OMNIPIANO: DIVERSE DEXTEROUS PIANO-PLAYING CHALLENGES FOR STANDARD, ROBUST, SAFE, AND MULTI-AGENT RL

Anonymous authors
Paper under double-blind review

ABSTRACT

Dexterous robotic manipulation remains a major challenge in robotics, exemplified by piano playing, which requires coordinated control of fingers and provides a demanding testbed for reinforcement learning (RL). Although previous work established a benchmark for robotic piano playing, it is restricted to two hands and does not support unified evaluation. To address this gap, we introduce ***OmniPiano***, a benchmark that provides one to up to five Shadow Hands and supports standard, robust, safe, and multi-agent RL within a shared piano-playing task family. ***OmniPiano*** is developed for scalable and diverse piano-playing tasks with configurable perturbations, explicit safety constraints, and decentralized cooperation settings. Particularly, to facilitate usability and extendability, ***OmniPiano*** adopts a highly modular design and provides comprehensive tasks to support RL study of task performance, robustness, safety, and cooperation in dexterous control. With at least 912 task settings and 36 baseline algorithms, ***OmniPiano*** further incorporates LLM-based agents to broaden the evaluation scope and reveal new insights. Extensive evaluations show that state-of-the-art RL algorithms and frontier LLM-based agents still struggle on challenging piano-playing tasks, even under standard RL settings, highlighting substantial room for improvement in dexterous control. The code, dataset, tutorial, and demo are available at `https://omnipiano.site`.

1 INTRODUCTION

Reinforcement learning (RL) has made substantial progress in games (Mnih et al., 2015; Vinyals et al., 2019), continuous control (Duan et al., 2016), and robotic manipulation (Levine et al., 2016), yet learning precise, coordinated behavior in high-dimensional, contact-rich systems remains challenging. Dexterous piano playing exemplifies this challenge and provides a suitable testbed for evaluating RL methods under precise temporal and coordination requirements. In this task, RL methods learn to coordinate finger movements to press, hold, and release keys at the times specified by a Musical Instrument Digital Interface (MIDI) score (Xu et al., 2022), which provides a precise temporal target for evaluating RL performance through key-activation accuracy and timing.

The representative benchmark, RoboPianist, uses simulated Shadow Hands for dexterous piano playing (Zakka et al., 2023). However, its limited two-hand setting hinders scalability as the number of hands and control dimensions increase. More broadly, a unified dexterous piano playing benchmark for robust RL, safe RL, and multi-agent RL is still lacking. These gaps motivate a benchmark that extends the previous work to multi-hand configurations and supports systematic evaluation under perturbations, explicit safety constraints, and decentralized control, with shared musical objectives and performance measures.

To address this gap, we introduce ***OmniPiano***, a benchmark for standard, robust, safe, and multi-agent RL through dexterous piano playing. As illustrated in Figure 1, ***OmniPiano*** comprises four tracks built on a shared piano-playing core, each varying a different aspect of the same task: *standard* scales repertoire and hand configurations, *robust* injects interaction and environment perturbations, *safe* separates the task reward from safety costs, and *multi-agent* distributes the hands across decentralized agents. Rather than combining every factor within a single environment, this design pre-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

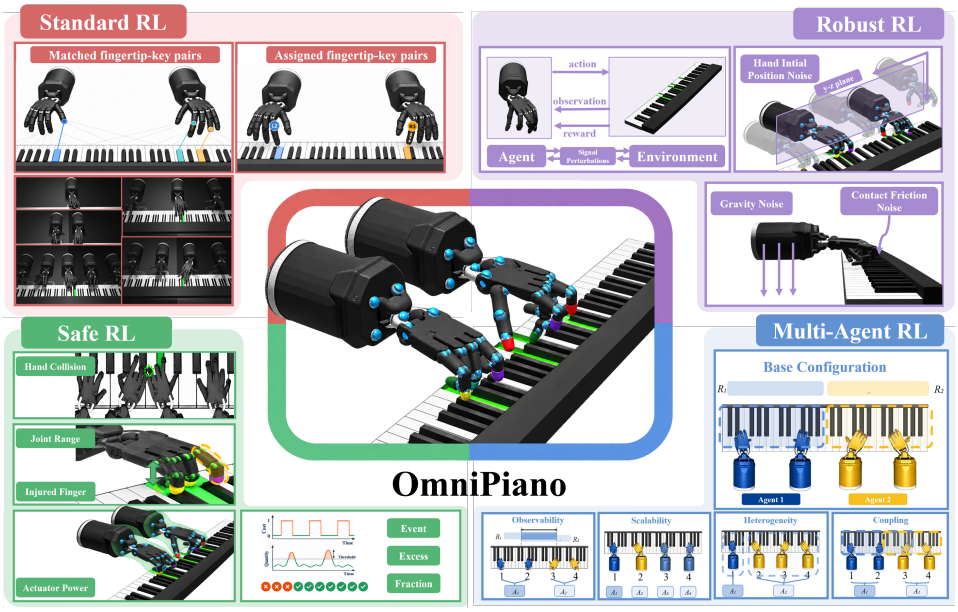


Figure 1: The overview of ***OmniPiano***. A unified benchmark for learning dexterous piano playing.

serves the core piano-playing task while varying one factor at a time, so that performance differences can be attributed to a single design choice. As shown in Table 1, ***OmniPiano*** leverages challenging dexterous control with broad coverage of RL evaluation dimensions. Its unified task setting supports robust, safe, and multi-agent RL alongside LLM-agent evaluation, bringing together capabilities that remain fragmented across existing benchmarks.

OmniPiano is implemented in MuJoCo and exposes all four task families through consistent Gymnasium (single-agent) and PettingZoo (multi-agent) interfaces (Todorov et al., 2012; Terry et al., 2021; Towers et al., 2025), enabling algorithms to be compared across settings without task-specific wrappers. The benchmark comprises 72, 216, 480, and 144 task settings across the standard, robust, safe, and multi-agent tracks, respectively. Our evaluation covers 10 standard RL methods, 4 robust RL methods, 14 safe RL methods, and 8 multi-agent RL methods, together with 4 LLM-based agents, for a total of 912 task settings and 36 baselines (as shown in Appendix Table 6). Overall, these tracks provide a foundation for studying RL in high-dimensional dexterous control.

Our main contributions are as follows:

- **A unified evaluation framework beyond single evaluation track.** We extend the piano-playing task into four task families covering standard, robust, safe, and multi-agent RL, with at least 912 tasks in total. This design broadens the scope of piano-playing benchmarks from task proficiency to the conditions under which that proficiency can be maintained.

- **A comprehensive and reproducible foundation for algorithm comparison.** We provide 36 representative RL baselines and 4 LLM-based agents, along with task-specific metrics, standardized evaluation protocols, and unified Gymnasium and PettingZoo interfaces, enabling systematic and reproducible comparison across the benchmark.

- **A scalable benchmark for multi-hand dexterous control.** We introduce ***OmniPiano***, providing one to up to five Shadow Hands and 111 continuous action dimensions. This expansion enables systematic investigation of how RL scales with control dimensionality, workspace coupling, and coordination demands.

Due to space limits, we provide preliminaries in Appendix A and related work in Appendix B.

Table 1: The comparison of ***OmniPiano*** with related reinforcement-learning benchmarks.

Feature	RoboPianist	Robust-Gym.	Safety-Gym.	***OmniPiano***
Action dimension	45	1–30	2–17	**23–111**
Dexterous piano playing	✓	✗	✗	✓
Robust RL track	✗	✓	✗	✓
Safe RL track	✗	✓	✓	✓
Multi-agent RL track	✗	✓	✓	✓
Unified task setting	✗	✗	✗	✓
LLM-agent evaluation	✗	✗	✗	✓

2 OMNIPIANO

We now introduce ***OmniPiano***, a unified benchmark for standard, robust, safe, and multi-agent reinforcement learning through dexterous piano playing. Each benchmark task comprises three components: an embodied agent configuration comprising one to five Shadow Hands, a musical objective specified by a MIDI score, and a task-specific design that modifies the agent–environment interaction. This modular construction yields four complementary settings: standard RL measures task capability; robust RL introduces controlled perturbations to observations, actions, rewards, and the physical environment; safe RL augments the task with explicit costs for joint, collision, power, and injury constraints; and multi-agent RL distributes the hands across decentralized agents that pursue a shared musical objective. In the following sections, we describe the task and environment foundations, with the design of each benchmark setting.

2.1 TASK AND ENVIRONMENT FOUNDATIONS

In this section, we describe the shared task and environment foundations of ***OmniPiano***. We use the RoboPianist (Zakka et al., 2023) framework as our starting point, built on the MuJoCo physics engine (Todorov et al., 2012; Tunyasuvunakool et al., 2020). Based on this framework, we introduce a new piano-playing environment that supports multiple Shadow Hands and diverse task designs. The environment provides a unified interface for users to configure the number of hands, songs to play, and task designs for standard, robust, safe, and multi-agent reinforcement learning. And the environment is compatible with popular reinforcement learning libraries such as Stable-Baselines3 (Raffin et al., 2021), RLlib (Liang et al., 2018), OmniSafe (Ji et al., 2024) and torchRL (Bou et al., 2023) enabling users to easily train and evaluate their policies.

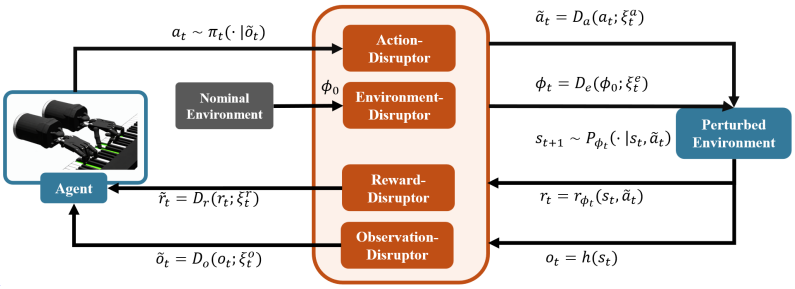


Figure 2: The overview of the Robust RL task design. Robustness perturbations are introduced through four independent disruptors acting on the agent–environment interaction loop.

2.2 STANDARD RL BENCHMARK DESIGN

Hand settings. ***OmniPiano*** includes up to 150 songs and eight hand settings, with the latter varying in both the number of hands (from one to five) and their mobility configurations. Specifically, *unrestricted* denotes the fully mobile setting, in which each hand can move freely across the keyboard, whereas *restricted* assigns each hand to a fixed region. We introduce restricted configurations to improve the learnability of multi-hand tasks: because with the increasing of the number of hands, full

mobility enlarges the joint action and coordination space, making exploration and credit assignment increasingly difficult for reinforcement learning.

Reward design. Following RoboPianist, we formulate the reward as Equation 1:

$$r_t = r_t^{\text{key}} + r_t^{\text{match}} - \lambda_{\text{energy}} r_t^{\text{energy}}, \tag{1}$$

where r_t^{key} rewards pressing the target keys while penalizing false-positive key activations, r_t^{match} encourages assigned fingertips to approach their target keys, and r_t^{energy} regularizes excessive joint actuation. We provide two variants that differ only in the fingertip-to-key assignment, *fingering-annotated reward* and *optimal transport (OT) reward*. Details can be seen in D.1.1. And evaluation metric F1 is described in Appendix D.1.2.

2.3 Robust RL Benchmark Design

OmniPiano evaluates robustness in the closed-loop agent–environment interaction. Illustrated in Figure 2, we consider two types of perturbation: *(i) signal perturbations*, which affect the observations, actions, or rewards exchanged during interaction; and *(ii) environment perturbations*, which alter the simulated physical conditions, including gravity, fingertip–key contact-friction, and the initial pose of each hand. Table 4 summarizes the resulting perturbation axes and canonical levels.

Signal perturbations. The operators D_o, D_a, and D_r perturb observations, actions, and rewards, respectively. D_o perturbs the observed hand and piano states, while keeping the fingering targets unchanged; D_a models actuation uncertainty before execution; and D_r perturbs the reward signal.

Environment perturbations. The operator D_e modifies environment through gravity, contact-friction, and initial hand-pose perturbations, affecting fingertip–key interaction, and initial reach. Gravity and friction can vary per episode or step, while hand-pose offsets are sampled at reset.

Composable perturbations. All perturbation channels can be combined flexibly, with independently specified distributions, magnitudes, and frequencies, enabling controlled evaluation of both isolated and compound robustness effects under the same underlying musical objective.

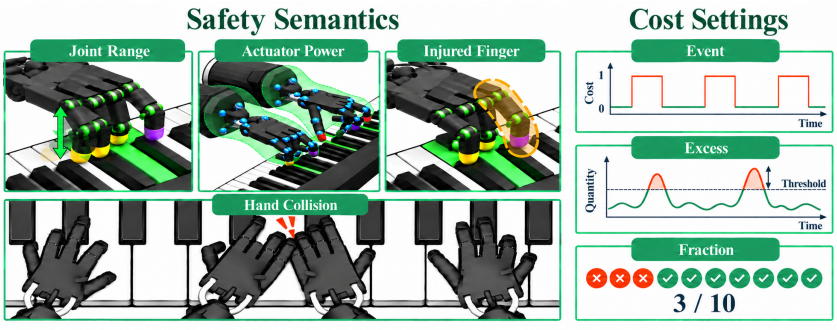


Figure 3: The overview of the safety task design in ***OmniPiano***. Four safety semantics, Joint Range, Actuator Power, Injured Finger, and Hand Collision, define the constrained behaviors. Event, Fraction, and Excess provide alternative measures of constraint violations.

2.4 Safe RL Benchmark Design

Accurate piano playing does not necessarily imply safe control. Policies with similar musical performance may differ substantially in their joint configurations, contact forces, and actuation demands. ***OmniPiano*** therefore separates task reward from safety cost to enable systematic study of musical performance under explicit safety constraints and different safety-performance trade-offs.

Reward design. All safety tasks share a common musical objective and use a fingering-free OT reward as Equation 1. By matching fingertips to target keys without prescribing a fixed fingering, this reward allows policies to adapt finger assignments to the safety constraint. We remove the actuator-power penalty from the base reward so that power is handled through the independent safety cost, making the trade-off between musical performance and constraint satisfaction explicit.

Safety semantics. As illustrated in Figure 3, ***OmniPiano*** considers four safety semantics: (1) **Joint Range** restricts selected joint positions to prescribed intervals; (2) **Actuator Power** limits instantaneous mechanical power; (3) **Injured Finger** imposes additional restrictions on a designated finger to represent an injury-related control limitation; and (4) **Hand Collision** limits unintended contact forces between hands in the shared keyboard workspace. These constraints capture different physical safety aspects, with thresholds chosen as benchmark parameters for controlled evaluation.

Cost settings. Violations are quantified through three cost formulations: (1) **Event** records whether a violation occurs at the current control step; (2) **Fraction** measures the proportion of monitored elements exceeding their limits; and (3) **Excess** measures the normalized deviation beyond an admissible interval or threshold. They capture violation occurrence, extent, and severity, respectively, enabling comparisons across both physical constraints and representations of the same constraint. The applicable formulation depends on the structure of the safety signal; in particular, Fraction is defined only for constraints involving a set of monitored elements.

Task construction and evaluation. Each task comprises a musical piece and hand configuration with the shared OT reward, one selected safety semantic, one selected cost setting, and an episodic cost limit. Policies maximize musical return subject to this limit, and evaluation jointly assesses playing quality and constraint satisfaction.

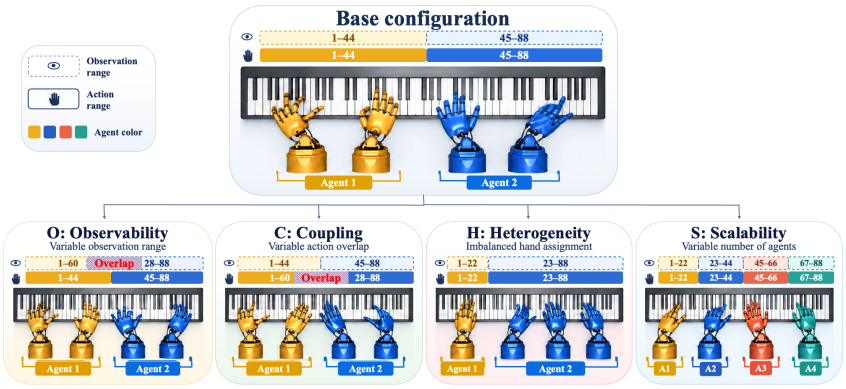


Figure 4: The overview of the multi-agent RL task design. The configuration shown is a typical example. In practice, users can freely configure the MIDI score, number of hands and agents, hand-to-agent assignments, and agent-specific observation and action ranges and overlaps.

2.5 Multi-Agent RL Benchmark Design

OmniPiano organizes MARL evaluation around four configurable dimensions, **Scalability, Coupling, Heterogeneity**, and **Observability (SCHO)**. Under the same hand configuration, MIDI objective, and shared reward, these dimensions vary different aspects of multi-agent control setup. Specifically, **Scalability** varies how many agents control the same set of Shadow Hands. More agents will split the same physical task across more independent decision makers. **Coupling** varies the overlap between agents' keyboard action ranges. More overlap lets multiple agents act on the same keys, increasing coordination demands and the chance of interfering actions. **Heterogeneity** varies how the Shadow Hands are divided among agents. Balanced assignments give agents similar control loads, while imbalanced assignments create unequal action dimensions and keyboard responsibilities. Finally, **Observability** varies the scope of each agent's local keyboard and teammate observations. Narrower observations give agents less information about the shared task without changing what they control. An illustrative 4-hands example is provided as Figure 4.

3 Experiments and Analysis

We will first introduce the task settings, evaluation results and key findings for standard RL, robust RL, safe RL, and multi-agent RL in Sections 3.1, 3.2, 3.3 and 3.4. Due to space constraints, additional experiments and detailed settings are provided in Appendix C and Appendix D, respectively.

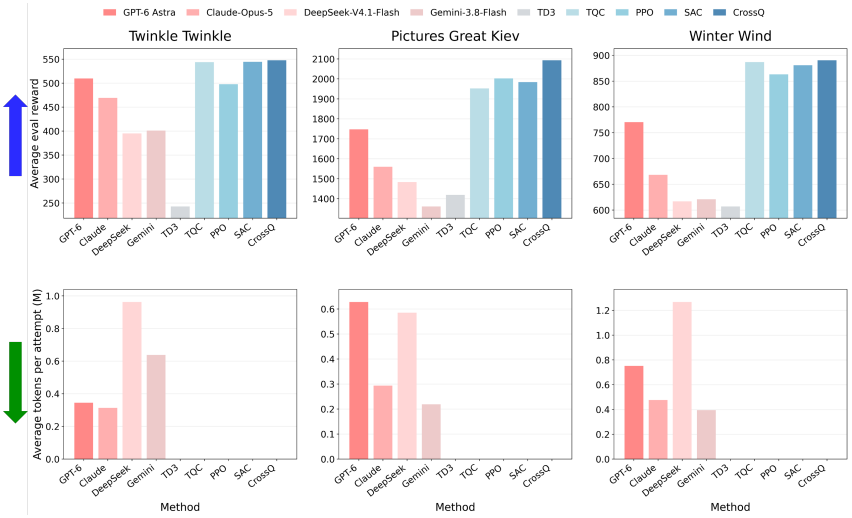


Figure 5: Average evaluation **reward** (↑: top row, higher is better) and mean **token usage** (↓: bottom row, lower is better) per optimization attempt for LLMs across ten keyframe-optimization iterations and RL methods over 5M training steps in the two-hand setting on three piano pieces.

3.1 Evaluation of Standard RL Baselines

We first examine the relationship between song difficulty and task performance across representative piano pieces (Table 2). We compare episode length, measured in control steps at 20 Hz, with note density, measured by Notes Per Second (NPS). Episode length alone does not reflect difficulty consistently: *Gymnopédie No.1* is longer than *French Suite No.1 Allemande* (657 vs. 517 steps) yet achieves higher F1 (0.88 vs. 0.59), while its lower NPS (1.77 vs. 10.89) better matches the performance gap, suggesting that NPS is more informative of task difficulty than duration alone.

We then evaluate PPO (Schulman et al., 2017) and SAC (Haarnoja et al., 2018) as standard RL baselines and compare them with large language models (LLMs). Experiments are conducted on three pieces of increasing difficulty: *Twinkle Twinkle* (161 steps), *Pictures of Great Kiev* (720 steps), and *WinterWind* (314 steps), with one to five hands. For three- to five-hand tasks, both unrestricted and restricted hand assignments are considered (Figure 6). Additional details on the LLM-based agents are provided in Appendix D.1.3. Across these settings, increasing the number of hands does not consistently improve performance. Meanwhile, LLM-based agents remain competitive with RL baselines and achieve higher F1 on challenging pieces (*WinterWind*) as the number of hands increases.Figure 10 further compares the two assignment schemes under standard RL. Restricted assignments achieve faster reward convergence and higher final returns than unrestricted assignments.

For the two-hand setting, we further compare PPO, SAC, CrossQ (Bhatt et al., 2024), TD3 (Qian et al., 2025b), TQC (Kuznetsov et al., 2022), and LLM-based agent methods (Figure 5, Figure 11). Over 5M environment steps, RL methods exploit the reward signal more consistently and achieve more stable F1 improvements. In contrast, LLM-based agents can obtain F1 gains within only 10 optimization attempts, but with higher variance across attempts and substantial token consumption.

Findings:

- ***Song duration alone does not explain task difficulty.*** *Note density is more informative, while difficulty also depends on fingering complexity, hand coordination, and spatial key transitions.*
- ***More hands do not guarantee better F1 score****, suggesting that additional control freedom can increase exploration and coordination difficulty.*
- ***Both RL and LLM-based agents struggle on* OmniPiano***, but exhibit distinct optimization behaviors. RL improves more steadily, whereas LLM-based agents can achieve rapid F1 gains within a few attempts at substantial token cost.*

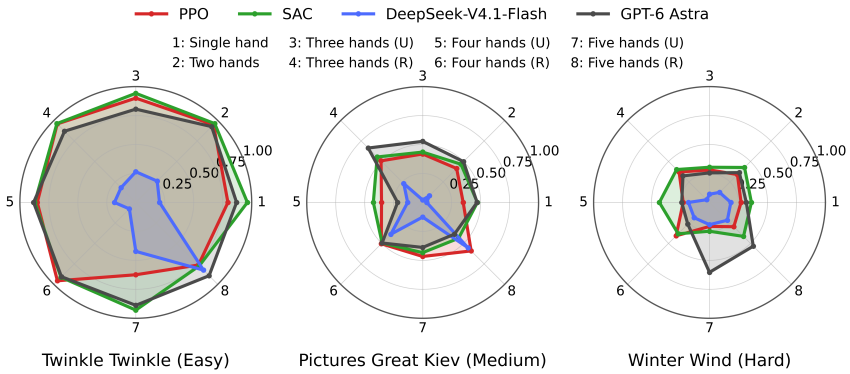


Figure 6: F1-score comparison of PPO, SAC, DeepSeek-V4.1-Flash, and GPT-6 Astra across eight hand settings on three piano pieces. The details of F1 metric can be found in Appendix D.1.2.

3.2 Evaluation of Robust RL Baselines

In this section, we evaluate robust RL under different perturbation channels, distributions, and magnitudes on two tasks: two-hand *Clair de Lune* and three-hand *Für Elise*. We begin by evaluating PPO under perturbations to action, observation, reward, gravity, contact-friction, and hand-pose, considering Gaussian, uniform, and shift distributions to characterize sensitivity across different disturbance (Figure 7). Among these perturbations, action and hand-pose disturbances produce the largest performance degradation, while contact-friction and gravity are generally less disruptive.

We then compare PPO, SAC, A2PSAC (Liu et al., 2024), SCPO (Kuang et al., 2021), and EPPO (Akgül et al., 2025) on two-hand *Clair de Lune* under Gaussian action and reward perturbations (Figure 12). A2PSAC and SAC maintain higher performance, whereas EPPO performs weakest overall, revealing substantial differences in robustness across algorithms. Finally, we move beyond isolated perturbations and evaluate composable disturbances on three-hand *Für Elise*, progressively combining Observation (O), Reward (R), Action (A), Gravity (G), and Hand-Pose (HP) perturbations to assess robustness under increasingly complex distribution shifts (Figure 13). Detailed analysis of algorithms and disturbance compositions are provided in Appendix C.2.

***Finding: Action and hand-pose perturbations are most disruptive**, highlighting actuation and initialization as key robustness challenges.*

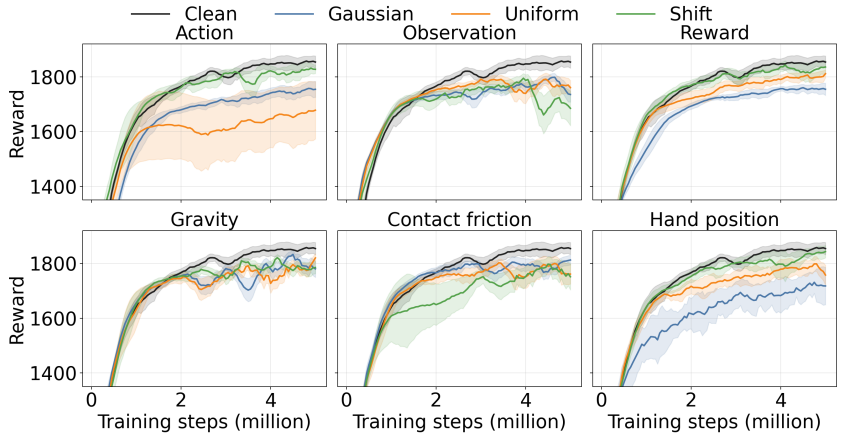
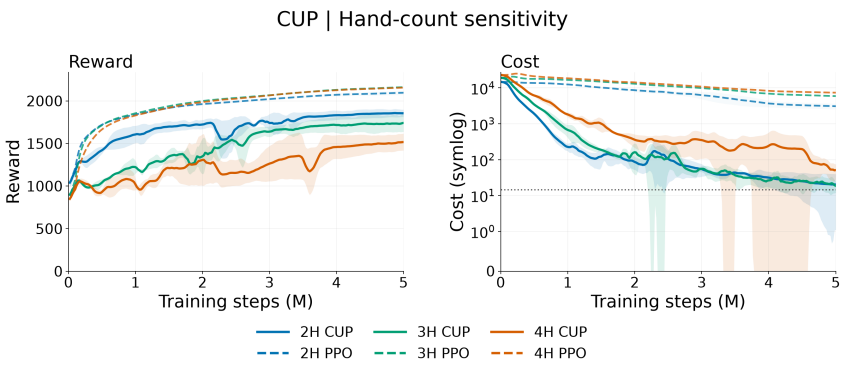
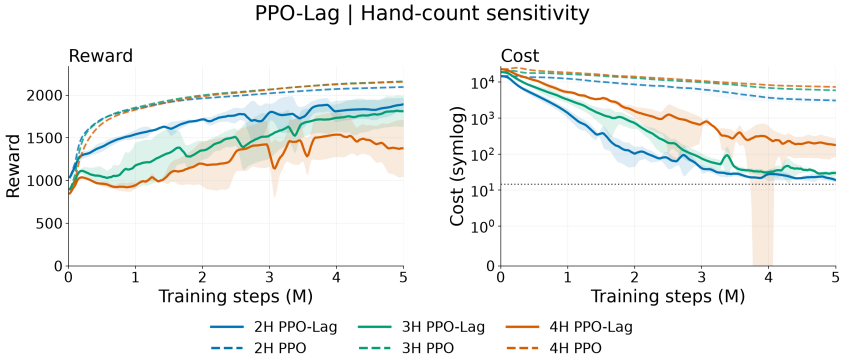


Figure 7: Periodic evaluation reward of PPO across six robustness channels under Gaussian, uniform, and shift perturbations, with the clean environment shown as a reference.



(a) Hand count sensitivity of CUP.



(b) Hand count sensitivity of PPO-Lag.

Figure 8: Effect of hand count on safety RL: training reward and safety cost for 2 to 4 hands. Solid lines denote CUP / PPO-Lag; dashed lines denote unconstrained PPO. The dotted line marks the cost budget. In this task, four hands policies achieve lower rewards and show greater difficulty reducing safety costs than two or three hands policies.

3.3 Evaluation of Safe RL Baselines

We compare four safe RL algorithms, PPO-Lag, OnCRPO (Xu et al., 2021), CPO (Achiam et al., 2017), and CUP (Yang et al., 2022) on ***OmniPiano*** tasks with two to four hands, using PPO as a reference. All methods use the OT reward and are trained for 5M environment steps with three random seeds, as shown in 8 and Appendix C.3. We examine reward and safety cost jointly to assess whether agents learn to reduce constraint violations while maintaining piano playing performance. Figure 8 shows hand count sensitivity results for CUP and PPO-Lag; results for OnCRPO and CPO are provided in Appendix C.3.1. The key conclusion is summarized in the finding.

We further study sensitivity to the cost budget, as shown in Appendix C.3.3. The budget study varies the cost limit while keeping the task fixed, examining how each algorithm balances reward and safety. The hand count study compares two, three, and four hands configurations with the same piece, safety semantic, and cost definition. Additional results and experimental settings are provided in Appendix C.3 and D.3.

> ***Finding: Increasing the number of hands can make safe RL more difficult.*** *In the evaluated task, policies controlling four hands struggle more to reduce safety costs and achieve lower rewards than those controlling two or three hands.*

3.4 Evaluation of Multi-Agent RL Baselines

We evaluate representative MARL algorithms under a base task and four SCHO variants. The tasks are build on the same four-hand piece, *WinterWind*. Specifically, the **Base task** uses two agents

with a balanced two-hand assignment, disjoint keyboard action ranges, and localized partial observations; the **Observability variant** keeps the same balanced assignment and action ranges as Base, but provides broader local observations; the **Coupling variant** uses two balanced agents with overlapping keyboard action ranges while retaining the Base observation ranges; the **Heterogeneity variant** assigns one hand to one agent and three hands to the other, creating asymmetric control loads and action dimensions; and the **Scalability variant** uses four agents to control the same four hands, with one hand assigned to each agent. All settings share the same physical hands, MIDI objective, team reward, and training budget. For algorithms, we compare IPPO(Schroeder de Witt et al., 2020a), MAPPO(Yu et al., 2022), HAPPO(Kuba et al., 2022), A2PO(Wang et al., 2023), and FACMAC(Peng et al., 2021), and include centralized PPO(Schulman et al., 2017) as a base reference with a single agent accessing to the full observation and action space. Figure 9 shows the task-wise evaluation return for each algorithm. Evaluation uses the reward in Equation 1 for benchmark consistency, while during training we add an additional shared per-step penalty for inter-agent hand contact. Detailed task setup and experiment analysis are provided in Appendix C.4.

Finding: More agents can be beneficial for learning dexterous piano playing. With the same four-hand setting, performance improves from centralized PPO with one agent, to the two-agent Base setting, and further to the four-agent Scalability setting. This suggests that distributing the task across more agents can lead to better learning.*

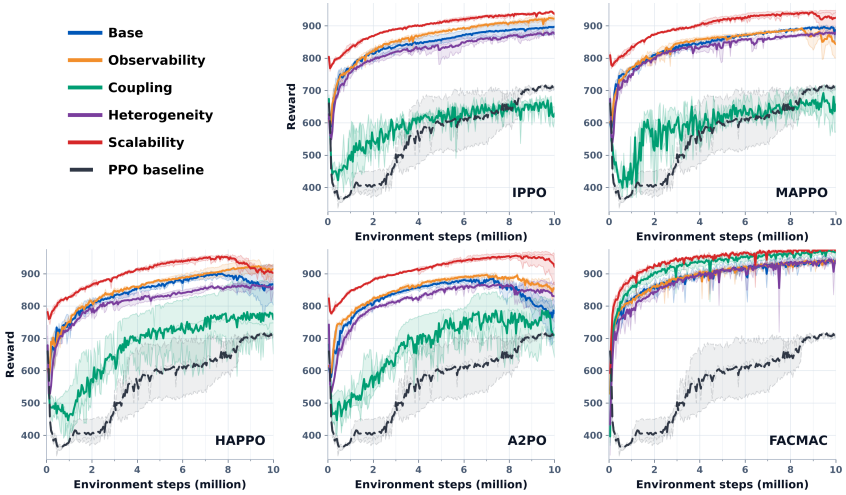


Figure 9: Task-wise training performance of representative MARL algorithms across the five settings. Compared with Base, Scalability and Observability variant generally improve performance, while Heterogeneity and especially Coupling variant make policy learning more difficult. Among the variants, Coupling most clearly differentiates the algorithms.

4 Conclusion

We introduced ***OmniPiano***, a unified benchmark for dexterous piano playing that supports one to five hands, up to 111 continuous action dimensions, 912 task settings, and 36 baseline algorithms across standard, robust, safe, and multi-agent RL. By integrating scalable control, configurable perturbations, explicit safety constraints, decentralized cooperation, and LLM-based agents within a shared task family, ***OmniPiano*** enables systematic evaluation beyond task proficiency. Our experiments show that challenging piano-playing tasks remain difficult for current RL methods and frontier LLM-based agents, even under standard conditions. Together with its modular architecture and public release, ***OmniPiano*** has the potential to become a versatile testbed for evaluating diverse RL algorithms in dexterous control.

*Here, "more agents" means more controllers, not increasing the number of hands or the total action space. Instead, the term "more hands" used earlier refers to increasing the number of hands and the corresponding action space of the task.

REFERENCES

Josh Achiam and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning. 2019. URL https://api.semanticscholar.org/CorpusID:208283920.

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International conference on machine learning*, pp. 22–31. Pmlr, 2017.

Abdullah Akgül, Gulcin Baykal, Manuel Haußmann, and Melih Kandemir. Overcoming non-stationary dynamics with evidential proximal policy optimization, 2025. URL https://arxiv.org/abs/2503.01468.

Qinbo Bai, Amrit Singh Bedi, Mridul Agarwal, Alec Koppel, and Vaneet Aggarwal. Achieving zero constraint violation for constrained reinforcement learning via primal-dual approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 3682–3689, 2022.

Nolan Bard, Jakob N. Foerster, Sarath Chandar, Neil Burch, Marc Lanctot, H. Francis Song, Emilio Parisotto, Vincent Dumoulin, Subhodeep Moitra, Edward Hughes, Iain Dunning, Shibl Mourad, Hugo Larochelle, Marc G. Bellemare, and Michael Bowling. The Hanabi challenge: A new frontier for AI research. *Artificial Intelligence*, 280:103216, 2020. doi: 10.1016/j.artint.2019.103216. URL https://arxiv.org/abs/1902.00506. Peer-reviewed Artificial Intelligence journal version preferred; the benchmark preprint first appeared as arXiv:1902.00506 in 2019.

Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of artificial intelligence research*, 47: 253–279, 2013.

Felix Berkenkamp, Matteo Turchetta, Angela Schoellig, and Andreas Krause. Safe model-based reinforcement learning with stability guarantees. *Advances in neural information processing systems*, 30, 2017.

Aditya Bhatt, Daniel Palenicek, Boris Belousov, Max Argus, Artemij Amiranashvili, Thomas Brox, and Jan Peters. Crossq: Batch normalization in deep reinforcement learning for greater sample efficiency and simplicity, 2024. URL https://arxiv.org/abs/1902.05605.

Albert Bou, Matteo Bettini, Sebastian Dittert, Vikash Kumar, Shagun Sodhani, Xiaomeng Yang, Gianni De Fabritiis, and Vincent Moens. Torchrl: A data-driven decision-making library for pytorch, 2023. URL https://arxiv.org/abs/2306.00577.

Yann Bouteiller, Simon Ramstedt, Giovanni Beltrame, Christopher Pal, and Jonathan Binas. Reinforcement Learning with Random Delays. In *International Conference on Learning Representations*, 2021. URL https://mlanthology.org/iclr/2021/bouteiller2021iclr-reinforcement/.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

Le Chen, Yi Zhao, Jan Schneider, Quankai Gao, Simon Guist, Cheng Qian, Juho Kannala, Bernhard Schölkopf, Joni Pajarinen, and Dieter Büchler. Dexterous robotic piano playing at scale. *arXiv preprint arXiv:2511.02504*, 2025.

Yuanpei Chen, Tianhao Wu, Shengjie Wang, Xidong Feng, Jiechuan Jiang, Zongqing Lu, Stephen McAleer, Hao Dong, Song-Chun Zhu, and Yaodong Yang. Towards human-level bimanual dexterous manipulation with reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 5150–5163. Curran Associates, Inc., 2022. doi: 10.52202/068431-0372. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/217a2a387f52c30755c37b0a73430291-Abstract.html. Peer-reviewed NeurIPS 2022 Datasets and Benchmarks Track version preferred; it introduces the Bi-DexHands benchmark. The citation key is retained as chen2023bidexhands for compatibility with the draft.

Karl Cobbe, Christopher Hesse, Jacob Hilton, and John Schulman. Leveraging procedural generation to benchmark reinforcement learning, 2020. URL https://arxiv.org/abs/1912.01588.

Wenhao Ding, Laixi Shi, Yuejie Chi, and DING ZHAO. Seeing is not believing: Robust reinforcement learning against spurious correlation. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 66328–66363. Curran Associates, Inc., 2023. doi: 10.52202/075280-2895.

Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. In *International conference on machine learning*, pp. 1329–1338. PMLR, 2016.

Benjamin Ellis, Jonathan Cook, Skander Moalla, Mikayel Samvelyan, Mingfei Sun, Anuj Mahajan, Jakob N. Foerster, and Shimon Whiteson. SMACv2: An improved benchmark for cooperative multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 37567–37593. Curran Associates, Inc., 2023. doi: 10.52202/075280-1634. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/764c18ad230f9e7bf6a77ffc2312c55e-Abstract-Datasets_and_Benchmarks.html. Peer-reviewed NeurIPS 2023 Datasets and Benchmarks Track version preferred; preprint available as arXiv:2212.07489.

Jakob N. Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, volume 32, pp. 2974–2982. AAAI Press, 2018. doi: 10.1609/aaai.v32i1.11794. URL https://ojs.aaai.org/index.php/AAAI/article/view/11794. Peer-reviewed AAAI 2018 version preferred; the preprint first appeared as arXiv:1705.08926 in 2017.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4rl: Datasets for deep data-driven reinforcement learning, 2021. URL https://arxiv.org/abs/2004.07219.

Javier Garcıa and Fernando Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(1):1437–1480, 2015.

Jiayuan Gu, Fanbo Xiang, Xuanlin Li, Zhan Ling, Xiqiang Liu, Tongzhou Mu, Yihe Tang, Stone Tao, Xinyue Wei, Yunchao Yao, et al. Maniskill2: A unified benchmark for generalizable manipulation skills. *arXiv preprint arXiv:2302.04659*, 2023a.

Shangding Gu, Jakub Grudzien Kuba, Yuanpei Chen, Yali Du, Long Yang, Alois Knoll, and Yaodong Yang. Safe multi-agent reinforcement learning for multi-robot control. *Artificial Intelligence*, 319:103905, 2023b.

Shangding Gu, Long Yang, Yali Du, Guang Chen, Florian Walter, Jun Wang, and Alois Knoll. A review of safe reinforcement learning: Methods, theories, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):11216–11235, 2024.

Shangding Gu, Laixi Shi, Muning Wen, Ming Jin, Eric Mazumdar, Yuejie Chi, Adam Wierman, and Costas Spanos. Robust gymnasium: A unified modular benchmark for robust reinforcement learning. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/fcc22e5b7d5d2155d994da22d045f0a6-Abstract-Conference.html.

Caglar Gulcehre, Ziyu Wang, Alexander Novikov, Tom Le Paine, Sergio Gomez Colmenarejo, Konrad Zolna, Rishabh Agarwal, Josh Merel, Daniel Mankowitz, Cosmin Paduraru, Gabriel Dulac-Arnold, Jerry Li, Mohammad Norouzi, Matt Hoffman, Ofir Nachum, George Tucker, Nicolas Heess, and Nando de Freitas. Rl unplugged: A suite of benchmarks for offline reinforcement learning, 2021. URL https://arxiv.org/abs/2006.13888.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor, 2018. URL https://arxiv.org/abs/1801.01290.

Sandy Huang, Nicolas Papernot, Ian Goodfellow, Yan Duan, and Pieter Abbeel. Adversarial attacks on neural network policies, 2017. URL https://arxiv.org/abs/1702.02284.

Jiaming Ji, Borong Zhang, Jiayi Zhou, Xuehai Pan, Weidong Huang, Ruiyang Sun, Yiran Geng, Yifan Zhong, Josef Dai, and Yaodong Yang. Safety Gymnasium: A Unified Safe Reinforcement Learning Benchmark. In *Neural Information Processing Systems*, 2023. URL https://mlanthology.org/neurips/2023/ji2023neurips-safety/.

Jiaming Ji, Jiayi Zhou, Borong Zhang, Juntao Dai, Xuehai Pan, Ruiyang Sun, Weidong Huang, Yiran Geng, Mickel Liu, and Yaodong Yang. Omnisafe: An infrastructure for accelerating safe reinforcement learning research. *Journal of Machine Learning Research*, 25(285):1–6, 2024.

Woosik Kim, Wonhyeok Choi, and Sunghoon Im. Cantabile: Learning expressive dynamics for robotic piano performance, 2026. URL https://arxiv.org/abs/2609.18213.

Bettina Könighofer, Roderick Bloem, Nils Jansen, Sebastian Junges, and Stefan Pranger. Shields for safe reinforcement learning. *Communications of the ACM*, 68(11):80–90, 2025.

Yufei Kuang, Miao Lu, Jie Wang, Qi Zhou, Bin Li, and Houqiang Li. Learning robust policy against disturbance in transition dynamics via state-conservative policy optimization, 2021. URL https://arxiv.org/abs/2112.10513.

Jakub Grudzien Kuba, Ruiqing Chen, Muning Wen, Ying Wen, Fanglei Sun, Jun Wang, and Yaodong Yang. Trust region policy optimisation in multi-agent reinforcement learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=EcGGFkNTxdJ. Peer-reviewed ICLR 2022 paper introducing HATRPO and HAPPO; the preprint first appeared as arXiv:2109.11251 in 2021.

Karol Kurach, Anton Raichuk, Piotr Stańczyk, Michał Zając, Olivier Bachem, Lasse Espeholt, Carlos Riquelme, Damien Vincent, Marcin Michalski, Olivier Bousquet, and Sylvain Gelly. Google Research Football: A novel reinforcement learning environment. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, volume 34, pp. 4501–4510. AAAI Press, 2020. doi: 10.1609/aaai.v34i04.5878. URL https://ojs.aaai.org/index.php/AAAI/article/view/5878. Peer-reviewed AAAI 2020 version preferred; the environment was first described in arXiv:1907.11180 in 2019.

Arsenii Kuznetsov, Alexander Grishin, Artem Tsypin, Arsenii Ashukha, Artur Kadurin, and Dmitry Vetrov. Automating control of overestimation bias for reinforcement learning, 2022. URL https://arxiv.org/abs/2110.13523.

Erwan Lecarpentier and Emmanuel Rachelson. Non-stationary markov decision processes, a worst-case approach using model-based reinforcement learning, extended version, 2020. URL https://arxiv.org/abs/1904.10090.

Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17(39):1–40, 2016.

Eric Liang, Richard Liaw, Philipp Moritz, Robert Nishihara, Roy Fox, Ken Goldberg, Joseph E. Gonzalez, Michael I. Jordan, and Ion Stoica. Rllib: Abstractions for distributed reinforcement learning, 2018. URL https://arxiv.org/abs/1712.09381.

Qianmei Liu, Yufei Kuang, and Jie Wang. Robust deep reinforcement learning with adaptive adversarial perturbations in action space, 2024. URL https://arxiv.org/abs/2405.11982.

Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in Neural Information Processing Systems*, volume 30, pp. 6379–6390. Curran Associates, Inc., 2017. URL https://arxiv.org/abs/1706.02275. Peer-reviewed NeurIPS 2017 version preferred; the same paper also introduced the Multi-Agent Particle Environments (MPE).

Ajay Mandlekar, Yuke Zhu, Animesh Garg, Li Fei-Fei, and Silvio Savarese. Adversarially robust policy learning: Active construction of physically-plausible perturbations. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3932–3939, 2017. doi: 10.1109/IROS.2017.8206245.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

Santiago Paternain, Miguel Calvo-Fullana, Luiz F. O. Chamon, and Alejandro Ribeiro. Safe policies for reinforcement learning via primal-dual methods. *IEEE Transactions on Automatic Control*, 68(3):1321–1336, 2023. doi: 10.1109/TAC.2022.3152724.

Bei Peng, Tabish Rashid, Christian Schroeder de Witt, Pierre-Alexandre Kamienny, Philip H. S. Torr, Wendelin Böhmer, and Shimon Whiteson. FACMAC: Factored multi-agent centralised policy gradients. In *Advances in Neural Information Processing Systems*, volume 34, pp. 12208–12221. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/hash/65b9eea6e1cc6bb9f0cd2a47751a186f-Abstract.html. Peer-reviewed NeurIPS 2021 version preferred; arXiv:2003.06709 was originally posted in 2020 under the title "Deep Multi-Agent Reinforcement Learning for Decentralized Continuous Cooperative Control" and was later substantially revised and retitled as FACMAC.

Xiyue Peng, Lingkai Zu, Ziyu Shao, and Xin Liu. Towards achieving optimal strong regret and constraint violation via computationally efficient model-free RL. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=y8rIspGlNQ.

Xue Bin Peng, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Sim-to-real transfer of robotic control with dynamics randomization. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–8, May 2018. doi: 10.1109/ICRA.2018.8460528.

Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 2817–2826. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/pinto17a.html.

Wilbert Pumacay, Ishika Singh, Jiafei Duan, Ranjay Krishna, Jesse Thomason, and Dieter Fox. The colosseum: A benchmark for evaluating generalization for robotic manipulation, 2024. URL https://arxiv.org/abs/2402.08191.

Cheng Qian, Julen Urain, Kevin Zakka, and Jan Peters. Pianomime: Learning a generalist, dexterous piano player from internet demonstrations. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard (eds.), *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 1194–1215. PMLR, 06–09 Nov 2025a. URL https://proceedings.mlr.press/v270/qian25a.html.

Jiahao Qian, Yinong Li, Huan Wu, Ling Zheng, and Hao Zheng. Twin-delayed deep deterministic policy gradient algorithm for active suspensions incorporating preview strategies. In *2025 17th International Conference on Computer and Automation Engineering (ICCAE)*, pp. 541–545, 2025b. doi: 10.1109/ICCAE64891.2025.10980556.

Rongjun Qin, Songyi Gao, Xingyuan Zhang, Zhen Xu, Shengkai Huang, Zewen Li, Weinan Zhang, and Yang Yu. Neorl: A near real-world benchmark for offline reinforcement learning, 2021. URL https://arxiv.org/abs/2102.00714.

Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8, 2021. URL https://jmlr.org/papers/v22/20-1364.html.

Tabish Rashid, Mikayel Samvelyan, Christian Schroeder de Witt, Gregory Farquhar, Jakob N. Foerster, and Shimon Whiteson. QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4295–4304. PMLR, 2018. URL https://proceedings.mlr.press/v80/rashid18a.html. Peer-reviewed ICML 2018 version preferred; a later expanded arXiv revision appears as arXiv:2003.08839.

Mikayel Samvelyan, Tabish Rashid, Christian Schroeder de Witt, Gregory Farquhar, Nantas Nardelli, Tim G. J. Rudner, Chia-Man Hung, Philip H. S. Torr, Jakob N. Foerster, and Shimon Whiteson. The StarCraft multi-agent challenge. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 2186–2188. International Foundation for Autonomous Agents and Multiagent Systems, 2019. URL https://arxiv.org/abs/1902.04043. Peer-reviewed AAMAS 2019 version preferred; an extended preprint is available as arXiv:1902.04043.

Christian Schroeder de Witt, Tarun Gupta, Denys Makoviichuk, Viktor Makoviychuk, Philip H. S. Torr, Mingfei Sun, and Shimon Whiteson. Is independent learning all you need in the StarCraft multi-agent challenge? *arXiv preprint arXiv:2011.09533*, 2020a. doi: 10.48550/arXiv.2011.09533. URL https://arxiv.org/abs/2011.09533. Canonical public version is the 2020 arXiv/CoRR paper; no peer-reviewed archival version was identified.

Christian Schroeder de Witt, Bei Peng, Pierre-Alexandre Kamienny, Philip H. S. Torr, Wendelin Böhmer, and Shimon Whiteson. Deep multi-agent reinforcement learning for decentralized continuous cooperative control. *CoRR*, abs/2003.06709, 2020b. URL https://arxiv.org/abs/2003.06709. Original 2020 preprint version introducing the Multi-Agent MuJoCo benchmark; the same arXiv record was later substantially revised and retitled as "FACMAC: Factored Multi-Agent Centralised Policy Gradients," published at NeurIPS 2021.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Carmelo Sferrazza, Dun-Ming Huang, Xingyu Lin, Youngwoon Lee, and Pieter Abbeel. Humanoidbench: Simulated humanoid benchmark for whole-body locomotion and manipulation, 2024. URL https://arxiv.org/abs/2403.10506.

Vasu Sharma, Prasoon Goyal, Kaixiang Lin, Govind Thattai, Qiaozi Gao, and Gaurav S. Sukhatme. CH-MARL: A multimodal benchmark for cooperative, heterogeneous multi-agent reinforcement learning. *CoRR*, abs/2208.13626, 2022. doi: 10.48550/arXiv.2208.13626. URL https://arxiv.org/abs/2208.13626. Canonical public version is the 2022 arXiv/CoRR paper; no peer-reviewed archival version was identified.

Joseph Suarez, Yilun Du, Phillip Isola, and Igor Mordatch. Neural MMO: A massively multiagent game environment for training and evaluating intelligent agents. *arXiv preprint arXiv:1903.00784*, 2019. doi: 10.48550/arXiv.1903.00784. URL https://arxiv.org/abs/1903.00784. Original Neural MMO paper; later platform revisions include Neural MMO v1.3 (arXiv:2001.12004), the 2021 platform paper (arXiv:2110.07594), and Neural MMO 2.0 (arXiv:2311.03736).

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

J K Terry, Benjamin Black, Nathaniel Grammel, Mario Jayakumar, Ananth Hari, Ryan Sullivan, Luis S Santos, Clemens Dieffendahl, Caroline Horsch, Rodrigo Perez-Vicente, et al. Pettingzoo: Gym for multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 34:15032–15043, 2021.

Chen Tessler, Yonathan Efroni, and Shie Mannor. Action robust reinforcement learning and applications in continuous control. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 6215–6224. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/tessler19a.html.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 23–30, 2017. doi: 10.1109/IROS.2017.8202133.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pp. 5026–5033. IEEE, 2012.

Mark Towers, Ariel Kwiatkowski, John Balis, Gianluca De Cola, Tristan Deleu, Manuel Goulão, Kallinteris Andreas, Markus Krimmel, Arjun KG, Rodrigo Perez-Vicente, J Terry, Andrea Pierré, Sander Schulhoff, Jun Jet Tai, Hannah Tan, and Omar G. Younis. Gymnasium: A standard interface for reinforcement learning environments. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference. Curran Associates, Inc., 2025. doi: 10.52202/085713-4916.

Saran Tunyasuvunakool, Alistair Muldal, Yotam Doron, Siqi Liu, Steven Bohez, Josh Merel, Tom Erez, Timothy Lillicrap, Nicolas Heess, and Yuval Tassa. dm_control: Software and tasks for continuous control. *Software Impacts*, 6:100022, November 2020. ISSN 2665-9638. doi: 10.1016/j.simpa.2020.100022. URL http://dx.doi.org/10.1016/j.simpa.2020.100022.

Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *nature*, 575(7782):350–354, 2019.

Xihuai Wang, Zheng Tian, Ziyu Wan, Ying Wen, Jun Wang, and Weinan Zhang. Order matters: Agent-by-agent policy optimization. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Q-neeWNVv1.

Muning Wen, Jakub Grudzien Kuba, Runji Lin, Weinan Zhang, Ying Wen, Jun Wang, and Yaodong Yang. Multi-agent reinforcement learning is a sequence modeling problem. In *Advances in Neural Information Processing Systems*, volume 35, pp. 16509–16521. Curran Associates, Inc., 2022. doi: 10.52202/068431-1201. URL https://arxiv.org/abs/2205.14953. Peer-reviewed NeurIPS 2022 version preferred; the paper introduces the Multi-Agent Transformer (MAT).

Amber Xie, Haozhi Qi, and Dorsa Sadigh. Handelbot: Real-world piano playing via fast adaptation of dexterous robot policies, 2026. URL https://arxiv.org/abs/2603.12243.

Huazhe Xu, Yuping Luo, Shaoxiong Wang, Trevor Darrell, and Roberto Calandra. Towards learning to play piano with dexterous hands and touch, 2022. URL https://arxiv.org/abs/2106.02040.

Tengyu Xu, Yingbin Liang, and Guanghui Lan. Crpo: A new approach for safe reinforcement learning with convergence guarantee. In *International Conference on Machine Learning*, pp. 11480–11491. PMLR, 2021.

Yang Xu, Swetha Ganesh, Washim Mondal, Qinbo Bai, and Vaneet Aggarwal. Global convergence for average reward constrained mdps with primal-dual actor critic algorithm. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 92592–92627. Curran Associates, Inc., 2025. doi: 10.52202/085713-3099.

Long Yang, Jiaming Ji, Juntao Dai, Yu Zhang, Pengfei Li, and Gang Pan. Cup: A conservative update policy algorithm for safe reinforcement learning. *arXiv preprint arXiv:2202.07565*, 2022.

Yaodong Yang, Rui Luo, Minne Li, Ming Zhou, Weinan Zhang, and Jun Wang. Mean field multi-agent reinforcement learning. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5571–5580. PMLR, 2018. URL https://proceedings.mlr.press/v80/yang18d.html. Peer-reviewed ICML 2018 version preferred; preprint available as arXiv:1802.05438.

Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. The surprising effectiveness of PPO in cooperative multi-agent games. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24611–24624. Curran Associates, Inc., 2022. doi: 10.52202/068431-1787. URL https://proceedings.neurips.cc/paper_files/

paper/2022/hash/9c1535a02f0ce079433344e14d910597-Abstract.html. Peer-reviewed NeurIPS 2022 Datasets and Benchmarks Track version preferred; the preprint first appeared as arXiv:2103.01955.

Peipei Yu, Hongcai Zhang, Yonghua Song, Zhenyi Wang, Huiyu Dong, and Liang Ji. Safe reinforcement learning for power system control: A review. *Renewable and Sustainable Energy Reviews*, 223:116022, 2025.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on robot learning*, pp. 1094–1100. PMLR, 2020.

Kevin Zakka, Philipp Wu, Laura Smith, Nimrod Gileadi, Taylor Howell, Xue Bin Peng, Sumeet Singh, Yuval Tassa, Pete Florence, Andy Zeng, et al. Robopianist: Dexterous piano playing with deep reinforcement learning. *arXiv preprint arXiv:2304.04150*, 2023.

Yves-Simon Zeulner, Simon Crämer, Sandeep Selvaraj, and Roberto Calandra. Learning to play piano in the real world, 2026. URL https://arxiv.org/abs/2503.15481.

Huan Zhang, Hongge Chen, Chaowei Xiao, Bo Li, Mingyan Liu, Duane Boning, and Cho-Jui Hsieh. Robust deep reinforcement learning against adversarial perturbations on state observations. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 21024–21037. Curran Associates, Inc., 2020.

Wenbo Zhang, Kaixuan Wang, Yutao Ouyang, Xiaoyu Huang, Liyang Li, Kailun Su, Weiyang Jin, Wenhao Chai, Haotian Liang, Zhiyang Dou, Yue Chen, and Tianxing Chen. An unexpected robot policy: Early evaluations of gpt-6 astra on robodojo and beyond, 2026. URL https://arxiv.org/abs/2609.24170.

Weiye Zhao, Yifan Sun, Feihan Li, Rui Chen, Ruixuan Liu, Tianhao Wei, and Changliu Liu. Guard: A safe reinforcement learning benchmark. *arXiv preprint arXiv:2305.13681*, 2023.

Yi Zhao, Le Chen, Jan Schneider, Quankai Gao, Juho Kannala, Bernhard Schölkopf, Joni Pajarinen, and Dieter Büchler. Rp1m: A large-scale motion dataset for piano playing with bi-manual dexterous robot hands. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard (eds.), *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 5184–5203. PMLR, 06–09 Nov 2025. URL https://proceedings.mlr.press/v270/zhao25d.html.

Lianmin Zheng, Jiacheng Yang, Han Cai, Ming Zhou, Weinan Zhang, Jun Wang, and Yong Yu. MAgent: A many-agent reinforcement learning platform for artificial collective intelligence. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, volume 32, pp. 8222–8223. AAAI Press, 2018. doi: 10.1609/aaai.v32i1.11371. URL https://ojs.aaai.org/index.php/AAAI/article/view/11371. Peer-reviewed AAAI 2018 demonstration paper preferred; the earlier arXiv:1712.00600 version has a slightly different author list.

Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Kevin Lin, Abhiram Maddukuri, Soroush Nasiriany, and Yifeng Zhu. robosuite: A modular simulation framework and benchmark for robot learning. *arXiv preprint arXiv:2009.12293*, 2020.

Adil Zouitine, David Bertoin, Pierre Clavier, Matthieu Geist, and Emmanuel Rachelson. Rrls : Robust reinforcement learning suite, 2024. URL https://arxiv.org/abs/2406.08406.

AI USE STATEMENT

During the preparation of ***OmniPiano***, generative AI tools were used to assist with language refinement and selected coding tasks. The authors reviewed and revised all AI-assisted content, verified the relevant implementations, and checked the technical statements. All scientific decisions, experimental design, interpretation of results, and final manuscript content remain the responsibility of the authors.

ETHICS STATEMENT

OmniPiano is a simulation benchmark designed to support research on dexterous piano playing. The experiments involve simulated robotic shadow hands and do not involve human participants or animal subjects. We do not identify specific ethical concerns arising from the reported experiments. The benchmark's safety and robustness evaluations are limited to the modeled constraints and perturbations and should not be interpreted as guarantees of safety or robustness in real-world deployment. Likewise, the multi-agent evaluations study simulated decentralized controllers and do not capture the full complexity of real-world multi-robot interaction.

REPRODUCIBILITY STATEMENT

We provide the information required to reproduce ***OmniPiano*** and the reported experiments throughout the main text and appendices. The benchmark task formulation, observation and action spaces, reward design, and evaluation metrics are described in Section 2, Section 3 and Appendix C. Experimental protocols, baseline implementations, and hyperparameter settings are documented in Appendix D. The accompanying code includes the benchmark environments, configuration files, and scripts for reproducing the reported experiments.

LIMITATIONS AND BROADER IMPACT

OmniPiano is limited to dexterous piano playing in simulation, and its findings may not directly generalize to other manipulation tasks or real-world robotic systems. Results may also depend on the simulated morphology, contact dynamics, and task-specific perturbations and constraints. Nevertheless, ***OmniPiano*** provides a unified platform for studying high-dimensional dexterous control, robustness, safety, and multi-agent coordination under consistent tasks and evaluation protocols.

A PRELIMINARIES

OmniPiano is built upon a common dexterous piano-playing formulation, in which robotic hands interact with a simulated piano to follow a MIDI-conditioned target sequence. This section introduces the shared state, observation, action, and reward definitions, and then formalizes standard RL as the base setting from which the robust, safe, and multi-agent variants are derived.

A.1 STANDARD REINFORCEMENT LEARNING

Given a MIDI score m with horizon H_m, the standard piano-playing task is formulated as a finite-horizon Markov Decision Process (MDP):

$$\mathcal{M}_m = (\mathcal{S}, \mathcal{A}, P_m, r_m, \rho_0, \gamma, H_m)\,, \tag{2}$$

where $\mathcal{S}$ and $\mathcal{A}$ denote the state and action spaces, ρ_0 is the initial-state distribution, P_m is the transition kernel, and r_m is the reward function. Let $\mathcal{H} = \{1, \ldots, N_h\}$ denote the set of robotic hands. The policy receives an observation $o_t := h(s_t) \in \mathcal{O}$ consisting of hand proprioception, piano-key states, and score-conditioned goals, while the action $a_t \in \mathcal{A}$ specifies the target joint positions of all hands and the sustain-pedal command:

$$\begin{aligned} o_t &= \left[\{q_t^h, x_t^h\}_{h\in\mathcal{H}}, k_t, g_{t:t+L-1}\right], \\ a_t &= \left[\{a_t^h\}_{h\in\mathcal{H}}, a_t^{\text{pedal}}\right]. \end{aligned} \tag{3}$$

where q_t^h and x_t^h denote the joint configuration and Cartesian forearm pose of hand h, respectively, k_t denotes the piano-key joint states, and g denote the key-activation targets over a lookahead horizon L. Under this formulation, the policy is optimized by

$$\pi^\star = \arg\max_\pi J_R(\pi), \qquad J_R(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{H_m-1} \gamma^t r_t\right]. \tag{4}$$

Here, r_t measures the agent's piano-playing performance at time t in terms of key accuracy, sustain control, and fingertip-to-key matching.

A.2 Robust Reinforcement Learning with Disruptions

Starting from the standard MDP, we formulate robustness using the modular disrupted-interaction framework of Robust-Gymnasium (Gu et al., 2025). Let:

$$\mathcal{M}_m^0 = (\mathcal{S}, \mathcal{A}, P_{m,\phi_0}, r_{m,\phi_0}, \rho_{\phi_0}, \gamma, H_m) \tag{5}$$

denote the nominal piano-playing task, where ϕ_0 contains the nominal environment parameters.

A disrupted interaction is described by observation, action, reward, and environment operators D_o, D_a, D_r, and D_e:

$$\begin{aligned} &\tilde{o}_t = D_o(o_t; \xi_t^o), \qquad a_t =\sim \pi(\cdot \mid \tilde{o}_t), \qquad \tilde{a}_t = D_a(a_t; \xi_t^a), \\ &\phi_t = D_e(\phi_0; \xi_t^e), \quad s_{t+1} \sim P_{m,\phi_t}(\cdot \mid s_t, \tilde{a}_t), \quad \tilde{r}_t = D_r(r_{m,\phi_t}(s_t, \tilde{a}_t, s_{t+1}); \xi_t^r)\,. \end{aligned} \tag{6}$$

The resulting objective is:

$$\pi^\star = \arg\max_\pi J_R(\pi), \qquad J_R(\pi) = \mathbb{E}_{\tau,\xi}\left[\sum_{t=0}^{H_m-1} \gamma^t \tilde{r}_t\right]. \tag{7}$$

Here, D_o, D_a, and D_r corrupt signals during interaction, whereas D_e may modify the transition dynamics or initial-state distribution. Setting all disruption operators to the identity recovers the standard piano-playing MDP.

A.3 Safe Reinforcement Learning

Safe piano playing is formulated as a constrained Markov decision process (CMDP) (Gu et al., 2024):

$$\mathcal{M}_m^C = (\mathcal{S}, \mathcal{A}, P_m, r_m, c, \rho_0, \gamma, H_m, d)\,, \tag{8}$$

where $c_t = c(s_t, a_t, s_{t+1}) \geq 0$ is the safety cost and d is the episodic cost limit. If multiple cost components are enabled, they are aggregated as:

$$c_t = \sum_{k=1}^{K} \alpha_k c_{k,t}. \tag{9}$$

The Safe-RL objective is:

$$\begin{aligned} \max_\pi \quad & J_R(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{H_m-1} \gamma^t r_t\right], \\ \text{s.t.} \quad & J_C(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{H_m-1} c_t\right] \leq d. \end{aligned} \tag{10}$$

OmniPiano considers safety costs related to excessive control or joint motion, inter-hand contact, actuator power, and joint-specific risk. Reward and cost are maintained as separate signals, so a policy must improve musical performance while satisfying the safety constraint. Benchmark feasibility is determined using the undiscounted episodic cost in Eq. equation 10.

A.4 Multi-Agent Reinforcement Learning

The multi-agent extension distributes the piano-playing task among multiple agents and is modeled as a cooperative partially observable Markov game:

$$\mathcal{G}_m = (\mathcal{I}, \mathcal{S}, \{\mathcal{O}_i\}_{i\in\mathcal{I}}, \{\mathcal{A}_i\}_{i\in\mathcal{I}}, P_m, r_m, \rho_0, \gamma, H_m)\,, \tag{11}$$

where $\mathcal{I}$ is the set of agents. Each agent selects an action from its local observation, and the joint policy factorizes as:

$$\boldsymbol{\pi}(\boldsymbol{a}_t \mid \boldsymbol{o}_t) = \prod_{i\in\mathcal{I}} \pi_i(a_t^i \mid o_t^i). \tag{12}$$

All agents share the same team reward and optimize:

$$\boldsymbol{\pi}^{\star} = \arg\max_{\boldsymbol{\pi}} \mathbb{E}_{\tau\sim\boldsymbol{\pi}} \left[\sum_{t=0}^{H_m-1} \gamma^t r_t \right]. \tag{13}$$

A constrained Markov game could additionally introduce agent-specific or team-level safety costs and budgets. However, the current ***OmniPiano*** benchmark evaluates safety and multi-agent learning as separate task families and does not yet combine them into a safe multi-agent setting.

B Related Work

B.1 Robotic Piano-Playing and Benchmarks

The closest foundations for ***OmniPiano*** are simulated, score-conditioned piano-playing tasks. Early work formulated the problem as reinforcement learning for dexterous hands (Xu et al., 2022), while RoboPianist established a reproducible MuJoCo benchmark with two Shadow Hands, 150 pieces, and note-level metrics (Zakka et al., 2023). These works primarily evaluate nominal bimanual performance, leaving control beyond two hands and evaluation under perturbations, safety constraints, or decentralized control outside their scope.

Subsequent studies have extended individual aspects of piano playing: RP1M and PianoMime scale demonstration data and cross-song generalization (Zhao et al., 2025; Qian et al., 2025a); real-world systems study dynamics calibration, domain randomization, and residual adaptation (Zeulner et al., 2026; Xie et al., 2026); and CANTABILE enriches musical evaluation with strike-velocity accuracy (Kim et al., 2026). These advances remain separated across data scale, physical deployment, and musical evaluation. ***OmniPiano*** instead brings multi-hand scaling, controlled perturbations, explicit safety costs, and decentralized cooperation into a shared piano-playing benchmark.

B.2 Standard Reinforcement Learning Benchmarks

Standard RL benchmarks cover a wide range of evaluation settings. General-purpose and data-oriented benchmarks standardize tasks, datasets, and generalization protocols (Bellemare et al., 2013; Cobbe et al., 2020; Fu et al., 2021; Gulcehre et al., 2021; Qin et al., 2021). Continuous-control benchmarks provide standardized physics-based environments and interfaces (Brockman et al., 2016; Tassa et al., 2018; Sferrazza et al., 2024). Robotic manipulation and specialized benchmarks further study multi-task learning, generalization, safety, and robustness (Yu et al., 2020; Zhu et al., 2020; Gu et al., 2023a; Ji et al., 2023; Pumacay et al., 2024).***OmniPiano*** complements these efforts with a unified dexterous piano-playing task family spanning standard, robust, safe, and multi-agent RL.

B.3 Robust Reinforcement Learning Benchmarks

Robust RL studies uncertainty entering different components of the agent–environment interaction, including observations, actions, transition dynamics, rewards or costs, timing, and visual or semantic task factors (Huang et al., 2017; Tessler et al., 2019; Pinto et al., 2017; Peng et al., 2018; Bouteiller et al., 2021; Tobin et al., 2017). Such uncertainties may also evolve over time or occur jointly (Lecarpentier & Rachelson, 2020; Mandlekar et al., 2017).

Among benchmarks explicitly targeting robust RL, RRLS (Zouitine et al., 2024) standardizes evaluation on six MuJoCo tasks with uncertain physical parameters and external forces, while Robust-Gymnasium (Gu et al., 2025) provides modular disruptions across a broad task collection. Robustness tests also appear in broader RL benchmarks and algorithm papers (Duan et al., 2016; Pinto et al., 2017; Tessler et al., 2019; Zhang et al., 2020; Kuang et al., 2021; Ding et al., 2023), but they are typically one-off evaluations with task- and method-specific protocols, limiting direct comparison. In contrast, ***OmniPiano*** evaluates robustness in piano playing, where high-dimensional continuous control and multi-hand coordination provide a more demanding testbed for revealing differences in algorithmic robustness.

B.4 Safe Reinforcement Learning Benchmarks

Safe RL is commonly formulated as a constrained Markov decision process, where policies optimize task return subject to safety-cost constraints (Garcıa & Fernández, 2015; Berkenkamp et al., 2017; Yu et al., 2025; Könighofer et al., 2025). Representative methods include CPO (Achiam et al., 2017) and primal-dual (Paternain et al., 2023; Bai et al., 2022; Xu et al., 2025; Peng et al., 2026) or Lagrangian approaches. Benchmark suites such as Safety Gym (Achiam & Amodei, 2019), Safety-Gymnasium (Ji et al., 2023), GUARD (Zhao et al., 2023), and OmniSafe (Ji et al., 2024) provide standardized environments and evaluation tools for studying safety in navigation, locomotion, and manipulation, while safe multi-agent RL extends these ideas to constrained Markov games (Gu et al., 2023b). Compared with prior Safe-RL benchmarks, ***OmniPiano*** focuses on long-horizon dexterous piano playing and evaluates safety under physically meaningful constraints while preserving the underlying musical objective.

B.5 Multi-Agent Reinforcement Learning and Benchmarks

Multi-agent Reinforcement Learning (MARL) studies coordination among decentralized policies under a shared objective. IPPO (Schroeder de Witt et al., 2020a) provides a simple decentralized baseline, while methods such as MADDPG (Lowe et al., 2017), COMA (Foerster et al., 2018), QMIX (Rashid et al., 2018), MAPPO (Yu et al., 2022), and FACMAC (Peng et al., 2021) use Centralized information during Training while retaining Decentralized Execution (CTDE). Other methods target specific challenges, including Mean Field MARL (Yang et al., 2018) for large agent populations, HAPPO (Kuba et al., 2022) for heterogeneous agents, and MAT (Wen et al., 2022) for sequence-based multi-agent coordination. Together, these methods highlight scalability, coupling, heterogeneity, and observability as recurring challenges in MARL.

Existing MARL benchmarks emphasize different subsets of these challenges. MAgent (Zheng et al., 2018) and Neural MMO (Suarez et al., 2019) focus on large agent populations; SMAC (Samvelyan et al., 2019), cooperative MPE (Lowe et al., 2017), and Google Research Football (Kurach et al., 2020) emphasize coordinated interaction; Hanabi (Bard et al., 2020) and SMACv2 (Ellis et al., 2023) emphasize partial observability; and CH-MARL (Sharma et al., 2022) introduces heterogeneous agents. Multi-Agent MuJoCo (Schroeder de Witt et al., 2020b) and Bi-DexHands (Chen et al., 2022) further extend MARL evaluation to continuous robotic control and dexterous manipulation. These benchmarks typically cover only a subset of scalability, coupling, heterogeneity, and observability, and some entangle multiple factors together. ***OmniPiano*** instead makes all four SCHO dimensions separately configurable within the same dexterous-control environment, enabling controlled comparison under a shared embodiment and musical objective.

C Additional Experiments and Analysis

C.1 Standard RL

C.1.1 Sweep study.

We examine the relationship between song difficulty and task performance through a sweep study across representative piano pieces (Table 2). Here, NPS denotes note density in notes per second, while steps is computed at a 20 Hz control frequency. Table 2 shows that song duration alone does not explain task difficulty. Short but note-dense pieces such as *Winter Wind* can be substantially harder than longer pieces with lower NPS, suggesting that note density is a more informative indicator than episode length. However, the variation among pieces with similar NPS also indicates that difficulty depends on additional factors such as fingering, spatial transitions, and coordination.

C.1.2 Restricted vs. Unrestricted Learning

We compare PPO and SAC under restricted and unrestricted hand configurations across the three songs (Figure 10). Restricting each hand to a predefined workspace tends to lead to faster and more stable learning, despite reducing the reachable space. This suggests that the unrestricted setting introduces additional exploration and coordination complexity, while workspace constraints can simplify policy learning by reducing unnecessary motion and inter-hand interference.

Table 2: Task statistics and performance across piano pieces.

Song	Steps	NPS	F1 Score
Gymnopédie No. 1	657	1.77	0.88
Twinkle Twinkle (Rousseau)	451	1.51	0.86
Clair de Lune	597	2.65	0.63
French Suite No. 1, Allemande	517	10.89	0.59
French Suite No. 5, Sarabande	962	3.56	0.59
Piano Sonata No. 23, 2nd mov.	313	12.80	0.49
Pictures at an Exhibition: Great Kiev	720	8.39	0.45
Kreisleriana Op. 16 No. 8	591	9.78	0.44
Piano Sonata No. 2, 1st mov.	361	6.94	0.44
Piano Sonata K279 in C major, 1st mov.	579	9.54	0.44
Für Elise	399	10.44	0.42
French Suite No. 5, Gavotte	962	6.80	0.40
Piano Sonata D845, 1st mov.	1000	9.04	0.34
Winter Wind	314	16.64	0.28
Bagatelle Op. 3 No. 4	550	9.24	0.26
Partita No. 2, 6th mov.	425	11.91	0.25

NPS denotes note density in notes per second. Steps are computed at a 20 Hz control frequency.

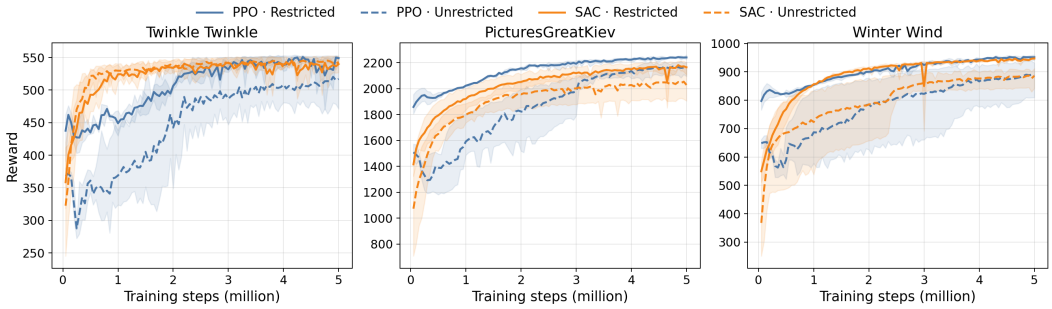


Figure 10: Periodic evaluation reward during training for PPO and SAC under Restricted and Unrestricted configurations across the three songs.

C.2 Robust RL

C.2.1 Cross-Algorithm Comparison.

We compare five RL methods on two-hand *Clair de Lune* under Gaussian action and reward perturbations. As shown in Figure 12, the comparison covers Gaussian action perturbations (Figure 12a) and Gaussian reward perturbations (Figure 12b). EPPO performs the weakest overall, while A2P-SAC and SAC maintain higher and more stable performance than PPO and SCPO. One possible explanation is that A2P-SAC is explicitly designed to improve robustness to action-space perturbations, while SAC also benefits from relatively stable off-policy learning.

C.2.2 Composable Perturbation Evaluation

Moreover, we also demonstrate the ability to leverage different perturbation channels. Figure 13 shows the performance of PPO under composable Gaussian perturbations from observation (O), observation + reward (O + R), observation + reward + action (O + R + A), observation + reward + action + gravity (O + R + A + G) and observation + reward + action + gravity + hand-pose (O + R + A + G + HP) on *ForElise* with three hands. The results show that the performance of PPO decreases as more perturbation channels are added, indicating that the composability of perturbations can significantly affect the robustness of RL algorithms.

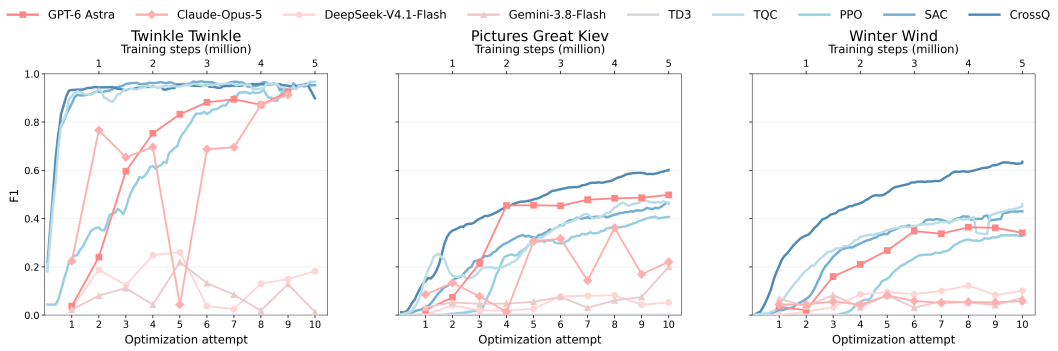
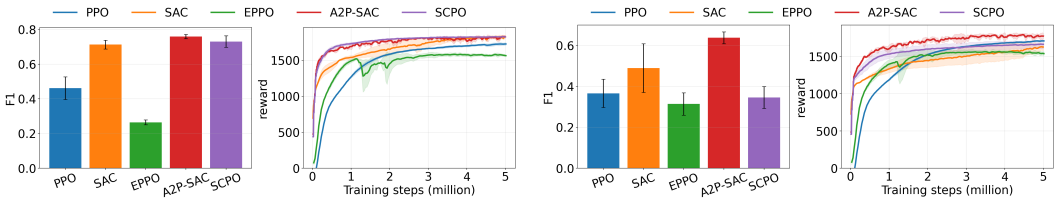


Figure 11: F1 of LLMs over ten keyframe-optimization iterations and RL method over 5M steps for the two-hand setting on three piano pieces.



(a) Under Gaussian action perturbations.

(b) Under Gaussian reward perturbations.

Figure 12: Cross-algorithm robustness comparison on *Clair de Lune*.

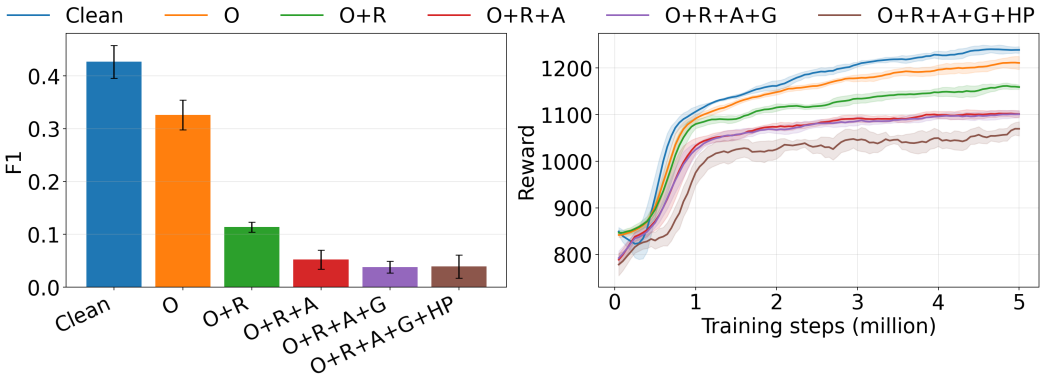


Figure 13: Composability of Gaussian perturbations on *Für Elise* with three hands.

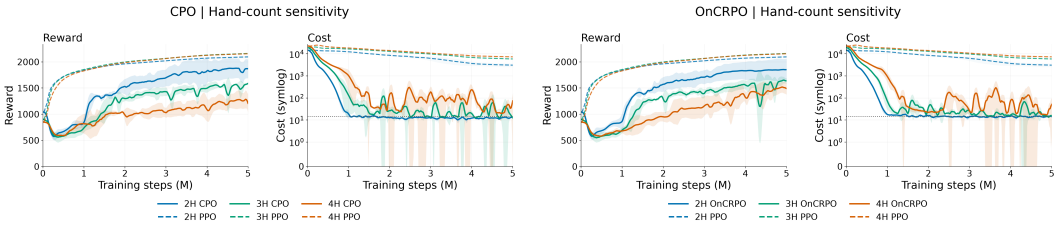
C.3 Safety RL

C.3.1 Additional Hand-Count Results

Figure 14 extends the CUP and PPO-Lag results in the main text with CPO and OnCRPO. We compare two-, three-, and four-hand configurations on the same piece, Pictures at an Exhibition: The Great Gate of Kiev, using the same actuator-power excess cost and episodic budget of 14.4. The aggregate power allowance is fixed rather than scaled with the number of hands. Each algorithm, including the unconstrained PPO reference, is trained for 5M steps with three seeds.

Both methods substantially reduce training cost relative to PPO. However, the four-hand configuration achieves lower reward and exhibits more persistent cost fluctuations than the two-hand configuration. These results show that additional hands do not necessarily facilitate safe policy learning

under a shared aggregate constraint. Because changing the hand count also changes the available interactions and control space, this comparison does not isolate action dimensionality alone.



(a) The hand count results of CPO.

(b) The hand count results of OnCRPO.

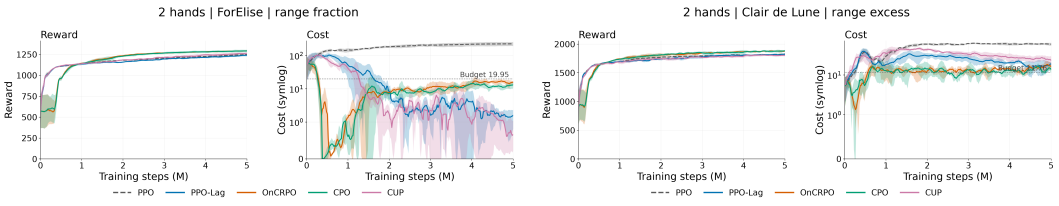
Figure 14: Additional hand count results for safe RL. Solid lines denote CPO / OnCRPO; dashed lines denote PPO. Training reward and cost show means with ± 1 standard deviation across three seeds. The dotted line marks the cost budget.

C.3.2 Overall Performance Evaluation

We compare PPO-Lag, OnCRPO, CPO, and CUP against unconstrained PPO on six tasks spanning two to four hands. The tasks combine different pieces with joint-range fraction, joint-range excess, or actuator-power excess costs. All methods use the OT reward and are trained for 5M environment steps. Reward and cost are compared within each task under a common cost definition and budget, rather than averaged across tasks with different cost scales.

Figures 15–17 report checkpoint-evaluation reward and undiscounted episodic cost. The corresponding key-press F1 curves appear in Figures 18–20. Reporting all three metrics distinguishes cost reduction from successful piano playing: lower cost alone does not establish that a policy preserves musical performance.

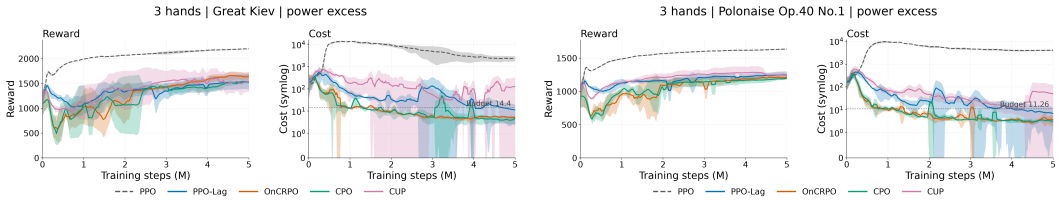
Curves use the available completed evaluation runs. Each task–algorithm group contains three seeds. For trajectory plots, we apply a trailing five-point average to each seed before computing the mean and sample standard deviation. The algorithms learn to reduce safety costs across the six tasks, but their playing performance depends on the constraint. Under joint range constraints, CPO and OnCRPO reduce cost without sacrificing reward or F1 relative to PPO. Under power constraints, the algorithms also reduce cost, but achieve lower reward and F1. Therefore, adding safety constraints can make learning more difficult, as the agent must learn to play the piano while also reducing safety costs.



(a) 2 hands, ForElise, joint-range fraction.

(b) 2 hands, Clair de lune, joint-range excess.

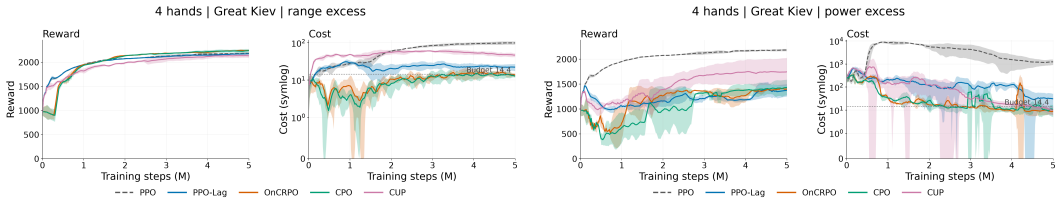
Figure 15: Overall performance on 2 hand safety tasks: evaluation reward and cost. Shading indicates ± 1 standard deviation across available seeds; dotted lines mark task-specific cost budgets.



(a) 3 hands, the Great Gate of Kiev, actuator-power excess.

(b) 3 hands, Polonaise Op. 40 No. 1, actuator-power excess.

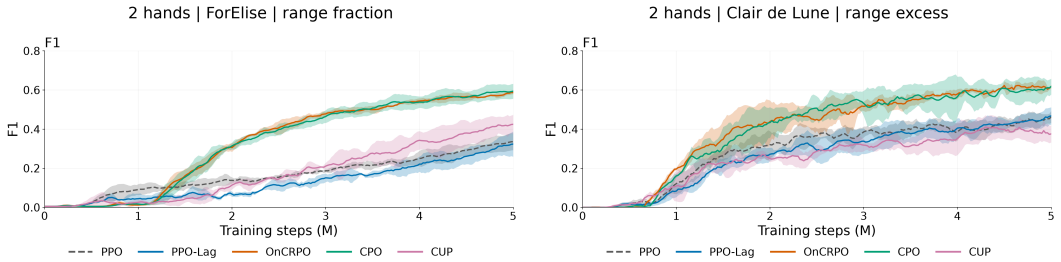
Figure 16: Overall performance on 3 hand safety tasks: evaluation reward and cost. Shading indicates ± 1 standard deviation across available seeds; dotted lines mark task-specific cost budgets.



(a) 4 hands, the Great Gate of Kiev, joint-range excess.

(b) 4 hands, the Great Gate of Kiev, actuator-power excess.

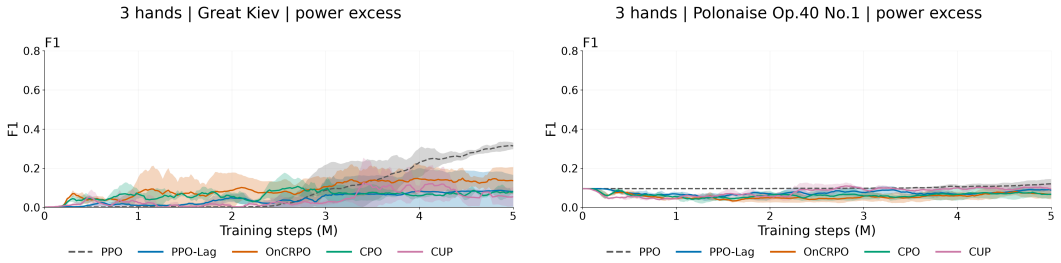
Figure 17: Overall performance on 4 hand safety tasks: evaluation reward and cost. Shading indicates ± 1 standard deviation across available seeds; dotted lines mark task-specific cost budgets.



(a) 2 hands, ForElise, joint-range fraction.

(b) 2 hands, Clair de lune, joint-range excess.

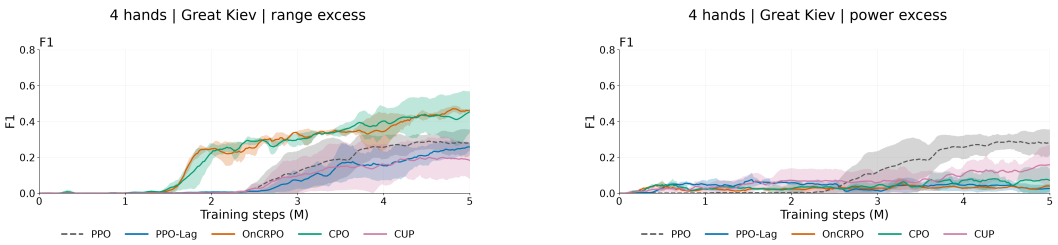
Figure 18: Key-press F1 on the 2-hand safety tasks. Curves show means with ± 1 standard deviation across available seeds.



(a) 3 hands, the Great Gate of Kiev, actuator-power excess.

(b) 3 hands, Polonaise Op. 40 No. 1, actuator-power excess.

Figure 19: Key-press F1 on the 3-hand safety tasks. Curves show means with ± 1 standard deviation across available seeds.



(a) 4 hands, the Great Gate of Kiev, joint-range excess.

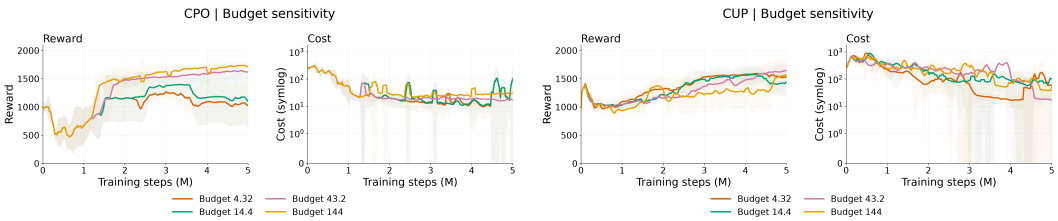
(b) 4 hands, the Great Gate of Kiev, actuator-power excess.

Figure 20: Key-press F1 on the 4-hand safety tasks. Curves show means with ± 1 standard deviation across available seeds.

C.3.3 Sensitivity to the Cost Limit

We vary the cost limit while keeping the piece, hand configuration, safety semantic, and cost definition fixed. A lower limit imposes a stricter safety requirement by allowing less cumulative cost per episode. We consider four limits, $d \in \{4.32, 14.4, 43.2, 144\}$, on the actuator power task with four hands. CPO and CUP are each trained for 5M steps with three seeds per limit.

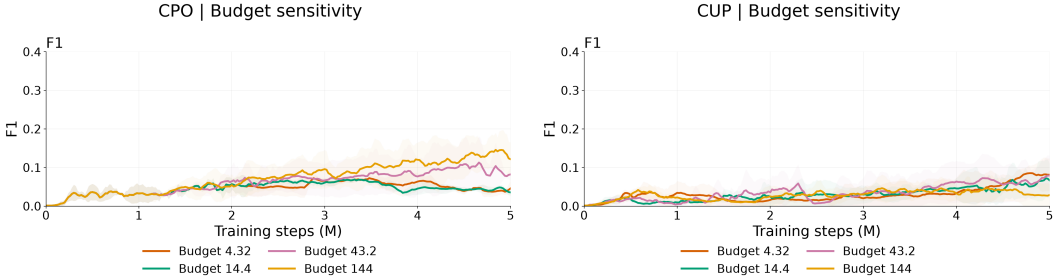
Figures 21 and 22 show how these limits affect reward, cost, and F1. For CPO, the stricter limits of 4.32 and 14.4 produce lower reward and F1 than the more permissive limits of 43.2 and 144. This suggests that stricter safety requirements can make accurate piano playing harder to learn. For CUP, however, relaxing the limit does not consistently improve performance: the largest limit produces the lowest final mean F1. Both algorithms also finish training with mean costs above the strictest limit of 4.32, indicating that this requirement remains difficult to satisfy within 5M steps.



(a) The budget sensitivity results of CPO.

(b) The budget sensitivity results of CUP.

Figure 21: Sensitivity to the episodic cost limit. Evaluation reward and cost are shown for five budgets; shading indicates ± 1 standard deviation across three seeds.



(a) The budget sensitivity F1 of CPO.

(b) The budget sensitivity F1 of CUP.

Figure 22: Key-press F1 under different cost limits, shown as means with ± 1 standard deviation across three seeds.

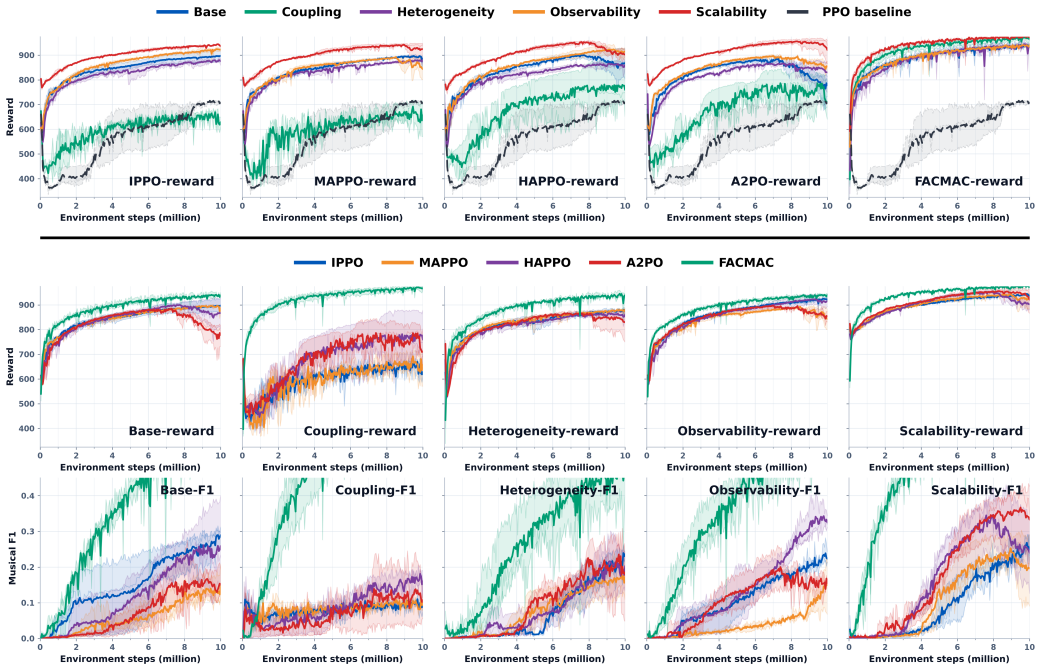


Figure 23: Detailed MARL results across the five ***OmniPiano*** cooperation settings. The first row compares task-wise team return for each MARL algorithm, while the second and third rows compare team return and musical F1 across algorithms for each task.

C.4 Multi-Agent RL

Table 3: MARL test settings.

Setting	Agent	Hands	Action	Observation	Visible hands
Base	A1	0,1	1–44	1–49	2
	A2	2,3	45–88	40–88	1
Observability	A1	0,1	1–44	1–70	2
	A2	2,3	45–88	19–88	1
Coupling	A1	0,1	1–70	1–49	2
	A2	2,3	19–88	40–88	1
Heterogeneity	A1	0	1–22	1–27	1
	A2	1,2,3	23–88	18–88	0
Scalability	A1	0	1–22	1–27	1
	A2	1	23–44	18–49	0,2
	A3	2	45–66	40–71	1,3
	A4	3	67–88	62–88	2

This section provides detail MARL experiment setup , results and analysis. Table 3 summarizes the five MARL task configurations. The **Base task** uses two agents, each controlling two hands and a disjoint 44-key action range, with only 10-key overlap between their local observations. The **Observability Variant** keeps the same hand and action assignments but broadens the observation ranges from 49 keys to 70 keys, so that both agents observe a much larger shared keyboard region. The **Coupling Variant** instead keeps the Base observations while expanding the action ranges to 1–70 and 19–88, creating a 52-key overlap that both agents can control. The **Heterogeneity Variant** assigns one hand and 22 actions to the first agent, but three hands and 66 actions to the second, producing a strongly imbalanced control load. The **Scalability Variant** divides the same four hands among four agents, each controlling one hand and 22 actions, with locally centered observation

ranges that overlap by 10 keys between neighboring agents. During training, we additionally discourage physical interference between agents with an inter-agent collision penalty. At each control step, if any physical contact occurs between hand parts controlled by different agents, we subtract 0.1 once from the shared team reward, regardless of the number or force of contacts. The resulting training reward is

$$r_t^{\text{train}} = r_t - \lambda_c c_t^{\text{inter}},$$

where r_t is the original benchmark reward in Equation 1, c_t^{inter} indicates inter-agent contact, and λ_c controls the penalty strength and is set to 0.1 in our experiments. The same shaped reward is shared by all agents, while evaluation uses the original benchmark reward.

The first row of Figure 23 compares how each MARL algorithm responds to the five task configurations. Across algorithms, the setting with **Increased Scalability** consistently achieves the highest or near-highest return. The physical four-hand task and total action space remain unchanged, but control is divided among four agents instead of two. Compared with Base, this reduces each agent's assignment from two hands and 44 actions to one hand and 22 actions, while increasing the number of independently acting controllers from two to four. The higher return suggests that the benefit from reduced per-agent control burden appears to outweigh the drawback brought by additional coordination. **Increased Observability** also generally improves or matches Base, showing that broader actor observations at execution time can improve coordination by exposing more keyboard and teammate information. In contrast, the **Heterogeneity** setting usually reduces performance, possibly because one agent must control three hands and 66 actions, making its local control problem much harder. The setting with **Increased Coupling** produces the largest degradation for A2PO, HAPPO, IPPO, and MAPPO, which combines overlapping control with severe partial observability. FACMAC is the clear exception and remains strong under Coupling, possibly because its centralized factored critic and joint policy gradient better capture the effects of interacting agent actions.

The second and third rows of Figure 23 summarize algorithm performance across the five MARL tasks using team return and musical F1. Because the return curves are close for several methods, we additionally report musical F1 following RoboPianist (Zakka et al., 2023), which is more sensitve to algorithms' performance and more directly measures note execution accuracy. At each timestep, precision measures how many pressed keys are correct, recall measures how many target keys are successfully pressed, and F1 is their harmonic mean.

Across the five tasks, **Coupling** provides the clearest separation between algorithms. The two agents share a 52-key action overlap, while 21 keys in each agent's action range lie outside its own observation range, combining strong cross-agent action interference with severe partial observability. Under this setting, HAPPO and A2PO achieve higher returns than IPPO and MAPPO. FACMAC performs best overall on this task, consistent with the sample efficiency of its off-policy training.

Similar differences persist across the other tasks as well, especially in F1 even when returns are close. IPPO remains a surprisingly strong baseline, matching or outperforming MAPPO in most settings and often achieving higher F1, indicating that MAPPO's centralized critic does not consistently improve performance in ***OmniPiano***. One possible explanation is that its larger critic input is harder to fit, although our experiments do not isolate this effect. Since HAPPO and A2PO also use global critics without showing the same weakness, centralized value information may be most useful when combined with policy updates that better account for inter-agent interactions. Finally, centralized PPO performs markedly worse than the MARL methods across all tasks, supporting the benefit of factorizing this high-dimensional control problem across multiple agents.

Finally, we further test the Base IPPO setting under Gaussian observation noise with $\sigma = 0.05$, applied at every control step to hand joint states, piano states, and sustain states while leaving the MIDI goal unchanged. As shown in Figure 24, the reward curves remain close to the clean baseline and largely fall within normal seed variation. In contrast, musical F1 drops substantially under observation noise, reaching about 0.15 compared with roughly 0.29 without noise. This shows that observation perturbations can noticeably degrade note-execution quality even when the training return appears similar. This is consistent with the Robust RL results in Section 3.2, where policy performance also degrades under injected perturbations, with the severity depending on the perturbation channel.

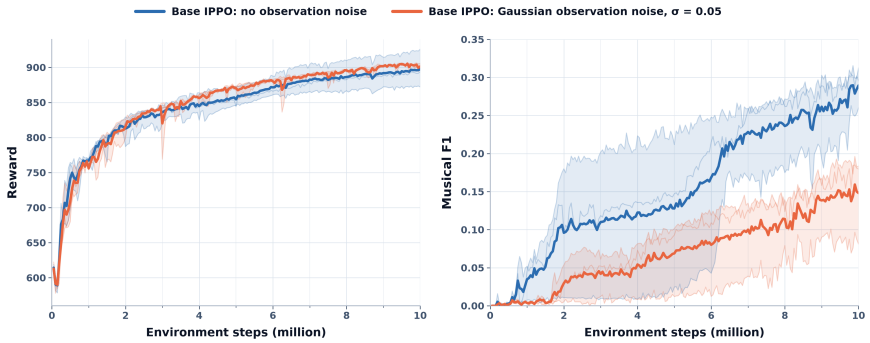


Figure 24: IPPO performance in the Base MARL setting with and without Gaussian observation noise ($\sigma = 0.05$). Reward remains similar, while musical F1 degrades substantially under observation noise.

D Detailed Experimental Settings

D.1 Standard RL

D.1.1 Reward Setting

The *fingering-annotated reward* directly uses fingering annotations from the MIDI score. In contrast, the *optimal transport (OT) reward* constructs a fingertip-to-key cost matrix from their spatial distances and solves the resulting minimum-cost one-to-one assignment using the Hungarian algorithm.(Chen et al., 2025) The OT formulation removes the dependence on fingering annotations and therefore supports arbitrary MIDI files and hand configurations, whereas the annotation-based reward provides explicit finger-level supervision but is limited to two-hand settings.

D.1.2 Evaluation Metric

We use the F1 score to evaluate key-activation accuracy by comparing the keys played by the policy with the target keys specified by the MIDI score at each control step. A true positive (TP) denotes a target key that is correctly activated, a false positive (FP) denotes a non-target key that is incorrectly activated, and a false negative (FN) denotes a target key that is not activated. Precision therefore measures how many played keys are correct, while recall measures how many target keys are successfully played. The F1 score is computed from precision and recall as

$$\begin{aligned} \text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \qquad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \\ \text{F1} = \frac{2\,\text{Precision}\,\text{Recall}}{\text{Precision} + \text{Recall}}. \end{aligned} \tag{14}$$

The F1 score balances the two complementary error modes: pressing incorrect keys and missing required keys. A high F1 score therefore requires both accurate and complete execution of the MIDI targets, making it a direct measure of task performance in our piano-playing setting.

D.1.3 LLM Setting

LLM-based agent framework. Our agent follows the code–feedback–optimization paradigm, adapted to dexterous piano playing with Shadow Hands. Rather than directly predicting low-level actions, the LLM synthesizes a complete Python controller that generates an open-loop actuator trajectory for each performance. The controller is executed in simulation and evaluated using task-level metrics, including F1 score, reward, note-timing errors, and trajectory diagnostics. These results are appended to the conversation as feedback, enabling iterative in-context refinement of the controller.
Tool configuration. Following Zhang et al. (2026), we provide the agent with a forward-kinematics tool for inspecting candidate hand poses. Beyond this, we introduce several task-specific tools for dexterous piano playing. The agent receives structured information about the MIDI score and tim-

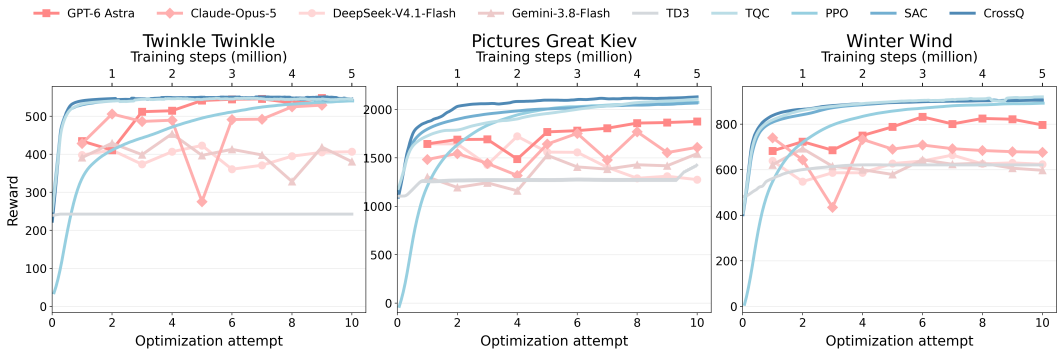


Figure 25: Rollout reward of llms over ten keyframe-optimization iterations and RL method over 5M steps for the two-hand setting on three piano pieces.

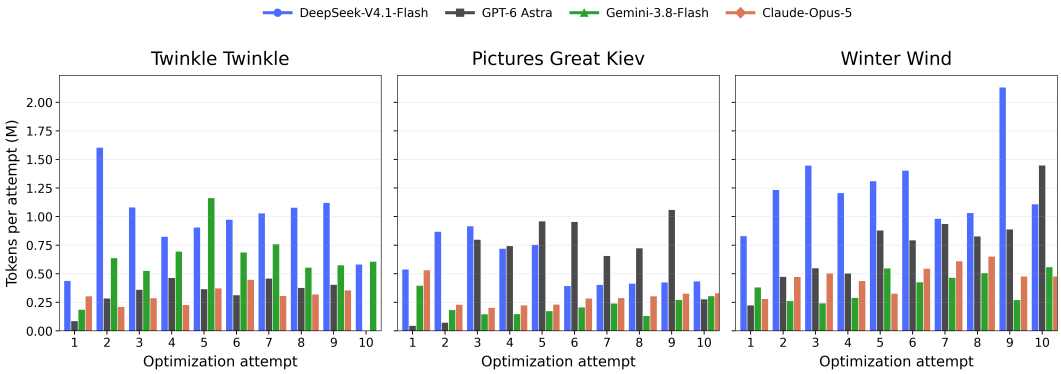


Figure 26: token consumption over ten keyframe-optimization iterations for the two-hand setting on three piano pieces.

ing, hand configuration, actuator bounds, and robot geometry, and can query Shadow-Hand reference information and retrieve local trajectory segments around erroneous time steps. These additions expose task structure and execution diagnostics to the LLM, enabling more targeted controller refinement from simulator feedback.

Optimization protocol. For each piano piece, the agent performs ten controller-optimization attempts. Each attempt uses feedback from previous executions to revise the controller, while model parameters remain fixed. We report the controller achieving the highest F1 score across the ten attempts.The reward results and F1 results can be seen in Figure 11 and Figure 25. GPT-6 Astra shows a tendency to improve reward across attempts, whereas the other LLM-based agents exhibit limited or inconsistent reward gains. In contrast, RL methods achieve stronger and more stable reward improvement through iterative training.

Token Consumption. The token consumption over ten keyframe-optimization iterations for the two-hand setting on three piano pieces can be seen in Figure 26. GPT-6 Astra allocates noticeably more tokens to more challenging pieces, whereas Gemini-3.8-Flash and DeepSeek-V4.1-Flash show weaker adaptation of token usage to task difficulty. Claude-Opus-5 maintains the lowest overall token consumption among the evaluated LLM-based agents.

D.2 Robust RL

We evaluate robustness over six perturbation targets: action, observation, reward, gravity, contact-friction, and initial hand-pose. Each target is tested under Gaussian, uniform, and shift perturbations with three canonical magnitudes (low, medium, and high), as summarized in Table 4. Signal perturbations are applied at each control step, while physical perturbations are sampled either per episode or per step depending on the setting. Hand-pose perturbations are applied once at reset. This design enables controlled comparison across perturbation type, distribution, magnitude, and frequency.

Table 4: ***OmniPiano*** robustness axes and canonical low/medium/high levels. Levels are standard deviations for Gaussian noise, half-widths for symmetric uniform noise, and magnitudes for constant shifts. Hand-pose entries report (y, z) offsets.

Target	Perturbed quantity	Family	Frequency	Low / Medium / High
Action	Canonical actuator target	G/U/S	Step	$0.05/0.10/0.15$
Observation	Proprioception and piano state	G/U/S	Step	$0.05/0.10/0.15$
Reward	Scalar reward received by the learner	G/U/S	Step	$0.10/0.30/0.50$
Gravity	Vertical acceleration g_z	G/U/S	Episode/step	$0.50/1.00/1.50\,\mathrm{m\,s^{-2}}$
Contact-friction	Fingertip–key sliding coefficient	G/U/S	Episode/step	$0.05/0.10/0.15$
Hand-pose	Per-hand initial (y, z) position	G/U/S	Episode	$(10, 5)/(25, 10)/(50, 20)\,\mathrm{mm}$

G/U/S denotes Gaussian, uniform, and shift. Signal shifts are fixed across steps; physical shift tasks use an episode-level constant. Step-wise physical variants are provided for Gaussian and uniform perturbations.

Table 5: Main safety experiments settings. Each task uses PPO, PPO-Lag, OnCRPO, CPO, and CUP, with three seeds and 5M training steps per run.

Hands	Song	Cost formulation	Constraint reference	T_{nom}	Budget d	Runs
2	Für Elise	Joint-range fraction	Central 50%	399	19.95	15
2	ClairDeLune	Joint-range excess	Central 50%	588	11.76	15
3	PicturesGreatKiev	Total-power excess	$P_{\max} = 12$	720	14.40	15
3	PolonaiseOp40No1	Total-power excess	$P_{\max} = 12$	563	11.26	15
4	PicturesGreatKiev	Joint-range excess	Central 50%	720	14.40	15
4	PicturesGreatKiev	Total-power excess	$P_{\max} = 16$	720	14.40	15

T_{nom} is the episode length in control steps. Joint-range costs monitor the J2 joints of all five fingers on every hand; the permitted interval is the central 50% of each joint's native range. $P_{\max}$ is the instantaneous total-power reference in simulator units, not the episode budget. Budgets are fixed as $d = 0.05T_{\mathrm{nom}}$ for event/fraction and $d = 0.02T_{\mathrm{nom}}$ for excess costs.

D.3 SAFETY RL

This section details the training parameters used in the safety experiments, summarized in Table 5.

Table 6: Baseline algorithms supported by ***OmniPiano*** across the four evaluation tracks. *Italic* denotes on-policy methods and underlined denotes off-policy methods.

Track	Baselines
Standard RL	*A2C*, ARS, CrossQ, DDPG, DroQ, *PPO*, SAC, TD3, TQC, *TRPO*
Robust RL	A2P-SAC, *EPPO*, OMPO, SCPO
Safe RL	*PPO-Lag*, *TRPO-Lag*, *RCPO*, *PDO*, *FOCOPS*, *CPO*, *PCPO*, *OnCRPO*, *IPO*, *P3O*, *CUP*, DDPG-Lag, TD3-Lag, SAC-Lag
Multi-Agent RL	*IPPO*, *MAPPO*, *HAPPO*, *MAT*, *A2PO*, HATD3, FACMAC, MACSAC

Italic: on-policy; underlined: off-policy. ARS is a derivative-free policy-search method.