processes

MDPI

Article

Digital Twin-Assisted Multi-Objective Optimization Method Based on Multi-Agent Reinforcement Learning for Five-Axis CNC Machining

Jialin Li [1,*], Jiang Li [1], Xin Zhou [2] and Jinliang An [1]

[1] Laboratory of Numerical Control Engineering and Applied Technology, Guiyang University, Guiyang 550005, China
[2] College of Computer Science, Guiyang University, Guiyang 550005, China
* Correspondence: gzyegui@126.com

Abstract

Five-axis CNC machining involves strong coupling among machining quality, material removal efficiency, and operational safety, making it difficult to obtain adaptive and feasible process parameters using conventional scalar-objective optimization methods. To address this problem, this study proposes a physics-constrained multi-objective multi-agent deep deterministic policy gradient framework, termed MOMADDPG, for Pareto-oriented optimization of five-axis machining parameters. A data-calibrated digital twin simulation environment is constructed to model five-axis kinematics, tool-workpiece engagement, cutting force, chatter tendency, spindle power, tool wear, actuator bounds, and collision risk. The PHM Society 2010 milling dataset is used to calibrate the cutting force and tool wear sub-models, while five-axis motion, tool orientation variation, and engagement conditions are generated within the digital twin environment. In the proposed framework, three heterogeneous agents are assigned to quality preservation, efficiency improvement, and safety assurance, respectively. A hierarchical attention Actor is designed to enhance feature extraction under partially observable machining conditions, while vector-valued dual Critics preserve objective-specific value information. Physical constraints are handled using adaptive Lagrangian multipliers, and a Pareto archive-guided preference curriculum is introduced to improve the diversity of feasible non-dominated solutions. Simulation results show that MOMADDPG achieves a task success rate of 98% and a hypervolume value of 0.674 after training. Compared with representative baselines, including DQN, MADDPG, MAAC, and MAPPO, the proposed method provides better Pareto-front approximation, higher task feasibility, and stronger robustness under process perturbations in the data-calibrated five-axis simulation environment. The results demonstrate the potential of combining digital twins and multi-objective multi-agent reinforcement learning for safe and adaptive parameter optimization in five-axis machining simulations. Further validation on physical five-axis CNC systems is still required before industrial deployment.

Keywords: five-axis CNC machining; digital twin; multi-agent reinforcement learning; multi-objective optimization; Pareto-optimal control; physical constraints

Check for updates

Academic Editor: Zhibin Lin

Received: 25 May 2026
Revised: 22 June 2026
Accepted: 25 June 2026
Published: 1 July 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Five-axis computer numerical control (CNC) machining is a key enabling technology for manufacturing high-value components with complex freeform surfaces, such as impellers, turbine blades, and lightweight structural parts. These components usually

involve rapidly varying surface curvature, thin-walled local structures, and difficult-to-cut materials, which impose stringent requirements on dimensional accuracy, surface integrity, machining efficiency, and process stability [1]. Compared with three-axis machining, five-axis machining provides higher geometric flexibility by continuously changing the tool orientation along the tool path. However, this advantage also introduces strong process uncertainty. The variation in tool posture changes the instantaneous tool-workpiece engagement, chip thickness, cutting force direction, local stiffness, and collision margin. As a result, surface quality, material removal efficiency, chatter stability, spindle power, and tool wear become strongly coupled during machining [2,3]. In industrial practice, process engineers often adopt conservative constant parameters generated by computer-aided manufacturing (CAM) systems to avoid unstable cutting or tool failure. Although such strategies improve safety, they reduce the utilization of high-performance machine tools and limit machining efficiency.

Considerable research has been conducted on cutting parameter optimization over the past decades. Empirical regression, response surface methodology, and orthogonal experiments have been widely used to establish mappings among process parameters, cutting forces, surface roughness, and tool wear. Based on these models, multi-objective evolutionary algorithms, such as non-dominated sorting genetic algorithm II (NSGA-II), have been adopted to search for trade-off solutions in the parameter space [4,5]. These methods are effective when the machining condition is close to the experimental design space, but their prediction accuracy decreases when tool orientation, engagement geometry, or material removal conditions change significantly. Finite element methods can provide detailed thermo-mechanical responses in the cutting zone, but their high computational cost makes them difficult to embed into iterative optimization or online decision-making loops [6]. Analytical and semi-analytical models, including cutting force models and frequency-domain chatter stability analysis, offer better computational efficiency [7]. Nevertheless, their generalization capability is still limited under complex five-axis conditions where the tool posture and engagement state vary continuously. Therefore, conventional optimization methods remain largely offline and are insufficient for adaptive parameter control along dynamic five-axis tool paths.

Digital twin technology provides a promising solution for this limitation by constructing a virtual environment that reflects the physical behavior of machining processes. Tao et al. proposed a five-dimensional digital twin framework for manufacturing systems, emphasizing the integration of physical entities, virtual models, services, twin data, and connection mechanisms [8]. Kritzinger et al. clarified the role of bidirectional data flow in distinguishing digital twins from digital shadows and digital models [9]. At the machine-tool level, Luo et al. developed digital twin models for CNC machine tools and explored their applications in process simulation and error compensation [10]. Söderberg et al. discussed the potential of digital twins in closed-loop manufacturing quality control [11]. These studies show that digital twins can provide a high-fidelity virtual platform for monitoring, prediction, and process evaluation. However, existing machining-oriented digital twins mainly focus on geometric accuracy, thermal error, or process monitoring. Their ability to simultaneously model cutting force, regenerative chatter, spindle power, tool wear evolution, actuator limits, and collision-related constraints remains insufficient. Moreover, when a digital twin is used as an interactive environment for reinforcement learning, its response models should be calibrated and validated with machining data whenever possible, so that state transitions, reward vectors, and constraint violation signals can provide physically meaningful feedback for policy learning.

Reinforcement learning has attracted increasing attention because it can learn sequential decision policies through interaction with an environment. In continuous control

domains, deep reinforcement learning has demonstrated strong performance in robotic control [12] and autonomous driving [13]. Deep Q-Network (DQN) introduced experience replay and target networks to stabilize value learning in high-dimensional state spaces [14]. Deep Deterministic Policy Gradient (DDPG) extended deterministic policy gradient methods to continuous action spaces through an Actor-Critic architecture [15]. Proximal Policy Optimization (PPO) further improved policy optimization stability by constraining policy updates [16]. In manufacturing, reinforcement learning has been applied to job shop scheduling [17] and robot trajectory planning [18]. Nevertheless, directly applying conventional reinforcement learning to cutting parameter optimization faces two main challenges. First, real-machine exploration is expensive and risky because unsafe actions may cause scrap parts, tool breakage, or machine damage. Second, machining parameter optimization is inherently multi-objective. Improving material removal efficiency often increases cutting load, chatter risk, and tool wear, while improving surface quality may require conservative cutting parameters. Traditional scalar reward formulations cannot fully represent these conflicts, and fixed weighting coefficients cannot ensure a diverse approximation of the Pareto frontier [19,20].

Multi-agent reinforcement learning offers a natural framework for decomposing complex control tasks into coordinated decision-making processes. Multi-Agent Deep Deterministic Policy Gradient (MADDPG) adopts the centralized-training-decentralized-execution paradigm, allowing agents to learn coordinated policies while executing actions based on local observations [19]. Multi-Agent Actor–Attention–Critic (MAAC) introduces attention mechanisms into multi-agent actor–critic learning, enabling agents to focus on more relevant interaction information [21]. Multi-Agent Proximal Policy Optimization (MAPPO) has also shown competitive performance in cooperative multi-agent tasks when carefully tuned [22]. In parallel, multi-objective reinforcement learning extends scalar rewards to vector-valued rewards and studies policy evaluation under conflicting objectives [20]. Yang et al. proposed generalized multi-objective reinforcement learning to support policy adaptation under different preferences [23]. These studies provide useful foundations for multi-agent coordination and multi-objective policy learning. However, their integration for safety-critical machining parameter optimization remains underdeveloped. In particular, few studies have addressed how heterogeneous agents can separately represent quality, efficiency, and safety objectives, how vector-valued value functions can preserve objective-specific information, how multiple objective-oriented agents can be coordinated to generate executable machining actions, and how hard physical constraints can be incorporated into Pareto-oriented policy learning.

The above analysis indicates several open issues. First, existing digital twin models for machining are not sufficiently designed as reinforcement learning environments that provide physically consistent and data-validated state transitions, reward vectors, and constraint violation signals. Second, traditional multi-objective optimization methods are mostly offline and cannot adapt process parameters according to dynamic machining states. Third, current reinforcement learning methods often rely on scalarized rewards and do not explicitly maintain the independent value information of different objectives. Fourth, the use of heterogeneous multi-agent coordination for quality–efficiency–safety optimization in five-axis machining remains insufficiently explored, especially under physical constraints such as chatter stability, spindle power, tool wear, actuator bounds, and collision margin. In addition, when public machining datasets are used to calibrate digital twin sub-models, the boundary between measured data, calibrated response models, and simulated five-axis variables should be clearly stated to avoid overgeneralizing simulation results to real machining conditions.

To address these challenges, this paper proposes a physics-constrained multi-objective multi-agent deep deterministic policy gradient framework, termed MOMADDPG, for Pareto-oriented parameter optimization of five-axis CNC machining in a data-calibrated digital twin simulation environment. The proposed framework combines a digital twin machining environment, heterogeneous multi-agent decision-making, vector-valued dual-Critic learning, Lagrangian constraint handling, and Pareto archive-guided preference curriculum. The novelty of this work lies in two aspects. From the methodological perspective, MOMADDPG integrates heterogeneous quality–efficiency–safety agents, vector-valued dual-Critic learning, adaptive Lagrangian constraint handling, and archive-guided preference learning into a unified constrained multi-objective reinforcement learning framework. From the application perspective, this framework is embedded into a five-axis machining digital twin simulation environment that incorporates kinematics, tool-workpiece engagement, machining response models, and safety-related constraints. The study is therefore positioned as simulation-based validation of a PHM-calibrated digital twin and multi-objective multi-agent reinforcement learning framework, while validation on physical five-axis CNC systems remains future work. The main contributions are summarized as follows.

First, a physics-informed and data-calibrated digital twin environment is constructed for five-axis machining parameter optimization. The environment integrates five-axis kinematics, tool-workpiece engagement, cutting force prediction, regenerative chatter evaluation, spindle power estimation, tool wear evolution, actuator bounds, and collision checking. The Prognostics and Health Management (PHM) Society 2010 milling dataset is used to calibrate and validate the cutting force and tool wear sub-models, while five-axis motion, tool orientation variation, engagement geometry, and collision-related constraints are generated within the digital twin environment. This setting enables the environment to provide physically meaningful state observations, vector-valued rewards, and constraint violation signals for safe offline reinforcement learning.

Second, a heterogeneous multi-agent control architecture is developed for quality–efficiency–safety collaborative optimization. Three specialized agents are assigned to quality preservation, efficiency improvement, and safety assurance, respectively. Each agent employs a hierarchical attention-based Actor to extract process-related, constraint-sensitive, and interaction features from local observations, thereby supporting decentralized execution under partially observable and strongly coupled machining conditions. The local action proposals of the three agents are further fused into executable machining parameter adjustments, and the detailed aggregation mechanism is described in Section 4.

Third, a physics-constrained Pareto learning mechanism is proposed by integrating vector-valued dual-Critic learning, adaptive Lagrangian constraint handling, and Pareto archive-guided preference curriculum. The dual Critics preserve objective-specific Q-value information instead of prematurely scalarizing rewards, while the Lagrangian penalties enforce machining safety constraints. Completed trajectories are evaluated at the trajectory level, and feasible non-dominated solutions are stored in an external Pareto archive to guide preference sampling toward sparse or under-explored trade-off regions. Comparative experiments against DQN, MADDPG, MAAC, and MAPPO are conducted in the same digital twin simulation environment to evaluate convergence behavior, Pareto-front quality, ablation performance, robustness under simulated process perturbations, and computational efficiency.

The remainder of this paper is organized as follows. Section 2 reviews related studies on digital twin-based machining modeling, multi-objective optimization, reinforcement learning, and multi-agent reinforcement learning. Section 3 describes the construction of the five-axis machining digital twin environment, including process response modeling,

physical constraint formulation, and reinforcement learning interfaces. Section 4 presents the proposed MOMADDPG framework, including heterogeneous agent modeling, hierarchical attention Actors, constrained dual-Critic learning, Pareto archive-guided curriculum, and the complete training procedure. Section 5 reports the experimental setup, digital twin validation, comparative results, ablation analysis, robustness evaluation, computational efficiency analysis, and industrial applicability discussion.

2. Related Works

The optimization of machining parameters in five-axis CNC milling has long been a central topic in advanced manufacturing, because machining quality, material removal efficiency, and operational safety are simultaneously affected by highly coupled process variables. In five-axis machining, the continuously varying tool orientation changes the instantaneous tool-workpiece engagement condition, which further influences cutting force, chatter tendency, tool wear, spindle power, and surface integrity. Consequently, static parameter selection based on empirical experience or offline process planning is often insufficient for complex freeform surface machining. With the development of digital twins and learning-based control, recent studies have begun to explore data-driven and physics-informed decision-making methods for machining process optimization. This section reviews the related literature from four aspects: digital twin-based machining process modeling, multi-objective optimization of machining parameters, reinforcement learning for manufacturing control, and multi-agent/multi-objective reinforcement learning.

Digital twin technology provides an effective means of linking physical machining processes with virtual models. Tao et al. proposed a five-dimensional digital twin framework for manufacturing systems, emphasizing the integration of physical entities, virtual models, services, twin data, and connections [8]. Kritzinger et al. further clarified the distinction between digital models, digital shadows, and digital twins, and pointed out that model fidelity and bidirectional interaction remain critical challenges in manufacturing applications [9]. At the machine-tool level, Liu et al. developed a cyber–physical machine tool platform using industrial communication protocols, enabling data acquisition and process monitoring [24]. Luo et al. investigated digital twin modeling strategies for CNC machine tools and discussed their potential in machining process simulation [10]. These studies have established the foundation for virtual representation and process-state monitoring. However, many existing digital twin models mainly focus on geometric accuracy, thermal error, or monitoring functions, while the coupled dynamics of cutting force, regenerative chatter, tool wear, spindle power, and collision-related safety constraints are not sufficiently integrated for reinforcement learning-based process optimization. In addition, when digital twins are used as learning environments, the credibility of the simulated responses depends on whether key sub-models are calibrated and validated with machining data. Existing studies have not sufficiently addressed how data-calibrated response models and five-axis simulation variables can be jointly used to construct a reinforcement-learning-oriented machining environment.

Multi-objective optimization has also been extensively studied in machining parameter selection. Since surface quality, machining efficiency, energy consumption, and tool life are usually conflicting objectives, Pareto-based optimization methods are commonly adopted. Yusup et al. reviewed evolutionary techniques for machining parameter optimization and showed that algorithms such as NSGA-II can obtain reasonable trade-offs between material removal rate and surface roughness [4]. Deb et al. developed NSGA-II as a representative elitist multi-objective evolutionary algorithm, which has been widely used in manufacturing optimization problems [5]. Decomposition-based methods such as multi-objective evolutionary algorithm based on decomposition (MOEA/D) have also been

applied to machining parameter optimization to improve convergence in high-dimensional objective spaces [25]. In addition, surrogate-assisted optimization has been used to reduce the cost of finite element simulation or physical experiments [26]. Despite these advances, most evolutionary or surrogate-based methods remain essentially offline optimization approaches. They usually require repeated evaluations of candidate solutions and are difficult to adapt to time-varying machining states along a complex five-axis tool path. More importantly, physical constraints such as chatter stability, spindle power limits, actuator saturation, and collision margins are often treated as external penalties rather than as explicit safety signals during sequential decision-making.

Reinforcement learning offers a promising paradigm for adaptive control because it learns control policies through interaction with an environment. Classical deep reinforcement learning methods, such as Deep Q-Network (DQN) [14] and Deep Deterministic Policy Gradient (DDPG) [15], have demonstrated strong performance in high-dimensional perception and continuous control tasks. In manufacturing, reinforcement learning has been applied to scheduling, robotic manipulation, and trajectory planning. Kuhnle et al. introduced reinforcement learning into adaptive order dispatching in job shop manufacturing systems [17], while Wang et al. used reinforcement learning-based control for robotic trajectory planning [27]. Earlier symbolic and learning-based planning studies also showed the potential of reinforcement learning for sequential decision problems in robotic systems [28]. However, direct application of conventional reinforcement learning to machining parameter optimization remains challenging. First, trial-and-error exploration on real machine tools is costly and unsafe, which requires a reliable simulation or digital twin environment. Second, most existing reinforcement learning methods rely on scalar rewards, whereas machining optimization is inherently multi-objective. A fixed weighted reward may hide the trade-off structure among quality, efficiency, and safety, and cannot guarantee a diverse approximation of the Pareto frontier.

Multi-agent reinforcement learning provides a useful framework for modeling distributed decision-making and objective-specific coordination. Lowe et al. proposed multi-agent deep deterministic policy gradient (MADDPG) based on the centralized-training-decentralized-execution paradigm, allowing agents to learn coordinated policies under non-stationary multi-agent interactions [19]. Iqbal et al. introduced attention mechanisms into multi-agent actor–critic learning, enabling agents to focus on the most relevant information from other agents [21]. Yu et al. showed that properly tuned PPO-based multi-agent methods can also achieve competitive performance in cooperative multi-agent tasks [22]. In manufacturing systems, multi-agent reinforcement learning has mainly been applied to production scheduling, resource allocation, and distributed system control [29]. Nevertheless, the use of heterogeneous agents for machining parameter optimization remains limited. In particular, how to assign specialized agents to quality preservation, efficiency improvement, and safety assurance, how to coordinate their action proposals into executable machining parameter adjustments, and how to maintain feasibility under physical constraints have not been sufficiently investigated.

Multi-objective reinforcement learning has been developed to overcome the limitations of scalar reward formulations. Vamplew et al. discussed the extension from scalar rewards to vector-valued rewards and summarized evaluation criteria for multi-objective reinforcement learning [20]. Mossalam et al. proposed a multi-objective deep reinforcement learning method based on decomposed value functions, aiming to approximate different trade-off policies [30]. Yang et al. further studied generalized multi-objective reinforcement learning and policy adaptation, showing that preference-conditioned learning can improve the flexibility of multi-objective policy search [23]. These studies provide useful theoretical and algorithmic foundations for Pareto-oriented policy learning. However,

most of them are validated in benchmark environments with relatively simple dynamics. Their direct application to five-axis CNC machining is non-trivial because the problem involves continuous machining parameter adjustment, heterogeneous objective-specific decision making, data-calibrated digital twin responses, and hard physical constraints. Therefore, comparisons with conventional scalarized reinforcement learning baselines should be interpreted together with the differences in multi-objective learning mechanisms and constraint-handling strategies.

Table 1 summarizes the main characteristics and limitations of representative related studies. Existing research has made substantial progress in digital twin modeling, evolutionary optimization, reinforcement learning, and multi-objective policy learning. However, these directions are often developed separately. Digital twins are usually used for monitoring or offline simulation; multi-objective optimization methods often lack sequential adaptability; and reinforcement learning methods frequently rely on scalar rewards or simplified constraints. Therefore, a unified framework is still needed to integrate a data-calibrated digital twin simulation environment, heterogeneous multi-agent coordination, explicit constraint handling, and Pareto-frontier exploration for five-axis machining parameter optimization.

Table 1. Summary of representative related studies and their relevance to this work.

Research Direction	Representative Studies	Main Contribution	Limitation	Difference in This Work
Digital twin for manufacturing and CNC systems	[8–10,24]	Establish virtual-physical mapping, process monitoring, and CNC digital twin modeling	Limited integration of cutting dynamics, chatter, tool wear, power, collision risk, and RL-oriented interaction	Constructs a data-calibrated five-axis digital twin simulation environment that provides state transitions, reward vectors, and constraint violation signals
Multi-objective machining optimization	[4,5,25,26]	Obtain Pareto trade-offs using evolutionary or surrogate-based optimization	Mainly offline; limited adaptability to time-varying machining states and explicit safety constraints	Performs sequential parameter optimization through reinforcement learning under physical constraints
Reinforcement learning for manufacturing control	[14,15,17,27,28]	Learn sequential decision policies for scheduling, robotics, and continuous control	Mostly scalar-reward based; costly real-machine exploration and weak Pareto-front representation	Uses vector-valued rewards and a digital twin simulation environment for safe offline policy learning
Multi-agent reinforcement learning	[19,21,22,29]	Enable coordinated learning under centralized training with decentralized execution and attention-based interaction	Rarely applied to heterogeneous objective-specific machining parameter control	Assigns heterogeneous agents to quality, efficiency, and safety objectives and fuses their action proposals into executable machining parameter updates

Based on the above review, three research gaps can be identified. First, existing digital twin models for machining are not sufficiently designed as physics-informed interaction environments for reinforcement learning, especially when cutting force, chatter, tool wear, power, actuator limits, and collision risks must be evaluated simultaneously. Second,

conventional multi-objective optimization methods can approximate Pareto solutions, but they are generally offline and cannot continuously adapt process parameters along a dynamic five-axis machining trajectory. Third, current reinforcement learning and multi-agent reinforcement learning methods have not fully addressed vector-valued objective preservation, heterogeneous objective-specific agents, and explicit physical constraint satisfaction in one unified framework.

To address these gaps, this paper proposes a physics-constrained multi-objective multi-agent deep deterministic policy gradient framework for five-axis CNC machining. A digital twin environment is constructed to provide physically meaningful state transitions, reward vectors, and constraint violation signals. Three heterogeneous agents are assigned to quality, efficiency, and safety objectives, respectively. A hierarchical attention Actor, a vector-valued dual-Critic mechanism, and an adaptive Lagrangian constraint penalty are integrated to support constrained Pareto-oriented policy learning. Furthermore, a Pareto archive-guided preference curriculum is introduced to improve the diversity and distribution of feasible non-dominated machining policies. This framework aims to provide a simulation-validated solution for safe, adaptive, and Pareto-oriented optimization of five-axis machining parameters, while physical CNC validation remains necessary before industrial deployment.

3. Multi-Objective Coupling Modeling and Twin Environment Construction

To support safe and data-efficient policy learning, a digital twin machining environment is constructed for five-axis CNC process optimization. The environment provides a closed-loop simulation platform that links process parameters, tool-workpiece engagement, machining responses, and physical constraints. Compared with direct trial-and-error learning on real machine tools, the digital twin allows reinforcement learning agents to explore different control strategies with lower risk and lower experimental cost.

The digital twin environment is designed to reproduce the main dynamic characteristics of five-axis machining, including cutting force variation, material removal rate, spindle power, tool wear, surface quality, chatter tendency, and collision risk. These machining responses are updated according to the current machining state and the control actions generated by the agents. Therefore, the environment serves not only as a simulator for state transition but also as a physics-informed evaluator for reward calculation and constraint verification.

In this study, the digital twin environment provides three key functions. First, it generates local observations for heterogeneous agents according to the current machining condition. Second, it evaluates the quality, efficiency, and safety objectives under different cutting strategies. Third, it checks whether the generated actions violate physical constraints, such as chatter stability, spindle power limitation, tool wear threshold, actuator boundary, and collision margin. The outputs of the digital twin are then used to train the proposed MOMADDPG framework.

3.1. Digital Twin Architecture

The proposed digital twin environment consists of four modules: the machining state module, the process response module, the constraint evaluation module, and the learning interface module. The overall structure is designed to connect physical machining knowledge with multi-agent reinforcement learning.

The overall interaction mechanism between the digital twin machining environment and the reinforcement learning controller is shown in Figure 1. The multi-agent controller first generates objective-specific action proposals according to the current observations. These proposals are fused into a final machining action by the action aggregation mecha-

nism described in Section 4.1. After parameter clipping within feasible machining bounds, the updated process parameters are evaluated by the physics-informed five-axis digital twin. The digital twin returns the next state, local observations, objective-specific rewards, and physical constraint signals to support off-policy multi-agent learning.

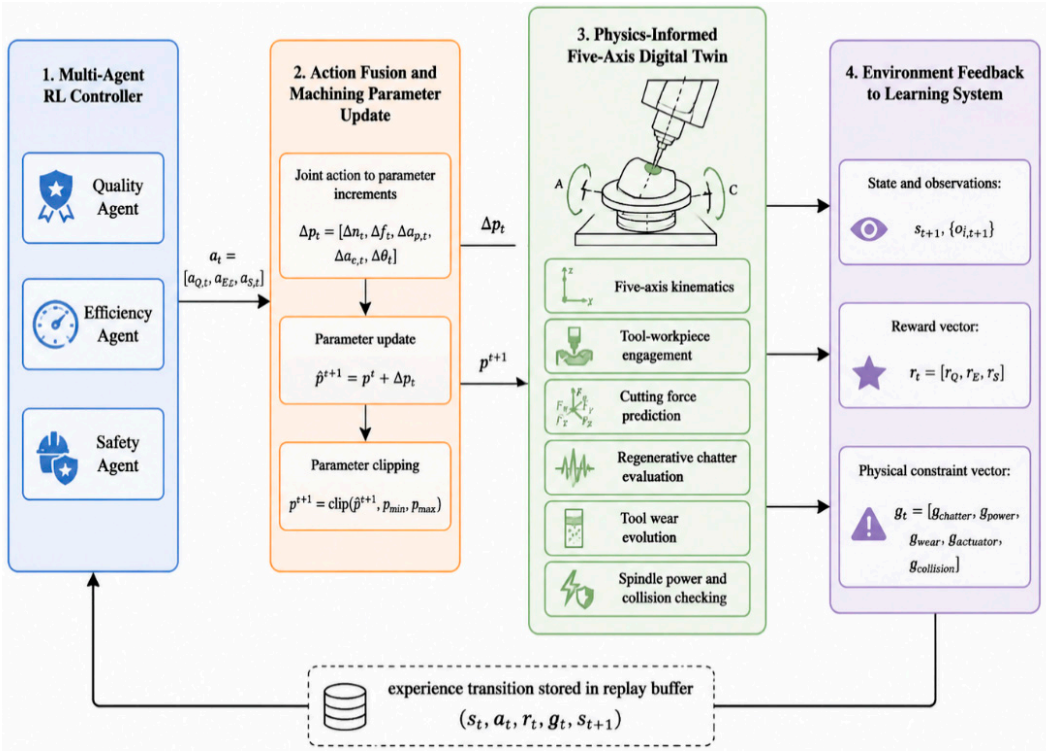


Figure 1. Digital Twin-Based Five-Axis Machining Environment and Reinforcement Learning Interface.

At time step t, the machining state is defined as

$$s_t = [\mathbf{x}_t, \mathbf{p}_t, \mathbf{y}_t, \mathbf{h}_t] \tag{1}$$

where $\mathbf{x}_t$ denotes the tool position and posture information, $\mathbf{p}_t$ represents controllable process parameters such as spindle speed, feed rate, and cutting depth, $\mathbf{y}_t$ denotes machining response variables including cutting force, vibration, power, and temperature, and $\mathbf{h}_t$ contains historical degradation information such as accumulated tool wear and previous stability states.

The action vector generated by the control policy is expressed as

$$\mathbf{a}_t = \Delta p_t = [\Delta n_t, \Delta f_t, \Delta a_{p,t}, \Delta a_{e,t}, \Delta\boldsymbol{\theta}_t], \tag{2}$$

where Δn_t, Δf_t, $\Delta a_{p,t}$, $\Delta a_{e,t}$ denote the adjustment of spindle speed, feed rate, cutting depth, and radial depth of cut, and $\Delta\boldsymbol{\theta}_t$ represents the adjustment of tool orientation. After receiving the action, the digital twin updates the machining parameters as

$$\mathbf{p}_{t+1} = \mathbf{p}_t + \mathbf{a}_t \tag{3}$$

The process response module calculates the machining outputs under the updated parameters. The material removal rate is estimated by

$$MRR_t = f_t a_{p,t} a_{e,t} \tag{4}$$

where f_t is the feed rate, $a_{p,t}$ is the axial depth of cut, and $a_{e,t}$ is the radial depth of cut. The spindle power is evaluated as

$$P_t = \frac{F_{c,t} v_{c,t}}{\eta} \tag{5}$$

where $F_{c,t}$ is the cutting force, $v_{c,t}$ is the cutting speed, and η is the transmission efficiency. Surface quality is represented by the predicted surface roughness:

$$R_{a,t} = \Phi_R(f_t, n_t, \mathbf{x}_t, \mathbf{q}_t) \tag{6}$$

where $\Phi_R(\cdot)$ denotes the data-driven or empirical surface roughness prediction model.

The constraint evaluation module checks whether the current state-action pair satisfies machining feasibility. The general constraint vector is written as

$$\mathbf{g}(s_t, \mathbf{a}_t) = [g_{\text{chatter}}, g_{\text{power}}, g_{\text{wear}}, g_{\text{collision}}, g_{\text{actuator}}] \tag{7}$$

where each component corresponds to chatter stability, spindle power limit, tool wear threshold, collision margin, and actuator saturation, respectively. A machining state is regarded as feasible only when

$$g_k(s_t, \mathbf{a}_t) \leq 0, k = 1, 2, \ldots, K \tag{8}$$

The learning interface module transforms the updated machining responses into observations, rewards, and constraint signals for reinforcement learning. For each agent, only task-related local information is provided during execution, while the global state and joint action are available to the centralized critics during training. This design makes the digital twin environment consistent with the centralized-training-decentralized-execution framework adopted in the proposed MOMADDPG method.

Through this architecture, the digital twin environment provides a unified platform for process simulation, constraint evaluation, and policy learning. It enables the reinforcement learning agents to optimize machining quality, efficiency, and safety under realistic physical limitations.

3.2. Machining Response Modeling

The machining response module is used to describe the dynamic relationship between process parameters, tool-workpiece engagement, and machining performance. In five-axis CNC machining, the cutting process is affected by nonlinear physical factors, such as cutting force variation, structural vibration, thermal deformation, and tool wear. Therefore, the digital twin environment should not only update the geometric tool path but also evaluate the physical responses caused by different control actions.

At time step t, the machining response vector is defined as

$$\mathbf{q}_t = [F_{c,t}, V_{b,t}, P_t, R_{a,t}, \chi_t, T_t] \tag{9}$$

where $F_{c,t}$ denotes the cutting force, $V_{b,t}$ is the flank wear value, P_t is the spindle power, $R_{a,t}$ represents the predicted surface roughness, χ_t denotes the chatter indicator, and T_t is the cutting temperature. These response variables are updated according to the current machining state and the joint action generated by the agents:

$$\mathbf{q}_{t+1} = \Phi_q(s_t, \mathbf{a}_t) \tag{10}$$

where $\Phi_q(\cdot)$ denotes the machining response model in the digital twin environment.

The cutting force is one of the key variables for evaluating machining load and stability. It is estimated according to the instantaneous cutting engagement and process parameters:

$$F_{c,t} = \Phi_F(n_t, f_t, a_{p,t}, a_{e,t}, \boldsymbol{\theta}_t, \mathbf{x}_t) \tag{11}$$

where n_t is the spindle speed, f_t is the feed rate, $a_{p,t}$ and $a_{e,t}$ are the axial and radial depths of cut, $\boldsymbol{\theta}_t$ denotes the tool orientation, and $\mathbf{x}_t$ represents the current tool position. In this study, $\Phi_F(\cdot)$ represents a mechanistic or data-calibrated cutting force sub-model. Its inputs include the controllable cutting parameters, tool orientation, and tool position, so that the influence of five-axis posture variation and local engagement changes can be reflected in the simulated cutting load. The specific parameter identification and validation procedure for this sub-model is described in Section 5.

Based on the cutting force, the spindle power is calculated as

$$P_t = \frac{F_{c,t} v_{c,t}}{\eta} \tag{12}$$

where $v_{c,t}$ is the cutting speed and η is the transmission efficiency of the spindle system. The material removal rate is expressed as

$$MRR_t = f_t a_{p,t} a_{e,t} \tag{13}$$

which is used as the main indicator of machining efficiency.

Surface quality is represented by the predicted surface roughness. Considering that surface generation in five-axis machining is affected by feed rate, spindle speed, tool posture, vibration, and local geometric features, the roughness model is formulated as

$$R_{a,t} = \Phi_R(f_t, n_t, \boldsymbol{\theta}_t, \mathbf{x}_t, \chi_t) \tag{14}$$

where $\Phi_R(\cdot)$ denotes the roughness prediction function. This formulation allows process parameters, tool posture, local tool position, and dynamic stability information to contribute to surface quality evaluation. In the present digital twin environment, $R_{a,t}$ is used as a quality-related simulation response for reward calculation and constraint-aware policy learning. Its dataset-specific validation is reported together with the digital twin validation procedure in Section 5 when applicable.

Tool wear is modeled as a cumulative degradation process. The wear state is updated by

$$V_{b,t+1} = V_{b,t} + \Delta V_{b,t} \tag{15}$$

$$\Delta V_{b,t} = \Phi_W(F_{c,t}, v_{c,t}, T_t, \Delta t) \tag{16}$$

where $\Delta V_{b,t}$ is the wear increment during the current time step, $\Phi_W(\cdot)$ is the tool wear evolution model, and Δt is the sampling interval. The cumulative formulation reflects the fact that tool degradation depends not only on the current cutting condition but also on the historical loading process. The parameters of $\Phi_W(\cdot)$ can be identified from tool-wear degradation data, and the dataset-specific calibration and validation procedure is described in Section 5.

Chatter tendency is evaluated using a stability indicator:

$$\chi_t = \Phi_C(n_t, a_{p,t}, F_{c,t}, \mathbf{q}_{t-1}) \tag{17}$$

where $\Phi_C(\cdot)$ represents the chatter evaluation model. A larger value of χ_t indicates a higher risk of unstable cutting. In this study, chatter is modeled as a stability-risk indicator derived

from cutting force, spindle speed, cutting depth, and previous response states. It is used to evaluate relative stability tendency in the simulation environment, while direct physical chatter validation is left for future CNC experiments. To clarify the physical or empirical basis of the response models, Table 2 summarizes the main inputs, model basis, and role of each response variable in the digital twin environment. The purpose of this table is to distinguish physics-based calculations, empirical response models, cumulative degradation models, and geometry-based safety evaluations. The dataset-specific calibration and validation procedure is reported in Section 5.

Table 2. Basis of machining response models in the digital twin environment.

Response Variable	Main Inputs	Model Basis	Role in This Study
Cutting force $F_{c,t}$	Spindle speed, feed rate, cutting depth, tool orientation, tool position	Mechanistic or data-calibrated cutting force model	Load estimation and power calculation
Tool wear $V_{b,t}$	Cutting force, cutting speed, temperature, time interval	Cumulative degradation model	Tool reliability and wear constraint evaluation
Spindle power P_t	Cutting force and cutting speed	Physics-based calculation from cutting power relation	Safety constraint evaluation
Material removal rate MRR_t	Feed rate, axial depth of cut, radial depth of cut	Geometric machining relation	Efficiency objective
Surface roughness $R_{a,t}$	Feed rate, spindle speed, tool orientation, tool position, chatter indicator	Physics-informed empirical response model	Quality-related simulation response
Chatter indicator χ_t	Spindle speed, cutting depth, cutting force, previous response state	Stability-risk indicator	Quality and safety-related simulation response
Collision clearance d_t	Tool posture, tool position, workpiece and machine geometry	Geometric clearance evaluation	Safety constraint

The above response models provide the physical basis for evaluating quality, efficiency, and safety objectives. Specifically, $R_{a,t}$ and χ_t are mainly related to machining quality, MRR_t reflects production efficiency, while P_t, $V_{b,t}$, and chatter risk are associated with operational safety. Therefore, the machining response module serves as the intermediate bridge between agent actions and multi-objective performance feedback.

Through this modeling strategy, the digital twin environment can map control actions into physically meaningful machining responses. This enables the reinforcement learning agents to evaluate different process strategies under realistic cutting conditions before being transferred to actual five-axis CNC machining.

3.3. Physical Constraint Formulation

In five-axis CNC machining, process optimization should not only improve machining quality and efficiency but also satisfy strict physical feasibility requirements. Excessive cutting load, unstable vibration, severe tool wear, actuator saturation, and insufficient collision clearance may lead to poor surface quality, tool failure, or even machine damage. Therefore, the digital twin environment explicitly formulates these machining limitations as physical constraints and provides constraint violation signals for reinforcement learning.

For a given state-action pair $(s_t, \mathbf{a}_t)$, the physical constraint vector is defined as

$$\mathbf{g}(s_t, \mathbf{a}_t) = \left[g_{\text{chatter}}, g_{\text{power}}, g_{\text{wear}}, g_{\text{actuator}}, g_{\text{collision}}\right] \tag{18}$$

where each component represents one type of machining feasibility constraint. The state-action pair is regarded as feasible only when

$$g_k(s_t, \mathbf{a}_t) \leq 0, k = 1, 2, \ldots, K \tag{19}$$

where K is the number of physical constraints. A positive value of g_k indicates that the corresponding constraint is violated.

First, chatter stability is considered as a key safety constraint because regenerative vibration can significantly deteriorate surface quality and reduce tool life. Based on the chatter indicator χ_t defined in Section 3.2, the chatter constraint is formulated as

$$g_{\text{chatter}}(s_t, \mathbf{a}_t) = \chi_t - \chi_{\max} \tag{20}$$

where $\chi_{\max}$ denotes the maximum allowable chatter indicator. The cutting state is considered stable when $g_{\text{chatter}} \leq 0$. This constraint prevents the policy from selecting aggressive cutting parameters that may cause unstable machining.

Second, spindle power should remain within the rated capacity of the machine tool. The power constraint is given by

$$g_{\text{power}}(s_t, \mathbf{a}_t) = P_t - P_{\max} \tag{21}$$

where P_t is the spindle power estimated by the digital twin, and $P_{\max}$ is the allowable power limit. This constraint avoids excessive cutting load and protects the spindle system from overload.

Third, tool wear is constrained to ensure machining reliability and dimensional accuracy. Since tool wear evolves cumulatively during machining, the wear constraint is expressed as

$$g_{\text{wear}}(s_t, \mathbf{a}_t) = V_{b,t} - V_{b,\max} \tag{22}$$

where $V_{b,t}$ is the current flank wear value, and $V_{b,\max}$ is the maximum allowable wear threshold. When this constraint is violated, the tool is considered unsuitable for continuing high-precision machining.

Fourth, the generated control action should satisfy actuator and process parameter limitations. Let the updated process parameter vector be

$$\mathbf{p}_{t+1} = [n_{t+1}, f_{t+1}, a_{p,t+1}, a_{e,t+1}, \boldsymbol{\theta}_{t+1}] \tag{23}$$

where n, f, a_p, a_e, and $\boldsymbol{\theta}$ denote spindle speed, feed rate, axial depth of cut, radial depth of cut, and tool orientation, respectively. The actuator constraint is formulated as

$$g_{\text{actuator}}(s_t, \mathbf{a}_t) = \mathbf{max}_j \left\{ \max \left(\mathbf{p}_{t+1,j} - \mathbf{p}_{\max,j}, \mathbf{p}_{\min,j} - \mathbf{p}_{t+1,j} \right) \right\} \tag{24}$$

where $\mathbf{p}_{\min}$ and $\mathbf{p}_{\max}$ are the lower and upper bounds of the allowable machining parameters, and j indexes the machining parameter dimension. This constraint ensures that the actions generated by the agents remain within the physical operating range of the CNC system.

Fifth, collision avoidance is essential in five-axis machining because the tool orientation changes continuously and may cause interference between the tool, holder, workpiece, and machine components. The collision constraint is defined according to the minimum clearance distance:

$$g_{\text{collision}}(s_t, \mathbf{a}_t) = d_{\min} - d_t \tag{25}$$

where d_t is the minimum distance between potential collision objects, and $d_{\min}$ is the required safety clearance. The machining state is considered collision-free only when $d_t \geq d_{\min}$, namely $g_{\text{collision}} \leq 0$.

To provide a unified constraint signal for reinforcement learning, the total violation degree is defined as

$$G(s_t, \mathbf{a}_t) = \sum_{k=1}^{K} \max(0, g_k(s_t, \mathbf{a}_t)) \tag{26}$$

where the hinge function $\max(0, g_k)$ penalizes only violated constraints. If $G(s_t, \mathbf{a}_t) = 0$, the current state-action pair satisfies all physical feasibility requirements. Otherwise, a larger value of G indicates a more severe violation.

The feasible machining domain can therefore be written as

$$\Omega_f = \{(s_t, \mathbf{a}_t) \mid g_k(s_t, \mathbf{a}_t) \leq 0, \forall k = 1, 2, \ldots, K\} \tag{27}$$

Only trajectories that remain within Ω_f are considered feasible candidates for Pareto archive updating. This formulation allows the digital twin environment to distinguish between high-performance but unsafe actions and truly feasible machining strategies.

By explicitly modeling physical constraints, the digital twin environment provides safety-aware feedback for the proposed reinforcement learning framework. These constraint functions are further incorporated into the Lagrangian actor objective in Section 4.3, enabling the policy to search for Pareto-oriented solutions within the feasible machining region rather than relying solely on implicit reward penalties.

3.4. State, Action, and Reward Design

To connect the digital twin environment with the proposed reinforcement learning framework, the machining process is formulated through state, action, reward, and constraint signals. At each time step t, the global state is defined as

$$s_t = [\mathbf{x}_t, \mathbf{p}_t, \mathbf{q}_t, \mathbf{h}_t] \tag{28}$$

where $\mathbf{x}_t$ denotes the tool position and posture information, $\mathbf{p}_t$ represents the process parameters, $\mathbf{q}_t$ contains machining responses, and $\mathbf{h}_t$ records historical information such as tool wear and previous stability states. Specifically

$$\mathbf{p}_t = [n_t, f_t, a_{p,t}, a_{e,t}, \boldsymbol{\theta}_t] \tag{29}$$

where n_t, f_t, $a_{p,t}$, $a_{e,t}$, and $\boldsymbol{\theta}_t$ denote spindle speed, feed rate, axial depth of cut, radial depth of cut, and tool orientation, respectively.

Since the proposed method adopts a multi-agent structure, the global state is decomposed into local observations:

$$o_{i,t} = \mathcal{O}_i(s_t), i \in \{Q, E, S\} \tag{30}$$

where Q, E, and S represent the quality-oriented, efficiency-oriented, and safety-oriented agents. The quality agent focuses on surface-related variables, the efficiency agent observes productivity-related variables, and the safety agent receives stability and constraint-related information.

The objective-specific action proposals generated by the three agents are expressed as

$$\mathbf{a}_t^{joint} = [a_{Q,t}, a_{E,t}, a_{S,t}]. \tag{31}$$

The three action proposals are fused into the executable machining parameter adjustment

$$\Delta \mathbf{p}_t = [\Delta n_t, \Delta f_t, \Delta a_{p,t}, \Delta a_{e,t}, \Delta \boldsymbol{\theta}_t]. \tag{32}$$

The detailed action aggregation mechanism is presented in Section 4.1. The process parameters are then updated by

$$\mathbf{p}_{t+1} = \mathrm{clip}(\mathbf{p}_t + \Delta \mathbf{p}_t, \mathbf{p}_{\min}, \mathbf{p}_{\max}) \tag{33}$$

where $\mathbf{p}_{\min}$ and $\mathbf{p}_{\max}$ are the allowable parameter bounds.

To preserve the multi-objective nature of the machining task, the environment returns a vector-valued reward:

$$\mathbf{r}_t = [r_{Q,t}, r_{E,t}, r_{S,t}] \tag{34}$$

where $r_{Q,t}$, $r_{E,t}$, and $r_{S,t}$ correspond to machining quality, efficiency, and safety. They are defined as

$$r_{Q,t} = -\alpha_R \tilde{R}_{a,t} - \alpha_\chi^Q \tilde{\chi}_t \tag{35}$$

$$r_{E,t} = \alpha_M \widetilde{MRR}_t - \alpha_T \tilde{T}_{m,t} \tag{36}$$

$$r_{S,t} = -\alpha_F \tilde{F}_{c,t} - \alpha_P \tilde{P}_t - \alpha_W \tilde{V}_{b,t} - \alpha_\chi^S \tilde{\chi}_t \tag{37}$$

where $\tilde{R}_{a,t}$, $\tilde{\chi}_t$, $\widetilde{MRR}_t$, $\tilde{T}_{m,t}$, $\tilde{F}_{c,t}$, $\tilde{P}_t$ and $\tilde{V}_{b,t}$ are normalized variables. The quality reward penalizes surface roughness and chatter tendency, the efficiency reward encourages higher material removal rate and shorter machining time, and the safety reward penalizes excessive cutting force, spindle power, tool wear, and stability risk.

The weighting coefficients in Equations (35)–(37) are used to balance the normalized reward components rather than to replace the multi-objective formulation. Their roles are summarized in Table 3. The selected coefficients keep the normalized terms at comparable magnitudes and reflect the engineering priority that safety-related penalties should not be ignored during efficiency improvement. The preference-conditioned learning and Pareto archive mechanism in Section 4 further reduce the dependence on a single fixed scalar reward.

Table 3. Interpretation of reward components.

Reward Component	Variable	Machining Meaning	Optimization Direction
Quality reward $r_{Q,t}$	$\tilde{R}_{a,t}$	Surface roughness	Minimize
Quality reward $r_{Q,t}$	$\tilde{\chi}_t$	Chatter tendency affecting surface quality	Minimize
Efficiency reward $r_{E,t}$	$\widetilde{MRR}_t$	Material removal efficiency	Maximize
Efficiency reward $r_{E,t}$	$\tilde{T}_{m,t}$	Machining time or ineffective cutting time	Minimize
Safety reward $r_{S,t}$	$\tilde{F}_{c,t}$	Cutting load	Minimize
Safety reward $r_{S,t}$	$\tilde{P}_t$	Spindle power consumption and overload risk	Minimize
Safety reward $r_{S,t}$	$\tilde{V}_{b,t}$	Tool wear state	Minimize
Safety reward $r_{S,t}$	$\tilde{\chi}_t$	Stability risk	Minimize

The normalization is performed as

$$\tilde{z}_t = \frac{z_t - z_{\min}}{z_{\max} - z_{\min} + \epsilon} \tag{38}$$

where z_t denotes a response variable, $z_{\min}$ and $z_{\max}$ are determined from feasible machining bounds, PHM-calibrated response ranges, and preliminary simulation runs, and ϵ is a small

constant for numerical stability. This normalization prevents variables with different physical units from dominating the reward calculation. The weighting coefficients in the reward functions are used to balance the normalized reward components rather than to replace the multi-objective formulation. Their roles are summarized in Table 3. These coefficients keep different normalized terms at comparable magnitudes and reflect the engineering requirement that efficiency improvement should not be achieved by sacrificing surface quality or machining safety.

In addition to the reward vector, the digital twin environment outputs the physical constraint vector:

$$\mathbf{g}(s_t, \mathbf{a}_t) = [g_{\text{chatter}}, g_{\text{power}}, g_{\text{wear}}, g_{\text{actuator}}, g_{\text{collision}}] \tag{39}$$

These constraints are not directly merged into the reward. Instead, they are separately provided to the learning algorithm and incorporated through the constrained learning mechanism in Section 4.3. This design separates performance evaluation from feasibility enforcement and avoids treating hard machining constraints only as soft reward penalties.

Therefore, the interaction between the agents and the digital twin environment can be summarized as

$$(s_t, \{o_{i,t}\}, \mathbf{a}_t) \rightarrow (s_{t+1}, \{o_{i,t+1}\}, \mathbf{r}_t, \mathbf{g}_t) \tag{40}$$

Through this formulation, the digital twin environment provides a compact and physically meaningful interface for multi-agent reinforcement learning, enabling the agents to optimize machining quality, efficiency, and safety under explicit physical constraints.

4. Physics-Constrained Heterogeneous Multi-Agent Pareto Learning Framework

Five-axis CNC machining is inherently a safety-critical and strongly coupled process, in which machining quality, material removal efficiency, and operational safety are simultaneously influenced by the same control variables, including spindle speed, feed rate, and axial depth of cut. Increasing the feed rate or depth of cut may improve production efficiency, but it also tends to intensify cutting forces, accelerate tool wear, raise spindle power consumption, and trigger regenerative chatter, thereby deteriorating surface integrity and process stability. This intrinsic physical conflict implies that no single control policy can optimize all objectives at once. Instead, the controller should identify a set of feasible and non-dominated operating policies distributed along the Pareto frontier.

Conventional reinforcement learning methods usually rely on scalarized reward functions to combine multiple objectives into a single optimization target. Although this strategy simplifies policy learning, it inevitably conceals nonlinear trade-offs among objectives and restricts the search process to a limited set of compromise solutions determined by manually assigned weights. Such a formulation is particularly inadequate for five-axis machining, where the objective conflicts are highly state-dependent and constrained by strict physical safety boundaries. Moreover, because the cutting process is only partially observable and dynamically non-stationary, a single-agent learning paradigm is often insufficient to capture the distributed interactions among heterogeneous objectives.

To address these challenges, this study proposes a Physics-Constrained Multi-Objective Multi-Agent Deep Deterministic Policy Gradient framework (MOMADDPG). The proposed method models five-axis machining as a heterogeneous multi-agent game in which three specialized agents are assigned to quality preservation, machining efficiency improvement, and safety assurance, respectively. Each agent learns its own control preference under decentralized execution, while centralized training enables the joint evaluation of physically coupled behaviors. To further improve feasibility and frontier coverage, two additional mechanisms are introduced.

First, a Physics-Constrained Lagrangian Optimization strategy is integrated into the learning framework to explicitly enforce safety-related constraints, such as chatter, spindle power, and tool wear limits, through adaptive Lagrange multipliers. Second, an Archive-Guided Preference Curriculum is designed to adaptively sample objective preferences according to the current distribution of non-dominated solutions, thereby promoting sparse-frontier exploration and staged trade-off learning.

Through the integration of heterogeneous preference decomposition, hierarchical state perception, multi-dimensional value estimation, explicit constraint handling, action aggregation, and adaptive Pareto exploration, the proposed framework provides a unified learning method for feasible Pareto-oriented parameter optimization in the digital twin-based five-axis machining environment.

4.1. Problem Formulation and Agent Modeling

The multi-objective optimization problem in five-axis CNC machining is formulated as a physics-constrained partially observable Markov game. This formulation is adopted because the machining process involves strongly coupled objectives, continuous control variables, and strict physical constraints. In particular, machining quality, production efficiency, and operational safety are affected simultaneously by spindle speed, feed rate, cutting depth, tool orientation, and the instantaneous tool-workpiece engagement state.

The proposed framework consists of three heterogeneous agents:

$$\mathcal{N} = \{Q, E, S\} \tag{41}$$

where Q, E, and S denote the Quality Agent, Efficiency Agent, and Safety Agent, respectively. The Quality Agent focuses on surface integrity and chatter suppression, the Efficiency Agent aims to improve material removal efficiency, and the Safety Agent is responsible for avoiding excessive tool wear, spindle overload, and collision-related risks. Each agent outputs a local continuous action proposal reflecting its objective-specific control preference. Since the CNC system can execute only one machining parameter adjustment at each control step, the three local action proposals are fused into one final five-dimensional executable action. Specifically, each agent outputs a five-dimensional action proposal.

$$a_{i,t} = \left[\Delta n_{i,t}, \Delta f_{i,t}, \Delta a_{p,i,t}, \Delta a_{e,i,t}, \Delta \theta_{i,t}\right], i \in \{Q, E, S\}, \tag{42}$$

where $a_{i,t}$ denotes the action proposal of $agent_i$, and $\Delta n_{i,t}$, $\Delta f_{i,t}$, $\Delta a_{p,i,t}$, $\Delta a_{e,i,t}$, and $\Delta \theta_{i,t}$ represent the proposed adjustments of spindle speed, feed rate, axial depth of cut, radial depth of cut, and tool orientation, respectively.

The final executable machining parameter adjustment is obtained by preference-weighted aggregation:

$$\Delta p_t = \omega_{Q,t} a_{Q,t} + \omega_{E,t} a_{E,t} + \omega_{S,t} a_{S,t}, \tag{43}$$

where $\omega_{Q,t}$, $\omega_{E,t}$, and $\omega_{S,t}$ are the preference weights corresponding to quality, efficiency, and safety. They satisfy

$$\omega_{Q,t} \geq 0, \omega_{E,t} \geq 0, \omega_{S,t} \geq 0, \omega_{Q,t} + \omega_{E,t} + \omega_{S,t} = 1, \tag{44}$$

The aggregated action is then applied to update the machining parameters within the feasible bounds:

$$p_{t+1} = \text{clip}(p_t + \Delta p_t, p_{min}, p_{max}), \tag{45}$$

where p_{min} and p_{max} denote the lower and upper bounds of the feasible machining parameter vector. This aggregation mechanism explicitly maps the three objective-specific

control proposals into one executable CNC parameter update. When the preference vector emphasizes quality, the final action is more influenced by the Quality Agent; when efficiency is emphasized, the Efficiency Agent contributes more strongly; and when safety risk increases, the Safety Agent can dominate the final action through larger safety-oriented preference weights and Lagrangian penalties.

The machining decision process is defined as

$$\mathcal{G} = \langle \mathcal{N}, \mathcal{S}, \{\mathcal{O}_i\}_{i\in\mathcal{N}}, \{\mathcal{A}_i\}_{i\in\mathcal{N}}, \mathcal{T}, \{r_i\}_{i\in\mathcal{N}}, \mathcal{C} \rangle \tag{46}$$

where $\mathcal{S}$ is the global machining state, $\mathcal{O}_i$ is the local observation of agent i, $\mathcal{A}_i$ is its continuous action space, $\mathcal{T}$ represents the state-transition function of the digital twin environment, r_i is the objective-specific reward, and $\mathcal{C}$ denotes the set of physical constraints. At time step t, each agent receives a local observation $o_{i,t}$ and generates a continuous action $a_{i,t}$. The joint action is expressed as

$$\mathbf{a}_t^{joint} = [a_{Q,t}, a_{E,t}, a_{S,t}], \tag{47}$$

which is fused into the final executable machining parameter adjustment Δp_t. The updated action is then applied to the digital twin environment to update the machining state:

$$s_{t+1} \sim \mathcal{T}(s_t, \mathbf{a}_t) \tag{48}$$

The three agents are assigned different optimization preferences. The Quality Agent is rewarded for reducing surface roughness deviation and suppressing chatter. The Efficiency Agent is rewarded for increasing material removal rate and reducing ineffective cutting time. The Safety Agent is rewarded for limiting tool wear, spindle power, and collision risk. Therefore, the reward vector of the system can be written as

$$\mathbf{r}_t = \left[r_Q(s_t, \mathbf{a}_t), r_E(s_t, \mathbf{a}_t), r_S(s_t, \mathbf{a}_t) \right] \tag{49}$$

Instead of directly scalarizing these objectives into a single reward, MOMADDPG preserves their independent value information during critic learning. This design enables the policy to search for multiple trade-off solutions rather than converging to a single weight-dependent optimum.

To guarantee machining feasibility, the optimization process is further constrained by physical safety limits:

$$g_k(s_t, \mathbf{a}_t) \leq 0, k = 1, 2, \ldots, K \tag{50}$$

where $g_k(\cdot)$ represents the k-th constraint function, such as chatter stability, spindle power limit, tool wear threshold, actuator saturation, or collision margin. These constraints define the feasible operating region of the machining process.

Accordingly, the objective of the proposed framework is to learn a set of feasible Pareto-optimal policies:

$$\Pi^* = \{\pi^* \mid \pi^* \text{ is non-dominated and satisfies } g_k(s_t, \mathbf{a}_t) \leq 0\} \tag{51}$$

Based on the above Markov game formulation, the overall architecture of the proposed physics-constrained MOMADDPG framework is illustrated in Figure 2. The framework follows the centralized-training-decentralized-execution paradigm. During decentralized execution, each heterogeneous agent receives only its local observation and lightweight inter-agent messages, encodes the information through a hierarchical attention module, and generates its own local continuous action. During centralized training, the global state representation and joint

action are used by the dual critics to estimate vector-valued Q-functions for quality, efficiency, and safety objectives. The clipped vector Q-value is then combined with the sampled preference vector and physical constraint penalty to form the actor objective.

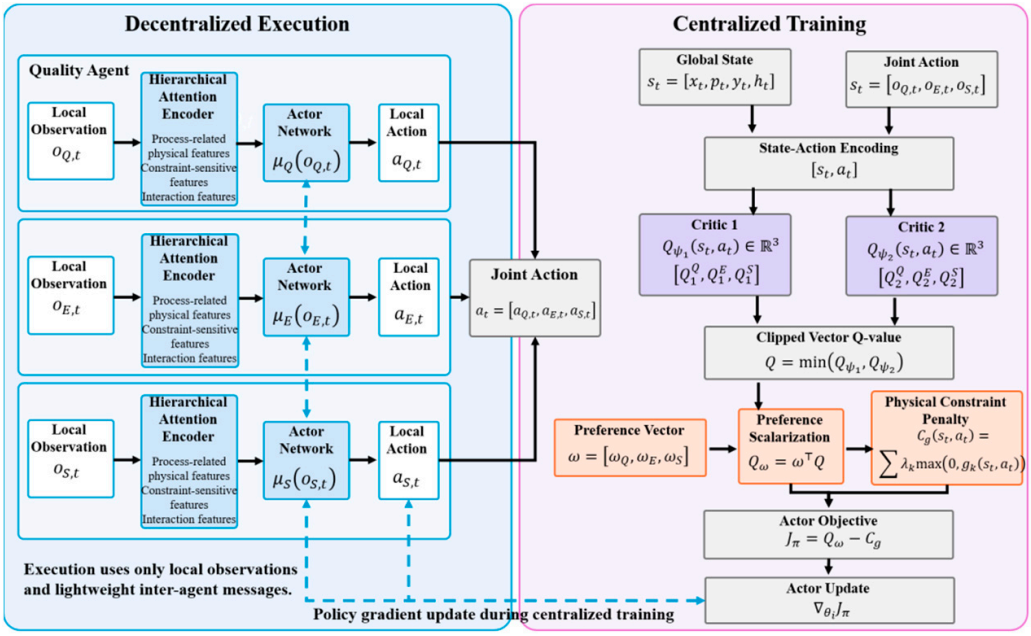


Figure 2. Overall architecture of the physics-constrained MOMADDPG framework under CTDE.

4.2. Hierarchical Attention Actor

As shown in the left part of Figure 2, each agent adopts a hierarchical attention-based Actor for decentralized execution. Although the centralized critics can access the global state during training, each Actor only uses its local observation and lightweight interaction information during execution. This design allows the learned policy to be deployed without requiring full global process information. In five-axis CNC machining, the state information perceived by each agent is highly heterogeneous and dynamically coupled. Local cutting engagement, tool orientation, vibration response, spindle load, and neighboring agents' control intentions may contribute differently to the final decision at different machining stages. Directly feeding all state variables into a fully connected policy network may weaken key transient features and reduce the sensitivity of the Actor to local instability or constraint-critical conditions. Therefore, a hierarchical attention Actor is designed to enhance state representation and support decentralized policy execution.

For each agent $i \in \{Q, E, S\}$, the local observation $o_{i,t}$ is first decomposed into three feature groups:

$$o_{i,t} = \left[x_{i,t}^p, x_{i,t}^c, x_{i,t}^n\right] \tag{52}$$

where $x_{i,t}^p$ denotes process-related physical features, such as spindle speed, feed rate, cutting force, power, and tool wear; $x_{i,t}^c$ denotes constraint-sensitive features, such as chatter index, collision margin, and actuator saturation; and $x_{i,t}^n$ denotes interaction features associated with the neighboring agents' actions or latent intentions. Each feature group is encoded through an independent nonlinear mapping:

$$h_{i,t}^m = \phi_m(x_{i,t}^m), m \in \{p, c, n\} \tag{53}$$

where $\phi_m(\cdot)$ is implemented by a lightweight multilayer perceptron. To adaptively distinguish the importance of different feature groups, an attention score is calculated as

$$\alpha_{i,t}^{m} = \frac{\exp\left(u_m^{\top}\tanh(W_m h_{i,t}^{m})\right)}{\sum_{l\in\{p,c,n\}}\exp\left(u_l^{\top}\tanh(W_l h_{i,t}^{l})\right)} \tag{54}$$

where W_m and u_m are trainable parameters. The final attention-enhanced state representation of agent i is obtained by weighted aggregation:

$$z_{i,t} = \sum_{m\in\{p,c,n\}} \alpha_{i,t}^{m} h_{i,t}^{m} \tag{55}$$

The representation $z_{i,t}$ is then passed to the deterministic policy network to generate the local continuous action:

$$a_{i,t} = \mu_i(z_{i,t};\theta_i) + \epsilon_t \tag{56}$$

where $\mu_i(\cdot)$ denotes the Actor network of agent i, θ_i represents its trainable parameters, and ϵ_t is the exploration noise. During training, the exploration noise follows a decaying Ornstein–Uhlenbeck process to encourage sufficient exploration in the early stage while improving policy smoothness during convergence.

The hierarchical attention structure provides two main advantages. First, it enables each agent to focus on the most relevant physical and constraint-sensitive features under different machining conditions. For example, when the process approaches a chatter boundary, the Quality Agent can assign larger attention weights to vibration-related features; when the spindle load approaches its rated limit, the Safety Agent can emphasize power and wear-related features. Second, the inclusion of interaction features allows each agent to implicitly consider the influence of other agents, thereby improving coordinated decision-making under decentralized execution.

Compared with a standard fully connected Actor, the proposed attention Actor improves feature selectivity without requiring global state information during execution. This property is consistent with the centralized-training-decentralized-execution paradigm and enhances the adaptability of MOMADDPG in strongly coupled and partially observable machining environments.

4.3. Constrained Dual-Critic Learning

To preserve objective-specific value information and improve policy feasibility, MOMADDPG adopts a constrained dual-critic learning mechanism. Unlike conventional scalarized reinforcement learning, which combines multiple objectives into a single reward before value estimation, the proposed critic directly estimates a multi-dimensional action-value vector. For each critic network, the value output is defined as

$$\mathbf{Q}_{\psi_j}(s_t,\mathbf{a}_t) = \left[Q_{\psi_j}^{Q}(s_t,\mathbf{a}_t), Q_{\psi_j}^{E}(s_t,\mathbf{a}_t), Q_{\psi_j}^{S}(s_t,\mathbf{a}_t)\right], j = 1,2 \tag{57}$$

where $Q_{\psi_j}^{Q}$, $Q_{\psi_j}^{E}$, and $Q_{\psi_j}^{S}$ correspond to the value estimates of quality, efficiency, and safety objectives, respectively. The use of two critics follows the clipped double-Q strategy to reduce overestimation bias.

The target action at the next state is generated by the target actors and the same action aggregation mechanism:

$$\Delta p_{t+1}^{-} = Agg\left(a_{Q,t+1}^{-}, a_{E,t+1}^{-}, a_{S,t+1}^{-}, \omega_t\right), \tag{58}$$

where $Agg(\cdot)$ denotes the action aggregation operation defined in Section 4.1. The target multi-objective value is computed as

$$y_t = r_t + \gamma \min_{j=1,2} Q_{\psi_j^-}\left(s_{t+1}, \Delta p_{t+1}^-\right), \tag{59}$$

where γ is the discount factor, ψ_j^- denotes the parameters of the target critic, and the minimum operation is applied element-wise to each objective dimension. The critic loss is then formulated as

$$\mathcal{L}_Q = \sum_{j=1}^{2} \mathbb{E}_{\mathcal{D}}\left[\| \mathbf{Q}_{\psi_j}(s_t, \mathbf{a}_t) - \mathbf{y}_t \|_2^2\right] \tag{60}$$

where $\mathcal{D}$ denotes the replay buffer. By maintaining a vector-valued critic, the proposed framework avoids premature scalarization and retains the gradient information of different objectives during policy optimization.

To support Pareto-oriented policy improvement, a preference vector is introduced:

$$\boldsymbol{\omega}_t = [\omega_Q, \omega_E, \omega_S], \omega_i \geq 0, \sum_i \omega_i = 1 \tag{61}$$

The preference-conditioned value used for actor updating is obtained as

$$Q_{\boldsymbol{\omega}}(s_t, \mathbf{a}_t) = \boldsymbol{\omega}_t^\top \mathbf{Q}_{\psi_1}(s_t, \mathbf{a}_t) \tag{62}$$

where $\boldsymbol{\omega}_t$ specifies the current trade-off preference among quality, efficiency, and safety. This design allows the actor to learn policies under different objective priorities while keeping the critic representation objective-decomposed.

In addition to multi-objective value learning, machining feasibility is explicitly considered through physics-constrained Lagrangian optimization. Let

$$\mathbf{g}(s_t, \mathbf{a}_t) = [g_1(s_t, \mathbf{a}_t), g_2(s_t, \mathbf{a}_t), \ldots, g_K(s_t, \mathbf{a}_t)] \tag{63}$$

represent the physical constraint functions, including chatter stability, spindle power, tool wear, actuator saturation, and collision margin. The constrained actor objective is formulated as

$$\mathcal{J}_\pi = \mathbb{E}_{\mathcal{D}}\left[Q_{\boldsymbol{\omega}}(s_t, \mathbf{a}_t) - \sum_{k=1}^{K} \lambda_k \max(0, g_k(s_t, \mathbf{a}_t))\right] \tag{64}$$

where $\lambda_k \geq 0$ is the adaptive Lagrange multiplier associated with the k-th constraint. The hinge term $\max(0, g_k)$ ensures that penalties are imposed only when the corresponding physical constraint is violated.

The actor parameters are updated by maximizing $\mathcal{J}_\pi$, while the Lagrange multipliers are updated according to the observed constraint violations:

$$\lambda_k \leftarrow [\lambda_k + \eta_\lambda \mathbb{E}_{\mathcal{D}}[\max(0, g_k(s_t, \mathbf{a}_t))]]_+ \tag{65}$$

where η_λ is the multiplier learning rate and $[\cdot]_+$ denotes projection onto the non-negative domain. When a constraint is frequently violated, its multiplier increases automatically, forcing the policy to reduce unsafe actions. Conversely, when the constraint is consistently satisfied, the corresponding penalty remains small, allowing the policy to focus more on Pareto performance improvement.

This constrained dual-critic mechanism provides two advantages. First, the vector-valued critic preserves the independent contribution of each objective and avoids the information loss caused by fixed scalar rewards. Second, the adaptive Lagrangian term transforms safety requirements from implicit reward penalties into explicit feasibility constraints. Therefore, the learned policy is guided toward feasible Pareto-oriented regions rather than merely high-reward but potentially unsafe operating points.

4.4. Archive-Guided Curriculum

To improve Pareto frontier coverage and avoid preference collapse, an archive-guided preference curriculum is introduced to adaptively regulate the objective preference during training. In multi-objective machining optimization, different preference vectors may lead the policy toward different trade-off regions. However, uniformly sampling preferences from a simplex often results in inefficient exploration, especially when the obtained Pareto solutions are unevenly distributed or when some frontier regions are difficult to reach due to physical constraints. Therefore, the preference distribution is adjusted according to the current Pareto archive.

An external archive A is maintained to store feasible non-dominated solutions discovered during training. For each sampled trajectory, the corresponding objective vector is defined as

$$\mathbf{F}(\pi) = \left[F_Q(\pi), F_E(\pi), F_S(\pi)\right] \tag{66}$$

where $F_Q(\pi)$, $F_E(\pi)$, and $F_S(\pi)$ denote the accumulated performance values related to machining quality, efficiency, and safety, respectively. A candidate solution is considered feasible only when its accumulated constraint violation is zero or below a predefined tolerance:

$$G(\pi) = \sum\nolimits_t \sum_{k=1}^{K} max(0, g_k(s_t, \Delta p_t)) \leq \delta_g, \tag{67}$$

where δ_g is a small feasibility tolerance. A feasible candidate solution is added to the archive only if it is not dominated by existing archive members:

$$\nexists F_a \in A \text{ such that } F_a \succ F(\pi), \tag{68}$$

where $F_a \succ F(\pi)$ means that the archived solution F_a dominates the candidate solution $F(\pi)$. After insertion, any archived solution dominated by the new candidate is removed:

$$A \leftarrow \{F_a \in A \mid F(\pi) \not\succ F_a\} \cup \{F(\pi)\}. \tag{69}$$

This update rule maintains a compact approximation of the feasible Pareto frontier.

To quantify the distribution of archived solutions, the crowding distance dc is used to identify sparse regions along the frontier. For an archived solution $\mathbf{F}_a$, its sparsity score is calculated as

$$\rho(\mathbf{F}_a) = \frac{d_c(\mathbf{F}_a)}{\sum_{\mathbf{F}_b \in \mathcal{A}} d_c(\mathbf{F}_b) + \epsilon} \tag{70}$$

where ϵ is a small constant to avoid numerical instability. A larger $\rho(\mathbf{F}_a)$ indicates that the corresponding region of the Pareto frontier is insufficiently explored and should receive more preference samples in subsequent training.

Based on the archive distribution, the preference vector $\boldsymbol{\omega}_t$ is sampled from an adaptive distribution:

$$\boldsymbol{\omega}_t \sim p_t(\boldsymbol{\omega}) \tag{71}$$

where $p_t(\boldsymbol{\omega})$ is updated according to both the training stage and the sparsity of the current archive. In the early stage, the curriculum assigns higher probability to safety-oriented preferences to guide the policy toward feasible operating regions. In the middle stage, preferences are gradually diversified to explore trade-offs between machining quality and efficiency. In the later stage, the sampling probability is increased around sparse Pareto regions identified by the archive, thereby improving frontier uniformity and coverage.

The adaptive sampling probability can be expressed as

$$p_t(\boldsymbol{\omega}) = (1 - \beta_t) p_{\text{base}}(\boldsymbol{\omega}) + \beta_t p_{\mathcal{A}}(\boldsymbol{\omega}) \tag{72}$$

where $p_{\text{base}}(\boldsymbol{\omega})$ is the base distribution, such as a Dirichlet distribution over the preference simplex, and $p_{\mathcal{A}}(\boldsymbol{\omega})$ is the archive-guided distribution biased toward under-explored trade-off regions. The coefficient $\beta_t \in [0,1]$ gradually increases with training progress, allowing the learning process to shift from broad exploration to archive-guided refinement.

This curriculum strategy provides two benefits. First, it reduces inefficient random preference exploration by using the Pareto archive as feedback. Second, it enhances sparse-frontier exploration and prevents the learned policies from concentrating only on a few easily reachable trade-off regions. As a result, the proposed method can obtain a more diverse and uniformly distributed set of feasible Pareto solutions for quality-efficiency-safety optimization in five-axis machining.

4.5. Training Procedure

The overall training procedure of the proposed MOMADDPG framework follows the centralized-training-decentralized-execution paradigm. During training, the critics have access to the global machining state, joint actions, objective-specific rewards, and physical constraint information. During execution, each agent only relies on its local observation and the hierarchical attention Actor to generate control commands. This design enables coordinated policy learning while preserving decentralized online control capability.

At each time step, each agent $i \in \{Q, E, S\}$ receives its local observation $o_{i,t}$ from the digital twin environment and generates an action according to

$$a_{i,t} = \mu_i(o_{i,t}; \theta_i) + \epsilon_t \tag{73}$$

where $\mu_i(\cdot)$ is the Actor network and ϵ_t denotes exploration noise. The joint action

$$a_t^{joint} = \left[a_{Q,t}, a_{E,t}, a_{S,t}\right]. \tag{74}$$

The action aggregation module then converts the joint action proposal into the executable machining parameter adjustment:

$$\Delta p_t = \text{Agg}\left(a_{Q,t}, a_{E,t}, a_{S,t}, \omega_t\right). \tag{75}$$

The executable action Δp_t is applied to the machining digital twin. The environment returns the next state s_{t+1}, local observations $\{o_{i,t+1}\}$, reward vector $\mathbf{r}_t$, and physical constraint values $\mathbf{g}_t$. The transition tuple is stored in the replay buffer:

$$\left(s_t, \{o_{i,t}\}, \mathbf{a}_t, \mathbf{r}_t, \mathbf{g}_t, s_{t+1}, \{o_{i,t+1}\}\right) \in \mathcal{D}. \tag{76}$$

For each training iteration, a mini-batch is sampled from D. The target joint action is generated by the target actors, and the vector-valued target $\mathbf{y}_t$ is computed using the clipped double-critic mechanism described in Section 4.3. The two critics are updated by minimizing the multi-objective temporal-difference loss:

$$\mathcal{L}_Q = \sum_{j=1}^{2} \mathbb{E}_{\mathcal{D}}\left[\| \mathbf{Q}_{\psi_j}(s_t, \mathbf{a}_t) - \mathbf{y}_t \|_2^2\right] \tag{77}$$

Then, a preference vector $\boldsymbol{\omega}_t$ is sampled according to the archive-guided curriculum introduced in Section 4.4. The Actors are updated by maximizing the preference-conditioned and constraint-penalized objective:

$$\mathcal{J}_\pi = \mathbb{E}_{\mathcal{D}}\left[\boldsymbol{\omega}_t^\top \mathbf{Q}_{\psi_1}(s_t, \mathbf{a}_t) - \sum_{k=1}^{K} \lambda_k \max(0, g_k(s_t, \mathbf{a}_t))\right] \tag{78}$$

where the first term guides policy improvement toward the current Pareto preference, and the second term penalizes violations of physical constraints. The Lagrange multipliers are adaptively updated according to the observed constraint violations:

$$\lambda_k \leftarrow [\lambda_k + \eta_\lambda \mathbb{E}_{\mathcal{D}}[\max(0, g_k(s_t, \mathbf{a}_t))]]_+ \tag{79}$$

After each episode, the accumulated objective vector of the generated trajectory is evaluated. If the trajectory satisfies all physical constraints and is not dominated by existing archive members, it is inserted into the Pareto archive. Dominated solutions are removed, and the archive distribution is used to update the preference sampling strategy for subsequent training.

Finally, the target networks are updated by soft replacement:

$$\psi_j^- \leftarrow \tau\psi_j + (1-\tau)\psi_j^- \tag{80}$$

$$\theta_i^- \leftarrow \tau\theta_i + (1-\tau)\theta_i^- \tag{81}$$

where $\tau \ll 1$ is the soft update coefficient. The training process continues until the Pareto archive becomes stable or the maximum number of episodes is reached.

The complete episode-level training loop is shown in Figure 3. At the beginning of each episode, a preference vector is sampled from the current preference distribution. The sampled preference guides policy rollout in the five-axis digital twin environment, and the collected transitions are stored in the replay buffer for off-policy learning. The MOMADDPG learner updates the dual critics, actors, and Lagrange multipliers using mini-batches sampled from the replay buffer. After each episode, the generated trajectory is evaluated at the trajectory level using the accumulated objective vector and constraint violation vector. Only feasible and non-dominated trajectories are inserted into the Pareto archive. The updated archive then reshapes the preference sampling distribution through the curriculum scheduler.

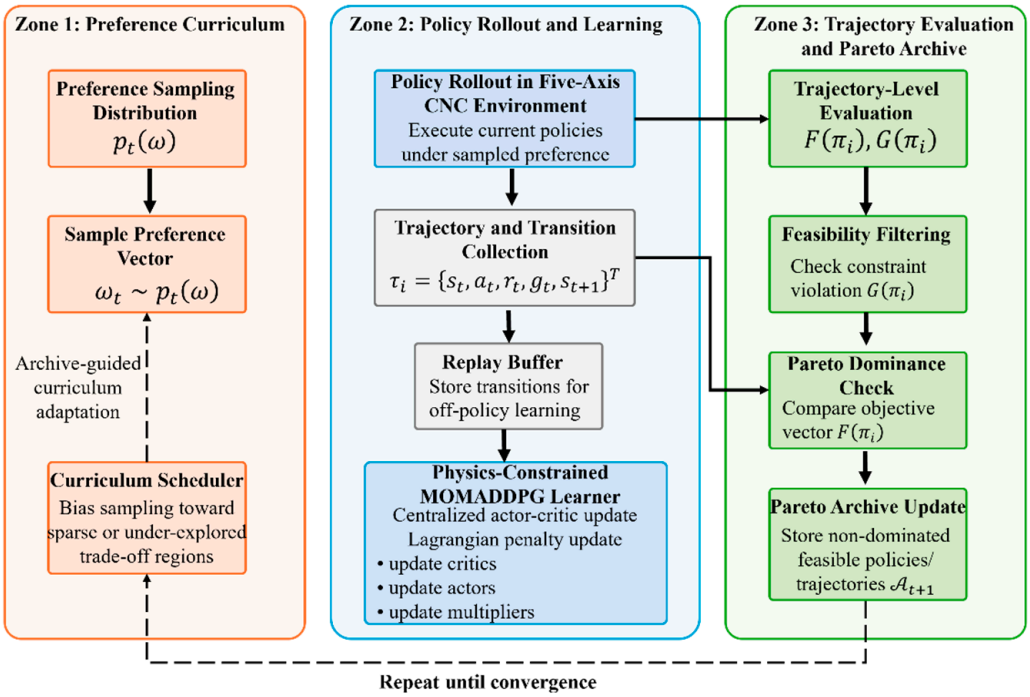


Figure 3. Pareto archive-guided curriculum training loop of the proposed framework.

Based on this loop, the complete training procedure is summarized in Algorithm 1.

Algorithm 1. Training procedure of the proposed MOMADDPG framework

```
Initialize Actor networks and target Actors for each agent i ∈ {Q, E, S}
Initialize dual Critics and target Critics
Initialize replay buffer D, Pareto archive A, and Lagrange multipliers
Initialize the preference sampling distribution
for episode = 1 to M do
    Reset the digital twin machining environment
    Obtain initial global state and local observations
    Sample preference vector from the current preference distribution
    for t = 1 to T do
      for each agent i ∈ {Q, E, S} do
          Generate local action proposal using the Actor network
      end for
      Aggregate the three action proposals into the executable machining action
      Execute the executable action in the digital twin environment
      Observe next state, local observations, reward vector, and constraint values
      Store transition in replay buffer D
      Sample a mini-batch from D
      Generate target action proposals using target Actors
      Aggregate target action proposals into target executable action
      Compute vector-valued target using clipped double Critics
      Update Critics by minimizing multi-objective temporal-difference loss
      Update Actors by maximizing the constrained preference-conditioned objective
      Update Lagrange multipliers according to constraint violations
      Soft update target Critics and target Actors
    end for
    Evaluate the accumulated objective vector and accumulated constraint violation
    if the trajectory is feasible and non-dominated then
      Insert it into Pareto archive A
      Remove dominated archive members
    end if
    Compute crowding-distance-based sparsity scores
    Update the preference sampling distribution using archive information
end for
Return trained Actors and feasible Pareto archive A
```

5. Simulation Experiments and Result Analysis

Five-axis CNC machining parameter optimization is a strongly coupled multi-objective decision-making problem. Machining quality, material removal efficiency, and operational safety are jointly affected by spindle speed, feed rate, axial depth of cut, radial depth of cut, and tool orientation. Increasing material removal efficiency may improve productivity, but it can also increase cutting force, chatter risk, spindle power, tool wear, and collision probability. Therefore, this problem cannot be sufficiently evaluated using a single scalar objective. A suitable control strategy should simultaneously maintain machining feasibility, improve process performance, and approximate a diverse Pareto frontier under physical constraints.

To evaluate the proposed physics-constrained MOMADDPG framework, a data-calibrated five-axis digital twin simulation environment was established. The experiments were designed to assess convergence behavior, multi-objective optimization capability,

Pareto-front quality, computational efficiency, and safety-related constraint performance. Four representative reinforcement learning algorithms, namely DQN, MADDPG, MAAC, and MAPPO, were selected as baselines. All algorithms were trained and tested under the same digital twin environment, hardware platform, and hyperparameter settings where applicable, ensuring the fairness and reproducibility of the comparison. Since all reported results are obtained in the data-calibrated digital twin simulation environment, the conclusions of this section should be interpreted as simulation-based validation rather than direct evidence of physical five-axis CNC machining performance.

5.1. Experimental Platform Configuration and Baseline Setup

The experimental environment was calibrated using the high-speed milling dataset from the PHM Society 2010 Data Challenge [31]. The dataset records the full-life degradation process of a three-flute carbide ball-end milling cutter during stainless steel machining. In this study, the PHM 2010 dataset was used to identify and calibrate key parameters in the cutting force and tool wear models rather than to serve as a complete five-axis machining dataset. The calibration procedure included signal preprocessing, model parameter identification, normalization range determination, and validation on held-out data. Specifically, the measured force and wear signals were filtered and aligned, the model parameters were fitted using the calibration subset, and the remaining runs were used to verify the consistency of the calibrated response models. The five-axis motion, tool orientation variation, and tool-workpiece engagement conditions were generated by the proposed digital twin environment. Therefore, the experiments were conducted in a data-calibrated five-axis simulation environment rather than on direct five-axis experimental measurements.

The digital twin integrates five-axis kinematic transformation, tool-workpiece engagement calculation, transient cutting force prediction, regenerative chatter evaluation, spindle power estimation, cumulative tool wear evolution, actuator-bound checking, and collision-risk evaluation. At each discrete control step, the environment outputs the global machining state, local observations for heterogeneous agents, vector-valued rewards, and physical constraint violation signals. Multi-source information from the PHM dataset, including cutting force, vibration, and acoustic emission signals, was used to support the calibration of the cutting force and tool wear models. The validation of the calibrated digital twin was carried out using quantitative error metrics such as MAE, RMSE, MAPE, and R^2. The corresponding measured-predicted comparisons are reported in Table 4, and the results show that the calibrated environment can provide physically meaningful feedback for reinforcement learning. Figure 4 illustrates the digital twin simulation environment and the closed-loop interaction between the virtual five-axis CNC machine and the MOMADDPG controller.

Table 4. Quantitative validation of the calibrated digital twin response models.

Response Variable	Data Source	MAE	RMSE	MAPE	R^2
Cutting force $F_{c,t}$	PHM 2010 force data	38.6 N	54.2 N	8.7%	0.91
Tool wear $V_{b,t}$	PHM 2010 wear data	0.011 mm	0.016 mm	9.4%	0.93
Spindle power P_t	Force-derived data	0.12 kW	0.18 kW	10.1%	—

The machining parameters were constrained within industrially feasible ranges. The spindle speed was set within 4000–12,000 rpm, the feed rate within 200–2000 mm/min, and the axial depth of cut within 0.3–1.5 mm. The radial depth of cut and tool orientation adjustment were also bounded according to the feasible operating domain of the simulated five-axis process. Each training episode corresponded to one complete tool-path segment containing approximately 500 control steps. All experiments were implemented using

PyTorch 2.0.1 on a workstation equipped with an Intel Core i9-13900K CPU and an NVIDIA RTX 4090 GPU under Ubuntu 22.04 LTS.

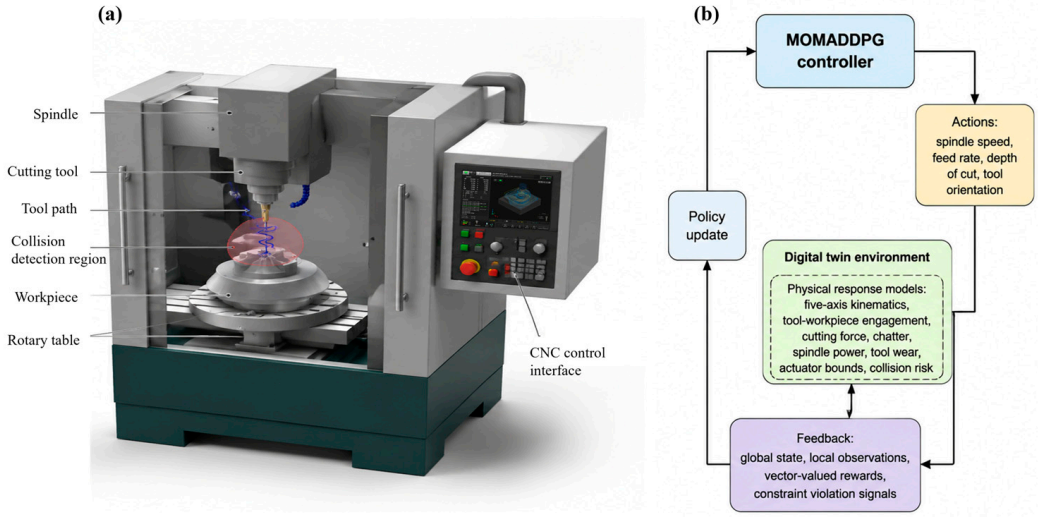


Figure 4. Digital twin–MOMADDPG interaction framework. (**a**) Virtual five-axis CNC machine model. (**b**) Closed-loop interaction between the digital twin environment and the MOMADDPG controller.

Four baseline algorithms were considered. DQN was used as a value-based baseline with discretized action spaces, where continuous machining parameters were divided into 121 joint action combinations. MADDPG was selected as a continuous-control multi-agent baseline under the centralized-training-decentralized-execution paradigm. MAAC was used to evaluate the influence of attention-based multi-agent critic learning. MAPPO represented an on-policy multi-agent reinforcement learning method and was used to compare sample efficiency and convergence stability. For a fair comparison in the multi-objective setting, the baseline algorithms adopted fixed-weight scalarization to aggregate quality, efficiency, and safety rewards. To partially address the fairness issue of scalarized comparison, each baseline was evaluated under multiple weight combinations, and the non-dominated solutions obtained from these runs were used to compute hypervolume.

In the proposed MOMADDPG framework, three heterogeneous agents were assigned to quality preservation, efficiency improvement, and safety assurance, respectively. Each agent adopted a hierarchical attention-based Actor followed by three fully connected layers with 256, 256, and 128 neurons. The centralized dual-Critic networks output three-dimensional Q-value vectors corresponding to quality, efficiency, and safety objectives. During actor updating, the vector-valued Q-function was scalarized using a sampled preference vector, while physical constraint violations were penalized through adaptive Lagrangian multipliers. Training was conducted for 3000 episodes. The standard deviation of the Ornstein–Uhlenbeck exploration noise was initialized as 0.3 and linearly decayed to 0.05. The Pareto archive capacity was set to 200, and a crowding-distance-based maintenance strategy was activated when the archive approached its capacity to preserve the diversity of feasible non-dominated solutions.

5.2. Multi-Objective Optimization Capability and Pareto Hypervolume Evaluation

Several metrics were used to evaluate policy performance. The success rate denotes the proportion of episodes in which the tool path was completed while satisfying the predefined task termination conditions. The collision rate which also represents as safety violation rate, denotes the proportion of episodes in which collision, actuator-limit, or other safety-boundary violations occurred at least once. The task reward component, denoted as

T, reflects the overall task-completion performance. The efficiency component, denoted as F, evaluates normalized material removal performance, while the safety component, denoted as S, measures the normalized safety-related performance associated with chatter, spindle power, tool wear, and collision risk. Since these indicators describe different aspects of the machining process, they are not strictly complementary.

After all algorithms were trained for 3000 episodes in the same digital twin environment, their convergence behavior and terminal multi-objective performance were evaluated. Figure 5 shows the evolution of task success rate and collision rate, where the shaded regions represent the standard deviation over five independent runs.

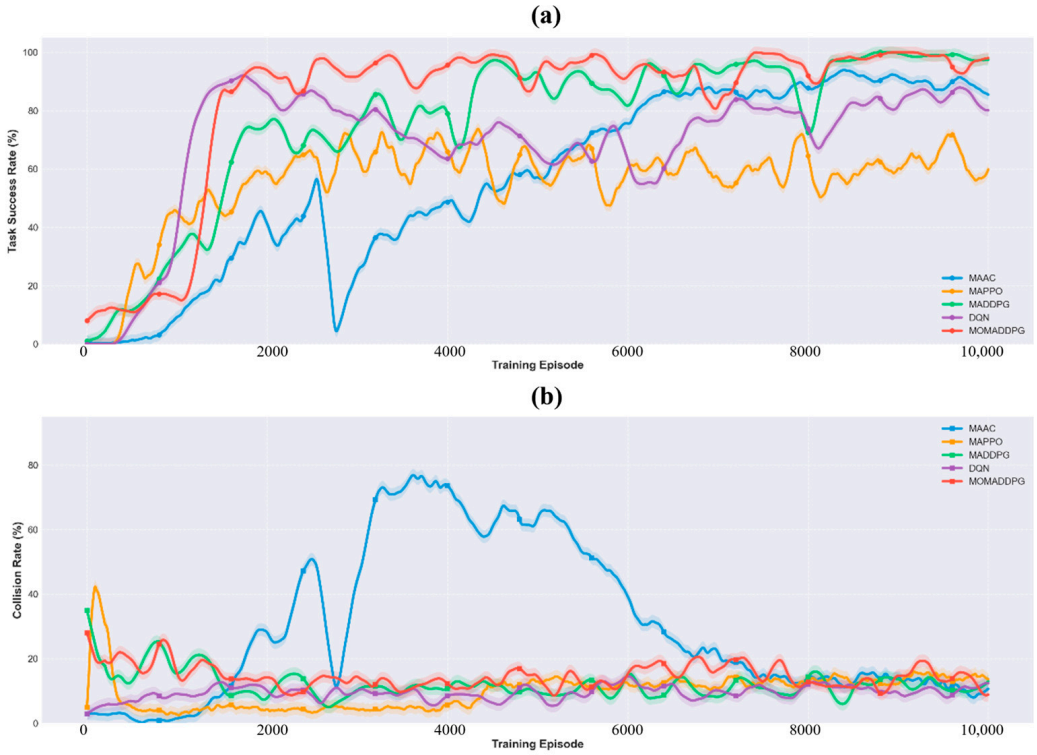


Figure 5. Comparison of training convergence across multiple algorithms. (**a**) Multi-Algorithm Training Convergence: Success Rate. (**b**) Multi-Algorithm Training Convergence: Collision Rate.

The learning process of MOMADDPG exhibits a clear staged pattern. During the early exploration stage, approximately before Episode 1300, the success rate remains low because the agents are still searching for feasible actions under coupled physical constraints. In this stage, the task reward is close to zero, while the efficiency component gradually improves as the policy begins to identify more productive cutting conditions. However, safety-boundary violations occur more frequently because stable constraint-aware behaviors have not yet been learned.

A significant improvement appears around Episodes 1400–1700. The success rate rises rapidly to above 90%, while the collision rate decreases substantially. This improvement indicates that the vector-valued dual-Critic mechanism and the Pareto archive-guided preference curriculum help improve the feasibility of the learned policy while preventing collapse toward a single objective. Importantly, this improvement is not achieved by sacrificing efficiency or safety, suggesting that MOMADDPG can maintain a more balanced quality–efficiency–safety trade-off during training. After approximately Episode 2000, the success rate stabilizes between 90% and 98%, and the collision rate remains within a controllable range. The evolution of actor loss, critic loss, and reward components is shown in Figure 6.

The baseline methods show different limitations. DQN benefits from discretized actions in some regular tool-path segments, but its limited action resolution restricts fine-grained control and Pareto-front exploration. MADDPG supports continuous actions but fixed scalarized rewards cause the loss of objective-specific value information during critic learning. MAAC improves inter-agent information utilization through attention-based critic learning, but it lacks the proposed hierarchical attention Actor and Pareto archive-guided curriculum. MAPPO shows stable policy updates, but its on-policy nature leads to lower sample efficiency under the same 3000-episode training budget.

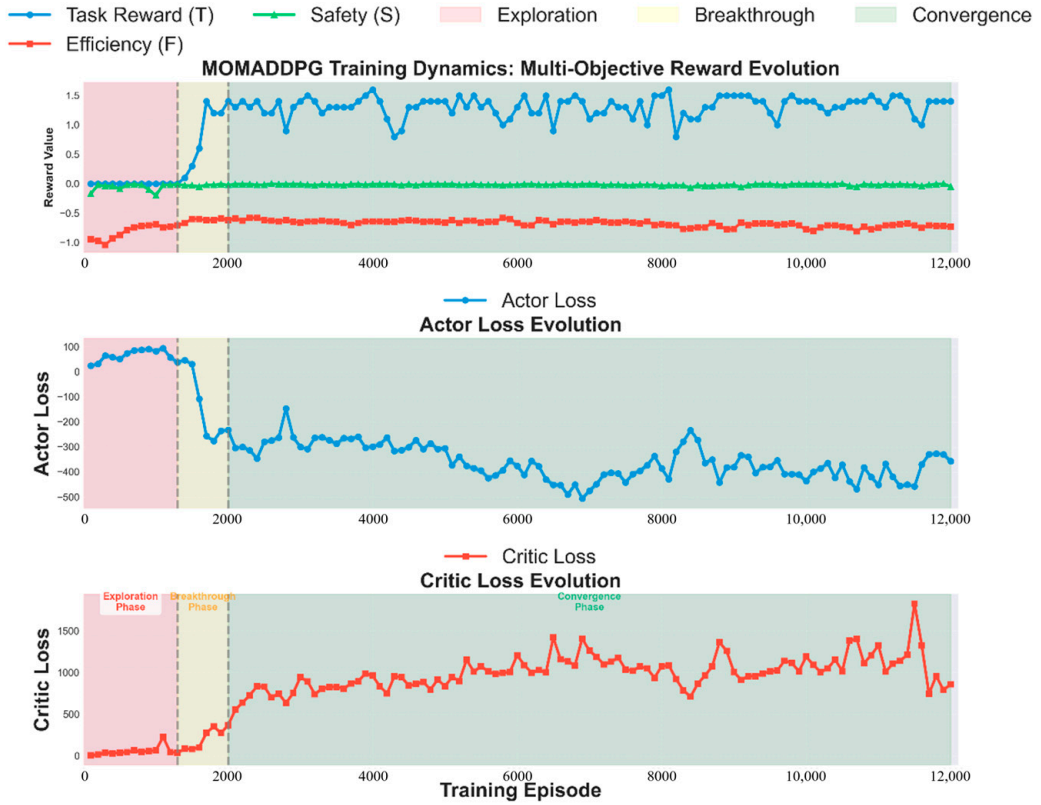


Figure 6. Training phase division and loss evolution of MOMADDPG.

To quantitatively evaluate Pareto-front approximation, the hypervolume indicator was adopted. The hypervolume was calculated in the normalized maximization objective space. The reference point was set slightly worse than the worst observed objective vector among all compared algorithms. For scalarized baseline methods, the non-dominated solution set obtained from independent runs with different weight combinations was used to calculate hypervolume. Figure 7 presents the terminal performance in the three-dimensional objective space using normalized task completion, efficiency, and safety indicators.

The final performance comparison is summarized in Table 5. The values in Table 5 are reported as the mean performance over five independent runs. MOMADDPG achieves the highest hypervolume value of 0.674, which is approximately 59% higher than MAAC, the best-performing baseline in terms of hypervolume. This result indicates that MOMADDPG obtains a broader and more uniformly distributed feasible non-dominated solution set. The improvement mainly comes from three aspects: the vector-valued dual Critics preserve independent value information for different objectives; the hierarchical attention Actor enhances feature extraction under partially observable machining conditions; and the Pareto archive-guided curriculum promotes exploration of sparse trade-off regions.

It should be noted that success rate and hypervolume evaluate different properties of policy quality. DQN obtains a relatively high success rate because discretized actions

can produce stable but conservative execution sequences in some cases. However, its low action resolution limits the diversity of quality–efficiency–safety trade-offs, resulting in a lower hypervolume. In contrast, MOMADDPG achieves both the highest success rate and the highest hypervolume, demonstrating its ability to maintain task feasibility while exploring a wider range of Pareto-optimal machining policies.

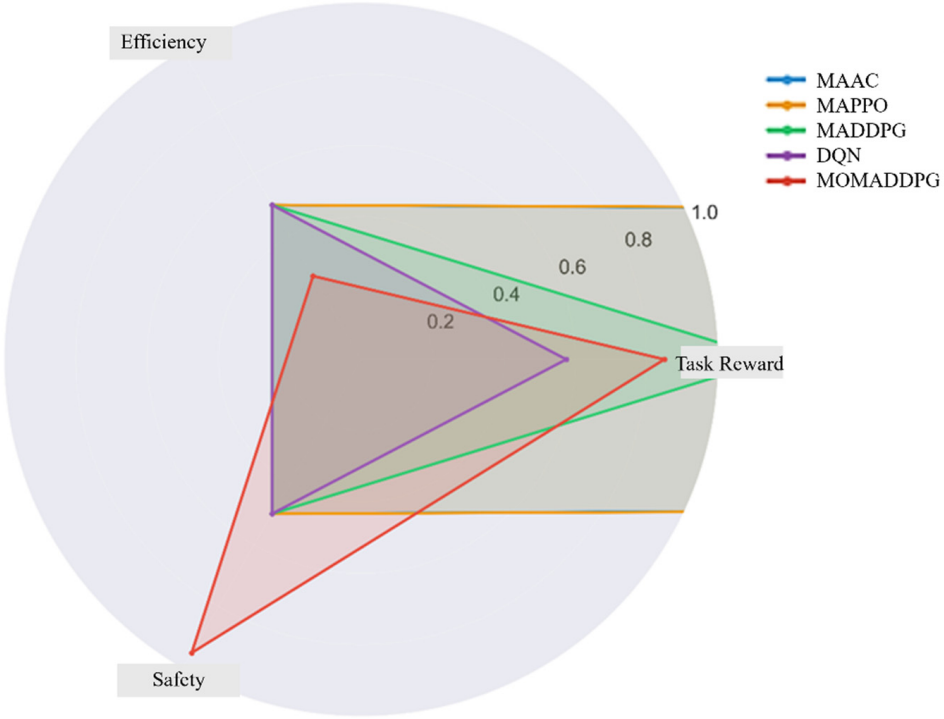


Figure 7. Radar chart of multi-dimensional reward components.

Table 5. Comparison of multi-objective performance metrics.

Metric	DQN	MADDPG	MAAC	MAPPO	MOMADDPG
Task Reward	0.86	1.12	0.91	0.58	1.40
Success Rate (%)	84.0	96.0	87.0	61.0	98.0
Collision Rate (%)	10.0	13.0	12.0	14.0	13.0
Hypervolume (HV)	0.310	0.397	0.423	0.351	0.674

The success rate is also not the inverse of the collision rate. A successful episode may still contain local transient violations if the policy subsequently recovers and completes the tool path according to the termination rule. Therefore, the coexistence of high success rate and non-zero collision rate is not contradictory. It reflects the fact that task completion, local safety violation, and Pareto-front quality are evaluated using different statistical definitions. Overall, the results verify that the proposed MOMADDPG framework improves convergence performance, Pareto-front quality, and constrained multi-objective control capability in the five-axis digital twin machining environment.

5.3. Computational Efficiency Analysis

To further evaluate the computational practicality of the proposed method, the training and inference costs of all compared algorithms were measured on the same workstation. Table 6 reports the total training time for 10,000 episodes, average time per episode, online inference time per control step, GPU memory usage, and the number of episodes required to first reach a 90% task success rate. Although MOMADDPG introduces additional computational overhead due to the hierarchical attention Actor, vector-valued dual Critics,

Lagrangian multiplier update, and Pareto archive maintenance, its online inference only requires forward propagation of the trained Actors and action aggregation. Therefore, the online inference cost remains much lower than the offline training cost and is compatible with offline-training and online-deployment scenarios.

Table 6. Computational efficiency comparison of the compared algorithms.

Algorithm	Total Training Time (h)	Average Time per Episode (s)	Inference Time per Step (ms)	GPU Memory (GB)	Episodes to 90% Success
DQN	4.9	1.76	1.3	4.8	1400
MADDPG	7.2	2.59	2.0	6.9	4600
MAAC	7.9	2.84	2.2	7.2	8900
MAPPO	9.1	3.28	2.6	7.5	—
MOMADDPG	11.4	4.10	2.9	8.8	1900

Note: "—" indicates that the algorithm did not reach a 90% task success rate within 10,000 training episodes.

5.4. Ablation Study of Core Components and Strategy Robustness Verification

To evaluate the individual contribution of each key module in the proposed MOMADDPG framework, four ablation variants were designed and compared with the complete model. In Variant A, the hierarchical attention module in the Actor was removed, and raw state features were directly processed by fully connected layers. In Variant B, the vector-valued dual-Critic structure was replaced by a single scalar Critic, which aggregated quality, efficiency, and safety rewards through fixed-weight scalarization. In Variant C, the Pareto archive-guided preference curriculum was removed, and preference weights were sampled without archive feedback during training. Variant D simultaneously removed the hierarchical attention Actor and the Pareto archive mechanism. In Variant E, the adaptive Lagrangian constraint handling mechanism was removed, and safety constraints were handled only through reward penalties. All variants were trained under the same random seeds, digital twin environment, and hyperparameter settings as the full MOMADDPG framework. The evaluation metrics include task reward, task success rate, collision rate, and hypervolume, as summarized in Table 7 and illustrated in Figure 8.

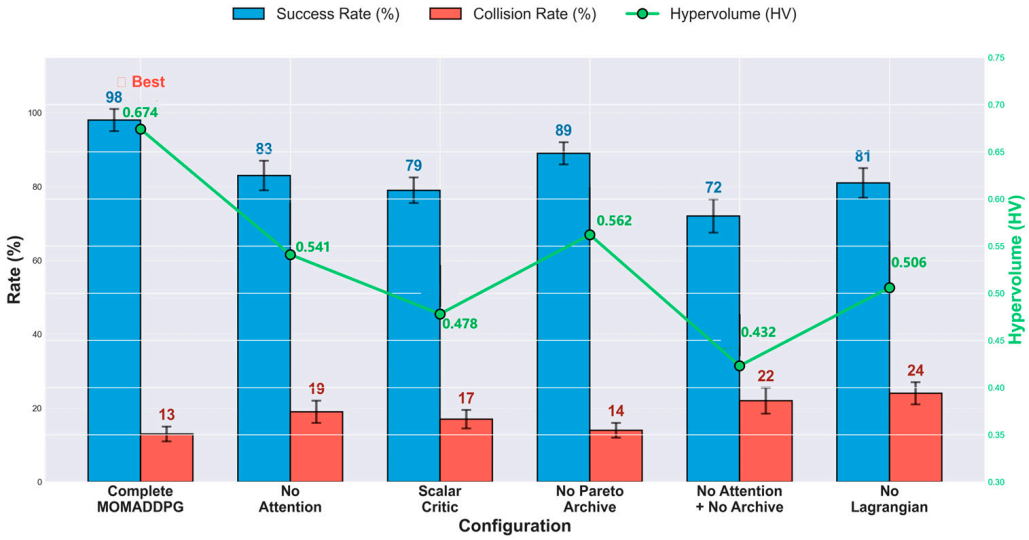


Figure 8. Performance comparison across ablation variants.

The results show that all three core components contribute positively to the final performance. Among the single-module ablations, Variant B produces the largest degradation. Its hypervolume decreases from 0.674 to 0.478, corresponding to a reduction of

approximately 29.1%, while the success rate drops to 79%. This result indicates that scalar value estimation is insufficient for quality–efficiency–safety optimization. Once the reward vector is compressed into a scalar before critic learning, objective-specific value information is lost. The resulting policy gradient is dominated by fixed preference weights and may converge to locally biased solutions, such as overly conservative cutting conditions or efficiency-oriented but unsafe parameter combinations. In contrast, the vector-valued dual-Critic structure preserves independent Q-value estimates for each objective, allowing the policy to adjust its behavior according to different trade-off preferences.

Table 7. Performance metrics for ablation study.

Configuration	Task Reward (T)	Success Rate (%)	Collision Rate (%)	Hypervolume
MOMADDPG	1.4	98	13	0.674
Variant A (No Attention)	1.1	83	19	0.541
Variant B (Scalar Critic)	0.9	79	17	0.478
Variant C (No Archive)	1.2	89	14	0.562
Variant D (A + C)	0.8	72	22	0.423
Variant E (No Lagrangian)	1.0	81	24	0.506

Variant A also shows a clear performance decline. Without the hierarchical attention Actor, the success rate decreases to 83%, and the collision rate increases to 19%. Five-axis machining involves continuously changing tool orientation, local engagement geometry, cutting load, and collision margin. These process variables have different importance at different tool-path positions. The hierarchical attention mechanism helps the Actor distinguish global trajectory information, transient cutting states, and constraint-sensitive features. When this module is removed, the policy becomes less capable of identifying high-risk regions, especially near sharp curvature changes or local engagement variations, leading to more frequent safety violations.

Variant C achieves a success rate of 89%, but its hypervolume decreases to 0.562. This shows that fixed preference weights can still guide the policy to feasible machining behaviors, but they are less effective in maintaining a diverse Pareto-front distribution. The Pareto archive-guided curriculum improves exploration by storing feasible non-dominated solutions and biasing preference sampling toward sparse trade-off regions. Therefore, it contributes more directly to Pareto-front coverage than to task completion alone. Variant D gives the worst overall performance, with a success rate of 72%, a collision rate of 22%, and a hypervolume of 0.423. The degradation is larger than that of either Variant A or Variant C alone, suggesting a synergistic relationship between state representation and archive-guided exploration. Accurate attention-based feature encoding provides reliable policy updates, while the archive mechanism guides the search toward diverse feasible regions. Variant E removes the adaptive Lagrangian constraint handling mechanism and relies only on reward penalties to discourage unsafe actions. Compared with the complete MOMADDPG, Variant E shows a clear increase in collision rate from 13% to 24%, while the success rate decreases from 98% to 81% and the hypervolume decreases from 0.674 to 0.506. This result indicates that explicit constraint handling is important for safety-critical machining optimization. Although reward penalties can partially guide the policy away from unsafe regions, they are less effective than adaptive Lagrangian multipliers in enforcing chatter, spindle power, tool wear, actuator, and collision-related constraints.

The robustness of MOMADDPG was further evaluated under two perturbation conditions. The first condition introduced initial parameter deviations, including a 15% spindle speed error and a 20% feed-rate error. The second condition introduced dynamic process disturbances, represented by 20% cutting-force pulses to simulate material inhomogeneity or local engagement fluctuation. The robustness comparison is shown in Figure 9. In Figure 9a, the initial-condition perturbation results are reported using percentage-based indicators, including nominal success rate, perturbed success rate, success degradation, and collision increase. In Figure 9b, the dynamic-process perturbation indicators are visualized after column-wise min-max normalization because perturbed reward, perturbed collision, and recovery time have different units and numerical scales. Therefore, the values in Figure 9b represent normalized metric magnitudes rather than directly comparable physical quantities.

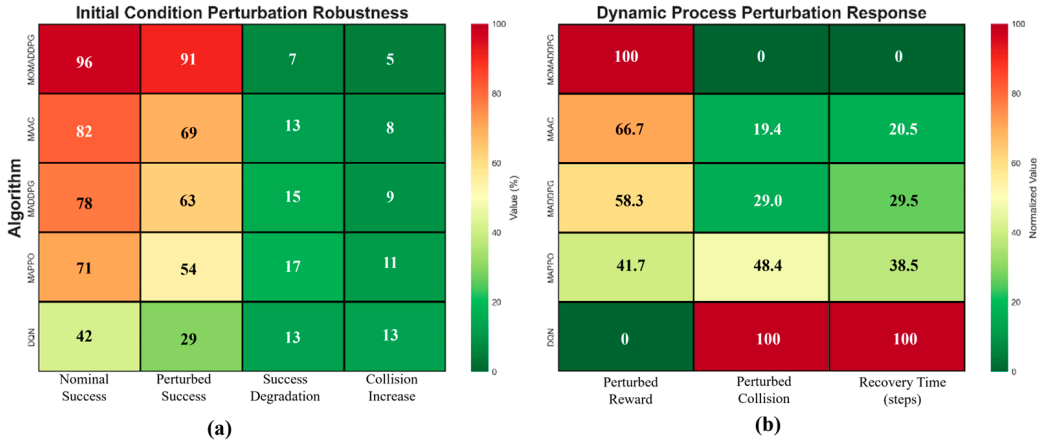


Figure 9. Robustness comparison under perturbation conditions. (**a**) Initial-condition perturbation robustness using percentage-based indicators. (**b**) Dynamic-process perturbation response using column-wise normalized metric values.

Under initial-condition perturbations, MOMADDPG maintains a success rate of 91%, showing only a moderate decrease compared with nominal conditions. In contrast, MAPPO shows a larger performance drop of approximately 17 percentage points. The better robustness of MOMADDPG is mainly attributed to its off-policy learning mechanism, diverse replay buffer, and preference-conditioned policy updates, which improve adaptability to shifted state distributions. Under dynamic force-pulse perturbations, MOMADDPG recovers to nominal performance within 22 control steps, whereas MADDPG requires approximately 45 steps, and DQN fails to recover reliably. This result indicates that heterogeneous multi-agent coordination improves disturbance response. When a perturbation primarily affects one objective dimension, the corresponding specialized agent can identify the abnormal state quickly, while the other agents adjust their actions through the learned interaction representation. In addition, the adaptive Lagrangian constraint penalty helps suppress unsafe actions during recovery, thereby improving constrained robustness.

Overall, the ablation and robustness results confirm that the proposed MOMADDPG framework benefits from the joint effect of hierarchical attention representation, vector-valued dual-Critic learning, adaptive physical constraint handling, and Pareto archive-guided curriculum. These components jointly improve convergence stability, Pareto-front diversity, and safety-aware policy robustness in the five-axis digital twin machining environment.

6. Conclusions

This paper proposed a physics-constrained multi-objective multi-agent deep deterministic policy gradient framework, termed MOMADDPG, for adaptive parameter optimization in five-axis CNC machining simulations. The method addresses the coupled optimization of machining quality, material removal efficiency, and operational safety in a data-calibrated digital twin environment. By assigning heterogeneous agents to different objective dimensions, the framework enables coordinated decision-making under strongly coupled machining dynamics and physical constraints.

A data-calibrated five-axis digital twin environment was constructed using the PHM Society 2010 milling dataset for cutting force and tool wear model calibration. The environment integrates five-axis kinematics, tool-workpiece engagement, transient cutting force prediction, regenerative chatter evaluation, spindle power estimation, tool wear evolution, actuator bounds, and collision-risk evaluation. It provides state observations, vector-valued rewards, and constraint violation signals for safe offline reinforcement learning. It should be noted that the PHM 2010 dataset is not a direct five-axis machining dataset; therefore, the dataset mainly supports the calibration and validation of the cutting force and tool wear sub-models, while five-axis motion and engagement variations are generated by the digital twin environment.

Simulation results show that MOMADDPG outperforms representative baseline algorithms, including DQN, MADDPG, MAAC, and MAPPO, in the constructed digital twin simulation environment. After 10,000 training episodes, the proposed framework achieves a task success rate of 98% and a hypervolume value of 0.674. Compared with MAAC, the best baseline in terms of hypervolume, MOMADDPG improves the hypervolume by approximately 59%, indicating better Pareto-front approximation and distribution diversity. Although the collision rate remains non-zero due to local transient violations, the framework maintains high task feasibility and superior multi-objective performance under the predefined simulation evaluation criteria.

The ablation study further verifies the effectiveness of the core modules. Removing the vector-valued dual-Critic structure causes the largest performance degradation, demonstrating the importance of preserving objective-specific value information. Removing the hierarchical attention Actor weakens state representation under complex tool-orientation and engagement variations, leading to a higher collision rate. Removing the Pareto archive-guided preference curriculum reduces Pareto-front diversity, although task completion can still be maintained to some extent. The combined ablation produces the worst performance, confirming the synergy among feature encoding, multi-objective value learning, and archive-guided exploration.

Robustness tests also demonstrate the disturbance rejection capability of the proposed framework in the simulation environment. Under initial spindle-speed and feed-rate deviations, MOMADDPG maintains a high success rate. Under dynamic cutting-force pulses, it recovers faster than MADDPG and DQN. These results suggest that heterogeneous agent specialization, off-policy learning, and adaptive constraint handling can improve policy robustness under uncertain simulated machining conditions. From the perspective of industrial applicability, the proposed framework is more suitable for an offline-training and online-inference deployment mode. During offline training, the agents interact with the PHM-calibrated digital twin simulation environment to learn feasible Pareto-oriented policies. During online deployment, only the trained Actors, the action aggregation module, and a lightweight safety monitor are required to generate machining parameter adjustments. In practical CNC systems, sensing signals such as spindle power, vibration, cutting force, acoustic emission, and tool wear estimation can be used to update the machining state. The communication with CNC controllers may be implemented through industrial

interfaces such as OPC UA, MTConnect, or machine-specific parameter override channels. However, several challenges remain before real industrial deployment, including sensing latency, signal noise, domain gap between simulation and real machining, machine-specific recalibration, and independent collision/safety supervision. Therefore, physical five-axis CNC validation is still required in future work.

Future work will focus on extending the framework to higher-dimensional objective spaces, including surface roughness, tool life, energy consumption, and dimensional accuracy. Further efforts will also be made to improve the computational efficiency of Pareto archive management and to investigate transfer learning or domain adaptation between different machine tools, materials, and machining platforms. In addition, integration with real-time sensing, CNC communication interfaces, and physical five-axis machining experiments will be an important step toward closed-loop intelligent machining. Overall, the proposed MOMADDPG framework provides a simulation-validated approach for safe, adaptive, and Pareto-oriented parameter optimization in a PHM-calibrated five-axis digital twin environment. Physical five-axis CNC validation and machine-specific recalibration remain necessary before practical industrial deployment.

Author Contributions: J.L. (Jialin Li): conceptualization, investigation, methodology, validation, formal analysis, writing—original draft, writing—review & editing, funding acquisition; J.L. (Jiang Li): investigation, data curation, formal analysis; X.Z.: supervised the project, discussed the experimental and theoretical aspects, investigation, data curation; J.A.: visualization, investigation. All authors have read and agreed to the published version of the manuscript.

Funding: Research was funded by Guiyang University, grant number GYU-KY-[2025]-18.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Lasemi, A.; Xue, D.; Gu, P. Recent development in CNC machining of freeform surfaces: A state-of-the-art review. *Comput.-Aided Des.* **2010**, *42*, 641–654. [CrossRef]
2. Altintas, Y.; Kersting, P.; Biermann, D.; Budak, E.; Denkena, B.; Lazoglu, I. Virtual process systems for part machining operations. *CIRP Ann.* **2014**, *63*, 585–605. [CrossRef]
3. Arrazola, P.J.; Özel, T.; Umbrello, D.; Davies, M.; Jawahir, I.S. Recent advances in modelling of metal machining processes. *CIRP Ann.* **2013**, *62*, 695–718. [CrossRef]
4. Yusup, N.; Zain, A.M.; Hashim, S.Z.M. Evolutionary techniques in optimizing machining parameters: Review and recent applications. *Expert Syst. Appl.* **2012**, *39*, 9909–9927. [CrossRef]
5. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [CrossRef]
6. Özel, T.; Zeren, E. Finite element modeling the influence of edge roundness on the stress and temperature fields induced by high-speed machining. *Int. J. Adv. Manuf. Technol.* **2006**, *35*, 255–267. [CrossRef]
7. Altintaş, Y.; Budak, E. Analytical prediction of stability lobes in milling. *CIRP Ann.* **1995**, *44*, 357–362. [CrossRef]
8. Tao, F.; Zhang, H.; Liu, A.; Nee, A.Y.C. Digital twin in industry: State-of-the-art. *IEEE Trans. Ind. Inform.* **2019**, *15*, 2405–2415. [CrossRef]
9. 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]
10. Luo, W.; Hu, T.; Zhang, C.; Wei, Y. Digital twin for CNC machine tool: Modeling and using strategy. *J. Ambient Intell. Humaniz. Comput.* **2019**, *10*, 1129–1140. [CrossRef]
11. Söderberg, R.; Wärmefjord, K.; Carlson, J.S.; Lindkvist, L. Toward a digital twin for real-time geometry assurance in individualized production. *CIRP Ann.* **2017**, *66*, 137–140. [CrossRef]
12. Levine, S.; Finn, C.; Darrell, T.; Abbeel, P. End-to-end training of deep visuomotor policies. *J. Mach. Learn. Res.* **2016**, *17*, 1–40.
13. Kiran, B.R.; Sobh, I.; Talpaert, V.; Mannion, P.; Sallab, A.A.A.; Yogamani, S.; Pérez, P. Deep reinforcement learning for autonomous driving: A survey. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 4909–4926. [CrossRef]

14. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533. [CrossRef] [PubMed]
15. Lillicrap, T.P.; Hunt, J.J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous control with deep reinforcement learning. *arXiv* **2015**, arXiv:1509.02971.
16. Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; Klimov, O. Proximal policy optimization algorithms. *arXiv* **2017**, arXiv:1707.06347. [CrossRef]
17. Kuhnle, A.; Schäfer, L.; Stricker, N.; Lanza, G. Design, implementation and evaluation of reinforcement learning for an adaptive order dispatching in job shop manufacturing systems. *Procedia CIRP* **2019**, *81*, 234–239. [CrossRef]
18. Li, Y. Deep reinforcement learning: An overview. *arXiv* **2018**, arXiv:1701.07274. [CrossRef]
19. Lowe, R.; Wu, Y.; Tamar, A.; Harb, J.; Abbeel, P.; Mordatch, I. Multi-agent actor-critic for mixed cooperative-competitive environments. *arXiv* **2020**, arXiv:1706.02275. [CrossRef]
20. Vamplew, P.; Dazeley, R.; Berry, A.; Issabekov, R.; Dekker, E. Empirical evaluation methods for multiobjective reinforcement learning algorithms. *Mach. Learn.* **2011**, *84*, 51–80. [CrossRef]
21. Iqbal, S.; Sha, F. Actor-attention-critic for multi-agent reinforcement learning. *arXiv* **2019**, arXiv:1810.02912. [CrossRef]
22. Yu, C.; Velu, A.; Vinitsky, E.; Gao, J.; Wang, Y.; Bayen, A.; Wu, Y. The surprising effectiveness of PPO in cooperative multi-agent games. *arXiv* **2022**, arXiv:2103.01955. [CrossRef]
23. Yang, R.; Sun, X.; Narasimhan, K. A generalized algorithm for multi-objective reinforcement learning and policy adaptation. *arXiv* **2019**, arXiv:1908.08342. [CrossRef]
24. Liu, C.; Vengayil, H.; Lu, Y.; Xu, X. A cyber-physical machine tools platform using OPC UA and MTConnect. *J. Manuf. Syst.* **2019**, *51*, 61–74. [CrossRef]
25. Pangestu, P.; Pujiyanto, E.; Rosyidi, C.N. Multi-objective cutting parameter optimization model of multi-pass turning in CNC machines for sustainable manufacturing. *Heliyon* **2021**, *7*, e06043. [CrossRef] [PubMed]
26. Wang, G.G.; Shan, S. Review of metamodeling techniques in support of engineering design optimization. *J. Mech. Des.* **2007**, *129*, 370–380. [CrossRef]
27. Wang, Y.; Kasaei, H. Fast trajectory planner with a reinforcement learning-based controller for robotic manipulators. *Eng. Appl. Artif. Intell.* **2025**, *162*, 112341. [CrossRef]
28. Chou, P.W.; Maturana, D.; Scherer, S. Improving stochastic policy gradients in continuous control with deep reinforcement learning using the beta distribution. In *Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017*; PMLR: New York, NY, USA, 2017; pp. 834–843.
29. Liu, Y.; Wang, L.; Wang, X.V.; Xu, X.; Zhang, L. Scheduling in cloud manufacturing: State-of-the-art and research challenges. *Int. J. Prod. Res.* **2019**, *57*, 4854–4879. [CrossRef]
30. Mossalam, H.; Assael, Y.M.; Roijers, D.M.; Whiteson, S. Multi-objective deep reinforcement learning. *arXiv* **2016**, arXiv:1610.02707. [CrossRef]
31. PHM Society. *2010 PHM Society Conference Data Challenge: High-Speed Milling Cutter Degradation Dataset*; Prognostics and Health Management Society: Rochester, NY, USA, 2010. Available online: https://www.phmsociety.org/competition/phm/10 (accessed on 15 June 2026).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.