machines

MDPI

Article

AI-Enabled Axiomatic Design for MES-Level Process Parameters Optimization in Cloud-Based Manufacturing Execution Systems

Xin Chen [1,2] and Kai Cheng [2,*]

[1] School of Intelligent Science and Information Engineering, Shenyang University, Shenyang 110044, China; xin.chen@brunel.ac.uk
[2] Department of Mechanical and Aerospace Engineering, Brunel University London, London UB8 3PH, UK
* Correspondence: kai.cheng@brunel.ac.uk

Abstract

In cloud manufacturing, Manufacturing Execution Systems (MES) must dynamically optimize MES-level process parameters under multi-objective and constraint-intensive conditions, including quality conformity, production-takt feasibility, energy consumption, and equipment-stability risk. Conventional rule-based approaches suffer from limited efficiency, weak handling of parameter coupling, and insufficient decision traceability. This paper presents an AI-enabled Axiomatic Design (AD) governance framework for MES-level process parameter optimization and recommendation in cloud-based MES. AI capabilities are embedded within a structured AD-based framework to decompose manufacturing requirements, estimate functional requirement-design parameter coupling, and organize constraint-first decision making. A manufacturing knowledge graph encodes equipment boundaries and process rules as computable hard constraints; a multi-agent system (MAS) separates candidate generation, evaluation, and constraint validation; and a large language model (LLM) provides an evidence-driven semantic interface. Experiments were conducted on a 10,000-sample public manufacturing-process dataset. The rule audit confirmed that the dataset-defined Historical Favorable State label follows explicit thresholds on temperature, machine speed, and vibration; the classifier output is therefore used as label-consistency evidence within the evaluation module. In the offline recommendation case, the proposed AD-governed feasible-domain ranking achieved 0/20 hard-constraint violations in the Top-20 candidates, whereas the ungated weighted energy-stability ranking and the historical-label/data-driven ranking each produced 17/20 violations under the same deterministic tie-break rule. These results support the internal consistency and engineering feasibility of the proposed governance-oriented workflow at the MES execution-parameter level, with online industrial deployment identified as future work.

Keywords: AI-enabled axiomatic design; cloud-based manufacturing execution systems; MES-level process optimization; knowledge graph; large language model; multi-agent system

Academic Editor: Shaoping Bai

Received: 29 May 2026
Revised: 3 July 2026
Accepted: 8 July 2026
Published: 13 July 2026

1. Introduction

As cloud computing, the industrial internet, and intelligent manufacturing converge, manufacturing systems are evolving from localized automation toward a data-driven cloud manufacturing paradigm [1]. By virtualizing, servitizing, and enabling on-demand allocation of manufacturing resources, cloud manufacturing empowers production systems with collaborative capabilities across regions, production lines/machines, and in-process

 https://doi.org/10.3390/machines14070787

decision-making and process optimization, thereby imposing new requirements on the functional positioning and operational modes of Manufacturing Execution Systems (MES). In this study, MES refers to the shop-floor-oriented manufacturing execution system that connects production planning with real-time manufacturing execution by collecting process data, monitoring equipment and production states, dispatching operational tasks, and supporting parameter configuration and process control decisions.

In cloud manufacturing environments, MES is increasingly required to configure and dynamically adjust manufacturing process parameters under multi-objective constraints. In practical manufacturing scenarios, key parameters such as machine operating speed, unit-per-hour takt, temperature, and vibration level are often tightly coupled. Variations in these parameters affect product quality, production takt, equipment stability, and energy consumption. Meanwhile, engineering constraints, including machine capability boundaries, safety thresholds, and process prohibition rules, evolve with machine conditions and process configurations. Establishing an MES-level process-parameter optimization and decision-making mechanism that satisfies multi-objective requirements, complies with complex manufacturing constraints, and adapts to changing operating states has become a critical challenge in the intelligent evolution of manufacturing systems.

To support the cloudification and intelligent upgrading of MES, extensive research and development efforts have been conducted worldwide from multiple perspectives, including system architecture, data-driven methods, knowledge-enhanced mechanisms, and specific application case studies, with a growing body of systematic reviews summarizing the landscape [2]. At the system level, existing studies have improved the scalability and collaborative capabilities of MES in cloud environments through cloud-native and distributed paradigms [3,4], as well as virtualization and programmable infrastructure [5]. At the methodological level, a large body of work has applied machine learning to model and predict manufacturing process states, with its applications in areas like additive manufacturing being extensively reviewed [6]. Foundational algorithms such as tree-based ensemble methods [7] and support-vector machines [8] are widely used to identify and evaluate product quality and equipment conditions [9], while how to address the inherent uncertainty of machine learning models has become a significant topic in engineering design [10]. Furthermore, to tackle the common issue of data imbalance in manufacturing, techniques like the Synthetic Minority Over-sampling Technique (SMOTE) have been explored [11], in addition to devolved manufacturing and platform exploration [12]. To address the challenges of data silos and privacy protection, emerging paradigms such as federated learning have been proposed to improve model generalization through multi-source collaborative training [13]. Building on these data-driven foundations, digital twin (DT) technology, as a core enabler bridging the physical world and the information space, has been systematically reviewed for its concepts, frameworks, and applications [14,15], and has seen the development of specific modeling frameworks [16]. DTs are widely adopted for state modeling, process monitoring, and optimization of manufacturing systems [17,18], and through deep integration with machine learning, have further enhanced the system's robust control capability under uncertainty [19]. However, purely data-driven methods often lack deep integration with engineering principles and domain knowledge. To address this, knowledge graphs and manufacturing ontologies have been employed to explicitly represent equipment capabilities, process rules, and expert knowledge, with their application prospects and research topics being comprehensively reviewed [20]. Specific applications include cognitive intelligent manufacturing [21], knowledge reuse [22], patent knowledge mining [23], carbon footprint traceability [24], innovative design for additive manufacturing [25], and tolerance type generation [26]. To cope with the challenges of distributed decision-making and collaborative optimization in complex manufacturing systems, multi-agent systems (MAS) have been introduced into design and manufacturing scenarios, exploring complex optimization paths through dynamic negotiation and evolution among agents [27–29]. More recently, the rise of large language models (LLMs) has brought transformative opportunities to the engineering domain, with their capabilities being actively explored for decision-making in manufacturing [30], multimodal perception [31], conceptual design [32], supply chain discovery [33], CAD modeling [34], and engineering education [35]. Furthermore, researchers are exploring the potential of multi-technology integration, such as combining LLMs with reinforcement learning in DT environments [36] and enhancing multi-agent reinforcement learning with knowledge graphs for adaptive scheduling [37]. Despite these significant advances, existing methods still commonly rely on implicit weighting schemes or end-to-end black-box optimization when handling multi-objective optimization problems [38], resulting in limitations in the interpretability, auditability, and engineering executability of manufacturing process parameter optimization and recommendation results. Moreover, when manufacturing objectives, constraints, or operating environments change, existing methods frequently require extensive model reconstruction or retraining, making them difficult to sustain long-term operation and continuous evolution in cloud-based MES scenarios.

This paper conceptualizes MES-level process parameter optimization in cloud-based MES as an engineering decision-making process that requires systematic governance. It presents an AI-enabled parameter optimization and recommendation approach for cloud-based MES governed by Axiomatic Design (AD). By structurally modeling manufacturing objectives, the proposed approach clarifies the priority relationships among quality, production efficiency, energy consumption, and equipment-stability risk, and maps them onto key operational parameters. By embedding machine capability boundaries, safety thresholds, and process prohibition rules into the parameter optimization workflow, the proposed offline implementation enforces manufacturing constraints before final ranking and supports executability and traceability. Through modular agent specialization and evidence-driven explanation mechanisms, the workflow provides a stable governance structure for candidate generation, validation, ranking, and explanation. The methodological novelty lies in transforming AD from a static requirement-parameter mapping tool into an executable, AI-enabled and constraint-governed decision procedure. In this procedure, data-driven models estimate coupling strength and candidate evidence, the knowledge graph enforces non-negotiable engineering constraints, MAS modularizes generation, evaluation, and validation operations, and the LLM renders evidence-bound explanations after numerical and rule-based decisions have been finalized.

2. AI-Enabled Axiomatic Design for MES-Level Parameter Optimization

2.1. Overall Approach and Its Framework from Decision-Governance Perspectives

To systematically address diverse configuration objectives, dense constraints, and continuously evolving operational states in cloud-based MES scenarios, this study develops a structured methodological framework for MES-level process parameter optimization, with parameter recommendation as its executable output. The proposed framework models parameter recommendation as a closed-loop decision-making process composed of requirement input, governance rules, candidate generation, adjudication and validation, and evidence output. Within this process, the objective of parameter recommendation is to ensure that the decision workflow remains controllable, interpretable, and engineering-feasible under changing objectives, evolving constraints, and long-term system operation. In the proposed method, AI operationalizes three AD steps that are difficult to execute manually in dynamic MES environments. First, data-driven models approximate FR-DP coupling relationships and support construction of an executable design matrix. Second, the estimated coupling structure is converted into a sequential decoupling strategy that determines which requirements should be constrained first and which objectives should be optimized later. Third, AI-based evaluation and knowledge-graph-based adjudication jointly transform AD outputs from conceptual design guidance into verifiable parameter candidates with explicit feasibility evidence. Therefore, AI-enabled AD in this study refers to an executable integration of AD decomposition, data-driven coupling estimation, constraint-governed candidate screening, and evidence-based semantic explanation. The unique methodological contribution is a constraint-governed sequential decoupling formulation for AI-enabled axiomatic design.

The overall architecture of the proposed AI-enabled axiomatic design approach is illustrated in Figure 1, which consists of four main layers: an axiomatic-design-driven structural layer, a data and knowledge support layer, a semantic interaction layer, and a multi-agent execution layer. The inputs to the parameter recommendation process are uniformly defined as structured decision configurations, including natural-language production requirements, quality thresholds specified by process standards, feasible operating ranges jointly determined by order takt and equipment capabilities, knowledge-graph constraints, and runtime parameters used to control the search budget. These inputs are internally transformed into computable functional requirements and parameter boundaries. The system output is an ordered set of Top-K parameter candidate solutions, each of which includes concrete parameter values, predicted performance indicators, constraint verification results, and a complete evidence chain to support MES execution, archival, and human auditing.

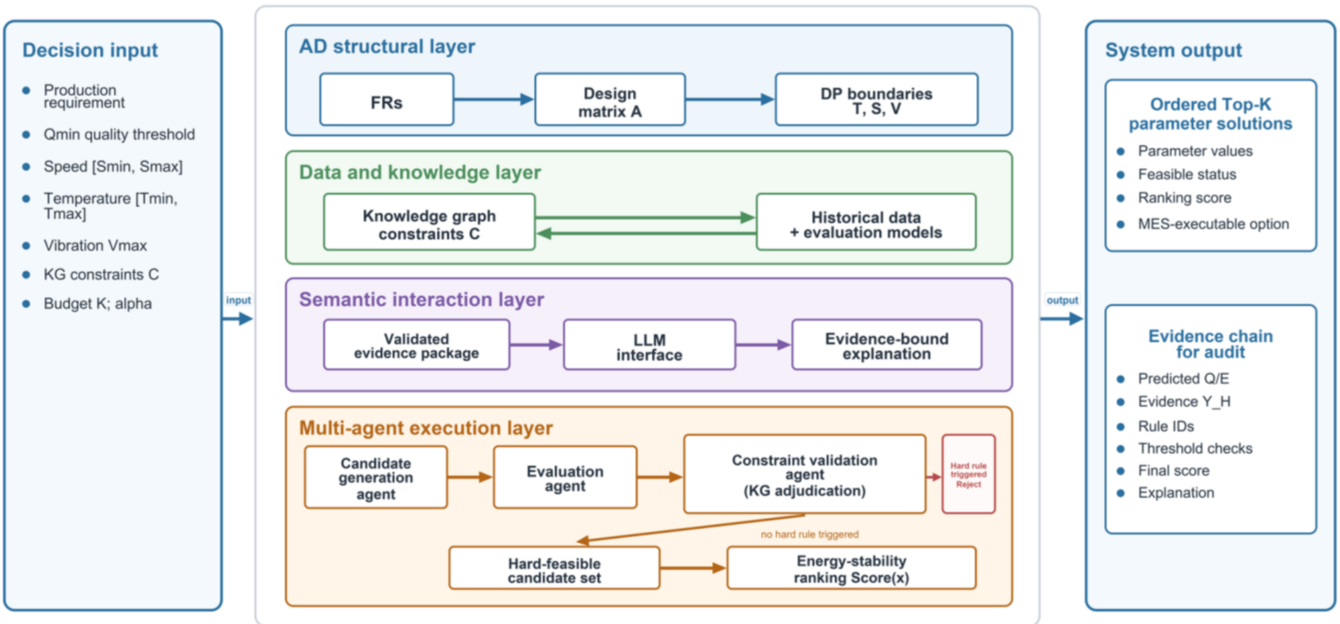

Figure 1. Overall framework of MES-level process parameter optimization and parameter recommendation.

In the overall workflow, parameter recommendation is first initiated by functional modeling of requirements through the axiomatic-design-driven structural layer, which explicitly defines the decision space and priority relationships. Subsequently, candidate solutions are generated within the constrained parameter space and evaluated using data-driven models, while a knowledge-graph-driven constraint adjudication mechanism performs hard-constraint verification on the candidates. Finally, under explicit evidence constraints, a large language model generates explanatory text to enable interpretable interaction between the system and human users. Through this hierarchical design, the proposed approach transforms parameter recommendation from a purely numerical computation task into a governable and auditable engineering decision-making process.

From an implementation perspective, the end-to-end decision procedure can be described as a sequential information-processing process with explicitly defined module inputs and outputs. The procedure starts from a production requirement, the quality threshold, the feasible speed interval, the applicable knowledge-graph constraint set C, the search budget, predictive models, and the manufacturing knowledge graph. The AD layer first transforms the production requirement into functional requirements and design parameter boundaries, thereby defining the structured decision space for subsequent agent operations. Based on these boundaries, the generation agent produces candidate parameter configurations within the allowed search scope and removes candidates that violate the feasible speed interval before model-based evaluation. Each remaining candidate is then passed to the evaluation agent, which estimates the predicted quality, energy consumption, and historical favorable-state probability using the corresponding predictive models. Candidates whose predicted quality is lower than the threshold are rejected by the quality gate, while quality-feasible candidates are transferred to the constraint validation agent.

The constraint validation agent converts each quality-feasible candidate into parameter triples and queries the manufacturing knowledge graph against the applicable constraint set. If any equipment capability boundary, safety threshold, or prohibited parameter combination is triggered, the candidate is immediately rejected through the short-circuit validation mechanism; otherwise, it is marked as constraint-feasible and passed to the final decision module. The coordination mechanism among agents is therefore based on structured evidence transfer rather than free-form negotiation: the generation agent outputs candidate parameter sets, the evaluation agent appends predicted indicators and quality-gate results, and the constraint validation agent appends feasibility labels, triggered-rule identifiers, threshold comparisons, and constraint sources. When model-based evaluation and knowledge-graph validation produce conflicting implications, the hard-constraint result has priority over predicted performance, and the candidate is rejected regardless of its predicted score. The final decision module then computes a composite ranking score based on the optimization objectives and risk-related evidence, ranks all constraint-feasible candidates, and selects the Top-K configurations as executable recommendations.

Within the offline implementation, the LLM is assigned to an evidence-driven semantic interface after the numerical and constraint-based decisions have been finalized. It receives the original requirement, selected Top-K configurations, predicted indicators, satisfied or triggered constraints, threshold comparisons, constraint sources, and final ranking scores as structured evidence. Based on this evidence package, the LLM generates a concise natural-language explanation for each recommended configuration, covering feasibility reasons, constraint-checking results, ranking logic, and risk warnings. The explanation includes checked quality and speed requirements, satisfied or triggered knowledge-graph constraints, energy- and stability-related ranking evidence, and a traceable justification for inclusion in the Top-K set. Feasibility and ranking decisions remain determined by the preceding evaluation and adjudication modules, while the LLM supports human–machine interaction, traceability, and auditability as a controlled semantic layer.

2.2. Functional Requirement Modeling Based on Axiomatic Design

Axiomatic Design emphasizes that functional requirements (FRs) should be defined prior to specifying implementation mechanisms, without presupposing solution forms [39–42]. In accordance with the operational objectives of cloud-based MES and the variables available in the experimental dataset, this study categorizes the functional requirements of the parameter recommendation problem into three classes: quality, production takt, and energy consumption, with equipment-stability risk handled as a safety-related constraint and ranking penalty. Quality and takt are treated as satisfaction-type requirements, while energy consumption is considered an objective-type requirement to be optimized within the feasible domain. Accordingly, the functional requirements are formalized as the following constraint and optimization relationships:

$$\begin{gathered} FR_1 \,:\, Q(x) \,\geq\, Q_{min} \\ FR_2 \,:\, S(x) \,\in\, [S_{min},\, S_{max}] \\ FR_3 \,:\, \min_{x \,\in\, \Omega} E(x) \end{gathered} \tag{1}$$

Here, x denotes a candidate parameter configuration, i.e., a set of design parameter values. $Q(x)$ represents the quality score associated with the configuration, which can be obtained from observed data or model predictions, while Q_{min} denotes the minimum acceptable quality threshold determined by process specifications or product-family settings. $S(x)$ denotes the machine operating speed corresponding to the configuration, and $[S_{\mathrm{min}},\, S_{\mathrm{max}}]$ defines the feasible speed interval jointly determined by order takt requirements and equipment capabilities. $E(x)$ represents the energy consumption associated with the configuration. F denotes the feasible domain further constrained by hard-constraint adjudication under the satisfaction of quality and speed requirements. The ordering of functional requirements follows an AD-based requirement hierarchy in which non-negotiable engineering requirements are separated from performance objectives. In the present MES parameter recommendation problem, quality conformity and takt feasibility are treated as satisfaction-type requirements because violating them makes a candidate configuration non-executable regardless of its energy performance, whereas energy consumption is treated as an optimization-type requirement within the feasible domain.

Energy consumption is selected as the explicit optimization objective in the experimental instantiation because it is directly observable in the available dataset and can be consistently evaluated for all candidate configurations. Throughput-related requirements are represented through the feasible speed interval, and robustness-related concerns are incorporated through the vibration-based stability risk term in the final ranking function. Cost, carbon emission, tool wear, and other scenario-specific indicators can be incorporated when reliable labels or calculation models are available. More generally, $E(x)$ can be extended from a single objective to a vector-valued performance objective. Within the proposed AD-governed framework, these objectives are evaluated after satisfaction-type FRs and hard constraints have defined the feasible domain, and the final recommendation can be obtained through a weighted ranking function or Pareto-based candidate selection. Through this formalization, the system explicitly distinguishes non-violable conditions from objectives to be optimized within the feasible region, thereby providing a consistent interface for subsequent sequential solution procedures and knowledge-graph-based constraint adjudication.

2.3. Design Parameters Modeling and Engineering Controllability Analysis

After the functional requirements have been specified, design parameters (DPs) are defined as process variables that can be directly set or indirectly regulated by the MES, influencing the degree to which the functional requirements are satisfied. Considering the available dataset attributes and manufacturing engineering controllability, this study selects temperature, machine speed, and vibration level as the core design parameters. Accordingly, the design parameter vector is expressed as:

$$DP = \begin{bmatrix} DP_1 \\ DP_2 \\ DP_3 \end{bmatrix} = \begin{bmatrix} T \\ S \\ V \end{bmatrix} \tag{2}$$

Here, T denotes the process temperature (Temperature), S denotes the machine speed (Machine Speed), and V denotes the vibration level (Vibration Level). Temperature and speed are explicitly controllable variables and can typically be directly specified through control systems or process recipes. Although vibration is not a directly settable variable, it reflects equipment stability and mechanical condition and can be indirectly influenced through maintenance strategies, fixture alignment, load distribution, and coordinated adjustment of process parameters.

2.4. Design Matrix and Sequential Decoupling Solution Strategy

Axiomatic Design employs a design matrix to represent the influence relationships between functional requirements and design parameters. In this study, this mapping is expressed in the following form:

$$FR = \begin{bmatrix} FR_1 \\ FR_2 \\ FR_3 \end{bmatrix} = A \begin{bmatrix} DP_1 \\ DP_2 \\ DP_3 \end{bmatrix} = A \begin{bmatrix} T \\ S \\ V \end{bmatrix} \tag{3}$$

Here, A denotes the design matrix, and the element aij represents the degree of influence of the j-th design parameter on the i-th functional requirement. In real manufacturing systems, quality and energy consumption are typically affected by the joint action of temperature, speed, and vibration; therefore, A usually exhibits a coupled structure. The independence axiom is used here as a governance principle for reducing uncontrolled functional coupling into an executable and ordered decision structure. Because a perfectly uncoupled diagonal design matrix is difficult to obtain in cloud-based MES parameter configuration, this study seeks to satisfy the independence axiom in a sequential and operational sense: non-negotiable FRs are first isolated as feasibility gates, and optimization-type FRs are handled after the satisfaction-type FRs have been enforced. In this way, the fulfillment of FR_1 and FR_2 is not traded off against FR_3, and energy optimization is performed within the feasible domain that already satisfies quality, takt, and hard engineering constraints. The coupled matrix is therefore transformed into a decoupled decision sequence rather than being solved as a simultaneous black-box multi-objective problem. Directly performing multi-objective optimization in such a coupled space often leads to engineering issues, including solution instability, difficulty in explicitly managing constraints, and poor interpretability of results.

To translate conceptual coupling relationships into an executable decision strategy, this study adopts a reproducible default estimation scheme for coupling strength. Trained prediction or classification models are used as approximate mapping functions, and feature permutation importance is computed on test data to estimate the global influence of each design parameter on different outputs. The row-normalized importance values form a computable coupling-evidence matrix. If the raw permutation importance of design parameter DP_j for requirement or evidence output FR_i is denoted as p_{ij}, the normalized coupling value is $a_{ij} = \frac{p_{ij}}{\sum_j p_{ij}}$ when the denominator is positive. Directly constrained requirements, such as speed feasibility and vibration safety, are represented as deterministic rows in which the corresponding controllable or observable parameter has unit coupling.

The resulting matrix provides an operational basis for deciding which requirements should be constrained first and which objective should be optimized later.

On this basis, a sequential decoupling procedure is constructed according to the priority hierarchy of the functional requirements. First, FR_2 is applied to constrain the feasible speed domain, thereby reducing the search space. Next, within this domain, FR_1 is used to perform quality feasibility screening. Finally, within the feasible region that satisfies the first two layers, energy consumption is optimized, and a stability risk penalty is introduced to obtain a comprehensive ranking. This process can be formally expressed as a progressively shrinking feasible domain:

$$\begin{aligned} \Omega_2 &= \{x \mid S(x) \in [S_{min}, S_{max}]\} \\ \Omega_1 &= \{x \in \Omega_2 \mid Q(x) \geq Q_{min}\} \\ \Omega &= \{x \in \Omega_1 \mid \forall c \in C, c(x) = True\} \end{aligned} \tag{4}$$

Here, Ω_2 denotes the speed-feasible domain constrained by the efficiency requirement, Ω_1 represents the quality-feasible domain further filtered by the quality requirement, and Ω is the final feasible domain obtained after hard-constraint adjudication based on the knowledge graph. $\mathcal{C}$ denotes the set of hard constraints defined in the manufacturing knowledge graph, and $c(x)$ indicates whether constraint c is satisfied by a candidate configuration x. A configuration x is included in the final feasible domain Ω only if all hard constraints are satisfied. In this way, a multi-objective problem with strong constraints is transformed into a series of engineering-executable steps following a "constrain first, optimize later" principle, thereby structurally reducing the solution complexity induced by coupling effects. To rank candidate solutions within the final feasible domain, a composite scoring function is defined as:

$$Score(x) = \alpha \cdot \tilde{E}(x) + (1 - \alpha) \cdot R_{stab}(x) \tag{5}$$

Here, $\tilde{E}(x)$ denotes the normalized form of the predicted energy consumption $E(x)$. Linear scaling can be adopted for normalization, i.e.,

$$\tilde{E}(x) = \frac{E(x) - E_{min}}{E_{max} - E_{min}} \tag{6}$$

where E_{min} and E_{max} are the minimum and maximum energy-consumption values obtained from historical data or from the current candidate set. The parameter α is a weighting coefficient applied after the quality, speed, temperature, and safety gates have been satisfied. In deployment, α can be set by the process engineer or MES planner to reflect the relative priority of energy saving versus stability margin within the feasible domain. A larger α gives greater weight to energy reduction, whereas a smaller α gives greater weight to the stability-risk penalty. In the offline recommendation case, $\alpha = 0.70$ is used to prioritize energy consumption while retaining a vibration-risk penalty. Because the reported Top-K case enforces $V \leq V_{\mathrm{max}}$ as a hard feasibility gate and defines R_{stab} as a threshold-exceedance penalty, R_{stab} is zero for all retained hard-feasible candidates; therefore, α primarily scales the energy component in this hard-gated case. The coefficient becomes ranking-active when a soft-risk variant retains candidates close to or beyond the vibration boundary, or when a within-boundary safety-margin penalty is enabled. R_{stab} denotes the stability-risk term, which penalizes candidate configurations whose vibration levels exceed the safety threshold. In this study, the stability-risk term is defined as follows:

$$R_{stab}(x) = max\left(0, \frac{V(x)}{V_{safe}} - 1\right) \tag{7}$$

Here, $V(x)$ denotes the vibration level associated with configuration x, and V_{safe} denotes the vibration safety threshold. When $V(x) \leq V_{\mathrm{safe}}$, the risk term is set to zero; when $V(x) > V_{\mathrm{safe}}$, the risk term increases linearly with the degree of threshold violation. The engineering implication of this definition is that, under the satisfaction of quality and takt requirements, configurations with lower energy consumption and lower vibration risk are prioritized, while configurations with excessive vibration are explicitly penalized in the ranking process. This ranking logic also provides an operational interpretation of the information axiom in the present AD-governed parameter recommendation problem. In classical axiomatic design, the information axiom favors the feasible design with the minimum information content, which can be understood as the design with the highest likelihood of satisfying the specified FRs. Since complete probability distributions for all FR satisfaction events are not available in the offline dataset, the information axiom is not implemented as a standalone probabilistic design-selection formula in this study. Instead, it is operationalized as an evidence-based uncertainty-reduction principle within the final feasible domain. After independence-oriented sequential decoupling has enforced quality, takt, and hard engineering constraints, candidate configurations are further ranked according to energy consumption, stability-related risk evidence, and traceable constraint-verification records. Configurations with lower energy demand, lower vibration-related risk, and complete constraint-verification records are treated as having lower decision uncertainty and lower effective information content. In this way, the independence axiom structures the decision sequence, while the information axiom guides the selection of better-supported and lower-risk recommendations among feasible candidates.

2.5. Constraint Adjudication, Evaluation, and Semantic Interface

In complex manufacturing environments, recommendations based exclusively on data-driven models may be insufficient to ensure safety and process feasibility. Therefore, this study introduces a manufacturing-process-oriented knowledge graph in which equipment capability boundaries, process rules, and forbidden parameter combinations are explicitly modeled as hard constraints. The knowledge graph is constructed using a rule-oriented manufacturing schema consisting of four types of nodes, namely Equipment, ProcessParameter, ConstraintRule, and ProcessState. Equipment nodes describe machine capability ranges, ProcessParameter nodes represent controllable or observable variables such as temperature, speed, and vibration, ConstraintRule nodes encode threshold limits and prohibited parameter combinations, and ProcessState nodes record the operating context in which a rule is applicable. The main relations include hasParameter, hasCapabilityBoundary, constrainedBy, triggersRule, and applicableTo, through which each candidate configuration can be linked to the corresponding equipment boundary, safety threshold, and process rule. Formally, the constraint knowledge is represented as typed triples such as (Equipment, hasCapabilityBoundary, ProcessParameter), (ProcessParameter, constrainedBy, ConstraintRule), and (ConstraintRule, applicableTo, ProcessState). Each ConstraintRule node contains computable attributes, including the constrained parameter, operator, threshold or interval, applicable process state, and rule source. For example, a vibration safety rule can be represented as a ConstraintRule connected to the Vibration parameter and the corresponding equipment node, with attributes such as operator = "$\leq$", threshold = V_{safe}, and source = "equipment specification". In this way, the knowledge graph does not simply store isolated if–then rules, but organizes constraints together with their equipment context, parameter dependency, applicability condition, and traceable engineering source. During decision-making, the constraint validation agent converts a candidate configuration into a set of parameter triples and queries the knowledge graph by matching the involved equipment, parameter names, value intervals, and applicable process states. The query process first retrieves all ConstraintRule nodes connected to the candidate-related equipment and parameters, then checks whether the current ProcessState satisfies the applicableTo relation, and finally evaluates the rule attributes against the candidate values. A constraint is triggered when the queried rule condition is satisfied, and the candidate is rejected immediately under the short-circuit mechanism; otherwise, it is passed to the subsequent evaluation and ranking stage with a structured evidence record. The use of the knowledge graph is therefore justified by its ability to provide contextualized, reusable, and auditable constraint retrieval, especially when different equipment, product families, or process states activate different subsets of constraints. For any candidate configuration x, the knowledge-graph-based adjudication module performs validation against the constraint set $\mathcal{C}$, which can be formalized as follows:

$$Feasible(x) = \begin{cases} 1, & \forall c \in C, \; c(x) = True \\ 0, & \exists c \in C, \; c(x) = False \end{cases} \tag{8}$$

Here, $\mathcal{C}$ denotes the constraint set, and $c(x)$ indicates whether constraint c is satisfied by configuration x. A short-circuit evaluation mechanism is adopted: once any constraint violation is detected, the configuration is immediately deemed infeasible. The adjudication module outputs structured evidence, including the identifier of the triggered constraint, the triggering condition, threshold comparisons, and the constraint source (e.g., equipment manuals, process specifications, or historically derived rules). At the implementation level, this design ensures that recommendation results are usable, interpretable, and traceable.

The evaluation agent is defined as a functional component within the multi-agent execution layer. It wraps a supervised classification module and provides standardized label-reconstruction evidence for candidate configurations, including historical favorable-state probability and risk-gated classification results. In this study, historical favorable state denotes the operating-state label provided by the dataset. The binary label is denoted as Y_H, where $Y_H = 1$ indicates a favorable state according to the dataset definition and $Y_H = 0$ indicates a non-favorable state. A rule audit of the experimental dataset shows that Y_H can be exactly reconstructed by $74 \leq T \leq 76$, $1475 \leq S \leq 1525$, and $V \leq 0.06$, yielding TP = 966, FP = 0, FN = 0, and TN = 9034 over all 10,000 records. The classifier output is therefore used as dataset-defined label-reconstruction evidence for the evaluation agent. This label evidence is separate from the Framework-Optimized Configuration produced by the sequential recommendation procedure. In the experimental implementation, the classifier receives candidate-related process variables as input, including temperature, machine speed, vibration level, and their temporal statistical features when time-window information is available. The model is trained offline using historical samples with stratified random sampling for training and testing, and then used to output a label-consistency probability for candidate evidence annotation. Optimization-defined recommendation results are determined after quality feasibility, speed feasibility, knowledge-graph-based hard constraints, and final energy-risk ranking have been evaluated. Thus, Y_H and $E(x)$ serve different roles: Y_H provides dataset-defined state evidence, whereas $E(x)$ participates in ranking feasible candidates within the optimization layer. This probability can be combined with a decision threshold to yield a binary classification outcome for risk-gated screening:

$$\hat{y}(x) = \begin{cases} 1, & \hat{p}(x) \geq \tau \\ 0, & \hat{p}(x) < \tau \end{cases} \tag{9}$$

Here, $\hat{y}$ denotes the predicted label, where 1 indicates the historical favorable-state class, and 0 indicates the non-favorable class, and τ denotes the classification threshold. In the recommendation workflow, $\hat{y}$ contributes model-based state evidence that is combined with quality thresholds, speed feasibility, knowledge-graph-based hard constraints, and the final ranking score.

3. Experiments and Analysis

3.1. Manufacturing Dataset Description and Experimental Setup

The dataset used in this study is derived from historical sampling records of the publicly available Smart Manufacturing Process Data [43], comprising 10,000 consecutive one-minute sampling instances from 1 April 2025 08:00:00 to 8 April 2025 06:39:00. Each instance records the timestamped process variables of a manufacturing system, including process temperature, machine operating speed, vibration level, energy consumption, and production quality score, together with the binary label named Optimal Conditions in the original dataset. In this paper, this binary label is denoted as Y_H and interpreted as the historical favorable-state label. The experimental validation is defined at the MES execution-parameter level, where temperature, speed, vibration, quality, and energy are used to evaluate the proposed process-optimization governance mechanism. Based on the above data modeling and experimental setup, the following section presents an experimental analysis of the historical favorable-state identification model and the role of key features. Let the observation at the i-th sampling instant be denoted as:

$$z(t) = [T(t), S(t), V(t), E(t), Q(t)] \tag{10}$$

Here, T_i, S_i, V_i, E_i, and Q_i denote the temperature, machine speed, vibration level, energy consumption, and quality score at time i, respectively. The corresponding historical favorable-state label is denoted as $Y_{H,i}$, where $Y_{H,i} = 1$ indicates a favorable operating state recorded in the dataset and $Y_{H,i} = 0$ indicates a non-favorable state. The dataset contains 966 favorable-state samples and 9034 non-favorable-state samples, corresponding to a favorable-state ratio of 9.66%. Regarding the experimental setup, the dataset is partitioned into a training set and a test set using stratified random sampling, with 70% of the samples allocated for model training and the remaining 30% for testing. A fixed random seed is used to ensure reproducibility. Table 1 provides a systematic summary of the physical meaning, units, and statistical characteristics of each variable in the dataset, establishing a unified data semantic basis for subsequent analysis and model development.

Table 1. Definitions and Statistical Characteristics of Dataset Variables.

Variable	Unit	Description	Min	Median	Max	Mean ± Std
Temperature	°C	Process temperature	67.58	75.00	82.47	74.99 ± 1.99
Machine Speed	RPM	Rotational speed of the machine	1450.0	1500.0	1549.0	1499.56 ± 29.06
Vibration Level	mm/s	Vibration level of the machine	0.03	0.07	0.10	0.07 ± 0.02
Production Quality Score	-	Quality score of the manufactured product	8.00	8.50	9.00	8.50 ± 0.29
Energy Consumption	kWh	Energy consumed during the process	1.00	1.50	2.00	1.50 ± 0.29

Note: The label-reconstruction component uses Temperature, Machine Speed, and Vibration Level, together with temporal statistical features when available, as input variables. Production Quality Score and Energy Consumption are treated as outcome-related indicators. The binary label Y_H is provided by the dataset and used as the reconstruction target for historical favorable-state evidence.

To connect the dataset-variable definitions with the AD-governed sequential decoupling strategy, Table 2 summarizes how feasibility evidence, dataset-defined state evidence, and ranking evidence are assigned to distinct decision roles within the constraint-first workflow.

Table 2. Decision-role matrix for constraint-first sequential decoupling.

Decision Role	Evidence Used in This Study	Role in the AD-Governed Sequence
Quality conformity	Required quality lower bound $Q \geq Q_{\min}$	Rejects candidates that fail the satisfaction requirement before ranking.
Production-takt feasibility	Required speed interval $S_{\min} \leq S \leq S_{\max}$	Applies the MES feasibility gate before any energy-oriented sorting.
Process-capability window	Admissible temperature interval $T_{\min} \leq T \leq T_{\max}$	Restricts candidate solutions to executable operating conditions.
Equipment stability and safety	Vibration threshold $V \leq V_{\max}$ and knowledge-graph safety rules	Prevents unstable or unsafe states from entering the ranking stage.
Historical favorable-state consistency	Dataset-defined Y_H reconstructed from the T, S, and V threshold structure	Provides dataset-state consistency evidence for the evaluation stage.
Feasible-domain ranking	Energy consumption, vibration-risk penalty, and Top-K budget	Ranks candidates after the hard constraints have been satisfied.

Note: The matrix summarizes how feasibility evidence, dataset-derived state evidence, and ranking evidence are assigned to different stages of the AD-governed workflow.

3.2. Exploratory Data Analysis and Time-Series Characteristics

Within the parameter recommendation governance framework, exploratory data analysis is used to identify process variables that have a critical impact on operating conditions and to provide data-driven evidence for constraint design and sequential decision-making. This study conducts a systematic analysis of manufacturing process data from three perspectives: statistical correlations, conditional distribution differences, and time-series dynamic behavior.

First, the Pearson correlation coefficients between the main continuous variables and the historical favorable-state label are computed. Let the sample sequences of random variables X and Y be denoted as $\{x_i\}$ and $\{y_i\}$, respectively. The Pearson correlation coefficient is defined as:

$$\rho_{X,Y} = \frac{Cov(X, Y)}{\sigma_X \sigma_Y} \tag{11}$$

Here, $Cov(X, Y)$ denotes the covariance between X and Y, and σ_X and σ_Y denote the standard deviations of X and Y, respectively. The correlation matrix shown in Figure 2 indicates that most process variables exhibit relatively weak overall linear correlations with the historical favorable-state label, suggesting that the favorable-state label cannot be determined by simple linear variations in any single variable. In contrast, vibration level shows a pronounced negative correlation with the historical favorable-state label, with the strongest correlation magnitude among all variables.

On this basis, further analysis is conducted to examine structural differences in variable distributions under the dataset-defined favorable and non-favorable states. Figure 3 presents comparative distributions of the main process variables. Temperature and machine speed show interval-constrained differences consistent with the threshold rule, while energy consumption is lower in many favorable samples but is not part of the label definition. Vibration level exhibits a clear separation around the 0.06 mm/s boundary, confirming its role in the dataset-defined favorable-state rule and in the subsequent stability-risk gate.

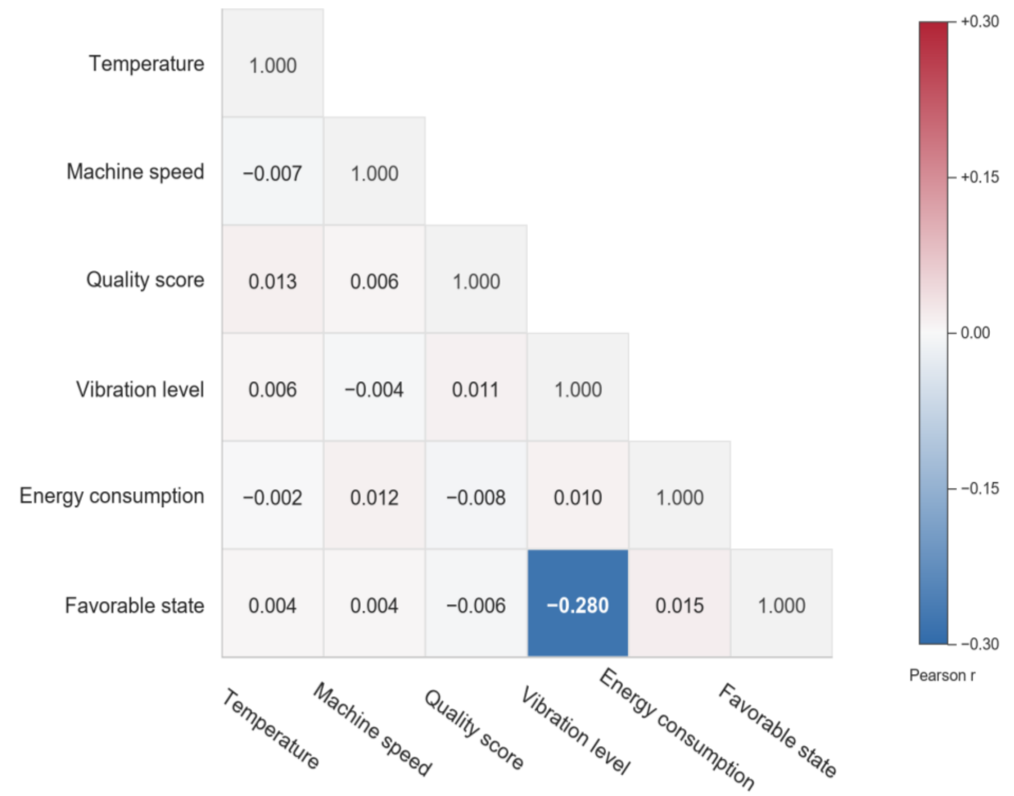

Figure 2. Pearson Correlation Matrix of Major Process Variables.

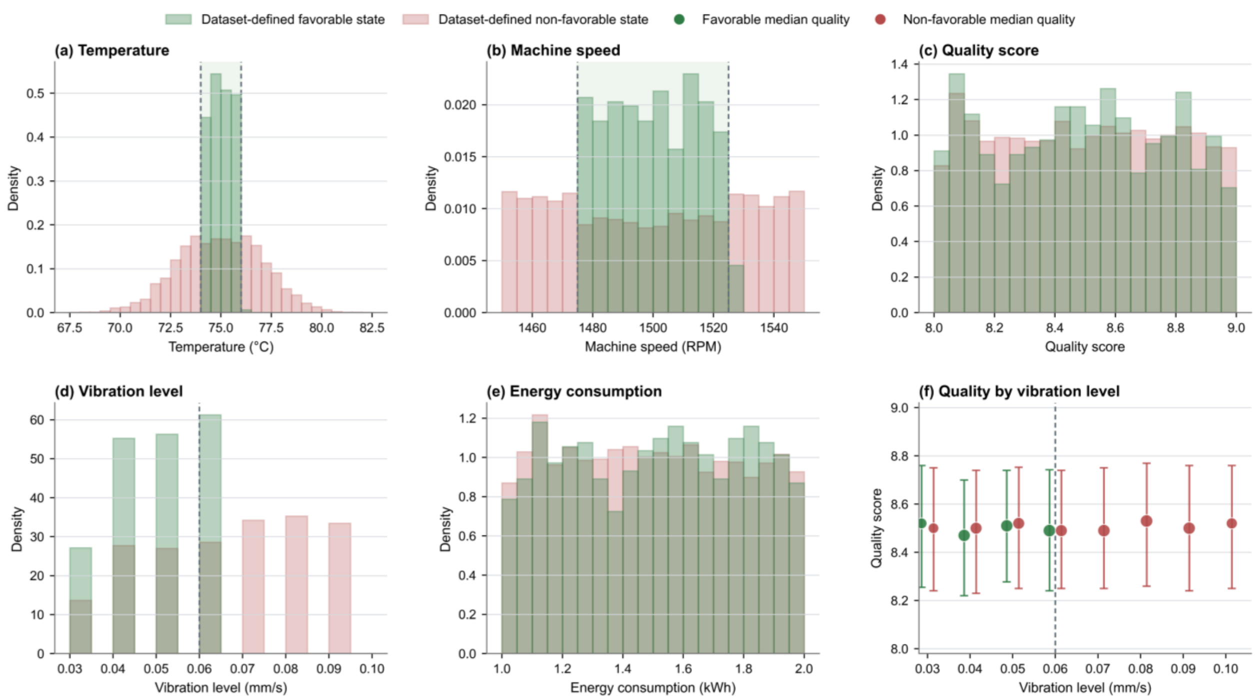

Figure 3. Distributions of key process variables by dataset-defined operating state.

To further examine the engineering operability of vibration as part of the dataset-defined threshold logic and as a stability-risk signal, vibration is analyzed from multiple perspectives, as illustrated in Figure 4. The boxplot results show clear differences in both the median and dispersion of vibration levels between historical favorable and non-favorable states, and statistical test results confirm that these differences are not attributable to random fluctuations. Threshold sensitivity analysis indicates that vibration thresholds materially affect the reconstruction of the dataset-defined favorable-state label. In addition, the ROC curve constructed using vibration as a single feature shows that vibration contains substantial label-relevant information, supporting its use as a stability signal within the constraint-adjudication workflow. The cumulative distribution function (CDF) analysis further reveals pronounced differences in cumulative probabilities between favorable and non-favorable samples under a given vibration threshold.

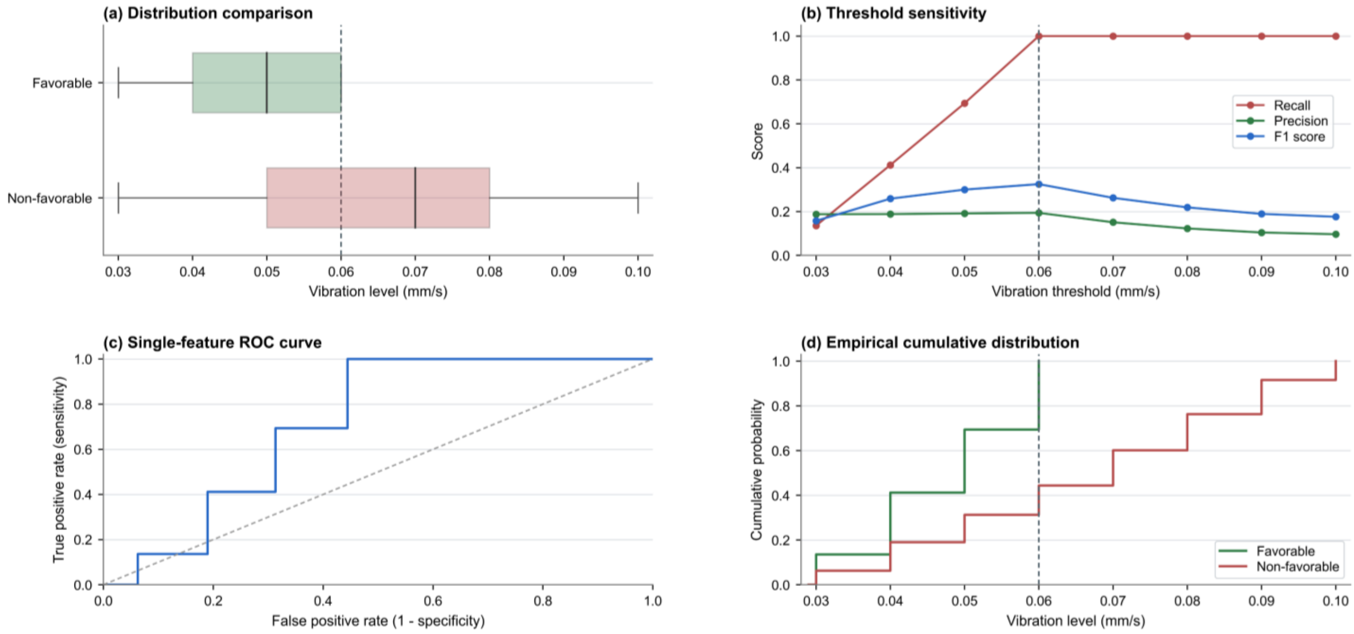

Figure 4. Vibration Characteristics Across Operating States.

In addition to static distribution analysis, this study further examines the dynamic behavior of manufacturing process parameters from a time-series perspective. Let the discrete-time sequence of a given process variable be defined as:

$$x(t),\ t = 1, 2, \ldots, T \tag{12}$$

where T denotes the total length of the sample sequence. To reduce high-frequency noise and highlight medium- to long-term trends, a moving average is applied to the time series, which is defined as:

$$\overline{x}(t) = \frac{1}{w}\sum_{i=0}^{w-1} x(t-i) \tag{13}$$

where w denotes the length of the sliding window. For the vibration sequence, the local fluctuation intensity within each sliding window is further computed as:

$$\sigma_V(t) = \sqrt{\frac{1}{w}\sum_{i=0}^{w-1}\left(V(t-i) - \overline{V}(t)^2\right)} \tag{14}$$

Here, v_t denotes the vibration level time series, and m_t represents its moving-average value. As shown in Figure 5, when the system operates within favorable-state intervals, both the vibration level and its local fluctuation intensity typically remain at low levels, while the quality score stays within a high range and energy consumption varies relatively smoothly. In contrast, during non-favorable operating intervals, the vibration level often exhibits sustained increases or frequent fluctuations. This dynamic behavior indicates that, under the given data conditions, vibration anomalies tend to precede noticeable changes in quality indicators in time. Therefore, vibration serves as an engineering-relevant stability risk signal and provides data-level justification for incorporating vibration into constraint adjudication or risk-penalty mechanisms in the parameter recommendation process.

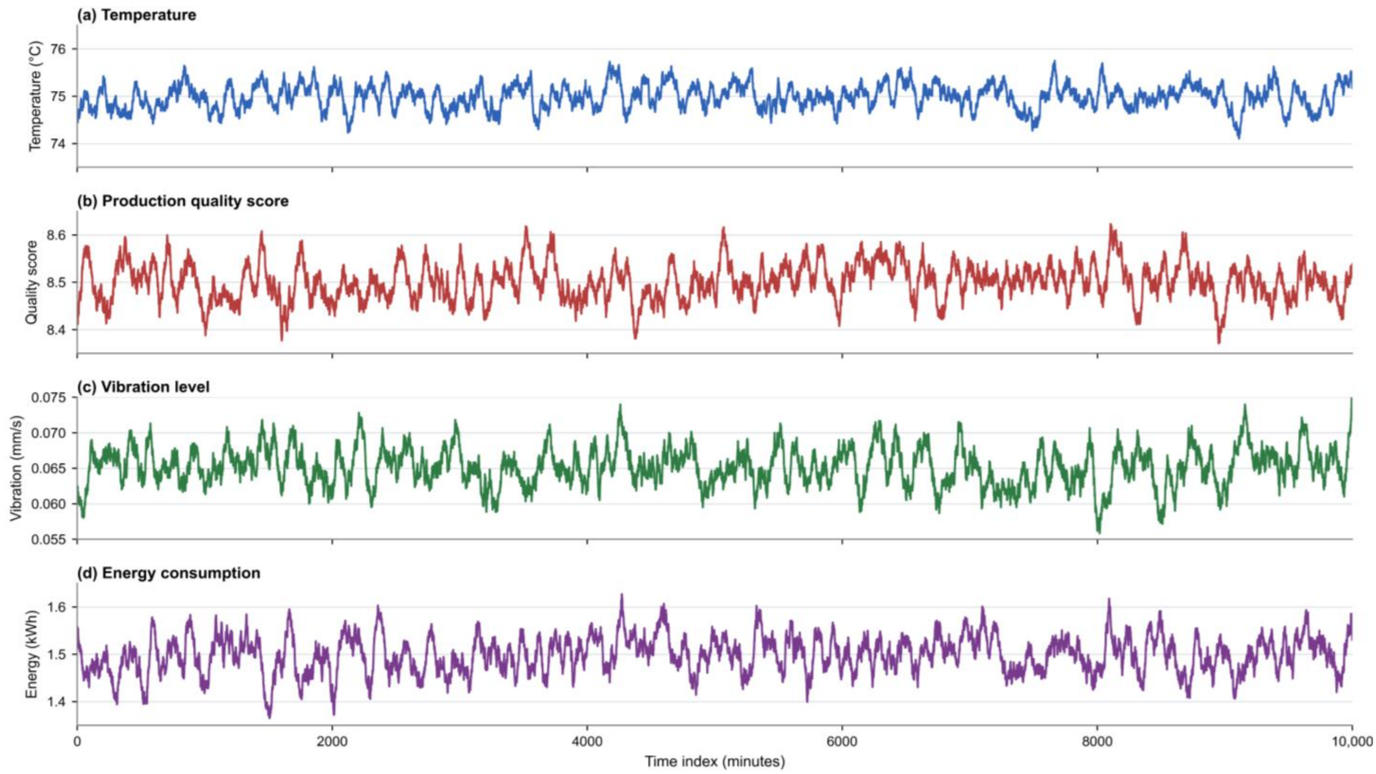

Figure 5. Time-Series Behavior of Manufacturing Process Parameters.

3.3. Dataset-Defined Favorable-State Label Reconstruction and Validation

Within the parameter recommendation framework, the evaluation agent uses the historical favorable-state indicator as dataset-defined evidence for candidate assessment. A rule audit was first conducted to characterize whether this label follows an independently learned state pattern or a deterministic threshold structure.

The threshold-rule reconstruction shows that the Historical Favorable State label can be exactly reproduced by the rule $74 \leq T \leq 76$, $1475 \leq S \leq 1525$, and $V \leq 0.06$. Across all 10,000 records, this rule yields TP = 966, FP = 0, FN = 0, and TN = 9034, indicating that the dataset-defined label follows a deterministic threshold structure. Table 3 summarizes the label-reconstruction results, in which AUC-ROC and average precision are retained as conventional threshold-independent binary evaluation metrics [44–48]. These results confirm that the evaluation module consistently reproduces the dataset-defined favorable-state label structure. Accordingly, the evaluation agent serves as a label-reconstruction and evidence-supply component within the recommendation workflow. The validation emphasis is placed on whether this reconstructed favorable-state evidence supports constraint-first feasible-domain construction, evidence-chain traceability, and Top-K recommendation consistency.

Table 3. Label-reconstruction validation for the dataset-defined favorable-state indicator.

Validation Item	Result	Interpretation
Dataset-defined threshold rule	TP = 966; FP = 0; FN = 0; TN = 9034; accuracy = 1.0000	The dataset label is exactly reconstructable from threshold conditions on T, S and V.
Evaluation-agent classification module (Extreme Gradient Boosting (XGBoost) implementation, 70/30 split)	Accuracy = 1.0000; AUC-ROC = 1.0000; AP = 1.0000	The result verifies that the evaluation module reproduces the dataset-defined label structure.
Role in the framework	Label-structure verification	Workflow validation is based on Top-K constraint compliance, KG adjudication, and evidence-chain traceability.

3.4. Recommendation Consistency and Offline Validation

Building upon the label-reconstruction evaluation, this study further examines whether the reconstructed historical favorable-state label is consistent with the threshold structure embedded in the dataset and with manufacturing stability considerations. Combined with the correlation analysis, conditional distribution analysis, and time-series analysis, vibration-related features reflect the stability boundary used by the dataset label rule. Temperature and machine speed also participate in the exact label definition through bounded intervals, although their isolated linear correlations with the label are weaker than that of vibration. The model behavior is therefore interpreted as reproduction of the dataset-defined state logic within the evaluation module.

An offline recommendation case was constructed from the 10,000-record dataset to validate the complete parameter recommendation workflow. The input requirements were $Q_{\min} = 8.70$, speed range [1480, 1520] RPM, temperature window [70.00, 80.00] °C, vibration safety threshold $V_{\max} = 0.06$ mm/s, search budget $K = 5$, and $\alpha = 0.70$ in the final energy-stability ranking score. Among the 10,000 candidate records, 605 samples satisfy the hard quality, speed, temperature, and vibration gates, and 215 samples also satisfy the Historical Favorable State evidence gate. Because the reported workflow treats vibration as a hard gate, the threshold-exceedance risk term is zero for the retained Top-K candidates, and the final ranking is mainly determined by normalized energy within the validated feasible domain. Table 4 reports the resulting Framework-Optimized Configurations and their execution-level evidence chains.

Table 4. Offline Top-K recommendation example with evidence chain.

Rank	Sample Time	Parameters	Quality	Energy	Evidence Chain and Score
1	6 April 2025 21:59:00	$T = 75.25$ °C; $S = 1508$ RPM; $V = 0.06$ mm/s	8.92	1.00 kWh	$Q \geq 8.70$ pass; $S \in [1480, 1520]$ pass; $T \in [70, 80]$ pass; $V \leq 0.06$ pass; Historical Favorable State = 1; score $= 0$
2	5 April 2025 01:12:00	$T = 74.06$ °C; $S = 1487$ RPM; $V = 0.04$ mm/s	8.83	1.01 kWh	$Q \geq 8.70$ pass; $S \in [1480, 1520]$ pass; $T \in [70, 80]$ pass; $V \leq 0.06$ pass; Historical Favorable State = 1; score $= 0.007$
3	1 April 2025 19:54:00	$T = 74.60$ °C; $S = 1486$ RPM; $V = 0.05$ mm/s	8.86	1.03 kWh	$Q \geq 8.70$ pass; $S \in [1480, 1520]$ pass; $T \in [70, 80]$ pass; $V \leq 0.06$ pass; Historical Favorable State = 1; score $= 0.021$

Table 4. *Cont.*

Rank	Sample Time	Parameters	Quality	Energy	Evidence Chain and Score
4	7 April 2025 23:58:00	$T = 75.37$ °C; $S = 1520$ RPM; $V = 0.05$ mm/s	8.96	1.04 kWh	$Q \geq 8.70$ pass; $S \in [1480, 1520]$ pass; $T \in [70, 80]$ pass; $V \leq 0.06$ pass; Historical Favorable State = 1; score $= 0.028$
5	4 April 2025 21:24:00	$T = 74.56$ °C; $S = 1500$ RPM; $V = 0.06$ mm/s	8.78	1.04 kWh	$Q \geq 8.70$ pass; $S \in [1480, 1520]$ pass; $T \in [70, 80]$ pass; $V \leq 0.06$ pass; Historical Favorable State = 1; score $= 0.028$

Table 5 compares the proposed AD-governed workflow with two ungated baselines using the same candidate pool and deterministic tie-breaking rules. The comparison focuses on whether each strategy satisfies the hard quality, speed, temperature, vibration, and knowledge-graph constraints before returning the Top-20 recommendations.

Table 5. Offline comparison between the proposed governance strategy and ungated baselines.

Method	Gate Order	Top-20 Hard-Constraint Violations	Violation Rate	Best Energy	Interpretation
AD-governed feasible-domain ranking	Hard gates before ranking	0/20	0%	1.00 kWh	Quality, speed, temperature, vibration, and knowledge-graph constraints are satisfied before final ranking.
Weighted energy-stability ranking prior to hard gates	Ranking before hard gates	17/20	85%	1.00 kWh	Low-energy candidates are selected, but many violate at least one hard requirement when audited after ranking under the deterministic tie-breaking rule.
Historical-label/data-driven ranking	Historical label before hard gates	17/20	85%	1.00 kWh	Historical favorable-state evidence alone is insufficient for enforcing hard engineering constraints.

Note: All Top-K and baseline rankings use deterministic tie-breaking for reproducibility. Score-based strategies are sorted by composite score in ascending order, then by quality in descending order, and then by timestamp in ascending order. The historical-label/data-driven baseline first filters $Y_H = 1$ and is sorted by energy consumption in ascending order, then by quality in descending order, and then by timestamp in ascending order. Under this common rule, the weighted-score baseline and the historical-label baseline each yield 17/20 hard-constraint violations.

The proposed workflow achieves a 0% hard-constraint violation rate in the audited Top-20 candidate set because hard requirements are applied before ranking. By contrast, the ungated strategies can return low-energy candidates that violate quality, speed, temperature, or vibration requirements when constraints are checked after ranking. The best energy value remains 1.00 kWh in the proposed feasible-domain ranking, indicating that sequential decoupling preserves the best observed energy level after hard constraints and Historical Favorable State evidence have defined the admissible decision domain.

In the implementation, the LLM is assigned to an evidence-driven semantic-interface role: it verbalizes the selected candidate, checked constraints, ranking logic, and risk warnings from the evidence chain, while feasibility and ranking decisions remain determined by the preceding numerical and rule-based modules.

This case demonstrates the distinction between the Historical Favorable State label used by the evaluation agent and the Framework-Optimized Configuration produced by the complete AD-governed workflow.

4. Discussion

This study re-examines MES-level process parameter optimization under cloud manufacturing environments from the perspective of engineering decision-making and system governance. As summarized in Figure 6, conventional approaches typically adopt an end-to-end optimization paradigm, where decision quality is primarily evaluated through model performance or numerical optimality. In such formulations, multiple objectives including quality, efficiency, and energy consumption are often compressed into a single objective function through implicit weighting schemes, while engineering constraints are treated as post hoc filters applied after optimization. This paradigm is mathematically tractable, but it weakens interpretability, auditability, and risk controllability in industrial contexts. Implicit weights lack clear engineering meaning, are difficult to reuse when production requirements change, and offer limited transparency for post-event analysis. Postponing constraint enforcement also increases the risk of generating recommendations that are numerically attractive but operationally infeasible or unsafe.

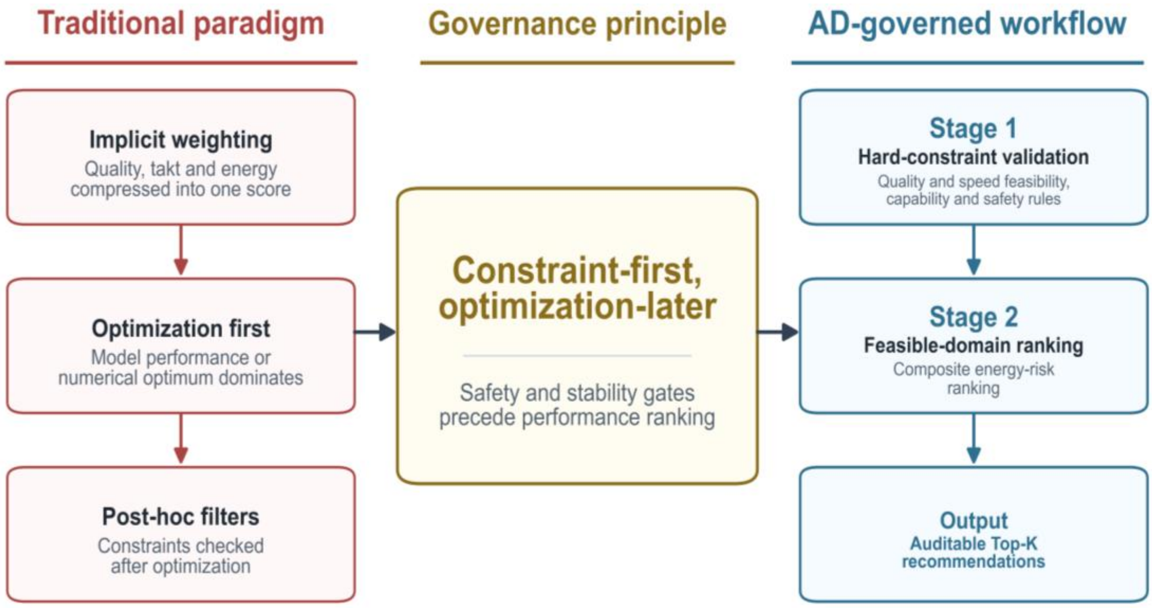

Figure 6. Staged engineering decision process for MES-level parameter recommendation.

The central insight highlighted in Figure 6 is that, in multi-objective and constraint-intensive manufacturing environments, stability and safety must be validated prior to performance-oriented optimization. This constraint-first, optimization-later perspective reframes parameter optimization and recommendation as a staged manufacturing decision-making process. Within the experimental settings of this study, vibration level demonstrates a consistent capability to distinguish between favorable and non-favorable operating states. From statistical correlation analysis, conditional distribution differences, threshold operability, and time-series dynamics, vibration reflects equipment operating stability and latent risk conditions. Vibration anomalies tend to emerge earlier than pronounced degradation in quality indicators, aligning vibration more closely with risk-indicative signals than with outcome-oriented performance metrics. Under the recorded data distribution and label definitions, vibration therefore serves as an engineering-meaningful proxy for stability and supports early-stage constraint validation.

Based on this insight, Figure 6 further illustrates a staged manufacturing decision-making framework for MES parameter optimization and recommendation. In the first stage, hard constraints related to quality thresholds, equipment capability boundaries, and safety rules are treated as non-negotiable conditions. These constraints are explicitly modeled through a knowledge-graph-based adjudication mechanism, which ensures that all candidate parameter configurations satisfy deterministic engineering requirements before entering subsequent evaluation stages. The output of this stage is a feasible candidate set that conforms to engineering executability and safety. In the second stage, optimization is performed within this feasible domain, where objectives such as energy consumption and stability-related risk penalties are evaluated and balanced through composite scoring and ranking. By structurally separating constraint validation from performance optimization, the proposed framework reduces coupling complexity among heterogeneous objectives and enhances the auditability and adaptability of the decision logic.

Within this governance-oriented framework, data-driven models serve as technical components that support specific stages of the decision process. In the present dataset, their classification outputs reconstruct a deterministic favorable-state label defined by temperature, speed, and vibration thresholds. Feature-importance analysis is therefore interpreted as consistency evidence for the dataset-defined label structure. The final recommendation is determined by the combined effects of feasibility gates, knowledge-graph constraints, and energy-risk ranking.

Model outputs are constrained to operate within the governance framework to preserve engineering controllability under data-distribution changes, sample-coverage uncertainty, and potential concept drift. The knowledge-graph-based adjudication mechanism provides the safeguarding layer by encoding equipment capability boundaries, prohibited process rules, and safety thresholds as computable constraints. This design preserves safety margins even when model confidence is high for a candidate that violates a deterministic rule. The effectiveness of this mechanism depends on the completeness and correctness of the underlying constraint knowledge, making knowledge acquisition, verification, and updating central tasks in practical deployment.

The multi-agent collaborative architecture is intended to support the scalability and evolvability of the proposed parameter recommendation workflow. By decoupling candidate generation, constraint validation, and performance evaluation, the implementation provides interfaces for incorporating new models, optimization strategies, or constraint types while preserving the overall decision logic. This modularity is relevant to cloud manufacturing environments characterized by frequent process adjustments and equipment updates. Within this architecture, large language models are assigned to evidence-driven semantic-interface functions, where generated explanations remain consistent with model evaluation results and constraint adjudications. This controlled usage is designed to improve system interpretability and human–machine collaboration efficiency while maintaining governance integrity.

Future work will extend the framework in three directions. First, online deployment studies will be conducted to evaluate long-term evolution phenomena such as equipment aging, maintenance strategy changes, raw-material variability, and concept drift. Second, semi-automated and human-in-the-loop mechanisms will be developed for extracting, validating, and updating knowledge-graph constraints. Third, product-specific validation will be conducted when raw CAD geometry, CAM toolpaths, process-plan files, tool-force measurements, and product identity are available, so that the workflow can be evaluated in process-specific smart machining and CAD/CAM execution settings [49]. Systematic quantitative evaluation of LLM explanation quality will also be added to measure faithfulness, completeness, and usefulness for human operators in MES decision workflows.

5. Conclusions

This paper addresses the challenges of diverse objectives, dense constraints, and continuously evolving operating states in MES parameter configuration under cloud manufacturing environments and proposes an AI-enabled MES-level process parameter optimization and recommendation approach governed by axiomatic design principles. The work conceptualizes MES parameter recommendation and optimization as a governed engineering decision-making process. By explicitly structuring functional requirements, design parameters, and manufacturing constraints, AI capabilities are embedded into a controlled and interpretable decision-making framework, thereby improving the executability, credibility, and engineering trustworthiness of MES parameter recommendation in dynamic manufacturing scenarios.

At the methodology level, axiomatic design is employed to organize manufacturing objectives into functional requirements with different priorities. Quality and production takt are treated as non-violable functional requirements, while energy consumption is evaluated and optimized within the feasible domain defined by hard constraints. These requirements are mapped to key design parameters, including temperature control, machine speed, and vibration level. Through approximate modeling of the design matrix and a sequential decoupling solution strategy, the coupled multi-objective parameter configuration problem is transformed into a series of engineering-executable decision steps following a constraint-first, optimization-later paradigm. This governance-oriented formulation avoids implicit weight tuning and end-to-end black-box optimization, resulting in a structurally transparent and behaviorally controllable MES parameter recommendation process.

Building on this foundation, the manufacturing knowledge graph is utilized as a hard-constraint adjudication mechanism, in which machine capability boundaries, prohibited process rules, and safety thresholds are explicitly modeled and embedded into the MES parameter recommendation workflow. Offline validation demonstrates that, by combining the axiomatic-design-based sequential decoupling strategy with knowledge-graph-based constraint adjudication, the proposed approach can avoid high-risk solutions associated with elevated vibration levels while satisfying hard requirements such as quality thresholds and production takt constraints. The resulting recommendations are constraint-compliant, interpretable, and auditable, highlighting the engineering feasibility and internal consistency of the proposed framework.

Systematic experiments conducted on the publicly available manufacturing process dataset support the internal consistency of the proposed approach and framework design. The dataset-defined Historical Favorable State label can be exactly reconstructed by the threshold rule $74 \leq T \leq 76$, $1475 \leq S \leq 1525$, and $V \leq 0.06$. Accordingly, the classification-module results verify label-structure reconstruction within the evaluation-agent implementation. The recommendation case and baseline comparison further show that the proposed AD-governed workflow can eliminate hard-constraint violations before ranking feasible candidates. Under deterministic tie-breaking, both ungated baselines have 17/20 Top-20 hard-constraint violations, while the AD-governed feasible-domain ranking has 0/20. These results provide data-supported evidence for the constraint-first governance logic in MES-level parameter recommendation.

https://doi.org/10.3390/machines14070787

Author Contributions: Conceptualization, X.C. and K.C.; Methodology, X.C. and K.C.; Software, X.C. and K.C.; Validation, X.C. and K.C.; Formal analysis, X.C. and K.C.; Investigation, X.C.; Resources, X.C.; Data curation, X.C. and K.C.; Writing – original draft preparation, X.C.; Writing – review and editing, X.C. and K.C.; Visualization, X.C. and K.C.; Funding acquisition, X.C. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by the grant from the basic research projects of the educational department of Liaoning province (Grant No. LJ212411035018) and China Scholarship Council (CSC) under Grant 202408210337.

Data Availability Statement: The data presented in this study are available in [Smart Manufacturing Process Data] at [https://www.kaggle.com/datasets/programmer3/smart-manufacturing-process-data], accessed on 19 December 2025, reference number [43].

Acknowledgments: The authors gratefully acknowledge the support provided by the funding agencies and host institutions.

Conflicts of Interest: The author declares no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

1. Tao, F.; Zhang, L.; Venkatesh, V.C.; Luo, Y.; Cheng, Y. Cloud manufacturing: A computing and service-oriented manufacturing model. *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.* **2011**, *225*, 1969–1976. [CrossRef]
2. Shojaeinasab, A.; Charter, T.; Jalayer, M.; Khadivi, M.; Ogunfowora, O.; Raiyani, N.; Yaghoubi, M.; Najjaran, H. Intelligent manufacturing execution systems: A systematic review. *J. Manuf. Syst.* **2022**, *62*, 503–522. [CrossRef]
3. Wu, D.; Rosen, D.W.; Wang, L.; Schaefer, D. Cloud-based design and manufacturing: A new paradigm in digital manufacturing and design innovation. *Comput.-Aided Des.* **2015**, *59*, 1–14. [CrossRef]
4. Wu, D.; Rosen, D.W.; Panchal, J.H.; Schaefer, D. Understanding communication and collaboration in social product development through social network analysis. *ASME J. Comput. Inf. Sci. Eng.* **2016**, *16*, 011001. [CrossRef]
5. Morariu, O.; Borangiu, T.; Răileanu, S.; Morariu, C. Redundancy and scalability for virtualized MES systems with programmable infrastructure. *Comput. Ind.* **2016**, *81*, 26–35. [CrossRef]
6. Deshmankar, A.P.; Challa, J.S.; Singh, A.R.; Regalla, S.P. A review of the applications of machine learning for prediction and analysis of mechanical properties and microstructures in additive manufacturing. *ASME J. Comput. Inf. Sci. Eng.* **2024**, *24*, 120801. [CrossRef]
7. Chen, T.; Guestrin, C. XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794. [CrossRef]
8. Cortes, C.; Vapnik, V. Support-vector networks. In *Machine Learning*; Springer: Berlin/Heidelberg, Germany, 1995; Volume 20, pp. 273–297. [CrossRef]
9. Xu, H.; Liu, R.; Choudhary, A.; Chen, W. A machine learning-based design representation method for designing heterogeneous microstructures. *ASME J. Mech. Des.* **2015**, *137*, 051403. [CrossRef]
10. Tootchi, A.; Du, X. Robust design under machine learning model uncertainty. *ASME J. Mech. Des.* **2026**, *148*, 021701. [CrossRef]
11. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]
12. Bateman, R.J.; Cheng, K. Extending the product portfolio with 'devolved manufacturing': Methodology and case studies. *Int. J. Prod. Res.* **2006**, *44*, 3325–3343. [CrossRef]
13. Yhdego, T.O.; Wang, H.; Chi, H.; Yu, Z. Ontology-guided data sharing and federated quality control with differential privacy in additive manufacturing. *ASME J. Comput. Inf. Sci. Eng.* **2025**, *25*, 011006. [CrossRef]
14. Kritzinger, W.; Karner, M.; Traar, G.; Henjes, J.; Sihn, W. Digital Twin in manufacturing: A categorical literature review and classification. *IFAC-PapersOnLine* **2018**, *51*, 1016–1022. [CrossRef]
15. Cimino, C.; Negri, E.; Fumagalli, L. Review of digital twin applications in manufacturing. *Comput. Ind.* **2019**, *113*, 103130. [CrossRef]
16. Wu, C.; Zhou, Y.; Pessôa, M.; Peng, Q.; Tan, R. Conceptual digital twin modelling based on an integrated five-dimensional framework and TRIZ function model. *J. Manuf. Syst.* **2021**, *58*, 79–93. [CrossRef]
17. Nath, P.; Mahadevan, S. Probabilistic digital twin for additive manufacturing process design and control. *ASME J. Mech. Des.* **2022**, *144*, 091704. [CrossRef]

18. He, B.; Song, Y.; Wang, Y. Special Issue: Digital twin-driven design and manufacturing. *ASME J. Comput. Inf. Sci. Eng.* **2021**, *21*, 030301. [CrossRef]
19. Chen, Y.P.; Tsai, Y.K.; Karkaria, V.; Chen, W. Uncertainty-aware digital twins: Robust model predictive control using time-series deep quantile learning. *ASME J. Mech. Des.* **2026**, *148*, 021702. [CrossRef]
20. Wan, Y.; Liu, Y.; Chen, Z.; Chen, C.; Li, X.; Hu, F.; Packianather, M. Making knowledge graphs work for smart manufacturing: Research topics, applications and prospects. *J. Manuf. Syst.* **2024**, *76*, 103–132. [CrossRef]
21. Liu, B.; Chen, C.-H.; Wang, Z. A multi-hierarchical aggregation-based graph convolutional network for industrial knowledge graph embedding towards cognitive intelligent manufacturing. *J. Manuf. Syst.* **2024**, *76*, 320–332. [CrossRef]
22. Chhim, P.; Chinnam, R.B.; Sadawi, N. Product design and manufacturing process based ontology for manufacturing knowledge reuse. *J. Intell. Manuf.* **2019**, *30*, 905–916. [CrossRef]
23. Siddharth, L.; Blessing, L.T.M.; Wood, K.L.; Luo, J. Engineering knowledge graph from patent database. *ASME J. Comput. Inf. Sci. Eng.* **2022**, *22*, 021008. [CrossRef]
24. Wu, T.; Li, J.; Bao, J.; Liu, Q.; Jin, Z.; Gao, J. CarbonKG: Industrial carbon emission knowledge graph-based modeling and application for carbon traceability of complex manufacturing process. *ASME J. Comput. Inf. Sci. Eng.* **2024**, *24*, 081001. [CrossRef]
25. Hagedorn, T.J.; Krishnamurty, S.; Grosse, I.R. A knowledge-based method for innovative design for additive manufacturing supported by modular ontologies. *ASME J. Comput. Inf. Sci. Eng.* **2018**, *18*, 021009. [CrossRef]
26. Zhong, Y.; Qin, Y.; Huang, M.; Lu, W.; Gao, W.; Du, Y. Automatically generating assembly tolerance types with an ontology-based approach. *Comput.-Aided Des.* **2013**, *45*, 1253–1275. [CrossRef]
27. Hanna, L.; Cagan, J. Evolutionary multi-agent systems: An adaptive and dynamic approach to optimization. *ASME J. Mech. Des.* **2009**, *131*, 011010. [CrossRef]
28. Cao, J.; Ming, Z.; Allen, J.K.; Mistree, F. Design of multi-agent systems based on bilevel closed-loop planning. *ASME J. Mech. Des.* **2025**, *147*, 081706. [CrossRef]
29. Chen, Q.; Ilami, S.; Lore, N.; Heydari, B. Adaptive information modulation: Designing governance mechanisms for multi-agent AI systems. *ASME J. Mech. Des.* **2026**, *148*, 041708. [CrossRef]
30. Du, K.; Yang, B.; Xie, K.; Dong, N.; Zhang, Z.; Wang, S.; Mo, F. LLM-MANUF: An integrated framework of fine-tuning large language models for intelligent decision-making in manufacturing. *Adv. Eng. Inform.* **2025**, *65*, 103263. [CrossRef]
31. Wang, T.; Zhang, B.; Jiang, D.; Li, D. A multimodal large language model framework for intelligent perception and decision-making in smart manufacturing. *Sensors* **2025**, *25*, 3072. [CrossRef] [PubMed]
32. Massoudi, S.; Fuge, M. Agentic large language models for conceptual systems engineering and design. *ASME J. Mech. Des.* **2026**, *148*, 051405. [CrossRef]
33. Li, Y.; Ko, H.; Ameri, F. Integrating graph retrieval-augmented generation with large language models for supplier discovery. *ASME J. Comput. Inf. Sci. Eng.* **2025**, *25*, 021010. [CrossRef]
34. Sun, Y.; Li, X.; Sha, Z. Large language models for computer-aided design fine tuned: Dataset and experiments. *ASME J. Mech. Des.* **2025**, *147*, 041710. [CrossRef]
35. Zhang, C.; Zhao, Y.F.; El Haddad, R. Understanding the impact of applying large language model in engineering design education. *ASME J. Comput. Inf. Sci. Eng.* **2025**, *25*, 021008. [CrossRef]
36. Fan, H.; Chow, E.; Lu, T.; Fuh, J.Y.H.; Lu, W.F.; Li, B. A unified framework for large language model-guided reinforcement learning in digital twin industrial environments. *Robot. Comput.-Integr. Manuf.* **2026**, *99*, 103215. [CrossRef]
37. Qin, Z.; Lu, Y. Knowledge graph-enhanced multi-agent reinforcement learning for adaptive scheduling in smart manufacturing. *J. Intell. Manuf.* **2025**, *36*, 5943–5966. [CrossRef]
38. Sun, Z.; Zheng, C.; Sun, X.; Hou, X.; Yang, Y. Dual multi-objective optimization design method for compliant guide mechanism. *Struct. Multidiscip. Optim.* **2024**, *67*, 75. [CrossRef]
39. Suh, N.P. Axiomatic design of mechanical systems. *ASME J. Mech. Des.* **1995**, *117*, 2–10. [CrossRef]
40. Suh, N.P. Design of systems. *CIRP Ann.* **1997**, *46*, 75–80. [CrossRef]
41. Hazelrigg, G.A. An axiomatic framework for engineering design. *ASME J. Mech. Des.* **1999**, *121*, 342–347. [CrossRef]
42. Rauch, E.; Matt, D.T.; Dallasega, P. Application of axiomatic design in manufacturing system design: A literature review. *Procedia CIRP* **2016**, *53*, 1–7. [CrossRef]
43. Python Developer. Smart Manufacturing Process Data. Available online: https://www.kaggle.com/datasets/programmer3/smart-manufacturing-process-data (accessed on 19 December 2025).
44. Pivin-Bachler, J.R.; van den Broek, E.L. SIMBA: A robust and generalizable measure of data imbalance. *Patterns* **2025**, *6*, 101395. [CrossRef] [PubMed]
45. Hanley, J.A.; McNeil, B.J. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology* **1982**, *143*, 29–36. [CrossRef] [PubMed]
46. DeLong, E.R.; DeLong, D.M.; Clarke-Pearson, D.L. Comparing the areas under two or more correlated receiver operating characteristic curves: A nonparametric approach. *Biometrics* **1988**, *44*, 837–845. [CrossRef] [PubMed]

47. Davis, J.; Goadrich, M. The relationship between Precision–Recall and ROC curves. In Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh, PA, USA, 25–29 June 2006; pp. 233–240. [CrossRef]
48. Saito, T.; Rehmsmeier, M. The Precision–Recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS ONE* **2015**, *10*, e0118432. [CrossRef] [PubMed]
49. Cheng, K.; Niu, Z.C.; Wang, R.C.; Rakowski, R.; Bateman, R.J. Smart cutting tools and smart machining: Development approaches, and their implementation and application perspectives. *Chin. J. Mech. Eng.* **2017**, *30*, 1162–1176. [CrossRef]