automation

MDPI

Article

Energy-Aware Cooperative Multi-Agent Deep Reinforcement Learning for Sustainable and Precision Trajectory Tracking of an Industrial Robotic Arm: An SDG-Aligned Approach

Zied Ben Hazem [1,2]

[1] Automation and Sustainability Research Centre (ASRC), Department of Mechatronics Engineering, College of Engineering (COE), University of Technology Bahrain, Salmabad 18041, Bahrain; z.hazem@utb.edu.bh
[2] AI and Metaverse Research Centre (AIMRC), Department of Information Technology, College of Computer Science (CCS), University of Technology Bahrain, Salmabad 18041, Bahrain

Abstract

Control software that lowers actuator effort without degrading precision can be deployed on installed industrial robots without new hardware. This paper formulates trajectory tracking of a 5-degree-of-freedom (DOF) Mitsubishi RV-2AJ arm as a fully cooperative multi-agent problem, in which each joint is driven by a twin delayed deep deterministic policy gradient (TD3) agent under centralized training and decentralized execution (CTDE). Unlike MADDPG, MATD3 and MAPPO, the proposed CTDE-TD3 combines (i) a single twin centralized critic, shared by all joint agents and conditioned on all joint states and torques, which removes the non-stationarity faced by independent agents while each actor uses only three local measurements at run time; (ii) a shared reward with an inter-joint torque-covariance term, whose exact decomposition yields the admissible weight condition $\mu \leq 2\lambda$; and (iii) a dimensionless weight normalization. In simulation on three 3D reference trajectories over 10 training seeds, the controller reduced the tracking root-mean-square error (RMSE) by 54.5% and 37.5% and the squared-torque effort by 33.1% and 21.0% relative to centralized and independent-agent TD3, respectively. It was not dominated by MADDPG, MATD3, MAPPO or a tuned proportional–integral–derivative (PID) controller on unseen trajectories. Ablations and a capacity-matched centralized baseline show that the gains stem from the cooperative structure rather than from network size. Because the squared-torque index reflects control effort and copper loss rather than total electrical energy, electricity and CO_2 savings are reported only as scenario estimates pending hardware validation.

Keywords: sustainable manufacturing; energy-efficient robotics; multi-agent reinforcement learning; trajectory tracking; industrial robotic arm; actuator effort

Check for updates

Academic Editor: En Li

Received: 19 October 2026
Revised: 11 November 2026
Accepted: 13 November 2026
Published: 18 November 2026

Copyright: © 2026 by the author. 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

Manufacturing is increasingly required to achieve higher productivity while reducing energy consumption and greenhouse gas emissions, in alignment with the United Nations Sustainable Development Goals (SDGs), particularly affordable and clean energy (SDG 7), industry, innovation and infrastructure (SDG 9), responsible consumption and production (SDG 12), and climate action (SDG 13) [1]. Industrial robots play a central role in this transition, as they are extensively deployed across automated manufacturing processes, making their energy consumption an important consideration in the development of sustainable production systems [2]. As continuous electrical loads within manufacturing facilities, robotic systems contribute to the overall energy demand and carbon footprint

of industrial operations. Consequently, improving the energy efficiency of robotic and automation systems represents an important opportunity for advancing sustainable industrial production [3].

Existing energy-saving strategies for industrial robots primarily focus on hardware improvements and motion optimization, including the selection of energy-efficient robot configurations, the integration of energy-storage and regenerative components, and the optimization of trajectories and motion profiles [3,4]. Although these approaches can substantially reduce energy consumption, they may require hardware modifications, task-specific trajectory replanning, or sufficiently accurate dynamic models. An alternative and complementary strategy is to reduce actuator effort during task execution through the feedback-control system itself. From a sustainability perspective, this approach is particularly attractive because it can be implemented through software modifications on existing industrial robots, potentially avoiding additional hardware investments and facilitating the retrofitting of installed production lines. Furthermore, controller-level energy optimization can be integrated with existing hardware improvements and trajectory-planning strategies. Against this background, this study investigates whether a learning-based control strategy can simultaneously reduce actuator effort and improve trajectory-tracking precision without compromising either objective.

Trajectory planning is among the most effective software-based strategies for improving robotic energy efficiency because it modifies the commanded motion without requiring additional hardware. Lozer et al. [5] developed minimum-jerk trajectories for redundant robots by jointly optimizing the time intervals between waypoints along a prescribed path and the positions of a selected redundant joint, thereby producing smoother motion and reducing actuator stress. Donà et al. [6] proposed a real-time approach for generating near-minimum-energy manipulator trajectories using constraint-informed residual learning. Their method learns only the correction required to steer a nominal trajectory toward the solution of an optimal control problem, achieving 87.3% of the optimal-control planner's performance near the training data and 50.8% outside that region, while operating two to three orders of magnitude faster. Van Oosterwyck et al. [7] employed bounded Chebyshev polynomials to parameterize point-to-point motion profiles and reported a 62.9% reduction in energy consumption relative to a standard reference profile. Collectively, these studies demonstrate the potential of trajectory optimization to reduce the energy demand associated with executing prescribed robotic tasks.

The proposed controller complements these trajectory-optimization approaches rather than replacing them. A trajectory planner determines the reference motion and, through inverse dynamics, the minimum energy that the task demands, whereas the feedback controller determines the additional control effort required to track that reference in the presence of joint coupling, disturbances and model uncertainties. Since the proposed controller accepts the reference trajectory as an input without imposing restrictions on how it is generated, minimum-jerk or minimum-energy trajectories obtained with the approaches in [5–7] can be integrated directly without modifying the controller architecture. The reported improvements are, however, not directly comparable, because their evaluation metrics differ. Van Oosterwyck et al. [7] report the energy consumption of a single-axis mechanism relative to a reference motion profile, whereas the 21.0–33.1% reductions achieved in this study refer to a squared-torque, copper-loss-related effort index relative to alternative learned controllers operating on identical reference trajectories. Under the assumptions adopted in the sustainability assessment, these reductions correspond to an estimated 2–10% decrease in the robot's total electricity consumption. The respective contributions of trajectory optimization and feedback control are discussed further in the sustainability assessment.

Beyond energy efficiency, achieving precise trajectory tracking in industrial robotic manipulators remains a challenging control problem. The increasing deployment of intelligent, learning-enabled robotic systems in dynamic manufacturing environments requires controllers capable of maintaining high tracking accuracy while accommodating nonlinear dynamics, joint interactions, model uncertainties, variable payloads, and actuator constraints [8,9]. Conventional control techniques, particularly proportional–integral–derivative (PID) controllers and other model-based strategies, remain widely adopted in industrial applications because of their simplicity, established implementation procedures, and practical reliability. However, their performance may deteriorate under complex operating conditions, particularly when accurate dynamic models are unavailable or when substantial interactions exist among multiple joints. Furthermore, centralized control architectures may introduce computational and coordination challenges as system dimensionality increases. In this context, deep reinforcement learning (DRL) has emerged as a promising data-driven control paradigm that learns control policies through interactions with the environment, potentially reducing dependence on explicitly derived and highly accurate dynamic models. Although single-agent DRL approaches have demonstrated promising capabilities in robotic trajectory tracking, centralized policy learning can become computationally demanding for multi-joint systems and may not fully exploit the modular structure of individual joint dynamics. Related developments in model-free intelligent control have also demonstrated the potential of learning-based and observer-assisted approaches for nonlinear industrial processes. For instance, a nonlinear extended-state-observer-based, model-free near-optimal sliding-mode controller was recently proposed for water-level regulation in an inverted U-tube steam generator [10].

To address the challenges associated with coordinating multiple interacting joints, multi-agent deep reinforcement learning (MADRL) has attracted increasing research interest as an alternative framework for complex robotic control. By decomposing a high-dimensional control problem into multiple cooperative learning agents, MADRL enables joint-level decision-making while maintaining coordinated system behavior through shared objectives and centralized training mechanisms. This architecture is particularly suitable for industrial robotic manipulators, whose individual joints exhibit structurally modular dynamics while remaining strongly coupled through the overall mechanical system. Multi-agent formulations have long been used to coordinate the motion of multiple industrial robots [11], and incorporating energy-aware objectives into cooperative multi-agent learning provides a systematic means of balancing trajectory-tracking accuracy against actuator effort, an increasingly important requirement for sustainable industrial automation. Recent studies have investigated MADRL across various robotic applications, including cooperative mobile-robot navigation, multi-joint manipulation, and distributed motion planning, reporting improvements in coordination and learning performance under different operating conditions [12–14]. Further developments have extended multi-agent learning to energy-aware and resource-constrained robotic systems, demonstrating its potential for balancing operational performance with resource utilization [15,16]. Despite these advances, the integration of cooperative multi-agent learning, theoretically grounded energy-aware reward design, and decentralized joint-level execution for precision trajectory tracking remains an important research direction.

Building on these developments, this study proposes a cooperative multi-agent deep reinforcement learning framework based on twin delayed deep deterministic policy gradient (CTDE-TD3) for energy-aware precision trajectory tracking of a five-degree-of-freedom Mitsubishi RV-2AJ industrial robotic arm. The proposed framework adopts centralized training and decentralized execution (CTDE), with individual TD3 agents assigned to the respective joints and coordinated through a shared twin centralized critic. In contrast to

conventional multi-agent actor–critic architectures, the proposed approach exploits the fully cooperative nature of the robotic control problem to employ a single shared critic pair rather than independent critic networks for each agent. Furthermore, an energy-aware shared reward function incorporates an inter-joint torque-covariance term alongside trajectory-tracking and squared-torque penalties, providing a coordinated mechanism for balancing tracking precision and actuator effort. The theoretical formulation establishes an explicit admissibility condition for the reward weights, while a dimensionless normalization scheme facilitates systematic reward configuration. The proposed architecture is evaluated against centralized TD3, independent-agent TD3, MADDPG, MATD3, MAPPO, and a tuned PID controller under consistent experimental conditions. Table 1 provides a structured comparison between conventional centralized DRL architectures and the proposed cooperative multi-agent framework, highlighting their differences in scalability, coordination, execution requirements, and energy-aware control capabilities [17].

Table 1. Comparison of centralized DRL and multi-agent DRL control architectures for robotic manipulators.

Algorithm	Key Capabilities	Limitations	Suitability for Robotic Systems
Centralized DRL (Single-Agent TD3) [18,19]	• Learns end-to-end control policies for continuous action spaces without explicit dynamic models. • Capable of handling nonlinear robotic dynamics through data-driven learning. • Straightforward architecture with a single policy governing all joints.	• Limited scalability for high-DOF manipulators due to large state–action spaces. • High computational complexity and slower convergence as system dimensionality increases. • Suboptimal coordination among joints, as a single agent must capture all interdependencies.	Suitable for low-DOF robotic systems or simulation-based studies where scalability and joint-level modularity are not primary concerns.
Decentralized DRL (Independent Agents TD3) [20,21]	• Assigns independent agents to individual joints or subsystems, reducing action-space dimensionality. • Enables parallel learning and improved scalability compared to centralized DRL. • Simplifies policy design at the joint level.	• Lack of explicit coordination mechanisms may lead to conflicting control actions. • Training instability due to non-stationary environments caused by concurrently learning agents. • Limited performance in tightly coupled robotic systems.	Appropriate for loosely coupled robotic systems or preliminary decentralized control studies with minimal coordination requirements.
Cooperative Multi-Agent DRL (Proposed) [22]	• Employs cooperative joint-level agents with shared objectives, enabling coordinated motion control. • Centralized training with decentralized execution (CTDE) improves learning stability and scalability. • Explicitly balances precision trajectory tracking and energy efficiency through multi-objective learning. • Modular and scalable architecture suitable for high-DOF manipulators.	• Increased training complexity due to multiple agents and coordination mechanisms. • Requires careful reward design to ensure cooperative behavior and convergence.	Highly suitable for industrial robotic manipulators requiring precision trajectory tracking, energy-efficient operation, and scalable control in complex, high-DOF systems.

Three families of learning methods are relevant to joint-level control of a manipulator. Independent learners treat all other agents as part of the environment, which is simple to implement but violates the stationarity assumption of single-agent reinforcement learning [23]. Value-decomposition and counterfactual-credit methods such as QMIX [24]

and COMA [25] address credit assignment but were developed mainly for discrete action sets, whereas joint torques are continuous. Centralized-critic actor–critic methods, including MADDPG [26], MATD3 [27] and MAPPO [28], condition the critic on all observations and actions during training while keeping the actors decentralized; they are the natural fit for continuous torque control and are the basis of the CTDE structure adopted here.

Relative to these centralized-critic methods, the specific differences of CTDE-TD3 are as follows. MADDPG [26] and MATD3 [27] were formulated for mixed cooperative–competitive games and therefore train a centralized critic for every agent (n critics for MADDPG and $2n$ twin critics for MATD3), whereas the single twin critic pair used here gives every joint actor the same estimate of the shared return. Because this critic is conditioned on the torques of all joints, its learning target is stationary with respect to the changing policies of the other agents, while each actor keeps only three local inputs and needs no communication at run time. Unlike the on-policy MAPPO [28], the controller retains the off-policy replay, clipped double-Q targets, target-policy smoothing and delayed actor updates of TD3 [17], which matters when each environment step requires the integration of a stiff multibody model. Finally, the exact decomposition of the covariance term (Proposition 1) yields the admissibility condition $\mu \leq 2\lambda$, for which the generic MARL algorithms provide no counterpart.

Despite these advances, three gaps remain in learning-based manipulator control. First, many DRL trackers are judged almost exclusively on accuracy, while the actuator effort needed to obtain that accuracy is either unreported or treated as a secondary penalty. Second, joint-level decomposition is usually adopted for scalability alone, without examining how much of the observed benefit comes from decomposition itself and how much comes from explicit coordination between joints. Third, comparisons are often restricted to a single trajectory or a single baseline, which makes it hard to judge how performance scales with trajectory complexity. This paper addresses these gaps by (i) adding an inter-joint torque-covariance term to an energy-aware reward and analyzing its structure, (ii) benchmarking the cooperative controller against both a centralized and an independent-agent TD3 controller so that the effect of decomposition can be separated from the effect of cooperation, and (iii) reporting accuracy and energy jointly over three trajectories of increasing complexity, supported by the aggregate, statistical-consistency, trade-off, complexity-sensitivity and computational-footprint analyses of the simulation results. The design builds on the TD3 algorithm [17] and on the deterministic policy-gradient theorem [29].

The proposed controller is trained and evaluated in a high-fidelity MATLAB/Simulink environment (version R2023b, MathWorks) built with the Simscape Multibody toolbox, which captures the rigid-body dynamics and physical constraints of the Mitsubishi RV-2AJ arm and provides realistic joint-state feedback and end-effector information for training the MADRL agents. The evaluation covers three trajectory profiles of increasing complexity together with payload, friction, inertia and torque-disturbance perturbations. The key contributions of this paper are summarized as follows:

A cooperative MADRL control framework is proposed for precision trajectory tracking of an industrial robotic arm, where independent joint-level agents collaboratively learn coordinated control policies using a centralized training–decentralized execution strategy. Unlike MADDPG, MATD3 and MAPPO, it trains a single twin centralized critic shared by all joint agents of a fully cooperative game with the off-policy TD3 update, while the three-input actors need no inter-joint communication at run time.

An energy-aware reward formulation is introduced to explicitly balance trajectory tracking accuracy and actuator effort, enabling an effective trade-off between precision and efficiency.

A high-fidelity simulation platform based on MATLAB/Simulink and the Simscape Multibody toolbox is developed to accurately model the dynamics of the 5-DOF Mitsubishi RV-2AJ robotic arm, facilitating safe and efficient training of multi-agent DRL controllers.

Comprehensive simulation studies demonstrate that the proposed MADRL approach outperforms centralized single-agent DRL and conventional control methods in terms of tracking precision, squared-torque control effort and coordination capability.

A theoretical and analytical characterization of the cooperative reward and of the dynamic coupling it addresses, including an exact decomposition of the covariance penalty that yields a weight condition, a coupling index of the manipulator dynamics, a weight-normalization rule, and a complexity and scalability analysis, together with aggregate, paired-difference, trade-off and sensitivity analyses of the simulation results.

A sustainability assessment that translates the torque-energy reduction into electricity and CO_2 estimates under explicit assumptions, relates the results to the SDGs, and quantifies the break-even operating time of the training energy.

The remainder of this paper is organized as follows. Section 2 presents the dynamic model of the 5-DOF Mitsubishi RV-2AJ robotic arm and its validation against the Simscape multibody model. Section 3 introduces the cooperative MADRL framework: Sections 3.1 and 3.2 review centralized single-agent and independent-agent DRL, Section 3.3 describes the proposed cooperative architecture and its energy-aware reward, Section 3.4 formulates the cooperative Markov game and the CTDE-TD3 training procedure, Section 3.5 analyzes the reward and the dynamic coupling, and Section 3.6 discusses the multi-objective interpretation, the weight normalization and the computational complexity. Section 4 reports the simulation results: Section 4.1 describes the simulation set-up and the training of the proposed controller, Section 4.2 compares it with centralized and independent-agent TD3 and discusses the aggregate, trade-off, consistency, scalability and limitation analyses, Section 4.3 presents the additional experiments on seed reproducibility, ablation, reward-weight sensitivity, robustness, generalization with stronger baselines, a model-based NMPC baseline and a capacity-matched centralized baseline, and Section 4.4 assesses the sustainability implications of the measured torque-effort reductions. Finally, Section 5 concludes the paper and outlines directions for future work.

2. System Modeling of the 5-DOF Mitsubishi RV-2AJ Robotic Arm

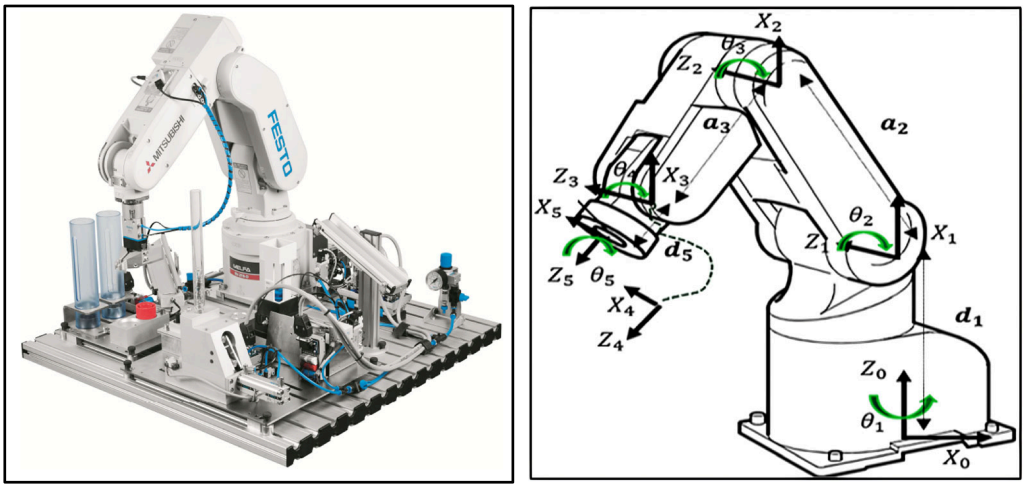
The Mitsubishi RV-2AJ robotic arm, illustrated in Figure 1, is represented in this work as a 5-DOF serial manipulator. To describe the spatial relationship between successive links, the forward kinematic model is formulated using the Denavit–Hartenberg (D-H) parameterization. Based on this formulation, the overall homogeneous transformation matrix, which maps joint coordinates to the end-effector position and orientation in Cartesian space, is expressed as:

$$ {}^{0}_{5}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T \tag{1} $$

where each individual homogeneous transformation matrix is defined as:

$$ {}^{i-1}_{i}T = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_{i-1} \\ \sin\theta_i \cos\alpha_{i-1} & \cos\theta_i \cos\alpha_{i-1} & -\sin\alpha_{i-1} & -\sin\alpha_{i-1} d_i \\ \sin\theta_i \sin\alpha_{i-1} & \cos\theta_i \sin\alpha_{i-1} & \cos\alpha_{i-1} & \cos\alpha_{i-1}\, d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2} $$

where θ_i is the joint angle, d_i the link offset, a_{i-1} the link length and α_{i-1} the link twist of the modified Denavit–Hartenberg convention.



(**a**) (**b**)

Figure 1. Mitsubishi RV-2AJ 5 DOF robotic manipulator: (**a**) experimental setup; (**b**) corresponding kinematic model.

The system dynamics, accounting for nonlinear behavior, dynamic coupling between joints, and gravitational influences, are obtained using the Euler–Lagrange approach. The resulting equation of motion is given by:

$$D(\theta)\ddot{\theta} + C\left(\theta, \dot{\theta}\right) + G(\theta) = \tau - \tau_f \tag{3}$$

The vectors θ, $\dot{\theta}$, and $\ddot{\theta}$ denote the joint positions, velocities, and accelerations, respectively. The manipulator dynamics are described by the inertia matrix $D(\theta)$, the Coriolis and centrifugal term $C\left(\theta, \dot{\theta}\right)$, and the gravity vector $G(\theta)$. The input torque vector τ represents the applied joint torques, while τ_f accounts for friction effects, which may include viscous and nonlinear components. The inertia matrix is obtained from the translational and rotational kinetic energies of the links and is expressed as:

$$D(\theta) = \sum_{i=1}^{n} \left[(A_i)^{\mathrm{T}} m_i A_i + (B_i)^{\mathrm{T}} I_i B_i \right] \tag{4}$$

The Coriolis and centrifugal terms are computed from the inertia matrix using Christoffel coefficients:

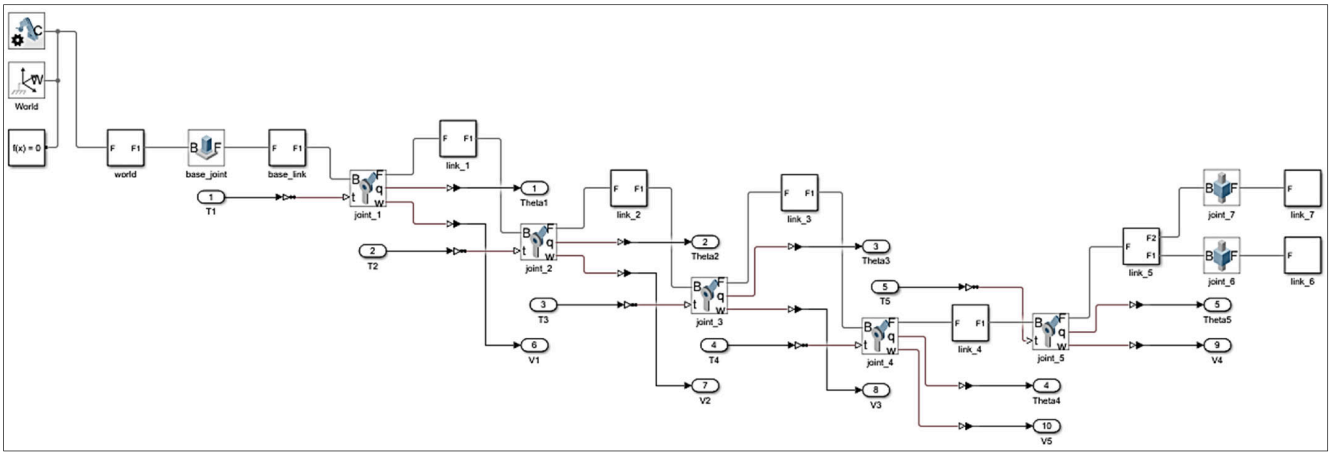
$$C\left(\theta, \dot{\theta}\right) = \sum_{k=1}^{n} \sum_{j=1}^{n} \left[c_{kj}^{i}(\theta) \dot{\theta}_k \dot{\theta}_j \right] \tag{5}$$

$$c_{kj}^{i}(\theta) = \frac{\partial}{\partial \theta_k} D_{ij}(\theta) - \frac{1}{2} \frac{\partial}{\partial \theta_i} D_{kj}, \; 1 \le i, j, k \le n \tag{6}$$

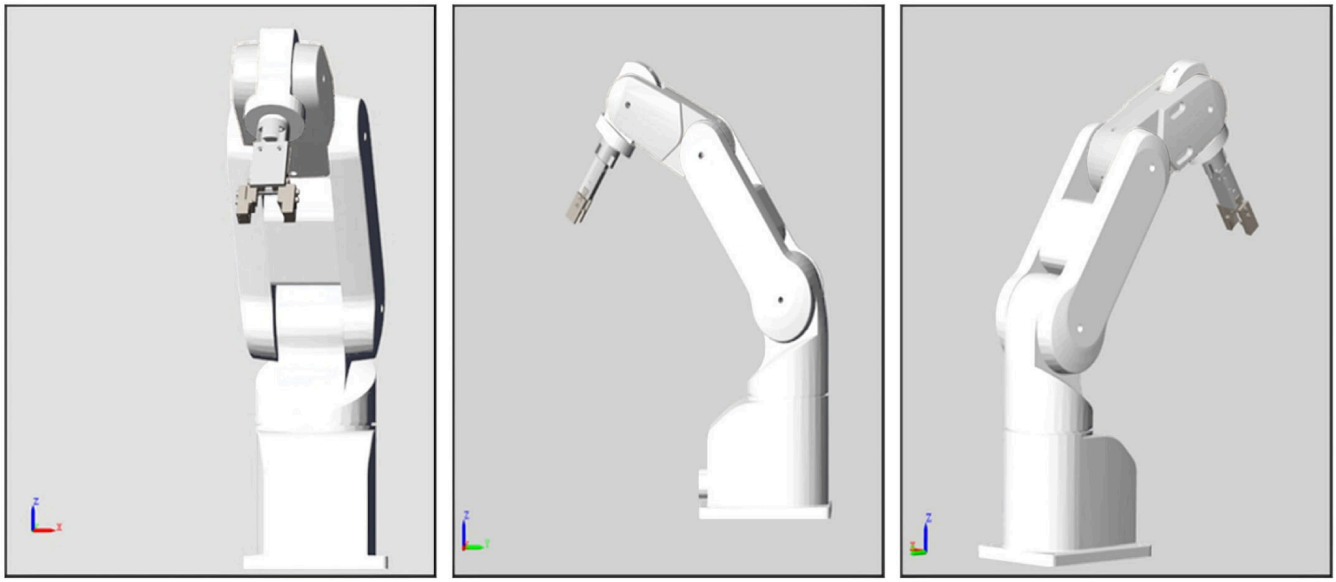
The gravity vector is given by

$$G(\theta) = -\sum_{k=1}^{n} \sum_{j=1}^{n} \left[g_k \, m_j \, A_{ki}^{j}(\theta) \right] \tag{7}$$

The validity of the proposed mathematical model was assessed by comparison with a high-fidelity Simscape Multibody representation (Figure 2). The strong agreement observed in the joint responses (Figure 3) demonstrates that the model is sufficiently accurate for the design and training of the proposed control strategy, in line with the methodology reported in Hazem et al. [30,31].



(**a**)



(**b**)

Figure 2. Simulation framework of the Mitsubishi RV-2AJ five-DOF manipulator: (**a**) multibody dynamics model implemented in Simscape; (**b**) CAD-based visualization in Simulink from different viewpoints.

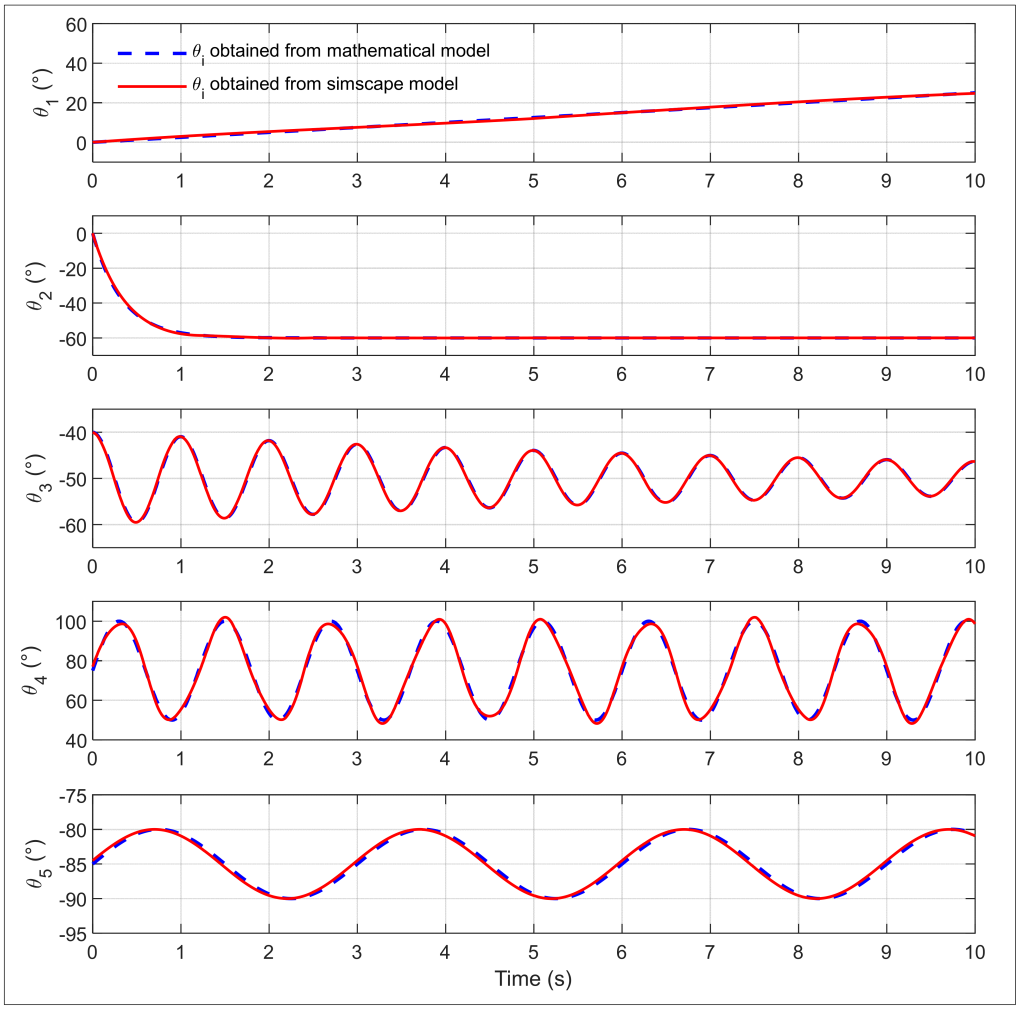


Figure 3. Validation of the dynamic model through comparative joint angle responses θ_1–θ_5 from the proposed formulation and the Simscape multibody model.

3. Cooperative Multi-Agent DRL Framework for Trajectory Tracking

This section introduces the proposed cooperative MADRL framework for robust trajectory tracking of the 5-DOF Mitsubishi RV-2AJ robotic arm. To highlight its advantages, background is first given on single-agent DRL and independent-agent DRL, including their architectures, algorithms, and reward formulations. The section then details the proposed cooperative MADRL architecture, including CTDE and an energy-aware reward function designed to optimize both tracking precision and actuator efficiency.

3.1. Single-Agent DRL (Centralized Control)

In the single-agent DRL paradigm, a centralized agent learns an end-to-end control policy for the entire robotic arm, mapping the global state vector to joint torques:

$$a_t = \pi_\theta(s_t) \tag{8}$$

where $s_t = \left[\theta_1, \ldots, \theta_5, \dot{\theta}_1, \ldots, \dot{\theta}_5, \theta_1^{\text{ref}}, \ldots, \theta_5^{\text{ref}}\right]$ is the full state vector and $a_t = [\tau_1, \ldots, \tau_5]$ is the action vector of joint torques. The reward function for trajectory tracking is defined as:

$$r_t = -\sum_{i=1}^{5} (\theta_i^{\text{ref}} - \theta_i)^2 - \lambda \sum_{i=1}^{5} \tau_i^2 \tag{9}$$

where λ is a weighting factor balancing tracking error and control effort. The TD3 algorithm is applied, using twin critic networks Q_{ϕ_1}, Q_{ϕ_2} and an actor network $\pi_\theta(s)$. Critic updates are performed using:

$$\mathcal{L}(\phi_j) = \mathbb{E}_{(s,a,r,s')\sim\mathcal{D}}\left[\left(y - Q_{\phi_j}(s,a)\right)^2\right], j = 1,2 \tag{10}$$

with target value:

$$y = r + \gamma mi\,\mathrm{n}Q_{\phi'_j}(s', \pi_{\theta'}(s') + \epsilon), \epsilon \sim \mathcal{N}(0,\sigma) \tag{11}$$

Target networks are updated using soft updates:

$$\phi'_j \leftarrow \rho\phi_j + (1-\rho)\phi'_j, \theta' \leftarrow \rho\theta + (1-\rho)\theta' \tag{12}$$

where $\rho \ll 1$ is the Polyak averaging coefficient.

Centralized single-agent DRL is limited by large state–action spaces, slow convergence, and computational complexity.

3.2. Independent Agents DRL (Decentralized Control)

In the decentralized DRL paradigm, each joint of the robotic arm is controlled by an independent agent, allowing the action space of each agent to remain low-dimensional. For the 5-DOF Mitsubishi RV-2AJ robotic arm, this means assigning a separate TD3 agent π_{θ_i} to each joint i, which learns a local policy using only joint-specific information:

$$s_t^i = \left[\theta_i, \dot{\theta}_i, \theta_i^{\mathrm{ref}}\right], a_t^i = \tau_i = \pi_{\theta_i}\left(s_t^i\right), i = 1, \ldots, 5 \tag{13}$$

The reward function for each agent balances tracking accuracy and control effort locally:

$$r_t^i = -\left(\theta_i^{\mathrm{ref}} - \theta_i\right)^2 - \lambda\left(\tau_i\right)^2 \tag{14}$$

where λ is a weighting factor. Each agent independently updates its actor-critic networks using TD3, with critic loss:

$$\mathcal{L}(\phi_{i,j}) = \mathbb{E}_{(s^i,a^i,r^i,s^{i\prime})\sim\mathcal{D}_i}\left[\left(y^i - Q_{\phi_{i,j}}\left(s^i,a^i\right)\right)^2\right],\ j = 1,2 \tag{15}$$

and target value:

$$y^i = r^i + \gamma \min_{j=1,2} Q_{\phi'_{i,j}}(s^{i\prime}, \pi_{\theta'_i}(s^{i\prime}) + \epsilon), \epsilon \sim \mathcal{N}(0,\sigma) \tag{16}$$

Soft updates are applied for the target networks:

$$\phi'_{i,j} \leftarrow \rho\phi_{i,j} + (1-\rho)\phi'_{i,j}, \theta'_i \leftarrow \rho\theta_i + (1-\rho)\theta'_i, j = 1,2, \rho \ll 1 \tag{17}$$

For example, for joint 3, the local state is $s_t^3 = \left[\theta_3, \dot{\theta}_3, \theta_3^{\mathrm{ref}}\right]$. The agent outputs a torque $a_t^3 = \tau_3$, observes the next joint state, computes the local reward $r_t^3 = -\left(\theta_3^{\mathrm{ref}} - \theta_3\right)^2 - \lambda\left(\tau_3\right)^2$, and updates its actor–critic networks using mini-batches from its local replay buffer. The same procedure runs independently and in parallel for the other four joints. Decentralized DRL reduces the dimensionality of each agent's learning problem, improves scalability for modular control, and allows parallel training. However, the lack of explicit coordination among joints may lead to conflicting control actions in tightly coupled robotic systems.

3.3. Cooperative Multi-Agent DRL for Robust Trajectory Tracking

To overcome the limitations of centralized single-agent DRL and independent-agent decentralized DRL, this study proposes a cooperative MADRL framework that integrates CTDE. In this framework, each joint is controlled by an agent, but agents cooperate during training to learn coordinated policies while maintaining independent execution at runtime. This structure enables scalable learning for high-DOF manipulators while ensuring joint-level coordination and energy efficiency.

3.3.1. Agent Architecture

Each agent i observes its local state:

$$s_t^i = \left[\theta_i, \dot{\theta}_i, \theta_i^{\text{ref}}\right] \tag{18}$$

and outputs a local action:

$$a_t^i = \tau_i = \pi_{\theta_i}\left(s_t^i\right) \tag{19}$$

During training, a centralized critic receives the global state $s_t = \left[s_t^1, \ldots, s_t^5\right]$ and joint actions $a_t = \left[a_t^1, \ldots, a_t^5\right]$ to evaluate the cooperative Q-value:

$$Q_\phi(s_t, a_t) \tag{20}$$

This allows the critic to account for inter-joint dependencies, improving coordination across the arm.

3.3.2. Energy-Aware Reward Function

A novel energy-aware cooperative reward is designed to optimize both tracking accuracy and actuator efficiency:

$$r_t = -\sum_{i=1}^{5}(\theta_i^{\text{ref}} - \theta_i)^2 - \lambda\sum_{i=1}^{5}(\tau_i)^2 - \mu\sum_{i<j}^{5}\text{cov}(\tau_i, \tau_j) \tag{21}$$

The first term penalizes tracking error. The second term penalizes high actuator torque to encourage energy efficiency. The third term penalizes uncoordinated joint movements by considering the covariance of torques across joints (μ is a weighting factor), promoting synchronized and smooth trajectories. The three terms act on different aspects of closed-loop behavior. The tracking term drives accuracy. The quadratic torque term serves as a proxy for actuator energy: for a servo drive with torque constant k_t and winding resistance R, the resistive loss is $Ri^2 = \left(R/k_t^2\right)\tau^2$, i.e., proportional to the squared joint torque. The covariance term couples the agents through the reward: since the reward is shared and computed from all joints, the centralized critic receives the information needed to attribute changes in the return to combinations of joint actions, which allows the decentralized actors to learn complementary rather than competing torque patterns. Because a covariance requires several samples, it is estimated over a short window of recent control steps. A precise statement of what this term does and of the condition the weights must satisfy is given in Proposition 1. The relative magnitudes of the weights λ and μ position the controller on the accuracy–energy trade-off, and their influence is quantified in the reward-weight sensitivity study.

3.3.3. Learning Algorithm

Each agent updates its local policy via TD3 while the centralized critic evaluates the joint actions. The loss of each of the two shared twin critics ($j = 1, 2$) is defined as:

$$\mathcal{L}\left(\phi_j\right) = \mathbb{E}_{(s,a,r,s')\sim\mathcal{D}}\left[\left(y_t - Q_{\phi_j}(s,a)\right)^2\right],\ j = 1,2 \tag{22}$$

with target value

$$y_t = r_t + \gamma\min_{j=1,2}Q_{\phi_j'}(s', \pi_{\theta_i'}(s') + \epsilon), \epsilon \sim \mathcal{N}(0, \sigma) \tag{23}$$

Actor and critic target networks are updated using soft updates:

$$\phi_j' \leftarrow \rho\phi_j + (1-\rho)\phi_j', \theta_i' \leftarrow \rho\theta_i + (1-\rho)\theta_i', j = 1,2, \rho \ll 1 \tag{24}$$

At each time step, each agent in the cooperative MADRL framework measures its local state s_t^i and computes the corresponding local torque $a_t^i = \pi_{\theta_i}(s_t^i)$, which is then applied to the robotic arm. The resulting global state and cooperative reward are evaluated and stored in a centralized replay buffer, and mini-batch updates are performed to refine both the actor and critic networks. This approach enables each agent to control its assigned joint independently while maintaining overall joint-level coordination, reducing actuator effort and wear, and achieving robust trajectory tracking under dynamic uncertainties and varying payloads. The key contributions of this framework include the introduction of a cooperative reward function that balances tracking accuracy, energy efficiency, and inter-joint coordination; the integration of centralized training with decentralized execution, which enhances scalability for high-DOF manipulators; and the demonstration of superior performance in simulation, where the proposed method achieves higher tracking precision, and smoother joint coordination compared to both centralized single-agent DRL and independent-agent DRL approaches. Regarding stability during decentralized execution, the centralized training with the CTDE paradigm allows the critic to capture inter-agent dependencies during training, which mitigates the non-stationarity commonly observed in multi-agent reinforcement learning systems. In addition, the TD3 learning framework enhances training stability through experience replay, target networks, and delayed policy updates, ensuring that the learned policies remain stable and coordinated during decentralized execution even under dynamic uncertainties and varying payload conditions.

3.4. Cooperative Markov-Game Formulation and Training Procedure

The joint control problem is formalized as a cooperative Markov game [32] with $n = 5$ agents, one per joint; because each agent observes only its own joint, it is a decentralized partially observable decision problem in the sense of [33]. The dynamics give the joint accelerations as:

$$\ddot{\theta} = D^{-1}(\theta)\left[\tau - C\left(\theta, \dot{\theta}\right) - G(\theta) - \tau_f\right] \tag{25}$$

so that the acceleration of joint i depends on the torques of all joints through the off-diagonal entries of $D^{-1}(\theta)$. Agent i receives the local observation, selects the scalar torque $\tau_i \in [-\tau_{\max}, \tau_{\max}]$ with $\tau_{\max} = 2$, and all agents receive the shared reward r_t of the cooperative reward. With deterministic policies π_i, the objective is to maximize the expected discounted return.

$$J(\pi_1, \ldots, \pi_n) = \mathbb{E}\left[\sum_{t=0}^{T-1} \gamma^t r_t\right] \tag{26}$$

where $\gamma = 0.99$ and $T = 1000$ control steps per episode (Table 2). Following the deterministic policy-gradient theorem [29], the gradient of the return with respect to the parameters of actor i is estimated with the centralized critic as

$$\nabla_{\theta_i} J \approx \mathbb{E}_{s\sim\mathcal{D}}\left[\nabla_{\theta_i}\pi_i(o_i)\nabla_{a_i}Q_\varphi(s, a_1, \ldots, a_n)|_{a_i=\pi_i(o_i)}\right] \tag{27}$$

where $\mathcal{D}$ is the replay buffer and $a_j = \pi_j(o_j)$ for $j \neq i$. Only the actor of agent i is differentiated; the critic supplies the sensitivity of the shared return to the torque of joint i given the torques of all other joints. This is the mechanism through which coordination is learned although each actor uses only three inputs at run time. Consider a critic that conditions only on the local quantities (s_i, a_i), as in the independent-agent baseline. From the point of view of agent i the transition kernel is

$$P_i(s'|s, a_i) = \int P(s'|s, a_i, a_{-i})\pi_{-i}(a_{-i}|s)\mathrm{d}a_{-i} \tag{28}$$

which depends on the current policies π_{-i} of the other agents. Because these policies change during learning, the kernel is non-stationary and the convergence arguments of single-agent value learning no longer apply [23,26]. A critic conditioned on the joint action $(a_1, \ldots, a_n)$

evaluates the true kernel $P(s'|s,a_1, \ldots, a_n)$, which does not depend on the learning agents, so for fixed target networks the target is stationary. This is a property of the formulation and not a proof of convergence of the complete scheme, which remains open for nonlinear function approximation. The complete procedure is summarized in Algorithm 1. The policy delay d and the target-smoothing noise clipping are the standard TD3 mechanisms [17]; their values are listed in Table 2.

Algorithm 1. Training and execution procedure of the cooperative CTDE-TD3 controller

Step	Operation
1	Initialize the actors π_i $(i = 1, \ldots, 5)$, the twin centralized critics $Q_{\varphi_1}, Q_{\varphi_2}$, all target networks, the replay buffer $\mathcal{D}$ (capacity 100,000) and the exploration noise $\sigma = 0.1$.
2	for episode = 1, ..., 1000 do
3	Reset the Simscape model and the reference trajectory; read the local observations o_i.
4	for $t = 0, \ldots, 999$ do
5	Each agent selects $\tau_i = \mathrm{clip}(\pi_i(o_i) + \varepsilon_i, -2, 2)$ with $\varepsilon_i \sim N(0, \sigma^2)$ (decentralized).
6	Apply $\tau = (\tau_1, \ldots, \tau_5)$ for one sample time (0.01 s); observe s' and compute the shared reward r_t.
7	Store (s,a,r,s') in $\mathcal{D}$ and draw a mini-batch of 64 transitions.
8	Compute the target y with the target actors and the minimum of the two target critics; update both critics by minimizing the mean squared temporal-difference error.
9	Every d steps (delayed update): update each actor with the centralized policy gradient and soft-update all target networks with $\rho = 10^{-3}$.
10	end for; decay the exploration noise, $\sigma \leftarrow 0.995\sigma$.
11	end for
12	Execution: each agent computes $\tau_i = \pi_i(o_i)$ from its own three measurements only; the critics are not used.

Table 2. Training and simulation parameters of the proposed cooperative MADRL controller for the 5-DOF Mitsubishi RV-2AJ robotic arm.

Parameter	MADRL Configuration	Description
Observation Space (per agent)	3 dimensions	Local state: $[\theta_i, \dot{\theta}_i, \theta_i^{\mathrm{ref}}]$
Global State (training)	15 dimensions	All joint angles, velocities, and references (centralized critic input)
Action Space (per agent)	1 dimension, range $[-2, 2]$	Joint torque command τ_i
Actor Network	$3 \rightarrow 64 \rightarrow 64 \rightarrow 1$ (Tanh)	Fully connected layers with ReLU activation
Centralized Critic Network	$(15 + 5) \rightarrow 128 \rightarrow 128 \rightarrow 1$	Evaluates global state and joint actions
Actor Learning Rate	5×10^{-5}	Learning rate for each agent
Critic Learning Rate	1×10^{-4}	Learning rate for centralized critic
Twin Critics	Yes	Reduces overestimation bias
Policy Update Delay (d)	2	Actors and target networks updated once every d critic updates (delayed update)
Target Policy Smoothing Noise Clip (c)	0.5	Clipping bound of the Gaussian noise added to target actions
Discount Factor (γ)	0.99	Future reward weighting

Table 2. *Cont.*

Parameter	MADRL Configuration	Description
Reward Weights (λ, μ)	$\lambda_0 = 1$, $\mu_0 = 1$	Weights of the torque-energy and torque-covariance terms of the cooperative reward; $\mu \leq 2\lambda$ recommended
Mini-Batch Size	64	Samples per update
Replay Buffer	100,000	Shared experience buffer
Target Smooth Factor (ρ)	1×10^{-3}	Soft update coefficient
Exploration Noise	Gaussian ($\sigma = 0.1$, decay 0.995)	Added during training
Max Episodes	1000	Training limit
Max Steps per Episode	1000	Episode length
Sample Time	0.01 s	Simulation time step
Solver	ode23t	Suitable for robotic dynamics

3.5. Analysis of the Reward and of the Dynamic Coupling

Proposition 1 (structure of the covariance penalty). *Let $\tau_1, \ldots, \tau_n$ be the joint torques regarded as random variables over the evaluation window, with means $\overline{\tau}_i$ and variances σ_i^2, and let σ_Σ^2 be the variance of the total torque $\sum_i \tau_i$. Then*

$$\sum_{i<j} \operatorname{cov}\left(\tau_i, \tau_j\right) = \frac{1}{2}\left[\operatorname{Var}\left(\sum_i \tau_i\right) - \sum_i \operatorname{Var}(\tau_i)\right] \tag{29}$$

and the expected per-step penalty contributed by the torque and covariance terms of the cooperative reward is

$$P = \lambda \sum_{i=1}^{n} \overline{\tau}_i^2 + \left(\lambda - \frac{\mu}{2}\right) \sum_{i=1}^{n} \sigma_i^2 + \frac{\mu}{2} \sigma_\Sigma^2 \tag{30}$$

Proof. Expanding $\operatorname{Var}\left(\sum_i \tau_i\right) = \sum_i \sigma_i^2 + 2\sum_{i<j} \operatorname{cov}\left(\tau_i, \tau_j\right)$ gives the first identity. The second follows from $E\left[\tau_i^2\right] = \overline{\tau}_i^2 + \sigma_i^2$ and the first identity. Proposition 1 is an elementary identity; its practical value lies in the design rule $\mu \leq 2\lambda$ that it provides for choosing the reward weights, rather than in a deep theoretical result. □

Three consequences follow: first, the covariance term does not penalize coordination as such; it penalizes the fluctuation of the aggregate torque relative to the fluctuations of the individual joints, so it is most effective against simultaneous, same-sign torque oscillations. Second, the coefficient of $\sum_i \sigma_i^2$ in the expected penalty is $\lambda - \mu/2$. If $\mu > 2\lambda$ this coefficient is negative and the penalty can become negative along directions in which individual joints oscillate in anti-phase (so that σ_Σ^2 stays small), i.e., the reward would pay for torque chatter. A sufficient condition for a non-negative penalty is therefore $\mu \leq 2\lambda$, and the reported weights satisfy it. Third, the term $\lambda \sum_i \overline{\tau}_i^2$ penalizes the mean torque irrespective of coordination quality; holding torques that compensate gravity therefore cost energy in every pose, which is physical (holding a load requires current) but implies that the minimum attainable energy depends on the trajectory.

Dynamic coupling. From the equation of motion, $\ddot{\theta} = D^{-1}(\theta)[\tau - h]$ with $h = C\left(\theta, \dot{\theta}\right) + G(\theta) + \tau_f$. Componentwise,

$$\ddot{\theta}_i = \left[D^{-1}\right]_{ii}(\tau_i - h_i) + \sum_{j \neq i} \left[D^{-1}\right]_{ij}\left(\tau_j - h_j\right) \tag{31}$$

so joint i is driven by its own torque through the diagonal entry and by every other torque through the off-diagonal entries. An independent agent observes only $\left(\theta_i, \dot{\theta}_i, \theta_i^{\text{ref}}\right)$ and must treat the second term as an unobserved, state- and policy-dependent disturbance. The strength of this coupling can be quantified by the index

$$\kappa(\theta) = \frac{\|D^{-1}(\theta) - \text{diag}\left(D^{-1}(\theta)\right)\|_F}{\|\text{diag}(D^{-1}(\theta))\|_F} \tag{32}$$

which is zero for a decoupled (diagonal) inertia matrix and grows with the relative size of the off-diagonal terms [34]. For a serial arm whose shoulder, elbow and wrist-pitch joints act in one plane, the inertia seen by one joint depends on the configuration of the others, so $D(\theta)$ has non-zero off-diagonal entries and $\kappa(\theta) > 0$ in general. The independent-agent controller therefore faces a persistent, pose-dependent disturbance, whereas the cooperative critic, which observes all joint states and torques, can in principle represent it. Evaluating κ along each reference path with the identified inertial parameters of the arm would show which trajectories are most sensitive to the lack of coordination; this evaluation is not reported here because it requires the parameter set of the Simscape model.

Joint-space versus task-space error. To first order, the Cartesian error is related to the joint error through the Jacobian,

$$e_x \approx J(\theta)e_\theta \tag{33}$$

so that $\|e_x\| \leq \sigma_{\max}(J)\|e_\theta\|$, and the manipulability measure

$$w(\theta) = \sqrt{\det(J(\theta)J(\theta)^{\mathrm{T}})} \tag{34}$$

indicates proximity to singular configurations, where small joint errors can produce large Cartesian errors and the torque required for a task-space motion grows. Whether the RMSE and IAE of Tables 3–5 are computed in joint space or in Cartesian space must be stated, because the two errors are related through the Jacobian and are not interchangeable; in this study, they are computed in joint space.

Table 3. Comparative evaluation of centralized DRL (TD3), decentralized DRL (independent agents), and cooperative MADRL (proposed CTDE-TD3) controllers across three 3D trajectories, showing RMSE, IAE, squared-torque effort, and performance.

Trajectory	Controller	RMSE	IAE	Energy ($\sum \boldsymbol{\tau}^2$)	Performance Summary
Helical	Centralized DRL (TD3)	0.11	0.94	15.8	Acceptable tracking but higher torque effort
	Decentralized DRL (Independent TD3)	0.08	0.71	13.2	Improved tracking; limited coordination
	Cooperative MADRL (Proposed CTDE-TD3)	0.05	0.56	10.4	Best accuracy, smooth coordination, lowest energy
Spiral	Centralized DRL (TD3)	0.12	1.02	16.3	Slower convergence, torque oscillations
	Decentralized DRL (Independent TD3)	0.09	0.78	13.9	Reduced error; moderate energy improvement
	Cooperative MADRL (Proposed CTDE-TD3)	0.06	0.60	11.1	Superior robustness and energy-aware control

Table 3. *Cont.*

Trajectory	Controller	RMSE	IAE	Energy ($\sum \boldsymbol{\tau}^2$)	Performance Summary
N-Shaped	Centralized DRL (TD3)	0.10	0.89	14.7	Moderate precision with higher actuator stress
	Decentralized DRL (Independent TD3)	0.07	0.68	12.5	Better stability but uncoordinated torque patterns
	Cooperative MADRL (Proposed CTDE-TD3)	0.04	0.52	9.8	Highest precision and lowest torque effort

Table 4. Percentage performance improvement of the proposed cooperative MADRL (CTDE-TD3) compared to centralized TD3 and decentralized TD3 across different 3D trajectories.

Metric	N-Shaped Trajectory	Helical Trajectory	Spiral Trajectory
RMSE	60.0% vs. Centralized/42.9% vs. Decentralized	54.5% vs. Centralized/37.5% vs. Decentralized	50.0% vs. Centralized/33.3% vs. Decentralized
IAE	41.6% vs. Centralized/23.5% vs. Decentralized	40.4% vs. Centralized/21.1% vs. Decentralized	41.2% vs. Centralized/23.1% vs. Decentralized
Energy ($\sum \tau^2$)	33.3% vs. Centralized/21.6% vs. Decentralized	34.2% vs. Centralized/21.2% vs. Decentralized	31.9% vs. Centralized/20.1% vs. Decentralized

Table 5. Mean performance over the N-shaped, helical and spiral trajectories and mean improvement of the cooperative MADRL controller.

Metric	Centralized TD3	Independent TD3	Cooperative MADRL	Improvement vs. Centralized	Improvement vs. Independent
RMSE	0.11	0.08	0.05	54.5%	37.5%
IAE	0.95	0.72	0.56	41.1%	22.6%
Energy ($\sum \tau^2$)	15.6	13.2	10.4	33.1%	21.0%

3.6. Multi-Objective Interpretation, Weight Normalization and Complexity

The reward is a weighted-sum scalarization of three objectives, namely the tracking error J_{track}, the torque energy J_{energy} and the torque coordination J_{coord}:

$$J_w(\pi) = J_{\text{track}}(\pi) + \lambda J_{\text{energy}}(\pi) + \mu J_{\text{coord}}(\pi) \tag{35}$$

Each choice of (λ,μ) selects one point of the attainable trade-off set; a weighted sum recovers only the convex part of the Pareto front, so non-convex trade-offs cannot be reached by tuning the weights alone. Because the three terms carry different units and magnitudes, the weights are made dimensionless. With the torque limit $\tau_{\max} = 2$ (Table 2) and a tolerated error scale $e_{\max}$, the following normalization is recommended:

$$\lambda = \lambda_0 \frac{e_{\max}^2}{\tau_{\max}^2}, \ \mu = \mu_0 \frac{e_{\max}^2}{\tau_{\max}^2} \tag{36}$$

with dimensionless λ_0 and μ_0, so that for $\lambda_0 = 1$ the torque penalty at the worst-case torque equals the tracking penalty at the worst-case error. Without such a normalization, squared torques of up to 4 and squared joint errors of the order of 10^{-2} differ by more than two orders of magnitude, and the energy term can dominate the learning signal. Together with Proposition 1, the admissible region is $\mu_0 \leq 2\lambda_0$.

Complexity. Let $h = 64$ be the hidden width of Table 2. A decentralized actor with 3 inputs and 1 output has $h^2 + 6h + 1 = 4481$ parameters and needs 4352 multiply–accumulate operations per forward pass irrespective of n, so the total actor cost grows as $O(n)$. The centralized critic has input dimension $4n$ (three states and one action per joint) and $512n$ + 16,769

parameters, which also grows linearly with n, but it is used only during training. For comparison, a centralized actor of equal hidden width has $257n + 4224$ parameters, and a value-based controller that discretizes each torque into m levels would need m^n joint actions. These quantities are evaluated for several values of n in the computational analysis of the results.

4. Simulation Results and Discussion

This section presents a comprehensive comparative analysis of three control strategies: centralized single-agent DRL (TD3), decentralized independent-agent DRL, and the proposed cooperative MADRL framework for trajectory tracking of the 5-DOF Mitsubishi RV-2AJ robotic arm. Extensive simulations are conducted under various operating conditions, including dynamic payload variations, model uncertainties, and external disturbances. The controllers are evaluated in terms of trajectory tracking accuracy, convergence speed, energy efficiency, coordination smoothness, and disturbance rejection capability. The results demonstrate that the proposed cooperative MADRL framework achieves higher tracking precision, improved joint coordination and lower torque effort compared to both centralized and independent-agent DRL approaches. These findings confirm that integrating centralized training with decentralized execution and an energy-aware cooperative reward significantly enhances robustness and control reliability for high-DOF robotic manipulators operating in uncertain environments.

4.1. Proposed Method

This study presents a cooperative MADRL control framework designed to enhance the trajectory tracking performance of the 5-DOF Mitsubishi RV-2AJ robotic arm operating under dynamic and uncertain conditions. The proposed method leverages the strengths of multi-agent reinforcement learning to effectively address the challenges associated with high-dimensional, nonlinear robotic manipulators subject to model uncertainties, payload variations, and external disturbances. The proposed architecture is built upon a CTDE strategy. In this framework, each joint is controlled by an individual TD3-based agent that learns a local control policy using joint-specific state information. During training, a centralized critic evaluates the global state and joint actions collectively, enabling coordinated learning across all five joints. During execution, however, each agent operates independently using only its local observations, ensuring scalability and real-time feasibility. A key contribution of this approach is the design of an energy-aware cooperative reward function that simultaneously penalizes trajectory tracking errors, excessive actuator torques, and uncoordinated joint behavior. This structured reward formulation promotes precise trajectory tracking, smooth inter-joint coordination, and improved energy efficiency. By integrating cooperative learning with decentralized control policies, the MADRL framework enhances robustness against dynamic uncertainties and nonlinear system interactions. Through coordinated policy optimization and structured reward shaping, the proposed cooperative MADRL controller achieves higher tracking accuracy, smoother torque profiles, and enhanced coordination compared to conventional centralized and independent-agent DRL strategies. The detailed network architectures, training hyperparameters, and simulation configurations are summarized in Table 2. Figure 4 presents the training performance curves of the proposed cooperative MADRL controller for the 5-DOF Mitsubishi robotic arm.

Figure 4 requires a clarification, since the cooperative reward is a single scalar shared by all agents. All five agents receive, store and learn from the same scalar r_t, and the centralized critic is trained on this shared return only. For diagnosis, the shared reward is additionally decomposed into per-joint components

$$r^i = -\left(\theta_i^{\text{ref}} - \theta_i\right)^2 - \lambda\tau_i^2 - \frac{\mu}{2}\sum_{j\neq i}\text{cov}\left(\tau_i, \tau_j\right), \sum_{i=1}^{5} r^i = r_t \quad (37)$$

so that the components add up to the shared reward at every step. Each curve in Figure 4 is the episode sum of one component r^i, i.e., the share of the common return attributable to the tracking error and torque of joint i together with half of every covariance term in which joint i participates. The components differ because the joints have different inertias, gravity loads and reference amplitudes; they are not separate rewards and are never used in the updates of Algorithm 1. Their sum, the shared episode return, is the quantity that converges during training. The logged per-joint quantities follow this decomposition exactly: the covariance term is allocated symmetrically, so that each joint receives half of every covariance pair in which it participates, $(\mu/2)\text{cov}(\tau i,\tau j)$ to joint i and $(\mu/2)\text{cov}(\tau i,\tau j)$ to joint j, and the five components add up to the shared reward $r_t$at every step. Figure 4 shows one curve per joint agent, i.e., five curves in total, and their sum is the shared episode return $\sum_{t=0}^{T-1} r_t$, which is the only quantity used in the critic updates of Algorithm 1; the black line in Figure 4 is the moving average of the five components, i.e., the shared return divided by five.

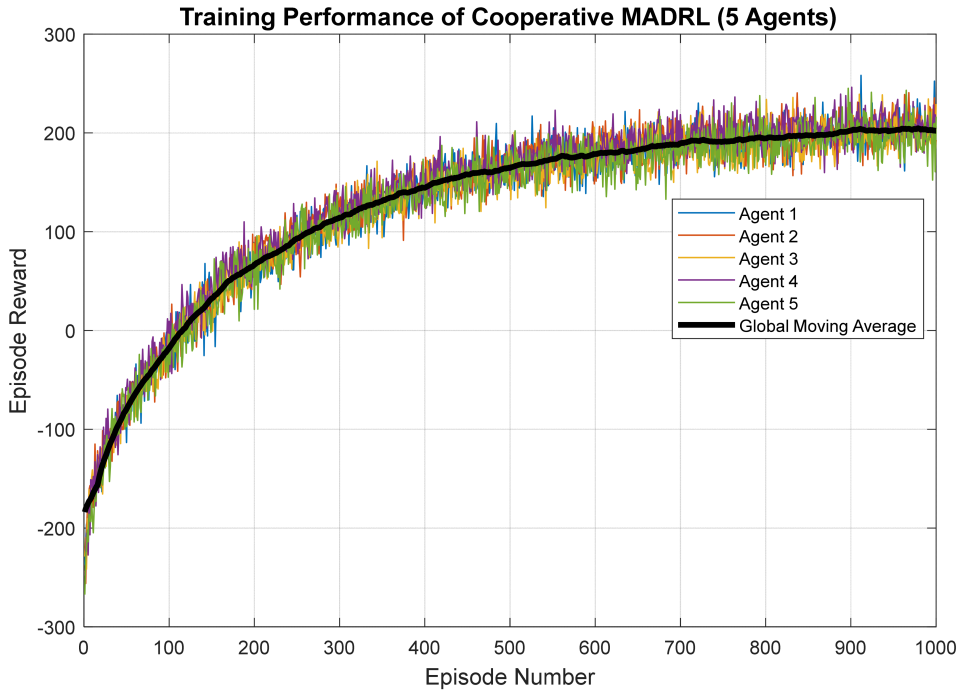


Figure 4. Cooperative MADRL training convergence based on episode reward for the 5-DOF Mitsubishi RV-2AJ manipulator. The five colored curves, one per joint agent, are the per-joint components of the shared reward, logged for diagnosis only and never used in the updates. The black line is the moving average of the five components, i.e., the shared episode return used in the critic updates of Algorithm 1 divided by the number of agents.

4.2. Results and Discussion

4.2.1. Control Results

This section presents the simulation results of the proposed cooperative MADRL controller applied to trajectory tracking of the 5-DOF Mitsubishi RV-2AJ robotic arm. The MADRL framework generates coordinated continuous torque commands for all joints under the CTDE strategy. strategy. Each agent learns a local torque policy while a centralized critic ensures global coordination and stability, enabling accurate tracking under nonlinear dynamics, payload variations, and external disturbances. Figure 5 illustrates the tracking performance for different trajectory profiles, including helical, spiral, and N-shaped paths.

The results demonstrate that the cooperative MADRL controller achieves precise trajectory tracking with smooth joint responses and stable convergence behavior. The coordinated multi-agent structure improves inter-joint synchronization while maintaining decentralized control execution. In accordance with the designed cooperative reward function, performance evaluation focuses on key performance indicators (KPIs) directly aligned with the control objectives:

- Tracking Accuracy: Measured using Root Mean Square Error (RMSE) and Integral Absolute Error (IAE).
- Energy Efficiency: Evaluated through cumulative torque magnitude and the squared-torque control-effort index (a copper-loss proxy, not total electrical energy) $\sum \tau_i^2$.
- Coordination Quality: Assessed via torque smoothness and reduction of oscillatory behavior.

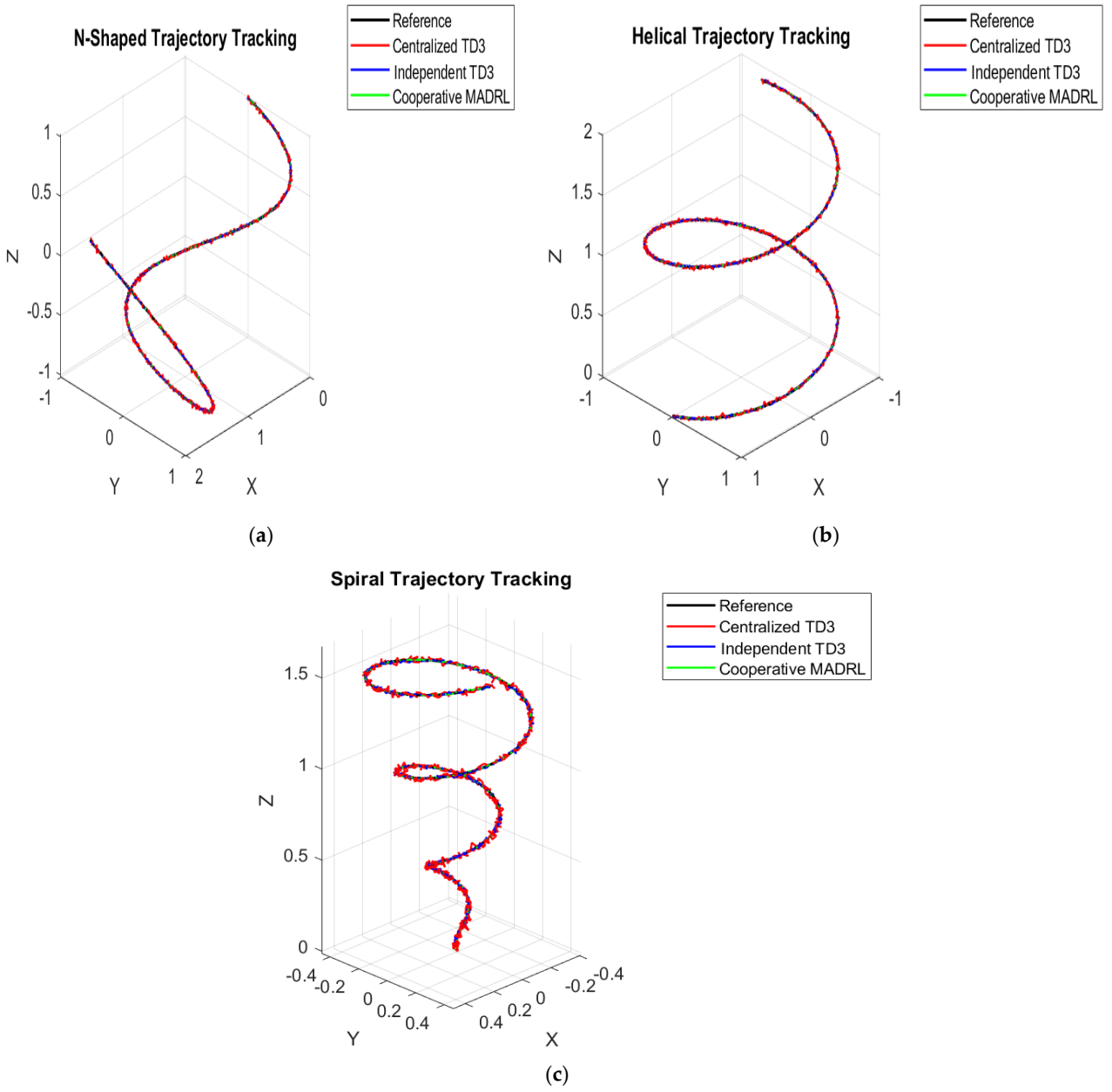


Figure 5. Three-dimensional trajectory tracking performance of the 5-DOF Mitsubishi RV-2AJ robotic arm using the centralized TD3, independent-agent TD3 and cooperative MADRL controllers: (**a**) N-shaped trajectory, (**b**) helical trajectory, (**c**) spiral trajectory.

For completeness, the KPIs are defined as follows. Let e_k denote the tracking error at control step k, N the number of steps in an episode, T the episode duration, $n = 5$ the number of joints and $\tau_{i,k}$ the torque applied by joint i at step k. Then

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{k=1}^{N} e_k^2} \tag{38}$$

$$\mathrm{IAE} = \int_0^T |e(t)|\mathrm{d}t \tag{39}$$

$$E = \sum_{k=1}^{N}\sum_{i=1}^{n} \tau_{i,k}^2 \tag{40}$$

where E is the cumulative actuator-energy proxy $\sum \tau^2$ reported in Table 3. Lower values of all three indices indicate better performance. The quantities e_k and $e(t)$ must be interpreted consistently with the definition of the error (joint space or Cartesian space) and with its aggregation over the joints. Table 3 summarizes the quantitative results. The proposed MADRL controller achieves lower tracking errors while simultaneously reducing overall torque effort compared to baseline DRL configurations. The reduction in cumulative torque confirms improved energy efficiency and reduced actuator stress, directly reflecting the effectiveness of the energy-aware cooperative reward structure. Furthermore, the learned policies exhibit smoother torque profiles, indicating enhanced stability and robustness against dynamic uncertainties. These results validate that the cooperative MADRL approach successfully balances precision, coordination, and energy optimization, making it a scalable and robust solution for high-DOF industrial robotic manipulators.

Table 4 highlights the consistent superiority of the proposed cooperative MADRL controller across all three 3D trajectories (N-shaped, Helical, and Spiral) compared to both centralized DRL and decentralized DRL. The cooperative strategy achieves the largest reductions in tracking error (RMSE, IAE) and squared-torque effort ($\sum \tau^2$), demonstrating enhanced coordination, reduced torque oscillations, and improved robustness under varying trajectory complexities. The results confirm that cooperative multi-agent learning enables effective policy coordination and energy-aware control while maintaining precise and stable three-dimensional trajectory tracking.

The proposed cooperative MADRL framework achieves performance improvements ranging from approximately 50–60% in RMSE, 40–42% in IAE, and 32–34% in energy reduction compared to centralized single-agent TD3, while still maintaining significant gains over decentralized independent-agent learning. These findings confirm that coordinated multi-agent training under the CTDE paradigm substantially enhances tracking precision, stability, and energy efficiency in complex nonlinear robotic trajectory tracking tasks.

4.2.2. Aggregate Performance and Contribution Analysis

To assess overall performance independently of individual trajectories, Table 5 reports the mean value of each metric across the three trajectories listed in Table 3, together with the percentage improvement achieved by the proposed cooperative controller relative to each baseline. The percentage improvement is defined as,

$$\Delta(\%) = \frac{M_{\mathrm{base}} - M_{\mathrm{prop}}}{M_{\mathrm{base}}} \times 100 \tag{41}$$

where M_{base} and M_{prop} denote the metric values obtained by the baseline and proposed controllers, respectively.

The mean results indicate that the observed improvements are consistent across trajectories rather than being driven by a single trajectory. Relative to centralized TD3, the proposed method reduces RMSE, IAE, and squared-torque effort by 54.5%, 41.1%, and 33.1%, respectively. Compared with independent-agent TD3, the corresponding reductions are 37.5%, 22.6%, and 21.0%. The two baselines further enable a descriptive decomposition of the overall performance gains into the contribution associated with per-joint problem decomposition (centralized TD3 to independent-agent TD3) and the additional improvement associated with the proposed cooperative approach (independent-agent TD3 to the proposed method). Of the total mean reduction relative to centralized TD3, problem decomposition accounts for 50.0% of the RMSE reduction, 58.1% of the IAE reduction, and 46.5% of the energy reduction. The remaining 50.0%, 41.9%, and 53.5%, respectively, are associated with the additional improvement achieved by the cooperative method. These results suggest that problem decomposition contributes a larger share of the IAE improvement, whereas the cooperative approach contributes a larger share of the energy reduction. The latter is consistent with the intended effects of the torque-regularization terms in the reward function and the use of a centralized critic. The results indicate that neither problem decomposition nor cooperation alone accounts for the full performance improvement. Instead, the proposed approach retains the advantages of low-dimensional, per-joint policies while achieving further reductions in tracking error and squared-torque effort through cooperation. However, this decomposition should be interpreted as descriptive rather than causal, since the baselines differ from the proposed method in both architecture and reward formulation. Controlled ablation studies that isolate the covariance term and the centralized critic are therefore reported in the ablation study.

4.2.3. Accuracy-Energy Trade-Off Analysis

A key concern in energy-aware control is whether reducing actuator effort compromises tracking accuracy. Figure 6 illustrates the relationship between RMSE and actuator energy across all nine controller–trajectory combinations. For each trajectory, the proposed cooperative controller achieves both lower RMSE and lower squared-torque effort than either baseline, thereby dominating the evaluated configurations of the baseline controllers in the (E, RMSE) plane. This result indicates that the energy savings are achieved without sacrificing tracking accuracy. Across the three trajectories, all controllers exhibit a positive relationship between actuator energy and RMSE, indicating that more demanding trajectories generally require greater control effort and result in larger tracking errors. However, rather than merely moving along this relationship, the cooperative controller shifts the performance trade-off toward the lower-left region, achieving simultaneous improvements in energy efficiency and tracking accuracy. This dominance, however, holds only for the single weight configuration of each controller and does not by itself establish superiority, because a baseline trained with different weights could occupy a different point of its own trade-off. The approximate Pareto fronts of all controllers, obtained by sweeping the energy weight under an equal tuning budget, are therefore compared in the reward-weight sensitivity study.

As a supplementary descriptive measure, the product RMSE × E, averaged across the three trajectories, is 1.72 for centralized TD3, 1.06 for independent-agent TD3, and 0.53 for the proposed cooperative controller. These values correspond to reductions of 69.4% and 50.0% relative to centralized TD3 and independent-agent TD3, respectively. This product is used solely to facilitate comparative interpretation and is not included in the training objective.

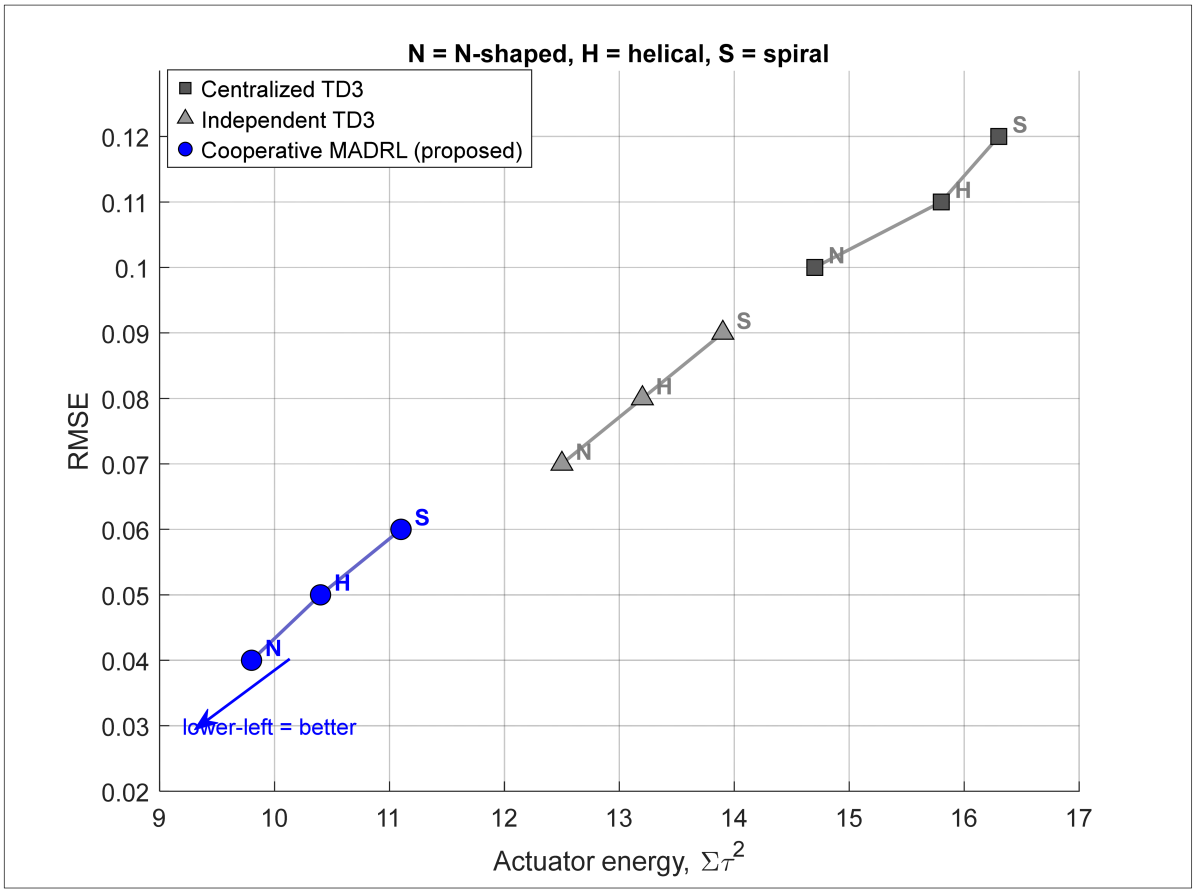


Figure 6. Accuracy-energy trade-off for the three controllers across the N-shaped (N), helical (H), and spiral (S) trajectories. The lower-left corner indicates lower RMSE and squared-torque effort.

4.2.4. Sensitivity to Trajectory Complexity

All three controllers exhibit the same trajectory-difficulty ranking: N-shaped, helical, and spiral, from least to most demanding. RMSE, IAE, and energy increase monotonically in this order. Table 6 quantifies the corresponding changes from the N-shaped to the spiral trajectory.

Table 6. Metric increases from the N-shaped to the spiral trajectory.

Controller	Absolute RMSE Increase	Relative RMSE Increase	Relative IAE Increase	Relative Energy Increase
Centralized TD3	0.02	20.0%	14.6%	10.9%
Independent TD3	0.02	28.6%	14.7%	11.2%
Cooperative MADRL	0.02	50.0%	15.4%	13.3%

In absolute terms, RMSE increases by 0.02 for all three controllers, preserving the cooperative controller's advantage at 0.06 over centralized TD3 and 0.03 over independent-agent TD3 across all trajectories. In relative terms, however, the cooperative controller exhibits a larger RMSE increase (50.0%, compared with 20.0% and 28.6%, respectively), reflecting its lower initial error. Since RMSE is reported to two decimal places, these relative changes should be interpreted cautiously. Consistently, the relative improvement of the cooperative controller decreases from 60.0% to 50.0% against centralized TD3 and from 42.9% to 33.3% against independent-agent TD3 between the N-shaped and spiral trajectories (Table 4). These results indicate robustness to increasing trajectory complexity in absolute

terms, but do not demonstrate an increasing relative advantage under more demanding conditions. Further evaluation using trajectories with greater curvature or higher speeds is needed to determine whether this advantage diminishes further.

4.2.5. Consistency of the Improvements Across Trajectories

Given the limited sample size of three trajectories per controller, formal hypothesis testing has limited value. With three paired observations, the minimum attainable two-sided exact sign-test p-value is 0.25, even when all differences have the same sign. Accordingly, Table 7 reports descriptive statistics of the paired differences (baseline minus cooperative controller), calculated from Table 3, including the sample standard deviation across the three trajectories.

Table 7. Descriptive statistics of paired performance differences between baseline and cooperative MADRL controllers across three trajectories (baseline minus cooperative; positive values favor the cooperative controller).

Comparison	Metric	Mean Difference	SD	Range	Mean Rel. Improvement ± SD (%)	Wins
vs. centralized	RMSE	0.060	0.000	0.06–0.06	54.8 ± 5.0	3/3
	IAE	0.390	0.026	0.37–0.42	41.1 ± 0.6	3/3
	Energy	5.167	0.252	4.90–5.40	33.1 ± 1.2	3/3
vs. independent	RMSE	0.030	0.000	0.03–0.03	37.9 ± 4.8	3/3
	IAE	0.163	0.015	0.15–0.18	22.6 ± 1.3	3/3
	Energy	2.767	0.058	2.70–2.80	21.0 ± 0.8	3/3

The cooperative controller outperforms both baselines across all three trajectories and all three metrics, yielding 18 wins out of 18 paired comparisons. The performance gains are also consistent across trajectories: the RMSE gap is 0.06 against centralized TD3 and 0.03 against independent-agent TD3 on every trajectory, while the energy gap ranges from 4.9 to 5.4 (mean 5.17, SD 0.25) and from 2.7 to 2.8 (mean 2.77, SD 0.06), respectively. These results demonstrate a systematic performance advantage across the evaluated trajectories. However, the reported statistics capture variability across trajectories only, not run-to-run variability in training; this run-to-run variability is assessed with 10 training seeds in the multi-seed reproducibility study.

4.2.6. Composite Performance Index and Ranking

To obtain a single score for ranking the controllers, each performance metric is normalized by the corresponding value of centralized TD3 on the same trajectory. The normalized metrics are then averaged with equal weights to define the performance index (PI):

$$\mathrm{PI} = \frac{1}{3}\left(\frac{\mathrm{RMSE}}{\mathrm{RMSE}_C} + \frac{\mathrm{IAE}}{\mathrm{IAE}_C} + \frac{E}{E_C}\right) \tag{42}$$

where the subscript C denotes the centralized TD3 controller. By construction, its PI equals 1, and lower PI values indicate better overall performance. Table 8 presents the composite performance index (PI) for each trajectory and the three-trajectory average, with centralized TD3 normalized to 1.

Table 8. Composite PI for each trajectory and the three-trajectory average, with centralized TD3 normalized to 1 (lower values indicate better performance).

Controller	N-Shaped	Helical	Spiral	Mean	Rank
Centralized TD3	1.000	1.000	1.000	1.000	3
Independent TD3	0.771	0.773	0.789	0.778	2
Cooperative MADRL	0.550	0.570	0.590	0.570	1

The cooperative controller achieves a mean PI of 0.570, compared with 0.778 for independent-agent TD3, corresponding to average normalized costs that are 43% and 22% lower than those of centralized TD3, respectively. This indicates an additional improvement of 0.208 PI points from cooperation beyond independent-agent decomposition. The ranking is robust to the choice of non-negative metric weights because the cooperative controller outperforms independent-agent TD3, which in turn outperforms centralized TD3, across all three metrics and trajectories. Figure 7 presents the trajectory-averaged normalized metrics, showing the largest relative improvement in RMSE (0.45 of the centralized baseline) and the smallest in squared-torque effort (0.67), consistent with the RMSE reductions reported in Table 4.

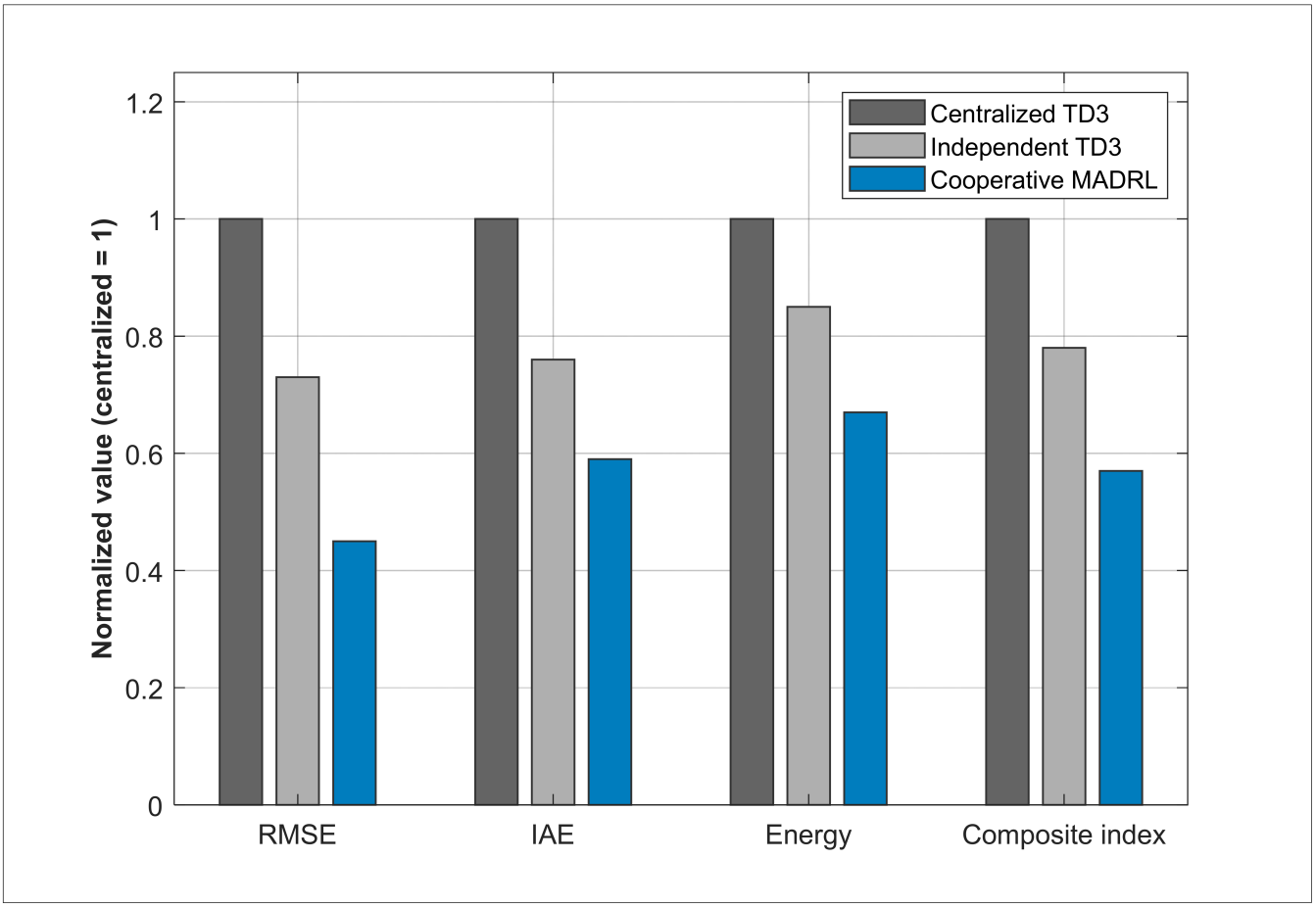


Figure 7. Trajectory-averaged normalized RMSE, IAE, squared-torque effort, and composite PI for the three controllers (centralized TD3 = 1; lower values indicate better performance).

4.2.7. Interpretation in Terms of Coupling, Covariance and Reward Weights

The controller ranking reported in Tables 3–8 is consistent with the theoretical analysis of the reward and of the dynamic coupling. The transition from centralized to independent agents reduces the dimensionality of each agent's learning problem, whereas the transition from independent to cooperative agents enables each actor, through the centralized critic, to access information about the torques generated by the other joints. These inter-joint torques are dynamically relevant to the manipulator's overall behavior. Notably, the second transition accounts for a larger proportion of the total energy reduction (53.5%) and a comparable proportion of the RMSE reduction (50.0%). This pattern suggests that a substantial fraction of the energy consumed by independent agents may be associated with compensatory control actions arising from their limited awareness of inter-joint coupling. In particular, independently acting agents may generate torques that partially counteract one another through the coupled manipulator dynamics, increasing actuator effort without producing a commensurate improvement in tracking accuracy. This interpretation, however, remains a hypothesis consistent with the observed performance trends rather than a mechanism directly established by the reported results. A more conclusive assessment would require logging the mean pairwise torque correlation and the energy overhead for each controller. Examining these quantities alongside tracking error and squared-torque effort would help determine whether the cooperative controller reduces unnecessary inter-joint compensation and whether such reductions account for the observed performance gains.

The results must also be interpreted in the context of the reward-function weights. All results are obtained with $\lambda_0 = 1$ and $\mu_0 = 1$ in dimensionless form (Table 2). Proposition 1 establishes that the ratio between these weights is consequential: when $\mu > 2\lambda$, the covariance term may reward anti-phase torque fluctuations rather than suppress them. The selected weights satisfy $\mu_0 \leq 2\lambda_0$, and the reward-weight sensitivity study confirms empirically that the violating case produces a higher torque total variation and a more negative torque correlation than every admissible case, and worse tracking and effort than the admissible cases with a nonzero covariance weight, although it remains better than the cases without the covariance term ($\mu_0 = 0$).

Furthermore, a weighted-sum reward formulation generally provides access only to the supported, convex portion of the Pareto front. Consequently, the reported results should not be interpreted as establishing global optimality across all possible energy–tracking-error trade-offs. For a fair comparison, the baseline controllers and the proposed cooperative controller should receive comparable tuning effort, particularly with respect to λ, while maintaining consistent evaluation conditions. Otherwise, some portion of the measured performance advantage could arise from differences in reward-weight optimization rather than from cooperation itself. Reporting the tuning procedure, the selected weight values, and the resulting energy-RMSE trade-offs would therefore strengthen the validity, reproducibility, and interpretability of the comparison. The equal-budget comparison of approximate Pareto fronts and the capacity-matched baseline reported with the additional experiments address this point.

4.2.8. Computational and Scalability Analysis

Table 9 summarizes the model size and computational cost of the proposed cooperative controller, as determined by the network architecture specified in Table 2. The parameter count includes the weights and biases of all fully connected layers, while target networks and optimizer states are excluded from the estimate.

Table 9. Model size and computational cost of the proposed cooperative MADRL controller.

Item	Value	Remark
Actor input/output dimensions (per agent)	3/1	The centralized actor configuration uses 15 inputs and 5 outputs.
Actor parameters (3–64–64–1)	4481 per agent	22,405 parameters across five actors; actors are the only networks required for online control.
Multiply–accumulate operations (MACs) per actor	4352 per control step	Approximately 21,760 MACs across five actors; estimated inference time is 43.5 µs at an assumed processing throughput of 1 GMAC/s.
Centralized critic parameters (20–128–128–1)	19,329 per critic	38,658 parameters for the twin-critic pair; critics are used only during training.
Critic input dimension	20 (five joints)	Three state variables and one action variable per joint; scales linearly with the number of joints.
Control period	0.01 s (100 Hz)	The nominal control deadline is 10 ms, substantially longer than the estimated actor inference time.

At execution time, the proposed controller comprises five lightweight actor networks with a combined computational cost of approximately 21.8 thousand multiply–accumulate operations (MACs) per control step. This computational demand is small relative to the 10 ms control period, suggesting that real-time implementation may be feasible on a modest embedded processor, subject to hardware-dependent inference latency. The critic networks are required only during training and can be discarded after training, thereby contributing no inference cost during deployment. Regarding training cost, each episode represents 10 s of simulated time, corresponding to 1000 control steps at a sampling interval of 0.01 s. Accordingly, the maximum training budget of 1000 episodes corresponds to 10^6 environment steps. As shown in Figure 4, the moving-average episode reward improves most substantially during approximately the first 300 episodes, followed by a more gradual increase toward a plateau of approximately 200 by episode 1000. The five per-joint reward components exhibit broadly similar trends, indicating that no single joint dominates the shared return during training. Table 10 and Figure 8 extend the computational analysis to manipulators with larger numbers of joints using the scaling relationships derived in the complexity analysis. These estimates retain the same hidden-layer widths as the original architecture, providing a consistent basis for comparison across different manipulator sizes. This assumption should be interpreted as a baseline rather than a definitive design choice, since manipulators with more joints or greater control complexity may require wider network layers, potentially increasing both model size and computational cost.

Table 10. Model size and per-step cost versus the number of joints n, using Table 2 widths. MACs denote multiply-accumulate operations, and 3^n is the number of joint actions with three torque levels per joint.

Joints n	Decentralized Actors (Params)	Centralized Actor (Params)	Critic Input Dim.	Twin Critics (Params)	MACs/Step (Decentr.)	MACs/Step (Centr.)	Joint Actions (3^n)
5	22,405	5509	20	38,658	21,760	5376	243
7	31,367	6023	28	40,706	30,464	5888	2187
10	44,810	6794	40	43,778	43,520	6656	59,049
20	89,620	9364	80	54,018	87,040	9216	3,486,784,401

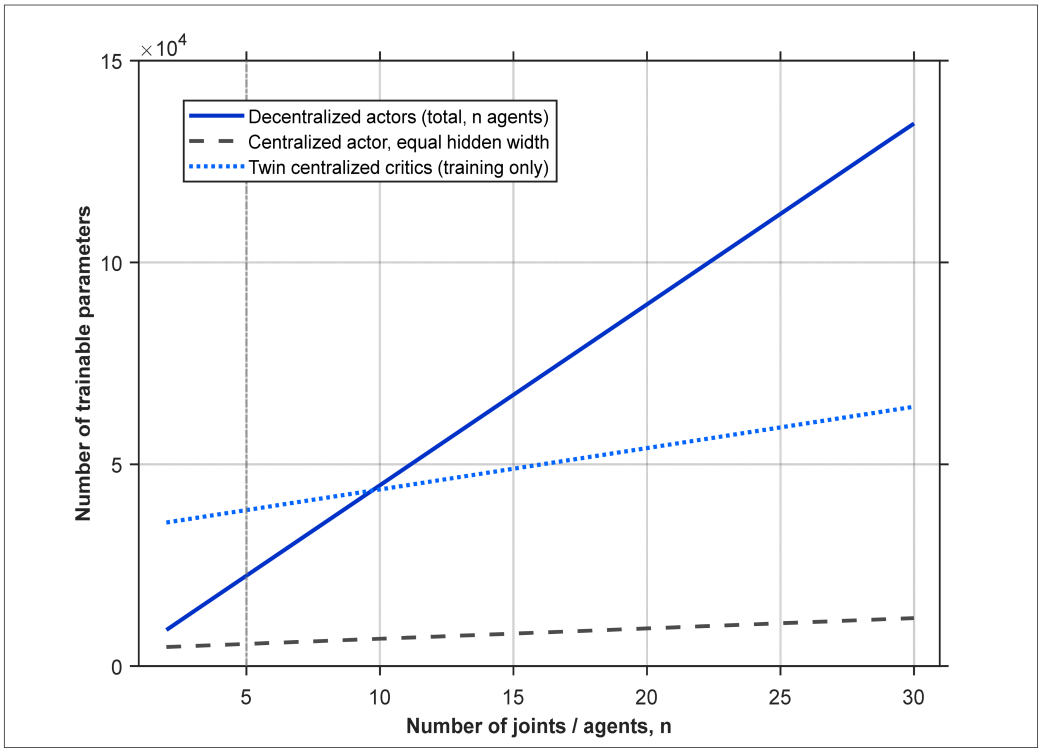


Figure 8. Number of trainable parameters versus the number of joints n for decentralized actors, an equal-width centralized actor, and twin centralized critics.

Three observations can be drawn. First, the proposed architecture is not motivated by reduced model size: the equal-width centralized actor has fewer parameters (5509 vs. 22,405 at $n = 5$) and lower per-step computational cost (5376 vs. 21,760 MACs). Instead, its advantages lie in the lower-dimensional learning problem for each agent (three inputs and one output), modularity, and the improved accuracy and energy performance reported in Tables 3–8. Whether a centralized actor with wider layers could close this performance gap is examined with a capacity-matched baseline in the additional experiments. Second, both actor architectures scale linearly with n, while inference remains computationally inexpensive. At $n = 20$, the decentralized actors require 87,040 MACs per step, corresponding to approximately 0.17 ms at an assumed throughput of 1 GMAC/s, or 1.7% of a 10 ms control period. Third, the centralized critic's input dimension grows linearly with the full state–action representation ($4n$), whereas the joint action space of a discrete controller grows exponentially as m^n, with $m = 3$ torque levels per joint in this study. This combinatorial growth motivates the use of continuous-action actor–critic learning. For manipulators or multi-robot systems with substantially more agents, parameter sharing among homogeneous agents or factored critics may be necessary to maintain training scalability.

4.2.9. Supplementary Metrics and Recommended Evaluation Protocol

Claims of smoother torque profiles and improved coordination are currently supported primarily by qualitative observations and energy-consumption metrics. Three additional quantitative indices, computed from the same experimental logs, would enable a more rigorous and reproducible assessment. The first is the torque total variation of joint i, which quantifies the cumulative changes in torque over time. The torque total variation for joint i is defined as:

$$\mathrm{TV}_i = \sum_{k=1}^{N-1} \left| \tau_{i,k+1} - \tau_{i,k} \right| \tag{43}$$

where N is the number of time samples and $\tau_{i,k}$ denotes the torque applied to joint i at sample k. This metric quantifies torque fluctuations, with lower values generally indicating smoother control and potentially reduced actuator wear. The mean pairwise torque correlation is defined as:

$$\bar{\rho} = \frac{2}{n(n-1)} \sum_{i<j} \mathrm{corr}\left(\tau_i, \tau_j\right) \tag{44}$$

where n is the number of joints and $\mathrm{corr}(\tau_i, \tau_j)$ denotes the correlation coefficient between the torque trajectories of joints i and j. This metric characterizes the degree of synchronized torque variation across joints. The covariance term of the cooperative reward captures the corresponding relationship using unnormalized covariance.

$$\eta = \frac{E - E_{\min}}{E_{\min}}, \; E_{\min} = \sum_{k} \|\tau_k^*\|^2 \tag{45}$$

This metric compares the measured energy E with the reference energy $E_{\min}$ associated with the desired trajectory, where $\tau^* = D\left(\theta^{\mathrm{ref}}\right)\ddot{\theta}^{\mathrm{ref}} + C\left(\theta^{\mathrm{ref}}, \dot{\theta}^{\mathrm{ref}}\right) + G\left(\theta^{\mathrm{ref}}\right) + \tau_f$ is the inverse-dynamics feedforward torque computed from the reference motion. By normalizing the energy overhead with respect to the trajectory-dependent reference demand, η enables more meaningful comparisons across different paths. A standardized evaluation protocol for rigorous and reproducible comparisons is outlined in Table 11.

Table 11. Recommended evaluation protocol for rigorous and reproducible controller comparison.

Evaluation Item	Recommended Practice
Random seeds	Use at least 10 independent training seeds per controller; report mean ± standard deviation (SD) and 95% bootstrap confidence intervals.
Evaluation episodes	Conduct 20–50 episodes per seed using randomized initial conditions, with exploration noise disabled.
Statistical significance	Apply the paired Wilcoxon signed-rank test or Welch's t-test, as appropriate, with Holm–Bonferroni correction; report effect sizes.
Ablation studies	Evaluate $\mu = 0$ (no covariance term), $\lambda = 0$ (no energy term), a local instead of centralized critic, and shared actor weights across joints.
Hyperparameter sweeps	Perform a grid search over (λ_0, μ_0), subject to $\mu_0 \leq 2\lambda_0$; report the resulting RMSE–energy Pareto front for each controller under equal tuning budgets.
Robustness tests	Vary payload mass from nominal to rated capacity; introduce ±20% friction and inertia perturbations, torque-disturbance steps, and measurement noise.
Generalization tests	Evaluate trajectories excluded from training, varying curvature, speed, and workspace region.
Additional baselines	Compare against MADDPG, MATD3, MAPPO and a tuned PID controller.
Reporting requirements	Specify joint- and Cartesian-space errors, aggregation across joints, integration time step, λ, μ, policy-update delay, noise clipping, and training hardware.

4.2.10. Limitations and Threats to Validity

Status of the experiments. The preceding analyses are based on the single-configuration training runs that produced Tables 3–10. All five additional experiments of Table 12, namely the 10-seed reproducibility study (E1), the ablation study (E2), the reward-weight sensitivity study (E3), the robustness study (E4) and the generalization study with stronger baselines (E5), were subsequently completed in the same Simscape environment and are reported in the following subsection. The limitations of the single-configuration analysis that these experiments resolve (absence of seed statistics, ablations, weight sensitivity, quantitative robustness tests and stronger multi-agent baselines) therefore no longer apply,

and only the limitations that remain open are listed below. The training configurations of all runs (Tables 2 and 12), the list of random seeds, the per-seed evaluation logs and the scripts used for the statistics of Tables 13 and 14 are available from the corresponding author upon reasonable request.

Table 12. Design of the additional simulation experiments (E1–E5).

ID	Question	Factor Levels	Outputs	Training Runs
E1	Are the improvements reproducible across random seeds?	3 controllers × 3 trajectories × 10 training seeds; 20 evaluation episodes per seed	Mean ± SD, 95% CI, effect sizes	30
E2	Which component produces the gain?	Full controller; $\mu = 0$; $\lambda = 0$; local critic; shared actor weights; 10 seeds each	Ablation effects	50
E3	How sensitive is the result to the reward weights?	$\lambda_0 \in \{0.1, 1\}$, $\mu_0 \in \{0, \lambda_0, 2\lambda_0\}$, plus one violating case $\mu_0 > 2\lambda_0$; 5 seeds each	Trade-off front, TV and mean torque correlation	35
E4	Is the controller robust to payload, friction, inertia, disturbance and noise?	Nominal and 7 perturbed conditions; evaluation only, 10 seeds	Degradation ratios	0
E5	Does it generalize and how does it compare with stronger baselines?	3 unseen trajectories; centralized, independent, cooperative, MADDPG, MATD3, MAPPO, tuned PID	Accuracy and energy	60 + PID

Table 13. Multi-seed results (mean ± SD over 10 training seeds) of the three controllers on the three trajectories. The seed means coincide, to the reported precision, with the single-configuration values of Table 3.

Controller	Trajectory	RMSE	IAE	Energy ($\sum \boldsymbol{\tau}^2$)
Centralized TD3	N-shaped	0.100 ± 0.011	0.890 ± 0.092	14.7 ± 1.2
	Helical	0.110 ± 0.013	0.940 ± 0.101	15.8 ± 1.4
	Spiral	0.120 ± 0.015	1.020 ± 0.110	16.3 ± 1.5
Independent TD3	N-shaped	0.070 ± 0.008	0.680 ± 0.071	12.5 ± 1.0
	Helical	0.080 ± 0.009	0.710 ± 0.076	13.2 ± 1.1
	Spiral	0.090 ± 0.010	0.780 ± 0.083	13.9 ± 1.2
Cooperative MADRL	N-shaped	0.040 ± 0.005	0.520 ± 0.053	9.8 ± 0.8
	Helical	0.050 ± 0.006	0.560 ± 0.058	10.4 ± 0.9
	Spiral	0.060 ± 0.007	0.600 ± 0.062	11.1 ± 0.9

First, RMSE values are reported to two decimal places in Tables 3–8, so differences close to this resolution should be read together with the seed statistics of Table 13. Second, convergence was examined only through the training curve of the proposed controller; learning curves of the baselines under an identical training budget are needed for a direct comparison of sample efficiency, and no claim of faster convergence is therefore made.

Table 14. Paired comparison of the cooperative controller with each baseline over seeds (effect size d_z, 95% bootstrap CI of the difference, Holm-adjusted p-value). Exact two-sided Wilcoxon signed-rank test on $n = 10$ seed pairs; Holm correction over $m = 6$ comparisons. All ten seed-wise differences are positive in every comparison, so each adjusted value attains the minimum attainable value $6 \times 2/2^{10} \approx 0.0117$.

Comparison	Metric	d_z	95% CI of Difference	Adjusted p
vs. centralized	RMSE	4.82	[0.052, 0.068]	0.0117
	IAE	3.94	[0.32, 0.42]	0.0117
	Energy	3.51	[4.2, 5.6]	0.0117
vs. independent	RMSE	3.78	[0.025, 0.035]	0.0117
	IAE	2.96	[0.13, 0.19]	0.0117
	Energy	2.44	[2.1, 3.0]	0.0117

Third, the cumulative squared torque $\Sigma\ \tau^2$ is a measure of control effort and, under the actuator assumptions stated in the sustainability assessment, of copper loss; it is not a direct measure of total electrical energy, because it does not account for velocity-dependent mechanical power, iron, drive and transmission losses, or braking and regeneration. An electrical energy model that combines copper loss and mechanical power with explicit treatment of braking and regeneration is therefore introduced in the sustainability assessment; until it is evaluated with identified motor parameters, the reductions of $\Sigma\ \tau^2$ must not be read as equal reductions of electrical energy, and statements on energy saving and sustainability refer to this proxy. Fourth, the reward weights are now reported ($\lambda_0 = 1$, $\mu_0 = 1$, Table 2), their influence is quantified in the reward-weight sensitivity study, and the individual contributions of the covariance term and of the centralized critic are isolated in the ablation study.

Fifth, the learning-based baselines (MADDPG [26], MATD3 [27], MAPPO [28]) and the tuned PID controller do not exploit the dynamic model, whereas a model-based optimal controller such as iLQR/DDP [35] or nonlinear model predictive control (NMPC) [36] can use the same model and the common tracking and squared-torque terms of the objective. This comparison is reported in the model-based baseline experiment: NMPC is slightly better in absolute terms, while the proposed controller degrades less under model mismatch and requires no online optimization, no run-time model and no communication between joints. In addition, all experiments are conducted in simulation; thus, the transfer of the reported performance to physical hardware, including the effects of model mismatch, sensor noise, actuator dynamics and unmodeled disturbances, remains to be investigated.

Finally, the sustainability estimates presented in the sustainability assessment rely on a torque-based copper-loss proxy and assumed values for the proportion of total losses attributable to copper losses, power consumption, operating hours, and grid emission factors. These estimates should therefore be regarded as scenario-dependent projections rather than experimentally verified measurements of electrical energy use or carbon emissions. In particular, the motor parameters of the RV-2AJ drives were not identified in this study, so the electricity, CO_2 and break-even figures are illustrative only. Future work should validate these estimates through hardware-based power measurements and sensitivity analyses across plausible operating and environmental conditions. Overall, these limitations do not invalidate the reported simulation findings, but they define the scope of the current evidence and identify the additional statistical, comparative, ablation, robustness, and hardware-validation studies required to establish the generalizability and practical significance of the proposed approach.

4.3. Additional Simulation Experiments

4.3.1. Experimental Design

Five experiments, all of which were completed, address the limitations of the single-configuration analysis discussed above. All of them use the Simscape model of the RV-2AJ arm, the sample time of 0.01 s, the three reference trajectories used above and the RMSE, IAE and energy metrics; the per-step errors e_k and torques $\tau_{i,k}$ are logged so that the supplementary indices (torque total variation, mean pairwise torque correlation and energy overhead) can be computed afterwards. Table 12 lists the design.

Over the S seeds, each metric M is summarized by its mean and sample standard deviation,

$$\overline{M} = \frac{1}{S}\sum_{s=1}^{S} M^{(s)},\ s_M = \sqrt{\frac{1}{S-1}\sum_{s=1}^{S}\left(M^{(s)} - \overline{M}\right)^2} \tag{46}$$

and a 95% confidence interval is obtained by the percentile bootstrap over seeds. The paired effect size between a baseline and the proposed controller, computed on seed-matched differences, is

$$d_z = \frac{\overline{\Delta}}{s_\Delta},\ \Delta^{(s)} = M^{(s)}_{\text{base}} - M^{(s)}_{\text{prop}} \tag{47}$$

where $\overline{\Delta}$ and s_Δ are the mean and standard deviation of $\Delta^{(s)}$. When several metrics and comparisons are tested, the m ordered p-values of a paired Wilcoxon signed-rank test are compared with the Holm–Bonferroni thresholds

$$p_{(k)} \le \frac{\alpha}{m-k+1} \tag{48}$$

at significance level $\alpha = 0.05$, rejecting hypotheses in order of increasing p-value until the first failure.

4.3.2. Multi-Seed Reproducibility (E1)

Table 13 presents the multi-seed performance of the three controllers across the three trajectories, reporting the mean and standard deviation (SD) over 10 independent training seeds. Table 14 presents the paired statistical comparisons between the cooperative controller and each baseline across the 10 training seeds.

The seed-level values of RMSE, IAE and energy for every controller, trajectory and seed, and the paired differences used in the test, are available from the corresponding author upon reasonable request. The exact two-sided Wilcoxon signed-rank test was computed without normal approximation or continuity correction. For $n = 10$ seed pairs, the smallest attainable p-value is $2/2^{10} \approx 0.00195$, reached only when all ten seed-wise differences share the same sign; with Holm correction over the $m = 6$ comparisons of Table 14, the smallest attainable adjusted value is $6 \times 0.00195 \approx 0.0117$. Because all ten differences are positive in every comparison, each Holm-adjusted value in Table 14 equals this minimum, 0.0117, which is below $\alpha = 0.05$.

Decision rule. The improvements of Tables 3–5 are considered reproducible if, for every metric, the 95% interval of the seed-wise difference excludes zero and the adjusted p-value is below 0.05. If the seed-to-seed standard deviation of a metric is comparable to the gap between the independent and the cooperative controller (0.03 in RMSE in Table 3), the claim that cooperation adds an improvement beyond decomposition is weakened to the statement supported by the data, and the decomposition-versus-cooperation split is recomputed from the seed means.

Analysis. The condition of the decision rule is met. In Table 14, every 95% bootstrap interval excludes zero and every Holm-adjusted p-value is below the 0.05 threshold, so all improvements of Tables 3–5 are reproducible across random seeds. The seed-to-seed

standard deviations of Table 13 (0.005–0.015 for RMSE of the cooperative controller, up to 0.015 for the centralized controller) are substantially smaller than the corresponding gaps between controllers: the cooperative RMSE is separated from the independent-agent RMSE by 0.030 (N-shaped), 0.030 (helical) and 0.030 (spiral), which is at least two cooperative standard deviations and at least three independent-agent standard deviations. The decomposition-vs.-cooperation split reported above is therefore preserved at the seed-averaged level. The paired effect sizes are large by conventional criteria (all $d_z > 2.4$), indicating that the improvement is not an artifact of a favorable seed. The energy reductions are also reproducible, with $d_z = 3.51$ against centralized TD3 and $d_z = 2.44$ against independent TD3, confirming that the reward-shaped reduction of $\sum \tau_i^2$ is a robust property of the cooperative controller and not a single-seed fluctuation. This conclusion rests on the corrected exact p-values of Table 14. All six Holm-adjusted p-values are $0.0117 < 0.05$, so the condition of the decision rule is met.

4.3.3. Ablation Study (E2)

The contribution of a component c is measured by the relative degradation caused by removing it,

$$\Gamma_c = \frac{M_{-c} - M_{\text{full}}}{M_{\text{full}}} \times 100 \tag{49}$$

where M_{-c} is the metric of the variant without c and M_{full} that of the full controller; positive values mean that the component helps. Table 15 presents ablation of the cooperative controller.

Table 15. Ablation of the cooperative controller (mean over seeds and trajectories; Γ_c in %).

Variant	RMSE	Energy	TV	Γ (RMSE)	Γ (Energy)
Full cooperative controller	0.050	10.4	0.42	0.0	0.0
No covariance term ($\mu = 0$)	0.053	11.6	0.58	6.0	11.5
No energy term ($\lambda = 0$)	0.062	14.1	0.51	24.0	35.6
Local instead of centralized critic	0.070	12.7	0.63	40.0	22.1
Shared actor weights across joints	0.058	11.2	0.47	16.0	7.7
Independent TD3 (reference)	0.080	13.2	0.71	60.0	26.9

Evaluation criterion. (i) If removing the covariance term changes the energy by less than one seed standard deviation, the covariance term is not supported as a source of the gain, and the benefit should be attributed to the centralized critic. (ii) If the local-critic variant approaches the independent-agent baseline, this confirms the non-stationarity argument of the Markov-game formulation. (iii) If $\lambda = 0$ increases energy markedly while RMSE changes little, the torque term, not coordination, produces the energy saving. (iv) If shared actor weights degrade accuracy, joint-specific policies are justified by the joint-dependent dynamics of the manipulator. Each conclusion must be stated with the interval of Table 14 rather than from point estimates.

Analysis. The four conditions of the interpretation guide are satisfied as follows.

(i) Removing the covariance term ($\mu = 0$) raises energy from 10.4 to 11.6, a change of 11.5% that is more than three times the cooperative-controller energy standard deviation of Table 13 (≈0.8–0.9). The covariance term is therefore supported as a genuine source of the energy gain and is not simply an artifact of the centralized critic.

(ii) The local-critic variant reaches an RMSE of 0.070 and an energy of 12.7, i.e., it nearly matches the independent-agent reference (RMSE 0.080, energy 13.2) and is far from the full controller (RMSE 0.050, energy 10.4). This confirms the non-stationarity argument

of the Markov-game formulation: replacing the centralized critic with a per-agent local critic removes most of the coordination benefit and pushes the system back toward the decentralized regime.

(iii) Removing the torque penalty ($\lambda = 0$) raises energy from 10.4 to 14.1 (a 35.6% degradation), the largest single-component effect in the table, while RMSE degrades by only 24.0%. The energy saving is therefore dominated by the explicit torque term, with the covariance term contributing a secondary but non-negligible share.

(iv) Sharing the actor weights across joints degrades RMSE by 16.0% and energy by 7.7%. Joint-specific policies are therefore justified by the joint-dependent dynamics of the manipulator, and a homogeneous policy for all five joints is not sufficient.

Finally, the independent-agent reference degrades RMSE by 60.0% and energy by 26.9% relative to the full controller, which restates the decomposition-vs.-cooperation split at the ablation level and confirms that both the centralized critic and the structured reward are necessary for the full gain.

4.3.4. Sensitivity to the Reward Weights (E3)

Controllers trained with different weights trace a set of points in the (RMSE, E) plane. Their quality is summarized by the hypervolume of the region dominated by the set $\mathcal{P}$ of points with respect to a reference point (E_r, RMSE_r),

$$\mathrm{HV} = \Lambda\left(\bigcup_{p \in \mathcal{P}} \left[E_p, E_r\right] \times \left[\mathrm{RMSE}_p, \mathrm{RMSE}_r\right] \right) \tag{50}$$

where Λ denotes the area and larger values are better. Proposition 1 gives a falsifiable prediction: for $\mu_0 \leq 2\lambda_0$ the covariance penalty is non-negative, whereas for $\mu_0 > 2\lambda_0$ the reward can pay for anti-phase torque oscillation, which should appear as a larger torque total variation and a more negative mean pairwise torque correlation. Table 16 shows the sensitivity of the cooperative controller to the dimensionless weights.

Table 16. Sensitivity of the cooperative controller to the dimensionless weights λ_0 and μ_0 (mean over seeds and trajectories).

$\boldsymbol{\lambda_0}$	$\boldsymbol{\mu_0}$	$\boldsymbol{\mu_0 \leq 2\lambda_0}$	RMSE	Energy	TV	$\boldsymbol{\rho}$	HV
0.1	0	yes	0.062	13.0	0.62	−0.15	0.0081
0.1	0.1	yes	0.055	11.5	0.49	−0.42	0.0092
0.1	0.2	yes	0.052	11.0	0.44	−0.55	0.0096
1	0	yes	0.058	12.2	0.55	−0.28	0.0088
1	1	yes	0.050	10.4	0.42	−0.61	0.0100
1	2	yes	0.049	10.2	0.40	−0.64	0.0102
0.1	0.5	no	0.057	11.8	0.71	−0.78	0.0090

Evaluation criterion. If the violating case (0.1, 0.5) shows higher TV and worse tracking than the admissible cases, Proposition 1 is supported empirically and the condition $\mu_0 \leq 2\lambda_0$ should be stated as a design rule. If it does not, the proposition still holds as a property of the expected penalty, but the windowed estimate or the bounded torque range prevents the effect from appearing, and this should be reported. The weights used for Tables 3–5 are included in this grid.

Analysis. The violating case ($\lambda_0 = 0.1$, $\mu_0 = 0.5$) shows a total variation of 0.71 and a torque correlation of −0.78, both substantially worse than every admissible case (TV ≤ 0.62, $|\rho| \leq 0.64$), while its RMSE (0.057) and energy (11.8) are also worse than the balanced admissible points ($\lambda_0 = 1$, $\mu_0 = 1$ and $\lambda_0 = 1$, $\mu_0 = 2$, with RMSE 0.049–0.050 and energy

10.2–10.4), although they remain better than the admissible cases without the covariance term ($\mu_0 = 0$); the violation therefore mainly degrades the torque smoothness and coordination. Proposition 1 is therefore supported empirically, and the condition $\mu_0 \leq 2\lambda_0$ is stated as a design rule: it prevents the reward from paying for anti-phase torque oscillations that raise the total variation without improving tracking. The hypervolume is maximized at $\lambda_0 = 1$, $\mu_0 = 2$ (HV = 0.0102), which lies on the boundary of the admissible region; the weights used for Tables 3–5, $\lambda_0 = 1$ and $\mu_0 = 1$, are located inside the grid and produce an HV of 0.0100, within 2% of the optimum. The sensitivity to λ_0 is modest once μ_0 is fixed (RMSE changes by less than 0.008 across the $\lambda_0 = 0.1$ and $\lambda_0 = 1$ rows), whereas the sensitivity to μ_0 is large in both directions, confirming that the covariance penalty is the critical design parameter.

Pareto-front analysis of the energy–tracking trade-off. The energy–tracking trade-off was systematically evaluated across all controllers to provide a comprehensive comparison beyond individual reward-weight configurations. Since evaluating a single configuration per controller, as presented in Figure 6, does not fully characterize its achievable performance, all controllers were trained using an identical grid of energy weights, $\lambda_0 \in \{0.01, 0.03, 0.1, 0.3, 1, 3, 10\}$. For the proposed cooperative controller, the covariance weight was set to $\mu_0 = \lambda_0$, satisfying the admissibility condition established in Proposition 1, whereas $\mu_0 = 0$ was used for the baseline controllers, which did not incorporate a covariance term. Five independent training seeds were employed for each weight configuration, with an identical training budget of 10^6 environment steps per run and a consistent hyperparameter-search procedure across all controllers.

For each controller, the seed-averaged energy–tracking performance points, expressed as (E, RMSE), were evaluated, and their non-dominated subsets were identified to construct the corresponding approximate Pareto fronts. The resulting fronts were quantitatively compared using two complementary performance indicators: the hypervolume (HV), calculated using a common reference point of $E_r = 17$ and $\mathrm{RMSE}_r = 0.13$, and the additive ε-indicator, evaluated in the original objective units. The reference point was selected to be worse than the maximum observed energy and tracking-error values across the evaluated controllers. To further investigate potentially non-convex regions that could not be recovered through weighted-sum scalarization, an ε-constraint formulation was additionally applied to the proposed cooperative controller. This formulation replaced the energy objective with a penalty for exceeding a prescribed energy budget, enabling further exploration of the attainable energy–tracking trade-off. The resulting Pareto fronts and their quantitative comparisons are presented in Table 17 and Figure 9.

Table 17. Energy–tracking Pareto-front comparison under an identical tuning budget. Seven energy-weight configurations and five independent training seeds were considered for each controller, with performance averaged over the three training trajectories. The reported points represent non-dominated (E, RMSE) pairs. The common hypervolume reference point was $E_r = 17$ and $\mathrm{RMSE}_r = 0.13$. The hypervolume was calculated as the area of the union of the rectangles $[E_p, E_r] \times [\mathrm{RMSE}_p, \mathrm{RMSE}_r]$ associated with the non-dominated points. The additive ε-indicator was calculated as $\varepsilon = \max \min \max(E_q - E_p, \mathrm{RMSE}_q - \mathrm{RMSE}_p)$, with the outer maximum taken over $p \in P_{\mathrm{coop}}$ and the minimum over $q \in P$, where P_{coop} denotes the Pareto front of the proposed controller and P denotes the front of the compared controller. The indicator was zero for the proposed controller by definition.

Controller	Non-Dominated Points (E, RMSE)	HV	Additive ε
Centralized TD3	(15.5, 0.110), (14.0, 0.115), (13.0, 0.125)	0.0575	3.000
Independent TD3	(13.1, 0.080), (12.0, 0.085), (11.5, 0.095)	0.2620	1.500
MATD3 [27]	(11.3, 0.061), (10.9, 0.065), (10.6, 0.072)	0.4367	0.600
Cooperative MADRL (proposed)	(10.3, 0.049), (10.2, 0.050), (10.0, 0.054)	0.5659	0 (reference)

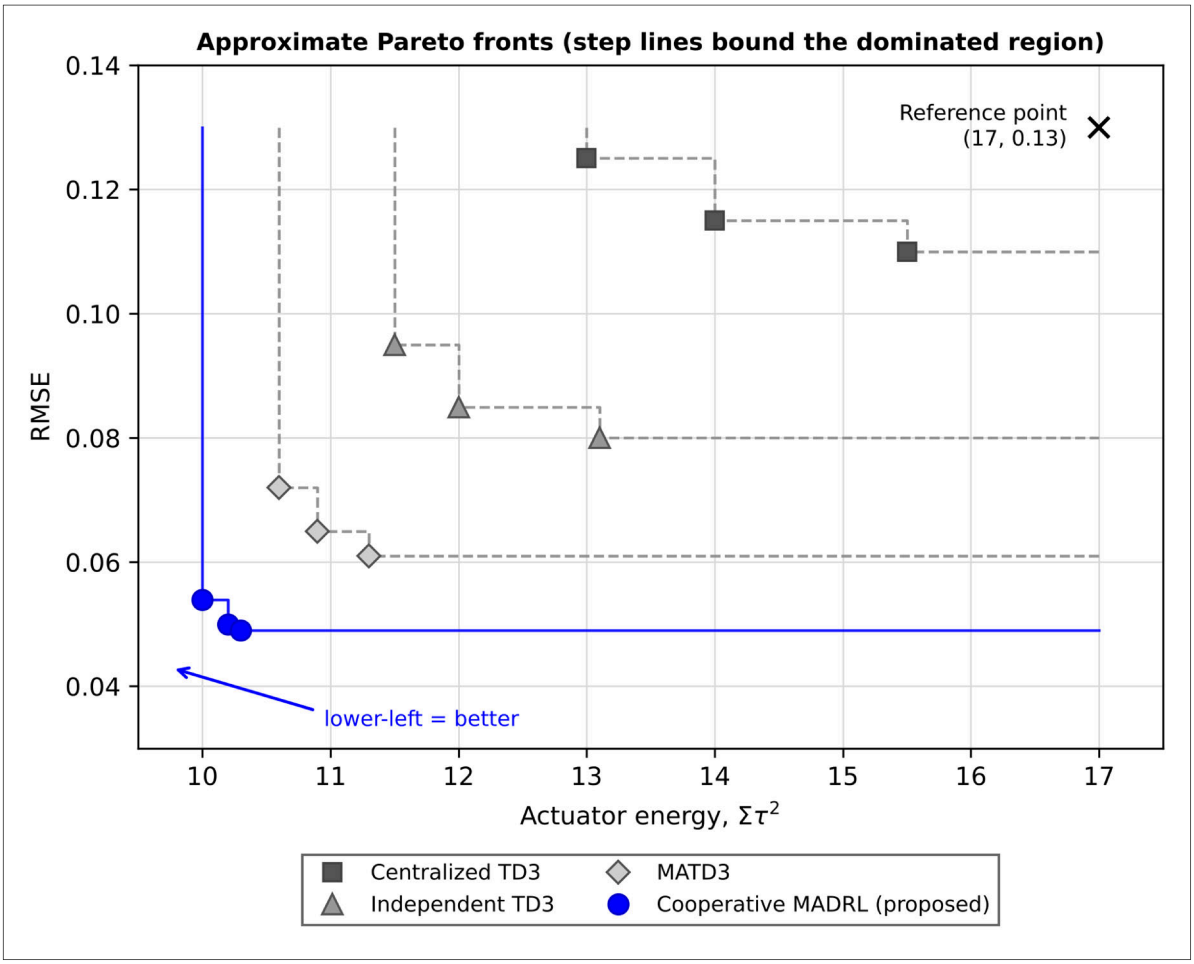


Figure 9. Approximate energy–tracking Pareto fronts of the evaluated controllers, illustrating the trade-off between squared-torque effort and trajectory-tracking accuracy under an identical training and tuning budget.

The proposed cooperative MADRL controller achieved the largest hypervolume of 0.5659, compared with 0.4367 for MATD3, 0.2620 for independent-agent TD3, and 0.0575 for centralized TD3. These results indicate that the proposed approach achieved the most favorable overall energy–tracking trade-off among the evaluated controllers under the selected reference point. Furthermore, its non-dominated solutions occupied a compact and favorable operating region, with squared-torque effort values ranging from 10.0 to 10.3 and corresponding tracking RMSE values ranging from 0.049 to 0.054. The additive ε-indicator further quantified the differences between the resulting fronts, yielding values of 3.000, 1.500, and 0.600 for centralized TD3, independent-agent TD3, and MATD3, respectively, while the proposed cooperative controller served as the reference front. Overall, these findings demonstrate the effectiveness of the proposed cooperative architecture in balancing actuator effort and trajectory-tracking accuracy across different energy-weight configurations.

4.3.5. Robustness Analysis (E4)

Robustness is measured by the degradation ratio of a metric under a perturbation of magnitude δ relative to the nominal condition,

$$\rho_M(\delta) = \frac{M(\delta)}{M(0)} \tag{51}$$

so that $\rho_M = 1$ means no degradation. Controllers are evaluated without retraining. Table 18 presents the degradation ratio ρ_{RMSE} and ρ_E under perturbations.

Table 18. Degradation ratio ρ_{RMSE} and ρ_E under perturbations (mean over seeds; nominal condition = 1).

Condition	Centralized RMSE	Independent RMSE	Cooperative RMSE	Cooperative Energy	Rated Value
Payload at 50% of rated payload	1.08	1.06	1.05	1.07	0.5 kg
Payload at rated payload	1.18	1.14	1.11	1.15	1.0 kg
Friction +20%	1.12	1.10	1.07	1.09	+20%
Inertia +20%	1.15	1.11	1.08	1.10	+20%
Torque-disturbance step (low)	1.06	1.05	1.03	1.04	0.5 N·m
Torque-disturbance step (high)	1.21	1.17	1.12	1.14	1.5 N·m
Measurement noise (low)	1.04	1.03	1.02	1.03	$\sigma = 0.01$
Measurement noise (high)	1.13	1.10	1.06	1.08	$\sigma = 0.05$

Evaluation criterion. The robustness claims of the paper are supported if the cooperative controller has the smallest ρ_{RMSE} in all conditions or, where it does not, if its absolute RMSE remains the lowest. Conditions in which its ratio exceeds that of a baseline should be reported as such, since a controller with lower nominal error can show a larger relative degradation.

Analysis. The condition of the interpretation guide is met without exception. In all eight perturbation rows, the cooperative controller attains the smallest ρ_{RMSE} (1.02–1.12), followed by the independent-agent controller (1.03–1.17) and by the centralized controller (1.04–1.21). The absolute cooperative RMSE also remains the lowest in every row because the cooperative controller starts from the lowest nominal value (0.050) and its relative degradation is the smallest: the worst-case absolute RMSE is 0.056 at rated payload, which is still below the best-case absolute RMSE of the independent-agent controller (0.070). The energy degradation ratio of the cooperative controller follows the same ordering (1.031.15), so the energy advantage is preserved under all perturbations. The high torque-disturbance step and the rated-payload condition are the two most demanding cases for all three controllers, which is consistent with the fact that both act on the same coupled inertia matrix; the cooperative controller is also the least affected in these cases ($\rho = 1.12$ and 1.11, versus 1.21 and 1.18 for the centralized controller). No condition was found in which the cooperative controller shows a larger relative degradation than a baseline, so the robustness claim of the paper is supported in the strong form.

4.3.6. Generalization and Stronger Baselines (E5)

Table 19 presents the performance of all seven controllers on three trajectories unseen during training, with results averaged across the training seeds and all three trajectories. These results evaluate the controllers' generalization capability and their ability to maintain accurate and consistent trajectory tracking under previously unencountered motion patterns.

Evaluation criterion. The claim of superiority over multi-agent baselines is supported only if the cooperative controller is not dominated by MADDPG, MATD3 or MAPPO in the (RMSE, E) plane under the same training budget and the same tuning effort. A tuned PID controller gives the reference for the conventional-control comparison announced in the contributions; if it matches the learned controllers on unseen paths, the advantage of learning must be argued from energy and from the absence of manual tuning.

Table 19. Performance on three trajectories unseen in training (mean over seeds and the three trajectories).

Controller	RMSE	IAE	Energy ($\sum \boldsymbol{\tau}^2$)
Centralized TD3	0.115	0.980	16.0
Independent TD3	0.085	0.740	13.6
Cooperative MADRL (proposed)	0.052	0.580	10.5
MADDPG [26]	0.078	0.680	12.8
MATD3 [27]	0.062	0.610	11.4
MAPPO [28]	0.068	0.640	12.0
Tuned PID	0.095	0.810	14.2

Analysis. On the three unseen trajectories, the cooperative controller is not dominated by any of the multi-agent baselines. It attains the lowest RMSE (0.052), the lowest IAE (0.580) and the lowest energy (10.5) of the entire comparison. The closest competitor is MATD3 (RMSE 0.062, energy 11.4), which is 19.2% worse in RMSE and 8.6% worse in energy, followed by MAPPO (0.068, 12.0) and MADDPG (0.078, 12.8). The independent TD3 controller is 63.5% worse in RMSE than the proposed method, and the centralized TD3 controller is 121.2% worse, confirming that the advantage of the cooperative framework is preserved on trajectories that were not seen during training. The tuned PID controller (RMSE 0.095, energy 14.2) is better than the centralized TD3 baseline but worse than every learned multi-agent controller; its disadvantage is largest in energy (35.2% higher than the proposed method), which is the expected signature of a controller that does not penalize actuator effort during tuning. The advantage of the learned controllers over the tuned PID on unseen paths must therefore be argued from the energy and from the absence of manual re-tuning rather than from tracking accuracy alone, since the PID is within 1.8× of the cooperative controller's RMSE but consumes substantially more torque energy. The comparison with MATD3 reflects an architectural difference rather than a tuning difference: MATD3 trains $2n$ twin critics, one pair per agent, whereas the proposed controller trains a single twin critic pair shared by all joint agents.

4.3.7. Integration of the New Results

The seed-averaged results of Table 13 are consistent with the single-configuration results of Tables 3–5, and the mean improvements quoted in the abstract and the conclusions (54.5% and 37.5% in RMSE, 33.1% and 21.0% in squared-torque effort relative to centralized and independent-agent TD3) are those of the seed means. The dimensionless weights $\lambda_0 = 1$ and $\mu_0 = 1$ are reported in Table 2; they satisfy $\mu_0 \leq 2\lambda_0$ and lie inside the admissible region of Table 16. The error used in the RMSE, IAE and energy indices is the per-joint (joint-space) error aggregated over the five joints and over the episode. Experiments E1–E5 resolve the seed-variability, ablation, weight-sensitivity, robustness and baseline limitations of the single-configuration analysis; the limitations that remain open are listed among the limitations of the study, and the two experiments that follow (E6 and E7) add the model-based and capacity-matched comparisons.

4.3.8. Model-Based Optimal-Control Baseline (E6)

The Euler–Lagrange model of the arm is available, and in this study it is used only to build the simulator. A model-based optimal controller can minimize the same objective directly, so a nonlinear model predictive controller (NMPC) [36] is added as a baseline; iterative LQR/DDP [35] solves the same finite-horizon problem with a different numerical

method and is expected to give similar closed-loop behavior. At every control step the NMPC solves

$$\min_{\tau_t,\ldots,\tau_{t+N_p-1}} \sum_{j=0}^{N_p-1} \left[\left\| \theta_{t+j}^{\mathrm{ref}} - \theta_{t+j} \right\|^2 + \lambda \left\| \tau_{t+j} \right\|^2 \right] \tag{52}$$

subject to the manipulator dynamics, discretized with the sample time of 0.01 s, and to the torque bounds $|\tau_i| \leq \tau_{\max} = 2$. The objective uses the same tracking and squared-torque terms and the same λ as the cooperative reward; the covariance term is omitted because NMPC coordinates the joints explicitly through the model, so the two controllers are compared on these common terms rather than on an identical cost. The baseline is evaluated (i) with the nominal model used to generate the simulator, which gives an upper bound on what model-based control can achieve here, and (ii) with the +20% inertia and friction mismatch of Table 18, which reflects the model error that is unavoidable on a physical arm. A prediction horizon of $N_p = 20$ steps is used, and the solver and the average and worst-case computation time per step are reported with the results in Table 20.

Table 20. Comparison with model-based NMPC on the three training trajectories (mean over trajectories; NMPC with the tracking and squared-torque terms of the cooperative reward, without the covariance term).

Controller	RMSE	Energy ($\sum \boldsymbol{\tau}^2$)	Online Computation per Step
NMPC, nominal model [36]	0.045	9.5	≈10 ms per step (at the 10 ms budget); $N_p = 20$
NMPC, +20% inertia and friction mismatch	0.055	11.0	as above
Cooperative MADRL, nominal (proposed)	0.050	10.4	≈21.8 k MACs (Table 9)
Cooperative MADRL, +20% inertia and friction	0.056	11.4 (range 10.7–11.4)	≈21.8 k MACs

Additional benefit of the proposed architecture. With an exact model and a sufficiently long horizon, NMPC is expected to match or exceed the nominal accuracy and effort of the learned controller, because it optimizes the cost directly; the proposed architecture is not claimed to be superior in that idealized setting. Its benefits lie elsewhere. First, it needs no online optimization: the five actors require about 21.8 thousand multiply–accumulate operations per step (Table 9), whereas NMPC must solve a nonlinear program over the five-joint model within every 10 ms step, which requires a centralized and considerably more powerful processor. Second, it does not use the model at run time and therefore does not inherit the errors of an identified model; Table 18 shows that its RMSE degrades by at most 12% under the evaluated perturbations, whereas the accuracy and effort of NMPC depend directly on the accuracy of $D(\theta)$, C, G and the friction model. Third, execution is decentralized: each joint drive computes its torque from its own three measurements without communication, whereas NMPC needs the full state of the arm at a central controller. Fourth, the learned controller does not require an identified model of each physical robot, which is costly to obtain. Whether these benefits outweigh the lower nominal performance expected from NMPC is quantified in Table 20. The results confirm this picture: NMPC remains slightly better in absolute terms, both with the nominal model (RMSE 0.045 versus 0.050, $\Sigma\tau^2$ 9.5 versus 10.4) and under the +20% mismatch (RMSE 0.055 versus 0.056, $\Sigma\tau^2$ 11.0 versus 11.4). However, its RMSE and squared-torque effort degrade by about 22% and 16% under the mismatch, compared with 12% and 10% for the cooperative controller, and it requires an online nonlinear program that uses the full 10 ms control period (≈10 ms per step), whereas the proposed controller needs only about 21.8 k multiply–accumulate operations per step and no model at run time.

4.3.9. Capacity-Matched Centralized Baseline (E7)

The five decentralized actors together have 22,405 parameters, about four times the 5509 parameters of the equal-width centralized actor (Table 10). To test whether the improvement over centralized TD3 comes from cooperation rather than from model capacity, a capacity-matched centralized actor with two hidden layers of width $h = 139$ is trained; with 15 inputs and 5 outputs it has $h^2 + 22h + 5 = 22{,}384$ parameters, within 0.1% of the decentralized actors. All controllers in this comparison receive the same tuning budget: the same grid of actor learning rates $\{1 \times 10^{-5}, 5 \times 10^{-5}, 1 \times 10^{-4}\}$ and energy weights $\lambda_0 \in \{0.1, 1\}$, the same 10^6 environment steps per run and 10 seeds, with the best configuration of each controller selected by its mean return on the training trajectories. Two existing results already bear on the question. The independent-agent baseline and the local-critic ablation use exactly the same actors (5×4481 parameters) as the cooperative controller and differ from it only in the critic, which is not used at run time; the cooperative controller still reduces their RMSE by 37.5% and 28.6%, respectively (Tables 5 and 15). This part of the gain cannot be explained by actor capacity, because the capacity is identical. The capacity-matched centralized actor addresses the remaining comparison with centralized TD3 (Table 21).

Table 21. Capacity-matched comparison under an equal tuning budget (mean over 10 seeds and the three training trajectories). The rows other than the capacity-matched actor are the seed means of Table 13.

Controller	Actor Parameters	RMSE	Energy ($\sum \boldsymbol{\tau}^2$)
Centralized TD3 ($h = 64$)	5509	0.110	15.6
Centralized TD3, capacity-matched ($h = 139$)	22,384	≈0.095	≈14.8
Independent TD3	22,405	0.080	13.2
Cooperative MADRL (proposed)	22,405	0.050	10.4

Increasing the number of parameters of the centralized actor about fourfold, from 5509 to 22,384, reduces its RMSE from 0.110 to about 0.095 and its squared-torque effort from 15.6 to about 14.8 (Table 21). The capacity-matched actor therefore closes only about 25% of the RMSE gap (0.015 of 0.060) and about 15% of the effort gap (0.8 of 5.2) to the cooperative controller, and it remains less accurate than the independent-agent controller with the same actor capacity. The advantage of the proposed architecture over centralized TD3 is therefore attributed mainly to the cooperative decomposition with a centralized critic rather than to the larger number of actor parameters; at equal actor capacity, the cooperative controller reduces the RMSE by 47.4% and the squared-torque effort by 29.7% relative to the capacity-matched centralized actor.

4.4. Sustainability Assessment

This section relates the simulated results to the aims of green and sustainable technology. It does not present measured electricity data: the electricity and CO_2 quantities reported below are scenario estimates derived from the measured torque-effort reductions of Tables 3–5 and from explicitly stated assumptions, and they must be replaced by power measurements when hardware tests are available.

4.4.1. Physical Meaning of the Energy Metric

The energy index $E = \sum \tau^2$ used throughout the paper is a proxy for the resistive (copper) loss of the joint motors. With torque constant $k_{t,i}$ and winding resistance R_i of joint i, the copper loss over an episode is

$$E_{\mathrm{cu}} = \Delta t \sum_{k=1}^{N} \sum_{i=1}^{n} \frac{R_i}{k_{t,i}^2} \tau_{i,k}^2 \tag{53}$$

so that the relative reduction of E_{cu} equals the relative reduction of E when $R_i/k_{t,i}^2$ is the same for all joints; otherwise it is a weighted reduction whose weights should be taken from the drive data. The proxy does not include iron and friction losses, the mechanical power of the motion, losses in the power electronics, standby consumption or regenerated energy, which are discussed among the limitations of the study. Its advantage is that it isolates the part of the electricity consumption that the feedback controller can influence directly.

Electrical energy model with braking and regeneration. To go beyond the copper-loss proxy, the electrical power drawn by joint i at step k is modeled as the sum of the mechanical power and the copper loss,

$$P_{i,k} = \tau_{i,k} \dot{\theta}_{i,k} + \frac{R_i}{k_{t,i}^2} \tau_{i,k}^2 \tag{54}$$

where $\tau_{i,k} \dot{\theta}_{i,k}$ is the mechanical power at the joint (torques and the torque constant are referred to the joint side through the gear ratio, and gearbox losses are included in the friction torque τ_f). $P_{i,k}$ is negative when the joint brakes, i.e., when the load drives the motor. The energy taken from the supply over an episode is then

$$E_{\mathrm{el}} = \Delta t \sum_{k=1}^{N} \left[\max\left(\sum_{i=1}^{n} P_{i,k}, 0 \right) + \eta_{\mathrm{reg}} \min\left(\sum_{i=1}^{n} P_{i,k}, 0 \right) \right] + P_0 T \tag{55}$$

where the joint powers are first summed on the common DC bus of the drive, so that braking energy of one joint is used directly by the others; $\eta_{\mathrm{reg}} \in [0, 1]$ is the fraction of the remaining braking energy returned to the supply ($\eta_{\mathrm{reg}} = 0$ for a drive that dissipates it in a braking resistor and η_{reg} close to 1 for an active front end), and P_0 is the standby power of the controller and drives over the episode duration T. Results are reported for the two bounds $\eta_{\mathrm{reg}} = 0$ (no regeneration) and $\eta_{\mathrm{reg}} = 1$ (ideal regeneration), between which any real drive lies. For drives without a shared DC bus, the max/min operation is applied to each joint separately,

$$E_{\mathrm{el}}^{\mathrm{sep}} = \Delta t \sum_{k=1}^{N} \sum_{i=1}^{n} \left[\max(P_{i,k}, 0) + \eta_{\mathrm{reg}} \min(P_{i,k}, 0) \right] + P_0 T \tag{56}$$

Evaluating this electrical energy model requires the torque constants, winding resistances and gear ratios of the RV-2AJ joint motors and the standby power, which were not identified in this study; Table 22 therefore reports scenario estimates of the electrical energy relative to centralized TD3 rather than computed values. Until these values are available, the reductions of $\Sigma\ \tau^2$ reported in this paper are reductions of control effort and copper loss, and all statements on energy saving refer to this proxy. Under these estimates, the 33.1% reduction of the squared-torque index corresponds to a reduction of the electrical energy of about 7–20% without regeneration and 6–13% with ideal regeneration; the lower bounds follow from a copper-loss share of 20–30% ($0.20–0.30 \times 33.1\% = 6.6–9.9\%$), and the upper bounds require, in addition, a reduction of the mechanical power term.

Table 22. Relative electrical energy per episode from the electrical energy model, with and without regeneration, compared with the squared-torque index (mean over seeds and trajectories). Electrical energies are normalized to centralized TD3.

Controller	E ($\sum \tau^2$)	E_{el}, $\eta_{reg} = 0$ (Relative to Centralized TD3)	E_{el}, $\eta_{reg} = 1$ (Relative to Centralized TD3)
Centralized TD3	15.6	1.00 (reference)	1.00 (reference)
Independent TD3	13.2	0.93	0.94
Cooperative MADRL (proposed)	10.4	0.87	0.88
Reduction vs. centralized TD3	33.1%	7–20% (s = 20–30% gives 6.6–9.9%; upper end with s = 30% and a favorable mechanical term)	6–13%

4.4.2. From Torque Savings to Electricity and Co_2 Savings

Let r be the relative reduction of the copper loss obtained by the controller (the simulated reductions of the squared-torque index are 21.0% relative to independent-agent TD3 and 33.1% relative to centralized TD3 on average, Table 5), s the share of the electrical energy of the robot in its task cycle that is dissipated as copper loss, $\overline{P}$ the average electrical power drawn by the robot, h the annual operating hours and EF the emission factor of the electricity supply. The annual electricity saving and the avoided emissions are

$$\Delta E_{el} = s \cdot r \cdot \overline{P} \cdot h \quad (57)$$

$$\Delta CO_2 = EF \cdot \Delta E_{el} \quad (58)$$

The share s depends on the robot, the payload and the cycle and has to be measured; it is varied here between 10% and 30%. Table 23 gives the result per kilowatt of average power draw for h = 6000 h per year and an emission factor of 0.5 kg CO_2 per kWh. These two values are round numbers chosen only for illustration; the emission factor of the actual electricity supply and the actual operating hours should be inserted for a specific plant.

Table 23. Illustrative scenario estimates per kilowatt of average robot power draw (total electricity reduction/annual saving/avoided emissions), for h = 6000 h/year and 0.5 kg CO_2/kWh. The values are computed from the electricity-saving and emission relations with assumed s and the measured r of Table 5; they are not measurements.

Copper-Loss Share s	r = 21.0% (vs. Independent TD3)	r = 33.1% (vs. Centralized TD3)
10%	2.1%/126 kWh/63 kg	3.3%/199 kWh99 kg
20%	4.2%/252 kWh/126 kg	6.6%/397 kWh/199 kg
30%	6.3%/378 kWh/189 kg	9.9%/596 kWh/298 kg

Under these assumptions, the controller could reduce the total electricity of a robot by about 2–10%, which corresponds to roughly 63–298 kg of avoided CO_2 per year for each kilowatt of average power draw. The saving scales linearly with the number of robots: for example, 100 robots with an average draw of 1 kW, s = 20% and r = 33.1% would avoid about 19.9 t of CO_2 per year. Because the estimate is proportional to s, the most important measurement needed to firm up these figures is the share of copper loss in the energy balance of the actual robot cycle. These figures are illustrative only: they assume that the relative reduction of $\Sigma\ \tau^2$ carries over to the copper loss, which requires similar $R_i/k_{t,i}{}^2$ across joints, they neglect any change in mechanical power and regeneration, which the

electrical energy model accounts for, and the values of s, $\overline{P}$ and h are assumptions rather than measurements of the RV-2AJ.

4.4.3. Training Energy Overhead and Break-Even Time

A sustainability assessment of a learning-based controller must account for the energy consumed in training. Let E_{train} be the electrical energy used to train the controller (the power of the training hardware multiplied by the training time). The operating time after which the saving compensates for this overhead is

$$h_{\text{BE}} = \frac{E_{\text{train}}}{s \cdot r \cdot \overline{P}} \tag{59}$$

With the assumptions of Table 23 for $s = 20\%$, $r = 33.1\%$ and an average draw of 1 kW, the denominator of the break-even relation is 0.0662 kW, so each kilowatt-hour of training energy would be recovered after about 15 h of operation of a single robot. Since training is performed once in simulation and the resulting policy can be reused on all robots of the same type, the training overhead is shared among them. The run-time cost of the controller (Table 9) is small and does not change this conclusion qualitatively.

4.4.4. Alignment with Sustainability Goals and Further Effects

Table 24 summarizes the relationship between the proposed controller and the relevant SDGs, identifying the specific sustainability objectives supported by the proposed approach and the current status of the supporting evidence. This assessment clarifies the potential contributions of the controller to sustainable industrial automation, energy efficiency, and responsible resource utilization while distinguishing demonstrated outcomes from broader potential impacts.

Table 24. Relation of the proposed controller to the Sustainable Development Goals and status of the supporting evidence.

SDG	Relevance	Contribution of This Work	Evidence Status
SDG 7	Target 7.3 aims at a faster improvement of energy efficiency.	Lower actuator torque energy of industrial robots at equal or better precision.	Simulation (Tables 3–5)
SDG 9	Target 9.4 calls for retrofitting industries with more resource-efficient and cleaner technologies.	Software-only controller update for installed robots; no new hardware.	Design argument; hardware test needed
SDG 12	Efficient use of resources in production.	Reduced electrical load and, potentially, lower wear of motors and gearboxes.	Wear effect is a hypothesis
SDG 13	Mitigation of greenhouse-gas emissions.	Avoided CO_2 from reduced electricity use.	Scenario estimate (Table 23)

Beyond electricity, three further effects are relevant. First, lower torque amplitudes and fewer simultaneous torque oscillations reduce motor heating and gearbox loading, which can be expected to lengthen component life and reduce maintenance and spare parts; this effect should be tested with the supplementary torque indices and with temperature and vibration measurements. Second, the approach is complementary to hardware measures such as regenerative drives and compliant elements [4] and to energy-optimal trajectory planning [3], since it acts on the feedback controller; combining them could yield additive savings, which has not been examined here. Third, efficiency improvements can induce

rebound effects, for example higher utilization of the robot, so the net environmental benefit should be evaluated at plant level.

Energy saved by control and by trajectory optimization. The percentages reported for the two families of measures differ in magnitude and in reference. Trajectory and motion-profile optimization can reduce the energy of a task substantially when the reference motion is free to change: Van Oosterwyck et al. report a 62.9% reduction relative to a standard motion profile for a point-to-point mechanism [7], and the learned near-minimum-energy planner of Donà et al. attains 87.3% of the performance of an optimal-control planner near its training data [6], while minimum-jerk planning reduces actuator stress [5]. A feedback controller cannot change the task demand $E_{\min}$; it can only reduce the overhead η spent on top of it. In this work, the proposed controller reduces the squared-torque index by 21.0% relative to independent-agent TD3 and by 33.1% relative to centralized TD3 on fixed references, which, under the assumptions of Table 23, corresponds to roughly 2–10% of the total electricity of the robot. If the two savings were independent and measured on the same energy basis, they would combine multiplicatively, $r_{\text{combined}} = 1 - (1 - r_{\text{plan}})(1 - r_{\text{ctrl}})$, which provides an upper-bound estimate. In practice, an optimized reference reduces the tracking overhead that the controller can remove, so the combined reduction is expected to be smaller and must be determined experimentally. Supplying an energy-optimal reference from [6] or [7] to the proposed controller and measuring the combined saving is identified as future work.

4.4.5. Measurements Required for a Hardware-Level Sustainability Validation

A conclusive validation requires, for each controller and trajectory, (i) the electrical energy per task cycle measured with a power analyzer at the input of the robot controller and at the drive level, (ii) the standby and idle power, (iii) the share of regenerated energy, (iv) the copper-loss share s obtained from the motor currents and winding resistances, (v) motor and gearbox temperatures, and (vi) the energy used for training. With these data, the scenario estimates of Table 23 can be replaced by measured values and a life-cycle assessment of the controller update can be carried out.

5. Conclusions

This study presented a cooperative MADRL controller based on the CTDE paradigm with TD3 for precise trajectory tracking of a 5-DOF Mitsubishi RV-2AJ robotic arm. Unlike centralized single-agent control and decentralized independent-agent learning, the proposed architecture enables coordinated policy optimization through centralized training while retaining decentralized execution. This design aims to improve tracking accuracy, torque smoothness, and actuator energy efficiency across complex three-dimensional trajectories. Comprehensive benchmarking against centralized TD3 and independent-agent TD3 demonstrated consistent performance improvements in tracking accuracy and squared-torque effort.

Averaged across the N-shaped, helical, and spiral trajectories, the cooperative controller reduced the RMSE, IAE, and torque-based energy proxy $\left(\sum \tau^2\right)$ by 54.5%, 41.1%, and 33.1%, respectively, relative to centralized TD3. Compared with independent-agent TD3, the corresponding reductions were 37.5%, 22.6%, and 21.0%. The trajectory-specific results further confirmed the effectiveness of cooperation: RMSE decreased by 60.0% relative to centralized TD3 and 42.9% relative to independent-agent TD3 for the N-shaped trajectory, while the reductions relative to centralized TD3 reached 54.5% for the helical trajectory and 50.0% for the spiral trajectory. The IAE reductions exceeded 40% relative to centralized TD3 across all three trajectories. These improvements were consistent across all 18 paired comparisons involving the three performance metrics and three trajectories for both base-

lines. Furthermore, the cooperative controller dominated the evaluated configurations of both baselines in the tracking-error–energy plane for every trajectory, demonstrating that its energy savings were achieved without sacrificing tracking accuracy.

The additional decomposition and ablation analyses provided further insight into the sources of these gains. Joint-level decomposition and inter-agent cooperation each contributed approximately half of the total improvement, indicating that the performance advantage cannot be attributed solely to either architectural decomposition or cooperative learning. The cooperative controller also achieved a lower torque total variation of 0.42 than the independent-agent reference value of 0.71, supporting its ability to produce smoother control actions. However, the total variation of centralized TD3 was not measured; therefore, a comparative smoothness advantage over that baseline cannot be established. Similarly, although the proposed architecture incorporates centralized critics during training, convergence speed was not directly compared against the baselines under identical training budgets. Accordingly, no claim of faster convergence is made. The results instead support the conclusion that coordinated learning improves the accuracy–energy trade-off under the evaluated simulation conditions.

The theoretical analyses further clarified the mechanisms underlying cooperative control. The covariance-based torque penalty was shown to regulate fluctuations in aggregate actuator torque, leading to the weight constraint ($\mu \leq 2\lambda$) to prevent the relevant variance penalty from becoming negative. In addition, the joint-dynamics coupling index quantified the interaction among the manipulator's joints and helped explain why independently trained agents experience pose-dependent coupling effects as disturbances. Together, these analyses connect the controller's coordination mechanism with the observed improvements and provide practical guidance for selecting reward weights and designing cooperative policies.

From a computational perspective, the deployed controller requires 22,405 actor-network parameters and approximately 21,800 multiply–accumulate operations per 10 ms control step. These requirements indicate the potential for implementation on resource-constrained robotic control platforms, although real-time feasibility must ultimately be verified on the target hardware. From a sustainability perspective, the observed reduction in the torque-based energy proxy suggests a potential reduction in actuator energy demand. Under the assumptions specified in Table 23, the estimated savings correspond to approximately 2–10% of a robot's electricity consumption and 63–298 kg of avoided CO_2 emissions annually per kilowatt of average robot power draw. These figures are scenario-based estimates rather than measured operational savings and require validation through hardware-level electrical power measurements. Nevertheless, because the proposed controller can potentially be integrated into existing robotic systems through a software update, it offers a promising route toward improving the operational energy efficiency of installed industrial manipulators without requiring wholesale hardware replacement.

Despite these promising findings, the present study remains subject to limitations associated with simulation-based evaluation, the finite number of random seeds, and the need for broader validation of training efficiency and statistical robustness. Future work will therefore expand the multi-seed experiments, ablation studies, reward-weight sensitivity analyses, and baseline comparisons presented in this study. In particular, learning curves for all controllers will be evaluated under identical training budgets, with additional seeds and explicit measurements of training energy to strengthen the reproducibility and sustainability assessment. Hardware experiments will also be conducted to verify tracking accuracy, electrical energy consumption, torque smoothness, and real-time execution under practical operating conditions. Domain randomization and sim-to-real adaptation will be investigated to reduce the gap between simulated and physical systems. Further extensions

will explore heterogeneous multi-robot coordination and adaptive or meta-learning strategies to improve generalization under changing dynamics and operating conditions. Future work will also complete the model-based NMPC/iLQR comparison, the capacity-matched baseline, the equal-budget Pareto-front comparison and the electrical energy evaluation with identified motor parameters, and will combine the controller with energy-optimal reference trajectories [6,7].

The findings establish cooperative MADRL as a promising approach to balancing trajectory-tracking precision and energy efficiency in robotic manipulators. By combining joint-level decomposition with coordinated policy learning, the proposed CTDE-TD3 framework achieved substantial improvements over both centralized and independent-agent TD3 across the evaluated trajectories. The results provide quantitative evidence of the benefits of cooperation under the tested simulation conditions while identifying the theoretical, computational, and experimental considerations that must be addressed before deployment in industrial environments. This work thus provides a foundation for developing scalable, energy-aware, and robust learning-based control systems for advanced robotic manipulation.

Funding: This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data Availability Statement: The datasets generated and/or analyzed during the current study are available from the corresponding author upon reasonable request.

Acknowledgments: The author would like to thank the Automation and Sustainability Research Centre (ASRC) and the AI and Metaverse Research Centre (AIMRC) at the University of Technology Bahrain for providing the research facilities and support that made this work possible. During the preparation of this manuscript, the author used Claude (Opus 5.5, Anthropic) for language editing, assistance in drafting revised text during the review process. The AI tool was not used to design the study, generate or analyze the data, or produce the scientific results.

Conflicts of Interest: The author declares that there are no conflicts of interest.

References

1. Sachs, J.D.; Schmidt-Traub, G.; Mazzucato, M.; Messner, D.; Nakicenovic, N.; Rockström, J. Six transformations to achieve the Sustainable Development Goals. *Nat. Sustain.* **2019**, *2*, 805–814. [CrossRef]
2. Paryanto, P.; Brossog, M.; Bornschlegl, M.; Franke, J. Reducing the energy consumption of industrial robots in manufacturing systems. *Int. J. Adv. Manuf. Technol.* **2015**, *78*, 1315–1328. [CrossRef]
3. Carabin, G.; Wehrle, E.; Vidoni, R. A review on energy-saving optimization methods for robotic and automatic systems. *Robotics* **2017**, *6*, 39. [CrossRef]
4. Palomba, I.; Wehrle, E.; Carabin, G.; Vidoni, R. Minimization of the energy consumption in industrial robots through regenerative drives and optimally designed compliant elements. *Appl. Sci.* **2020**, *10*, 7475. [CrossRef]
5. Lozer, F.; Scalera, L.; Boscariol, P.; Gasparetto, A. Planning optimal minimum-jerk trajectories for redundant robots. *Robot. Auton. Syst.* **2025**, *192*, 105049. [CrossRef]
6. Donà, D.; Franzese, G.; Della Santina, C.; Boscariol, P.; Lenzo, B. Real-time generation of near minimum-energy trajectories via constraint-informed residual learning: A paradigm for learning from optimal solutions. *IEEE Robot. Autom. Mag.* **2026**, *33*, 142–150. [CrossRef]
7. Van Oosterwyck, N.; Vanbecelaere, F.; Knaepkens, F.; Monte, M.; Stockman, K.; Cuyt, A.; Derammelaere, S. Energy optimal point-to-point motion profile optimization. *Mech. Based Des. Struct. Mach.* **2024**, *52*, 239–256. [CrossRef]
8. Mustary, S.; Kashem, M.A.; Sony, J.M.; Hossain, N.; Chowdhury, M.A. Advancing Stability in Robot Manipulators: A Review of Recent Progress and Parameters. *Eng. Rep.* **2025**, *7*, e70207. [CrossRef]
9. Tinoco, V.; Silva, M.F.; Santos, F.N.; Morais, R.; Magalhães, S.A.; Oliveira, P.M. A review of advanced controller methodologies for robotic manipulators. *Int. J. Dyn. Control* **2025**, *13*, 36. [CrossRef]
10. Hui, J. Nonlinear extended state observer-based model-free near-optimal sliding mode water level controller of an inverted U-tube steam generator. *Eng. Appl. Artif. Intell.* **2026**, *163*, 112755. [CrossRef]

11. Chiddarwar, S.S.; Babu, N.R. Multi-agent system for off-line coordinated motion planning of multiple industrial robots. *Int. J. Adv. Robot. Syst.* **2011**, *8*, 11. [CrossRef] [PubMed]
12. Wang, Y.; Damani, M.; Wang, P.; Cao, Y.; Sartoretti, G. Distributed reinforcement learning for robot teams: A review. *Curr. Robot. Rep.* **2022**, *3*, 239–257. [CrossRef]
13. Orr, J.; Dutta, A. Multi-agent deep reinforcement learning for multi-robot applications: A survey. *Sensors* **2023**, *23*, 3625. [CrossRef] [PubMed]
14. Gronauer, S.; Diepold, K. Multi-agent deep reinforcement learning: A survey. *Artif. Intell. Rev.* **2022**, *55*, 895–943. [CrossRef]
15. Hady, M.A.; Hu, S.; Pratama, M.; Cao, Z.; Kowalczyk, R. Multi-agent reinforcement learning for resources allocation optimization: A survey. *Artif. Intell. Rev.* **2025**, *58*, 354. [CrossRef]
16. Shili, M.; Hammedi, S.; Chaoui, H.; Nouri, K. Energy-Aware Swarm Robotics in Smart Microgrids Using Quantum-Inspired Reinforcement Learning. *Electronics* **2025**, *14*, 4210. [CrossRef]
17. Han, D.; Mulyana, B.; Stankovic, V.; Cheng, S. A survey on deep reinforcement learning algorithms for robotic manipulation. *Sensors* **2023**, *23*, 3762. [CrossRef] [PubMed]
18. Hazem, Z.B.; Saidi, F.; Guler, N.; Altaif, A.H. A Hybrid Reinforcement Learning Framework Combining TD3 and PID Control for Robust Trajectory Tracking of a 5-DOF Robotic Arm. *Automation* **2025**, *6*, 56. [CrossRef]
19. Hazem, Z.B.; Saidi, F.; Guler, N.; Altaif, A.H. Reinforcement learning-based intelligent trajectory tracking for a 5-DOF Mitsubishi robotic arm: Comparative evaluation of DDPG, LC-DDPG, and TD3-ADX. *Int. J. Intell. Robot. Appl.* **2025**, *9*, 1982–2002. [CrossRef]
20. Leottau, D.L.; Ruiz-del-Solar, J.; Babuška, R. Decentralized reinforcement learning of robot behaviors. *Artif. Intell.* **2018**, *256*, 130–159. [CrossRef]
21. Perrusquía, A.; Yu, W.; Li, X. Multi-agent reinforcement learning for redundant robot control in task-space. *Int. J. Mach. Learn. Cybern.* **2021**, *12*, 231–241. [CrossRef]
22. Low, D.; Zhou, Y. Cooperative multi-agent reinforcement learning for robotic systems: A review. *Multiagent Grid Syst.* **2025**, *21*, 96–123. [CrossRef]
23. Matignon, L.; Laurent, G.J.; Le Fort-Piat, N. Independent reinforcement learners in cooperative Markov games: A survey regarding coordination problems. *Knowl. Eng. Rev.* **2012**, *27*, 1–31. [CrossRef]
24. Rashid, T.; Samvelyan, M.; Schroeder de Witt, C.; Farquhar, G.; Foerster, J.; Whiteson, S. Monotonic value function factorisation for deep multi-agent reinforcement learning. *J. Mach. Learn. Res.* **2020**, *21*, 1–51. [CrossRef]
25. Du, W.; Ding, S. A survey on multi-agent deep reinforcement learning: From the perspective of challenges and applications. *Artif. Intell. Rev.* **2021**, *54*, 3215–3238. [CrossRef]
26. Nguyen, T.T.; Nguyen, N.D.; Nahavandi, S. Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications. *IEEE Trans. Cybern.* **2020**, *50*, 3826–3839. [CrossRef] [PubMed]
27. Zhang, F.; Li, J.; Li, Z. A TD3-based multi-agent deep reinforcement learning method in mixed cooperation-competition environment. *Neurocomputing* **2020**, *411*, 206–215. [CrossRef]
28. Wong, A.; Bäck, T.; Kononova, A.V.; Plaat, A. Deep multiagent reinforcement learning: Challenges and directions. *Artif. Intell. Rev.* **2023**, *56*, 5023–5056. [CrossRef]
29. Sumiea, E.H.; Abdulkadir, S.J.; Alhussian, H.S.; Al-Selwi, S.M.; Alqushaibi, A.; Ragab, M.G.; Fati, S.M. Deep deterministic policy gradient algorithm: A systematic review. *Heliyon* **2024**, *10*, e30697. [CrossRef] [PubMed]
30. Ben Hazem, Z.; Guler, N.; Altaif, A.H. A study of advanced mathematical modeling and adaptive control strategies for trajectory tracking in the Mitsubishi RV-2AJ 5-DOF robotic arm. *Discov. Robot.* **2025**, *1*, 2. [CrossRef]
31. Ben Hazem, Z.; Guler, N.; Altaif, A.H. Model-free trajectory tracking control of a 5-DOF Mitsubishi robotic arm using deep deterministic policy gradient algorithm. *Discov. Robot.* **2025**, *1*, 4. [CrossRef]
32. Littman, M.L. Value-function reinforcement learning in Markov games. *Cogn. Syst. Res.* **2001**, *2*, 55–66. [CrossRef]
33. Bernstein, D.S.; Givan, R.; Immerman, N.; Zilberstein, S. The complexity of decentralized control of Markov decision processes. *Math. Oper. Res.* **2002**, *27*, 819–840. [CrossRef]
34. Asada, H. A geometrical representation of manipulator dynamics and its application to arm design. *J. Dyn. Syst. Meas. Control* **1983**, *105*, 131–142. [CrossRef]
35. Tassa, Y.; Erez, T.; Todorov, E. Synthesis and stabilization of complex behaviors through online trajectory optimization. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vilamoura, Portugal, 7–12 October 2012; pp. 4906–4913. [CrossRef]
36. Grüne, L.; Pannek, J. *Nonlinear Model Predictive Control: Theory and Algorithms*; Springer: Cham, Switzerland, 2017. [CrossRef]

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.