m) = β·R_h(j,m) + (1−β)·R_t(j,m,t), β ∈ [0,1]. Notation is summarized in Table II.

Fusion Architecture

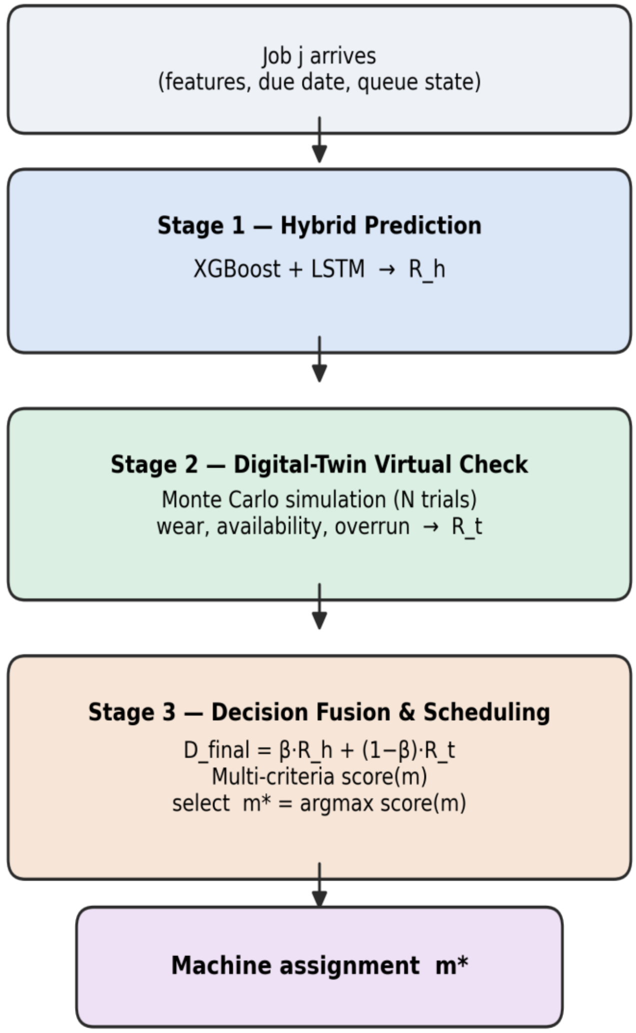

Fig. 1. Three-stage architecture: hybrid prediction, digital-twin virtual check, decision fusion, and risk-aware scheduling.

Symbol	Definition
j, M	A job; the set of machines capable of executing j
t	Current simulation time (job arrival instant)
R_h(j,m)	Hybrid XGBoost–LSTM predicted delay-risk probability ∈ [0,1]
R_t(j,m,t)	Digital-twin Monte Carlo virtual-check risk estimate ∈ [0,1]
N	Monte Carlo trials per R_t evaluation (N = 300)
β	Fusion weight on R_h; (1−β) is the weight on R_t
D_final(j,m)	Fused risk score = β·R_h + (1−β)·R_t
m*	Selected machine, m* = argmax_{m∈M} score(m)

TABLE 2. Notations used in this research paper.

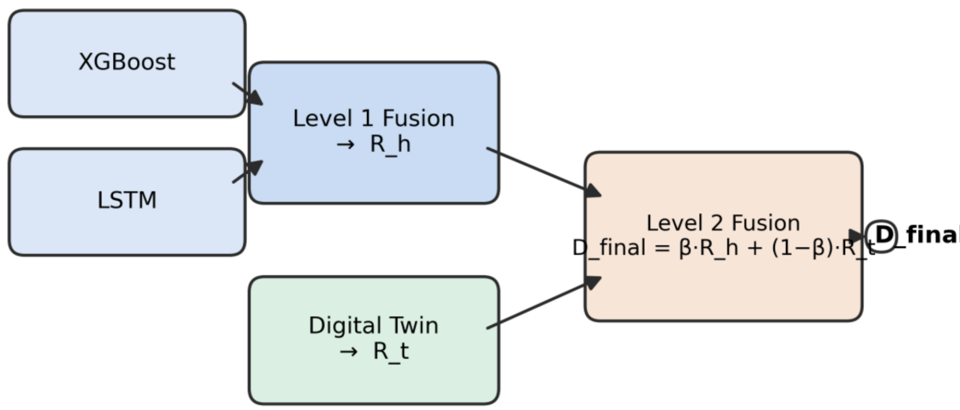

Fig. 2. Two-level fusion: Level 1 combines XG Boost and LSTM into R_h [1]; Level 2 combines R_h with the digital twin's R_t into D_final.

Algorithm

```
Input: job j, capable machines M, time t, weight β, trials N
Output: selected machine m*

for each machine m in M:
   R_h ← HybridModel.predict(m.features)
   late ← 0
   for trial = 1 to N:
      wear ← clip(m.tool_wear + Normal(0, 0.04), 0, 1)
      avail ← t
      for qjob in m.committed_queue where qjob.end > t:
         p_overrun ← 0.15 + 0.5 × wear
         if Random() < p_overrun: avail ← max(avail, qjob.end + Uniform(10,40))
      start ← max(t, avail) + j.setup_time
      proc ← max(5, j.processing_time × (1 + Normal(0, 0.10)))
      if start + proc > j.due_date: late ← late + 1
   R_t ← late / N
   D_final(m) ← β·R_h + (1−β)·R_t
   score(m) ← w_a·avail + w_q·queue + w_l·load +
            w_d·deadline + w_t·trust + w_r·(1−D_final(m))

return m* = argmax_{m in M} score(m)
```

Algorithm 1. Digital-twin virtual check and two-level fusion.

3.1 Design Rationale

Linear fusion won out for two reasons: it is interpretable, and it has just one parameter to tune. We do not assume it is the best option — Section VI-B tests it directly against Bayesian and logistic-regression alternatives. The weighting terms w_r and w_d scale with job priority, so urgent jobs weight predicted risk and deadline slack more heavily than routine ones do. R_t itself comes from Monte Carlo sampling rather than a closed-form calculation, because propagating queue-congestion risk through an arbitrary committed queue has no simple analytic solution once stochastic overrun and wear uncertainty are in the mix. $N = 300$ follows a Bernoulli-proportion standard-error bound (≈ 0.029 worst case), and Section VI-D confirms this empirically.

Algorithm 1 does one R_h inference and N Monte Carlo trials for each of $|M|$ candidate machines, which puts per-job complexity at $O(|M|\cdot(N\cdot\bar{q} + C_h))$, where $\bar{q}$ is average queue length and C_h is the cost of one hybrid-model inference. In measurements, R_h evaluation took 4–5 ms and R_t evaluation took roughly 2 ms at $N = 300$. Even in the worst case — six candidate machines — total latency stayed under 50 ms, comfortably within real-time scheduling requirements.

Predictive Model Adaptation and Decision Architecture. The predictive component of the proposed framework is based on the Hybrid XGBoost–LSTM architecture established in our previous study, where XGBoost is used for structured feature selection and LSTM is used to model sequential dependencies in event data. For the present CNC manufacturing application, the same hybrid architecture is adapted to the CNC-specific job and machine-state representation, and the manuscript should explicitly describe the actual CNC-specific training/retraining procedure used to obtain the delay-risk estimate Rh. The predictive output Rh represents the estimated probability of delay for assigning job j to candidate machine m. This predictive risk is subsequently complemented by an independently generated Digital-Twin-based Monte Carlo risk estimate Rt, representing the simulated operational uncertainty associated with the same assignment. The two risk signals are fused as $D_{final} = \beta R_h + (1-\beta)R_t$, where Dfinal is minimized as a risk measure. The resulting risk is incorporated into the overall scheduling score, where $(1-D_{final})$ represents the risk-related benefit and the complete scheduling score is maximized to select the candidate machine. The Digital Twin used in this study is a simulation-based virtual representation of the CNC manufacturing environment; therefore, the present experiments should not be interpreted as validation using a physically connected CNC Digital Twin or live shop-floor sensor data. The stochastic parameters used in the Monte Carlo simulation, including wear, queue overrun, machine availability and processing-time variation, should be explicitly defined and justified, and all scheduling weights should be reported with their numerical values, normalization procedure and rationale. The predictive risk and Digital-Twin risk should be described as complementary risk signals unless formal statistical independence has been demonstrated.

3.2 Experimental Setup

Predictive engine: a gradient-boosted classifier paired with a multilayer-perceptron regressor, trained on a synthetic 4,000-event structured-and-sequential log that mirrors [1] (ROC-AUC 0.755 and 0.719).

- Scenario: a six-machine MSME-style precision-CNC cluster, one machine carrying elevated wear as a simulated fault, 24 jobs per 8-hour shift with mixed priority and due-date slack — except where the robustness sweep (Section VI-C) varies these settings.
- Design: an exploratory sweep ($n = 40$) over $\beta \in \{0.0, 0.2, 0.4, 0.5, 0.6, 0.8, 1.0\}$ pointed to $\beta = 0.2$ for confirmatory testing. Two independent replications then followed ($n = 240$, $n = 230$; pooled $n = 470$), each evaluating baseline, R_h-only, and fused scheduling.
- Non-linear fusion baselines: a predictive-discrimination comparison on 1,440 held-out training and 1,440 held-out evaluation tuples (Bayesian log-odds pooling and logistic-regression meta-fusion against linear fusion), followed by a downstream scheduling-outcome comparison on a disjoint 40-trial held-out set.

- Multi-scenario sweep: six configurations combining job volume ∈ {16,24,32} with degraded machine ∈ {M1, M2}, each run for n = 30 paired trials.
- Monte Carlo convergence: R_t recomputed repeatedly at N ∈ {50,100,300,1000,3000,10000}.
- Every design is paired within condition. All data is synthetic — realistic and learnable in structure, but not drawn from real MSME field data.

3.3 Experimental Design and Statistical Validation.

The experimental evaluation should clearly define the unit of analysis for each trial, the number of independent simulation replications, the random seeds used, and the separation between model development, parameter selection and final evaluation. The β sweep should be treated as an exploratory/model-selection experiment, with β = 0.2 reported only as the best-performing value under the evaluated synthetic CNC conditions rather than as a universally optimal weight; multiple-comparison considerations should be addressed where applicable. The Bayesian and logistic fusion experiments should clearly identify their training, validation and held-out test datasets and demonstrate that no information from the final scheduling evaluation was used during model selection. The statistical comparison between scheduling strategies should include justification for the paired-test assumptions and, where appropriate, a non-parametric robustness analysis such as the Wilcoxon signed-rank test. The reported computational latency should be described as simulation/experimental-environment latency rather than demonstrated real-time CNC deployment. Finally, the proposed decision process should be explicitly interpreted as an autonomous decision-making mechanism: XGBoost–LSTM provides the data-analytics-based prediction, Digital-Twin simulation provides virtual risk assessment, risk fusion provides intelligent decision support, and the automated evaluation and selection of the highest-scoring feasible machine constitutes the autonomous machine-selection decision. Claims concerning the research gap should be stated conservatively as limited prior work rather than as an absolute absence of existing studies, and all citations and reference numbering should be re-audited to ensure that the previous Hybrid XGBoost–LSTM study is cited correctly.

4. RESULTS

We have obtained the results for Fused scheduling at β = 0.2 significantly outperformed R_h-only scheduling ($t = 4.820$, $p = 1.9\times10^{-6}$, $d = 0.222$, power = 0.998). β = 0.4, tested in just one of the two replications (n = 230), did not reach significance ($p = 0.223$) — so the confirmed benefit is best treated as specific to β ≈ 0.2 rather than the broader range. R_h and R_t were also uncorrelated in practice ($r = -0.005$, n = 480 decisions; R_h mean 0.20, SD 0.10; R_t mean 0.27, SD 0.42), which fits with the two carrying complementary rather than overlapping information.

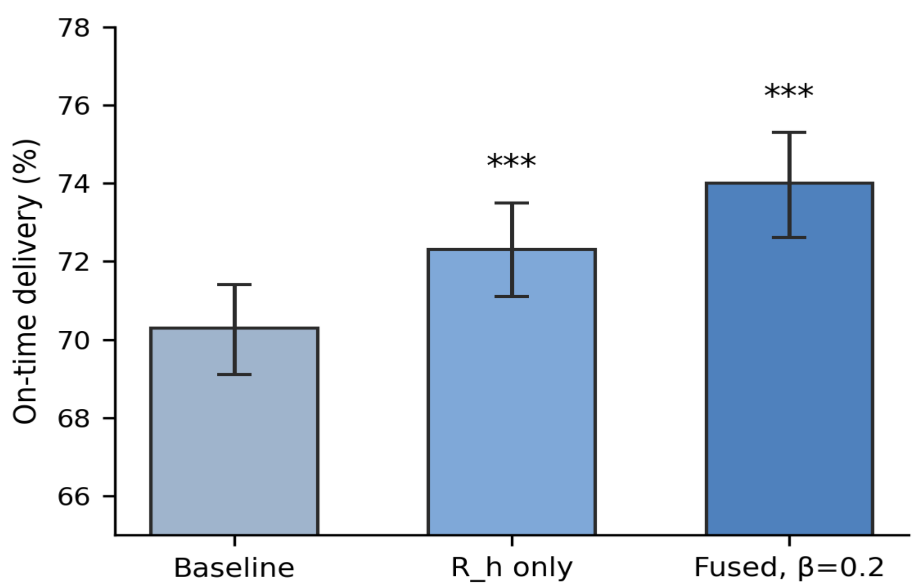

Fig. 3. On-time delivery by scheduling strategy (n = 470 pooled paired trials; *** p < 0.0001 vs. baseline).

Condition	On-time % (95% CI)	Cohen's d	p (vs. baseline)	Power
Baseline	70.3 [69.1, 71.4]	—	—	—
R_h only	72.3 [71.1, 73.5]	0.39	$<10^{-15}$	1.000
Fused, β = 0.2	74.0 [72.6, 75.3]	0.45	$<10^{-19}$	1.000

TABLE 3. Pooled results, n = 470 paired trials.

4.1 Non-Linear Fusion Alternatives

Both non-linear alternatives were fit on 1,440 held-out training tuples and evaluated on a disjoint set of 1,440: Bayesian log-odds pooling and logistic-regression meta-fusion, tested against linear fusion (β = 0.2). AUC came out at 0.983, 0.983, and 0.976 respectively. A separate logistic meta-fusion model, trained on 720 tuples and tried as the scheduler's risk term on 40 held-out trials, beat the baseline ($t = 2.05$, $p = 0.047$, $d = 0.33$) but not R_h-only ($p = 0.60$), and was itself beaten by linear fusion ($t = -2.05$, $p = 0.047$, $d = -0.32$).

Comparison	Metric	Linear	Bayesian	Logistic Meta
Predictive AUC (n = 1,440)	AUC	0.983	0.983	0.976
Scheduling outcome (n = 40)	On-time %	75.5%	—	74.4%

TABLE 4. Non-linear fusion alternatives: predictive and scheduling-outcome comparisons.

Configuration	Baseline	R_h-only	Fused	p (vs. baseline)
16 jobs, fault A	83.1%	84.4%	89.2%	6.5×10^{-6}
16 jobs, fault B	83.1%	84.4%	87.7%	1.3×10^{-4}
24 jobs, fault A	69.9%	72.4%	74.6%	2.8×10^{-4}
24 jobs, fault B	68.2%	69.0%	72.5%	0.0018
32 jobs, fault A	54.2%	54.3%	55.6%	0.32
32 jobs, fault B	52.7%	51.9%	55.7%	0.082

TABLE 5. Multi-scenario sweep, n = 30 paired trials per configuration.

At moderate job volume — 16 to 24 jobs per shift — the benefit is significant across both fault locations. Push to 32 jobs per shift, where cluster load and baseline utilization both climb, and it attenuates.

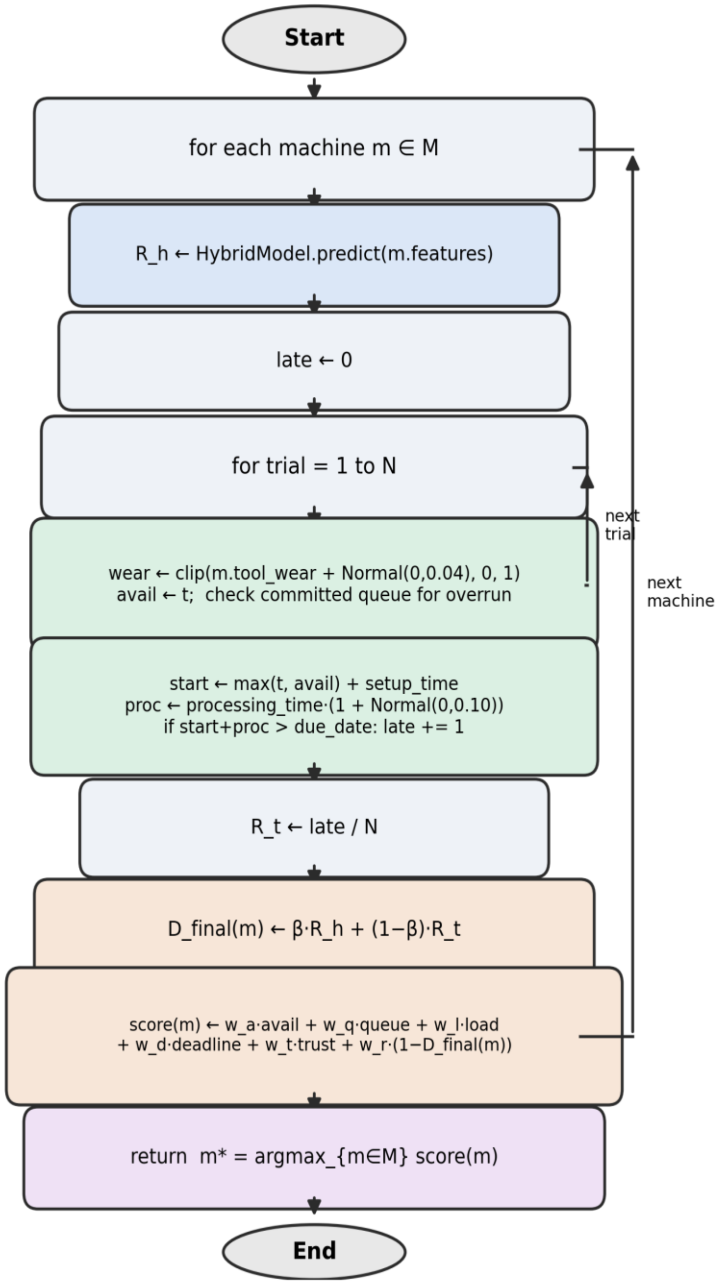

Fig. 4. Algorithm 1 flowchart, per candidate machine.

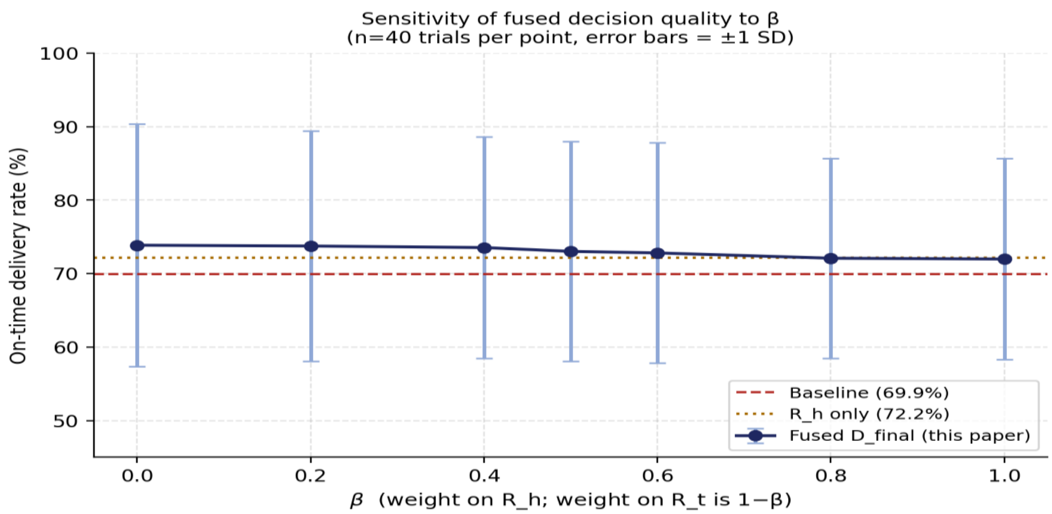

Fig. 5. Exploratory sweep (n = 40) across the full β grid.

4.2 Monte Carlo Convergence

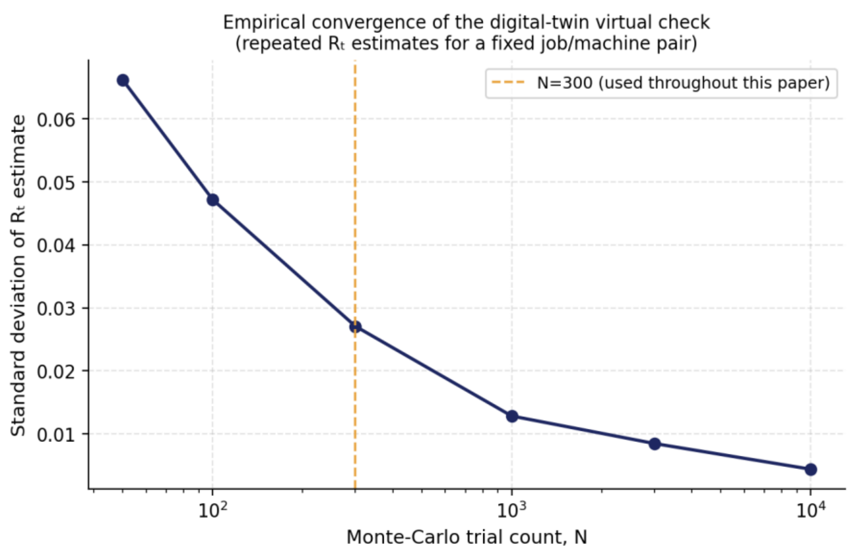

Fig. 6. Empirical standard deviation of the R_t estimate versus Monte Carlo trial count N.

At N = 300 the empirical standard deviation came out to 0.027, close to the 0.029 theoretical bound, and it kept falling as N grew further (0.013 at N = 1000, 0.004 at N = 10000). N = 300 looks like a reasonable balance of precision against cost.

5. DISCUSSION

The confirmed benefit is weight-specific, not universal: significant at β = 0.2, unconfirmed at β = 0.4; robust at moderate job volume, weaker under high cluster load. For deployment, that argues for fixing β near 0.2 rather than treating it as freely tunable, and for capacity planning that accounts for a shrinking fusion benefit as cluster utilization climbs. Neither Bayesian nor logistic-regression fusion beat the linear rule across two independent evaluation protocols, so the simpler, more interpretable rule looks like the better choice for operator-facing deployment — and it costs nothing in accuracy. The null result at β = 0.4 is probably about statistical power rather than a genuine absence of effect; the effect size at β = 0.2 is modest to begin with (d = 0.22), so a Type II error is still plausible. Latency, at least, is not the bottleneck: Section IV-D already shows the architecture is fast enough for real-time deployment. What remains is empirical validation on real machine data, not computational feasibility.

A live digital-twin virtual check needs per-machine sensor telemetry and a maintained model of each machine's committed queue. Many MSMEs do not have integrated IoT infrastructure [18], [19]; vibration or current-clamp sensors, combined with queue data already collected through facility-sharing coordination, offer a cheaper way in than full industrial IoT instrumentation. The paired trial design controls for job-stream variance across conditions, which covers internal validity, but the Monte Carlo virtual check's internal parameters were set by plausibility argument, not fit to external data. Every result here comes from synthetic evidence, so how well it maps onto real MSME shop-floor dynamics is assumed rather than measured — that is the external/ecological validity limitation. On-time delivery is also the only outcome measure used; it says nothing about cost or operator workload, which is a construct-validity gap. And statistically: paired t-tests assume roughly normal within-pair differences, which is unverified at these sample sizes, and the β-sweep was not corrected for multiple comparisons. It should be read as exploratory characterization, not a set of independent confirmatory tests.

5.1 LIMITATIONS

Every result here rests on synthetic evidence — no real MSME machine telemetry was involved. The confirmed fusion benefit is specific to $\beta \approx 0.2$ and attenuates at higher job volume, and that boundary was only characterized at $n = 30$ per configuration. Bayesian and logistic-regression alternatives were tested and did not beat linear fusion, though Dempster–Shafer combination remains untried. And the latency figures reported are order-of-magnitude measurements from one implementation on one platform, not a controlled benchmark.

6. CONCLUSION

This paper set out a two-stage fusion architecture pairing a hybrid XGBoost–LSTM predictor with a digital-twin Monte Carlo virtual check for risk-aware scheduling in MSME precision-CNC manufacturing. The main result: fused scheduling at $\beta = 0.2$ significantly beats the predictor used alone ($p = 1.9\times10^{-6}$, $d = 0.22$), a weight-specific but statistically confirmed improvement, and the two fused signals are empirically complementary rather than redundant ($r = -0.005$). On the practical side, the architecture needs no new machine hardware beyond the sensor telemetry a facility-sharing deployment already collects, and its measured sub-50-ms latency confirms it is fast enough for real-time scheduling. The linear fusion rule used here matched or beat two non-linear alternatives, which argues for a simpler, more interpretable system that operators can actually trust. Three things deserve priority next: validation on real MSME machine data, an adaptively learned fusion weight instead of a fixed β, and explainable-AI presentation of the fused risk score for shop-floor operators.

7. Acknowledgment

The authors used a large language model to assist with manuscript drafting, experimental software implementation, and statistical analysis, and reviewed and verified all reported content and results.

REFERENCES

1. W. Dai, V. N. Dubinin, J. H. Christensen, V. Vyatkin, and X. Guan, "Toward self-manageable and adaptive industrial cyber-physical systems with knowledge-driven autonomic service management," IEEE Transactions on Industrial Informatics, vol. 13, no. 2, pp. 725–736, 2017.
2. A. Ferraro, A. Galli, V. Moscato, and G. Sperlì, "Evaluating explainable artificial intelligence tools for hard disk drive predictive maintenance," Artificial Intelligence Review, vol. 56, no. 7, pp. 7279–7314, 2023.
3. P. Geragersian, I. Petrunin, W. Guo, and R. Grech, "Uncertainty-based sensor fusion architecture using Bayesian-LSTM neural network," in Proceedings of the AIAA SciTech Forum, 2023.
4. M. Ghobakhloo and M. Iranmanesh, "Digital transformation success under Industry 5.0: A need for digitally capable people and sustainable practices," Corporate Social Responsibility and Environmental Management, vol. 30, no. 3, pp. 964–985, 2023.

5. G. Herrera Granados, T. Misaka, J. Herwan, H. Komoto, and Y. Furukawa, "An experimental study of multi-sensor tool wear monitoring and its application to predictive maintenance," International Journal of Advanced Manufacturing Technology, vol. 133, pp. 3415–3433, 2024.
6. IBM Corporation, Autonomic Computing: IBM's Perspective on the State of Information Technology. IBM Research, 2001.
7. K. Kobayashi and S. B. Alam, "Explainable, interpretable, and trustworthy AI for an intelligent digital twin: A case study on remaining useful life," Engineering Applications of Artificial Intelligence, vol. 129, p. 107620, 2024.
8. H. Lee and H. Yang, "Cognitive digital twin for multiobjective production scheduling," Journal of Manufacturing Science and Engineering, vol. 147, no. 9, p. 091004, 2025.
9. J. Leng et al., "Industry 5.0: Prospect and retrospect," Journal of Manufacturing Systems, vol. 65, pp. 279–295, 2022.
10. J. Leng, H. Zhang, D. Yan, Q. Liu, X. Chen, and D. Zhang, "Digital twin-driven manufacturing cyber-physical system for parallel controlling of smart workshop," Journal of Ambient Intelligence and Humanized Computing, vol. 10, no. 3, pp. 1155–1166, 2019.
11. L. Liu, X. Wan, J. Li, W. Wang, and Z. Gao, "An improved entropy-weighted TOPSIS method for decision-level fusion evaluation system of multi-source data," Sensors, vol. 22, no. 17, p. 6391, 2022.
12. R. Liu, R. Piplani, and C. Toro, "A deep multi-agent reinforcement learning approach to solve dynamic job shop scheduling problem," Computers & Operations Research, vol. 159, p. 106294, 2023.
13. R. Liu, R. Piplani, and C. Toro, "Deep reinforcement learning for dynamic scheduling of a flexible job shop," International Journal of Production Research, vol. 60, no. 13, pp. 4049–4069, 2022.
14. L. Lv, C. Zhang, J. Fan, and W. Shen, "Deep reinforcement learning for job shop scheduling problems: A comprehensive literature review," Knowledge-Based Systems, vol. 321, p. 113725, 2025.
15. S. Moreira, H. S. Mamede, and A. Santos, "Business process automation in SMEs: A systematic literature review," IEEE Access, vol. 12, pp. 75832–75864, 2024.
16. K. Parameswari, S. Veni, and V. S. Thangarasu, "Optimizing business process automation using hybrid XGBoost-LSTM model," International Journal of Artificial Intelligence and Machine Learning, vol. 6, no. 3S, pp. 38–47, 2026.
17. Q. Qi, D. Zhao, T. W. Liao, and F. Tao, "Modeling of cyber-physical systems and digital twin based on edge computing, fog computing and cloud computing towards smart manufacturing," in Proceedings of the ASME International Manufacturing Science and Engineering Conference, 2018.
18. S. Sangarathas and S. Shanmugathas, "Human-centric digital transformation in Industry 5.0: A phenomenological study of family-owned SME in Sri Lanka," Electronic Journal of Information Systems in Developing Countries, vol. 91, no. 4, p. e70026, 2025.
19. F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang, and F. Sui, "Digital twin-driven product design, manufacturing and service with big data," International Journal of Advanced Manufacturing Technology, vol. 94, nos. 9–12, pp. 3563–3576, 2018.
20. L. Wan, L. Fu, C. Li, and K. Li, "Flexible job shop scheduling via deep reinforcement learning with meta-path-based heterogeneous graph neural network," Knowledge-Based Systems, vol. 296, p. 111940, 2024.
21. B. Wu, J. Zeng, H. Shi, X. Zhang, G. Shi, and Y. Qin, "Multi-sensor information fusion-based prediction of remaining useful life of nonlinear Wiener process," Measurement Science and Technology, vol. 33, p. 105106, 2022.
22. M. Xu, Y. Mei, F. Zhang, and M. Zhang, "Learn to optimise for job shop scheduling: A survey with comparison between genetic programming and reinforcement learning," Artificial Intelligence Review, vol. 58, no. 6, p. 160, 2025.
23. P. Zhang, T. Li, G. Wang, D. Wang, P. Lai, and F. Zhang, "A multi-source information fusion model for outlier detection," Information Fusion, vol. 93, pp. 192–208, 2023.