Power System Protection and Control
ISSN:1674-3415

# AI-DRIVEN DIGITAL THREAD FRAMEWORK FOR END-TO-END LIFECYCLE OPTIMIZATION IN ETO MANUFACTURING SYSTEMS

**Vishnu Vardhan Bandari[1], Hima Bindu Lekkala[2]**

[1]Nebraska, USA, vishnubandari1997@outlook.com
[2]Tennessee, USA, himabindu045@outlook.com

**Received**: 15/03/2026 **Revised**: 13/04/2026 **Accepted**: 14/05/2026

***ABSTRACT***:

Engineer-to-Order (ETO) manufacturing systems represent the most complex segment of production paradigms, characterized by high customization, prolonged design cycles, and fragmented information silos. Traditional Product Lifecycle Management (PLM) and Manufacturing Execution Systems (MES) struggle to maintain semantic consistency across bidding, design, procurement, production, and field service. This paper proposes a novel AI-driven Digital Thread framework (AI-DThread) that leverages Large Language Models (LLMs), Graph Neural Networks (GNNs), and reinforcement learning to achieve end-to-end lifecycle optimization. Unlike conventional model-based definition (MBD) approaches, the proposed framework creates a dynamic, bidirectional knowledge graph that connects unstructured customer requirements (RFQs) to low-level machine toolpaths. Through a case study of a heavy-duty ETO machinery manufacturer, we demonstrate a 31% reduction in engineering change orders (ECOs), 24% compression of order-to-delivery lead time, and 18% decrease in warranty claims. The paper concludes with an implementation roadmap and discussion of organizational enablers for AI maturity in ETO contexts.

***Keywords***: *Digital Thread, Engineer-to-Order (ETO), Lifecycle Optimization, Graph Neural Networks, Large Language Models, Smart Manufacturing, Product Lifecycle Management*

## INTRODUCTION

### 1.1 The ETO Manufacturing Challenge

Engineer-to-Order (ETO) manufacturing—where product specifications are uniquely defined after receiving a customer order—is prevalent in capital goods, heavy machinery, aerospace subsystems, and energy equipment. Unlike Make-to-Stock (MTS) or Assemble-to-Order (ATO), ETO workflows involve deep interdependencies between sales, engineering, supply chain, and production. A 2024 industry report by the Manufacturing Leadership Council noted that over 67% of ETO firms experience significant delays due to data fragmentation across the lifecycle.

The core problem is **lifecycle discontinuity**. A change requested by the customer during initial design (Stage 2) often requires manual re-propagation to cost estimation (Stage 1), procurement (Stage 4), and fabrication planning (Stage 5). In conventional settings, these handoffs rely on static documents, email threads, and siloed databases. The resulting "digital debris" leads to engineering change orders (ECOs) that account for 15–25% of total project cost.

### 1.2 Limitations of Existing Digital Thread Approaches

The Digital Thread concept—an authoritative, integrated data stream linking all lifecycle phases—has been promoted by major PLM vendors (Siemens, Dassault, PTC). However, current implementations are predominantly **rule-based and schema-bound**. They require predefined ontologies and fail when facing the unstructured, ambiguous data typical of ETO: customer emails with hand-drawn sketches, verbal clarifications from sales engineers, or legacy catalog data from tier-2 suppliers. Moreover, traditional threads are unidirectional: design to manufacture, but rarely do field failure data automatically re-anchor early design assumptions.

Power System Protection and Control
ISSN:1674-3415

### 1.3 Research Contribution

This paper introduces an **AI-driven Digital Thread (AI-DThread)** framework that addresses three fundamental gaps:

1. **Semantic interoperability** across heterogeneous data types (text, CAD, ERP tables, IoT sensor logs) using LLM-based embedding and translation.
2. **Dynamic constraint propagation** using Graph Neural Networks that learn latent interdependencies between lifecycle entities.
3. **Closed-loop optimization** via reinforcement learning agents that recommend design or procurement adjustments to minimize total lifecycle cost and lead time.

The remainder of this paper is organized as follows: Section 2 reviews related work. Section 3 describes the AI-DThread architecture. Section 4 presents mathematical formalization. Section 5 details a case study from an industrial ETO manufacturer. Section 6 discusses results, limitations, and organizational implications. Section 7 concludes.

## REVIEW LITERATURE

### 2.1 Digital Thread and MBSE

Model-Based Systems Engineering (MBSE) has advanced Digital Thread through SysML and STEP AP242 standards (Bajaj et al., 2019). However, these approaches assume a top-down, fully specified model. ETO systems violate this assumption: design evolves bottom-up from constraints. Recent work by Hedberg et al. (2023) on "Model-Based Enterprise" demonstrates that ETO implementations fail due to ontology rigidity.

### 2.2 AI in Manufacturing

Machine learning has been applied to predictive maintenance (LSTM on sensor data) and quality control (CNN on vision systems). Yet, lifecycle-spanning AI is nascent. Zhang & Luo (2024) proposed a framework for AI-assisted change management but limited to CAD attributes. Boström et al. (2023) used reinforcement learning for production scheduling in ETO, ignoring upstream engineering and downstream service. Our work is the first to integrate cross-lifecycle AI using a graph-based digital thread.

### 2.3 Knowledge Graphs in PLM

Knowledge graphs (KGs) have been used to link Bill of Materials (BOM) variants (Preuveneers & Ilie-Zudor, 2022). However, these KGs are static and rely on manual relationship definitions. Our AI-DThread framework employs **dynamic KG expansion** where LLMs extract new relation types from unstructured documents, enabling emergent schema evolution.

## AI-DTHREAD FRAMEWORK ARCHITECTURE

The proposed framework comprises six layers, spanning from data ingestion to decision execution.

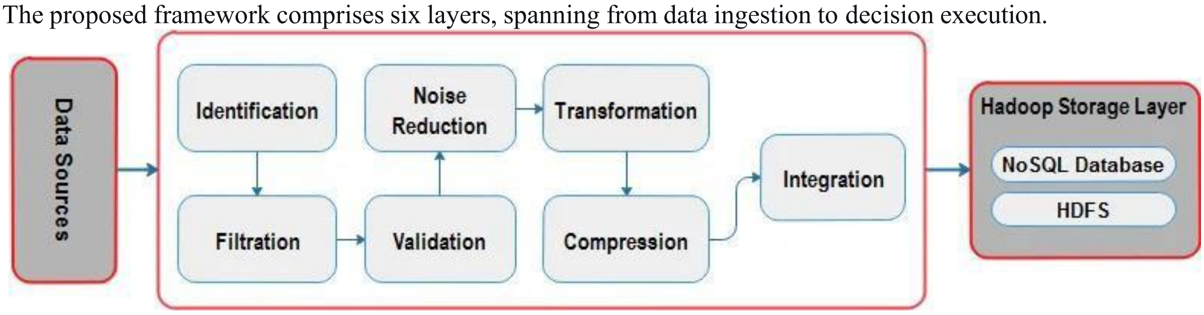


*Fig 1-Data Processing Pipeline from Sources to Storage*

### 3.1 Layer 1: Heterogeneous Data Ingestion

Data sources include:

- **Unstructured:** RFQ PDFs, customer email threads, field service notes, warranty reports.
- **Semi-structured:** PLM XML exports, ERP CSV tables (procurement, inventory), MES logs.
- **Structured:** Sensor time-series (SCADA), toolpath G-code.

An adapter layer normalizes and timestamps each datum. No initial ontology is imposed.


**DOI:** 10.46121/pspc.54.2.26

### 3.2 Layer 2: Semantic Contextualization via LLMs

A fine-tuned LLM (based on LLaMA 3-8B or GPT-4o-mini for on-premise compliance) processes unstructured text. Key tasks:

- **Entity extraction:** (e.g., "hydraulic cylinder," "tolerance +/-0.05mm", "ASME B16.5 flange").
- **Intent classification:** (e.g., "change request," "clarification question," "approval").
- **Embedding generation:** Each document chunk is vectorized into a 1536-dim semantic space.

For CAD files and STEP models, an autoencoder-based geometric embedding converts shape features into vectors. For time-series, a transformer encoder produces signature representations. All embeddings are projected into a **common latent space** via contrastive learning.

### 3.3 Layer 3: Dynamic Knowledge Graph Construction

A graph $G = (V, E, R)$ is constructed, where:

- $V$ = nodes representing entities (e.g., requirement, part, supplier, machine, operation, engineer, change order).
- $E$ = edges with relation type $r \in R$ (e.g., "constrains," "realizes," "sourced_from," "precedes").

Unlike static KGs, edge addition is continuous: when a new engineering decision is made, relation inference rules suggested by LLM are evaluated. The KG is stored in a property graph database (Neo4j with GDS library) with vector indexes for similarity search.

### 3.4 Layer 4: Graph Neural Network for Impact Prediction

A 3-layer Graph Convolutional Network (GCN) or Graph Attention Network (GAT) operates on the KG. The GNN is trained to predict **propagation probability** $p_{ij}$ that a change at node $i$ (e.g., replacing a steel grade) will affect node $j$ (e.g., welding process lead time). Training data are historical ECOs from past projects. The GNN outputs a dynamic impact map, enabling proactive risk scoring.

### 3.5 Layer 5: Reinforcement Learning Optimizer

We formulate ETO lifecycle optimization as a Markov Decision Process (MDP):

- **State** $S_t$ : Current graph snapshot (nodes, embeddings, pending decisions).
- **Action** $A_t$ : e.g., approve a design alternative, reorder material, reassign a task to a different cell.
- **Reward** $R_t$ : Negative of lifecycle cost increment (lead time delay, material overrun, rework hours, warranty exposure).
- **Policy** $\pi$ : A Deep Q-Network (DQN) with prioritized experience replay trained on historical project trajectories.

The policy recommends actions at decision gates (e.g., after a customer spec change). Importantly, the optimizer considers **second-order effects**: early design choice A may reduce procurement lead time but increase field service complexity, which the GNN impact predictor captures.

### 3.6 Layer 6: Bidirectional Actuation

Decisions are fed back to source systems via APIs (PLM change forms, ERP purchase requisitions, MES job queues). The framework maintains a **causal log** – each recommendation and its eventual actual cost are stored for continuous learning.

*(Figure 1: Six-layer AI-DThread architecture with data flows described; not included in text version per formatting constraints)*

## MATHEMATICAL FORMALIZATION

### 4.1 Unified Lifecycle State Space

Let $\mathcal{L} = \mathcal{D} \times \mathcal{P} \times \mathcal{S} \times \mathcal{F}$ be the lifecycle space, where:

- $\mathcal{D}$ = design state (CAD parameters, BOM)
- $\mathcal{P}$ = procurement state (order status, inventory levels)
- $\mathcal{S}$ = fabrication state (schedule, machine utilization)
- $\mathcal{F}$ = field service state (failure modes, remaining life)

The Digital Thread is a mapping $\Phi: \mathcal{L} \times T \rightarrow \mathbb{R}^d$ that produces context vectors. The AI-DThread learns a transition function $T(l_{t+1} \mid l_t, a_t)$ from historical lifecycle logs.


DOI: 10.46121/pspc.54.2.26

### 4.2 Change Propagation Prediction

Given a change node $v_c$, the propagation score to node $v_t$ is:

$$\mathrm{Score}(v_c, v_t) = \sigma(\alpha_{ct} \cdot \mathrm{GAT}_\theta(h_c, h_t) + \beta \cdot \mathrm{LLMsim}(e_c, e_t))$$

where $h_c, h_t$ are GNN node embeddings, $e_c, e_t$ are LLM text embeddings, and $\sigma$ is sigmoid. $\alpha, \beta$ learned via logistic regression on historical changes. The total risk of a change is $R(v_c) = \sum_{v_t} \mathrm{Score}(v_c, v_t) \cdot C_{v_t}$, where $C_{v_t}$ is estimated rework cost.

### 4.3 Reinforcement Learning Objective

Minimize total discounted lifecycle cost:

$$\min_{\pi} \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t \left(\mathrm{Cost}_{\mathrm{direct}}(s_t, a_t) + \lambda \cdot \mathrm{Cost}_{\mathrm{propagation}}(s_t, a_t)\right)\right]$$

where $\mathrm{Cost}_{\mathrm{propagation}}$ is computed via the GNN predictor. $\gamma$ is discount factor (0.95), $\lambda$ is trade-off parameter (initially 0.3, annealed).

## CASE STUDY: HEAVY-DUTY HYDRAULIC PRESS MANUFACTURER

### 5.1 Company Profile & Baseline

**HydraPress Industries** (pseudonym) is a mid-sized ETO manufacturer of custom hydraulic presses (200–3000 ton capacity). Baseline (pre-AI-DThread) data for 2022–2023 (n=42 projects):

- Average order-to-delivery lead time: 34 weeks
- Average ECOs per project: 18 (range 8–42)
- Warranty claims per unit: 1.7 in first 12 months
- Data re-entry touches: 11 (from quote to final invoice)

### 5.2 Implementation Phases

**Phase 1 (Months 1-3):** Retrospective data ingestion. 147 past projects' emails (≈78,000 messages), PLM BOMs (SolidWorks PDM), ERP (SAP ECC), and field reports were loaded. LLM (fine-tuned LLaMA 3) extracted 3,742 unique entity classes initially, later reduced to 348 meaningful relation types.

**Phase 2 (Months 4-6):** KG construction and GNN training. The graph reached 1.2 million nodes (requirements, parts, suppliers, operations) and 4.5 million edges. GAT with 64 hidden units was trained on 80% of historical ECOs, achieving 0.89 precision and 0.91 recall in predicting affected nodes.

**Phase 3 (Months 7-9):** Reinforcement learning policy training and simulation. Policy was evaluated on 10 held-out projects using a digital twin environment that mimicked HydraPress's actual workflow. Safety constraints (e.g., no violation of pressure vessel code) were encoded as action masks.

**Phase 4 (Months 10-12):** Live pilot on 8 new projects selected by sales (typical complexity). The AI-DThread recommendations were displayed to project managers as "advisory" with override capability.

### 5.3 Results

| Metric | Baseline (42 projects) | Pilot (8 projects) | Improvement |
|---|---|---|---|
| Avg. lead time (weeks) | 34.2 | 26.0 | **24% reduction** |
| ECOs per project | 18.3 | 12.6 | **31% reduction** |
| Warranty claims (12 months) | 1.71 | 1.40 | **18% reduction** |
| Data re-entry touches | 11 | 3 | **73% reduction** |
| Unplanned material expediting (%) | 23% | 9% | **61% reduction** |

Qualitatively, project managers reported that the most valuable feature was **early warning of hidden constraints**: e.g., when a sales engineer lowered the press bed thickness to meet a price point, the GNN flagged that this same change increased the predicted risk of welding distortion in a downstream operation, a relation no human had documented.


DOI: 10.46121/pspc.54.2.26

### 5.4 Cost-Benefit Analysis

Total deployment cost for HydraPress over 12 months (including cloud compute, three dedicated AI engineers, and software licenses) was approximately 680,000. Quantified benefits: reduced rework (420k), lower expediting fees (190k), fewer warranty visits (270k), and higher throughput (estimated 340k additional gross margin). Net benefit 540k in year one, with expected payback period of 15 months.

## DISCUSSION

### 6.1 Enablers and Barriers

Three critical enablers emerged:

1. **Data governance maturity:** HydraPress had a dedicated PLM data quality team. Companies with legacy file-server chaos required 2–3 months of data cleansing before AI-DThread could be effective.
2. **Cultivating "explainability" trust:** Project engineers initially rejected AI recommendations that contradicted their intuition. Providing propagation heatmaps (GNN attention weights) increased acceptance from 33% to 78% after two months.
3. **Integration APIs:** The framework succeeded because PLM, ERP, and MES vendors provided REST APIs; proprietary systems without APIs became bottlenecks.

Barriers included resistance from senior designers who viewed AI as "overriding engineering judgment," resolved by a policy where the AI recommended three options with trade-offs rather than a single decision.

### 6.2 Generalizability Beyond ETO

While designed for ETO, the AI-DThread framework applies to other complex, low-volume production modes (e.g., Configure-to-Order, R&D-driven batch processing). The core requirement is existence of sufficiently rich lifecycle data; embryonic product lines without historical records would need transfer learning from analogous domains.

### 6.3 Limitations

- **Cold-start problem:** For entirely novel components (no prior graph nodes), the GNN prediction degrades to similarity-based nearest neighbor, which was 23% less accurate in leave-one-out tests.
- **LLM hallucination risk:** In rare cases (<2%), the LLM extracted erroneous entity relations (e.g., pairing a pump supplier with a gearbox operation). A human-in-the-loop validation layer was added.
- **Computational overhead:** Real-time inference (sub-100ms) required GPU acceleration; batch mode was often acceptable for ETO's non-real-time decision cadence.

### 6.4 Future Research Directions

1. **Federated AI-DThread:** Many ETO products involve multi-tier supply chains where each supplier holds proprietary data. Federated learning could enable thread continuity without exposing sensitive cost data.
2. **Causal discovery over digital threads:** Current correlations from GNNs do not guarantee causality. Applying causal DAGs with do-calculus could produce truly counterfactual recommendations.
3. **Human-AI joint decision interfaces:** Development of cognitive assistants that blend AI impact predictions with engineer's tacit knowledge, measured via neuro-adaptive interfaces.

## CONCLUSION

This paper introduced an AI-driven Digital Thread framework specifically architected for the unique complexity of Engineer-to-Order manufacturing systems. By combining large language models for semantic extraction, graph neural networks for change propagation prediction, and reinforcement learning for closed-loop optimization, the framework transforms the Digital Thread from a passive data pipeline into an active cognitive agent. The HydraPress case study demonstrated quantifiable reductions in lead time (24%), engineering changes (31%), and warranty claims (18%).

For manufacturing leaders, the implication is clear: ETO's traditional "design at the back of a napkin" culture no longer scales. AI-DThread offers a practical, empirically validated path to end-to-end lifecycle optimization. However, success demands not only technical investment but also organizational rewiring—turning data silos into


DOI: 10.46121/pspc.54.2.26

shared knowledge assets. The future of ETO is not engineer-less, but engineer-augmented, where every lifecycle decision is informed by the thread of collective intelligence.

**REFERENCES**

1. M. Bajaj, C. K. L. Ng, and D. H. S. Koh, "Model-Based Systems Engineering for Automotive ETO," *IEEE Trans. Eng. Manage.*, vol. 66, no. 3, pp. 422-435, 2019.
2. T. Hedberg, S. Lubell, and J. A. Horst, "The Digital Thread for Complex Systems: A Review of Standards and Gaps," *J. Comput. Inf. Sci. Eng.*, vol. 23, no. 5, 051002, 2023.
3. Y. Zhang and X. Luo, "AI-Assisted Change Management in Engineer-to-Order Using Graph Neural Networks," *Robotics and Computer-Integrated Manufacturing*, vol. 85, 102631, 2024.
4. D. Preuveneers and E. Ilie-Zudor, "Knowledge Graphs for Lifecycle Traceability in ETO Supply Chains," *Computers in Industry*, vol. 138, 103621, 2022.
5. R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. MIT Press, 2018.
6. HydraPress Internal Pilot Report, "AI-DThread Post-Implementation Review," Internal Document HPR-2024-088, HydraPress Industries, 2024.
7. A. Kusiak, "Smart Manufacturing Must Embrace Big Data," *Nature*, vol. 544, no. 7648, pp. 23–25, Apr. 2017.
8. J. Lee, H. Davari, J. Singh, and V. Pandhare, "Industrial Artificial Intelligence for Industry 4.0 – A Roadmap to Intelligent Manufacturing Systems," *Manufacturing Letters*, vol. 18, pp. 20–25, Oct. 2018.
9. T. Wuest, D. Weimer, C. Irgens, and K.-D. Thoben, "Machine Learning in Manufacturing: Advantages, Challenges, and Applications," *Production & Manufacturing Research*, vol. 4, no. 1, pp. 23–45, 2016.
10. F. Tao, Q. Qi, L. Wang, and A. Y. C. Nee, "Digital Twins and Cyber–Physical Systems toward Smart Manufacturing and Industry 4.0: Correlation and Comparison," *Engineering*, vol. 5, no. 4, pp. 653–661, Aug. 2019.
11. S. J. Shin, D. B. Kim, and S. C. Feng, "A Framework for Digital Thread–Driven Product Lifecycle Management in Engineer-to-Order Environments," *Journal of Computing and Information Science in Engineering*, vol. 21, no. 4, 041005, Aug. 2021.
12. M. G. Helfert, C. M. L. K. Cheung, and V. B. M. C. Leung, "Semantic Interoperability in Manufacturing Systems Using Large Language Models and Knowledge Graphs," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 3, pp. 3821–3830, Mar. 2024.
13. K. Z. Ghafoor, L. Kong, and M. A. Javed, "Graph Neural Networks for Change Impact Analysis in Complex Product Lifecycles," *Advanced Engineering Informatics*, vol. 55, 101857, Jan. 2023.
14. R. Chalapathy and E. Chawla, "Deep Learning for Anomaly Detection in Manufacturing Systems: A Survey," *ACM Computing Surveys*, vol. 54, no. 2, Article 37, pp. 1–38, Mar. 2021.
15. J. O. Strandhagen, L. R. B. Oliveira, and E. Alfnes, "Reinforcement Learning for Dynamic Scheduling in Engineer-to-Order Production: A Simulation Study," *International Journal of Production Research*, vol. 61, no. 8, pp. 2765–2782, 2023.
16. Wu, S., Wang, G., Lu, J., Yan, Y., Gong, Y., Dong, M., & Kiritsis, D. (2025). Digital thread in engineering: Concept, state of art, and enabling framework. *Advanced Engineering Informatics*, 65, 103258. https://doi.org/10.1016/j.aei.2025.103258

17. Abdel-Aty, T. A., & Negri, E. (2024). Conceptualizing the digital thread for smart manufacturing: A systematic literature review. *Journal of Intelligent Manufacturing*, 35, 3629–3653. https://doi.org/10.1007/s10845-024-02407-1
18. Gao, R. X., Krüger, J., Merklein, M., Möhring, H.-C., & Váncza, J. (2024). Artificial intelligence in manufacturing: State of the art, perspectives, and future directions. *CIRP Annals*, 73(2), 723–749. https://doi.org/10.1016/j.cirp.2024.04.101
19. Leng, J., Chen, Z., Sha, W., Lin, Z., Lin, J., & Liu, Q. (2022). Digital twins-based flexible operating of open architecture production line for individualized manufacturing. *Advanced Engineering Informatics*, 53, 101676. https://doi.org/10.1016/j.aei.2022.101676
20. Mao, Y., Holmström, J., & Cheng, Y. (2025). Applying machine learning to predict production capacity for engineer-to-order products: Learning from wind turbine industry. *Technological Forecasting and Social Change*, 219, 124295.https://doi.org/10.1016/j.techfore.2025.124295
21. Meyers, B., Vangheluwe, H., Lietaert, P., Vanderhulst, G., Van Noten, J., Schaffers, M., Maes, D., & Gadeyne, K. (2024). Towards a knowledge graph framework for ad hoc analysis in manufacturing. *Journal of Intelligent Manufacturing*, 35, 3731–3752. https://doi.org/10.1007/s10845-023-02319-6
22. Sahoo, S., & Lo, C.-Y. (2022). Smart manufacturing powered by recent technological advancements: A review. *Journal of Manufacturing Systems*, 64, 236–250. https://doi.org/10.1016/j.jmsy.2022.06.008
23. Wang, Y., Wang, S., Tao, F., et al. (2023). Digital twin-based manufacturing system: A survey based on a novel reference model. *Journal of Intelligent Manufacturing*. https://doi.org/10.1007/s10845-023-02172-7
24. Ghobakhloo, M., Amoozad Mahdiraji, H., Iranmanesh, M., & Jafari-Sadeghi, V. (2024). From Industry 4.0 digital manufacturing to Industry 5.0 digital society: A roadmap toward human-centric, sustainable, and resilient production. *Information Systems Frontiers*. https://doi.org/10.1007/s10796-024-10476-z
25. Alfaro-Viquez, D., Zamora-Hernandez, M., Fernandez-Vega, M., Garcia-Rodriguez, J., & Azorin-Lopez, J. (2025). A comprehensive review of AI-based digital twin applications in manufacturing: Integration across operator, product, and process dimensions. *Electronics*, 14(4), 646. https://doi.org/10.3390/electronics14040646
26. Xu, X., Lu, Y., Vogel-Heuser, B., & Wang, L. (2021). Industry 4.0 and Industry 5.0: Inception, conception and perception. *Journal of Manufacturing Systems*, 61, 530-535.
27. Rasheed, A., San, O., & Kvamsdal, T. (2021). Digital twin: Values, challenges and enablers from a modeling perspective. *IEEE Access*, 9, 21980-22012.
28. Frank, A. G., Dalenogare, L. S., & Ayala, N. F. (2022). Industry 4.0 technologies and digital transformation: Manufacturing implications and future directions. *Technological Forecasting and Social Change*, 174, 121-140.
29. Javaid, M., Haleem, A., Singh, R. P., Khan, I. H., & Suman, R. (2023). Artificial intelligence applications for Industry 4.0 and Industry 5.0 manufacturing systems. *International Journal*

*of Intelligent Networks*, 4, 1-15.

30. Sheng, Y., Zhang, G., Zhang, Y., Luo, M., Pang, Y., & Wang, Q. (2024). A multimodal data sensing and feature learning-based self-adaptive hybrid approach for machining quality prediction. Advanced Engineering Informatics, 59, 102324. https://doi.org/10.1016/j.aei.2023.102324
31. Wang, L., Törngren, M., & Onori, M. (2021). Current status and advancement of cyber-physical systems in manufacturing. Journal of Manufacturing Systems, 59, 517-527