Learning social norms of cooperation under limited observability

Alexandre S. Pires*
University of Amsterdam
Amsterdam, The Netherlands
a.m.dasilvapires@uva.nl

Roman Chiva Gil*
University of Amsterdam
Amsterdam, The Netherlands
r.chivagil@uva.nl

Fernando P. Santos
University of Amsterdam
Amsterdam, The Netherlands
f.p.santos@uva.nl

ABSTRACT

In social dilemmas, cooperation leads to desirable collective outcomes, yet agents are incentivized to defect. Designing independent learning agents capable of cooperating in such settings remains a critical challenge. Reputation systems and reciprocity are potential mechanisms to incentivize cooperation; however, their practical implementation raises key difficulties, including how to assess agents in partial-information settings where only actions, rather than strategies, can be observed, and how to dynamically adapt reputation assignment rules to changing environments. To study these challenges, in this work-in-progress paper we develop a reinforcement learning environment grounded in evolutionary game theory in which a centralized agent learns to select reputation rules that maximize cooperation within an adaptive population of strategic actors. This environment captures a fundamental control problem applicable to domains such as human cooperation, content moderation, and multi-agent reinforcement learning, and poses three major challenges: (i) the agent's reward stems from the collective behavior of the population, which depends only indirectly on the agent's actions, (ii) the population's adaptation induces strong non-stationarity, and (iii) the agent lacks access to the population's strategy distribution, relying on a finite number of interactions for inference. We provide initial numerical and experimental results showing that belief formation, and therefore policy learning and cooperative stability, benefit directly from an increased number of observations. Our preliminary results also reveal a trade-off between information availability and learning difficulty.

KEYWORDS

Reinforcement Learning, Cooperation, Multi-Agent Systems, Indirect Reciprocity, Evolutionary Game Theory, Social Norms

INTRODUCTION

Cooperation is fundamental for successful multi-agent systems, from biological populations to societies of artificial agents. Whether it involves humans tackling climate change, a fleet of autonomous vehicles, or human-AI hybrid systems such as content moderation in social media, cooperation presents an inherent conundrum: agents must often incur a personal cost to provide a collective benefit. In the absence of stabilizing mechanisms, strategic actors are incentivized to free-ride, leading to the "tragedy of the commons" [55]. While various mechanisms help promote prosocial behavior [30], Indirect Reciprocity (**IR**) stands out as a primary driver of cooperation in large-scale decentralized systems [32].

Under **IR**, agents aim to maintain a "good" reputation, as those with high social standing are more likely to receive future benefits, creating a robust incentive for cooperation [2]. These reputations are governed by social norms — rules that dictate how an observer should judge an agent's action within a specific context. While social norms have been extensively studied through the lens of human evolution [35], the rise of artificial agents (AAs), including social bots, large language models (LLMs), and recommendation and moderation algorithms, has brought forward new applications of reputation systems and norms [1, 15]. In this capacity, AAs act not just as cooperators or defectors, but as mediators, judges, and influencers of the normative landscape [23, 53].

Under these positions, artificial agents actively shape social discourse and shift moral judgments in humans [4, 24] and artificial societies [40]. However, recent evidence suggests that the latent norms within AAs, such as current LLMs, are often inconsistent and, if adopted by a population, could actively compromise cooperative stability [42]. Given their systemic reach, it is important to determine if artificial agents can *learn* to select and promote social norms that sustain cooperation within adaptive populations.

Solving this problem is non-trivial from a learning perspective. An adaptive learning agent deciding how other actors must judge each other faces a unique control problem characterized by three distinct challenges. First, the agent is not a participant in the game but an influencer; it must derive its reward signal entirely from the observations of others' behaviors. Second, the environment is non-stationary and adaptive: other agents adjust their strategies over time based on evolutionary pressures, often operating on a different timescale from the learning agent. Third, and most critically, the environment is only partially observable. While an AA may observe the outcomes of interactions (who helped whom), the latent intentions and strategic distribution of the adaptive population remain hidden. Furthermore, artificial agents may only have access to a limited sample of interactions, creating a bottleneck in estimating the strategic landscape of the population. Our research aims to directly address these problems, for which we pose the following research questions: 1) Can a reinforcement learning agent learn to promote cooperation via indirect reciprocity under limited observability by selecting which social norms a population of adaptive agents should use? 2) How do observational constraints impact the agent's ability to infer strategies? 3) How do observational constraints impact the agent's ability to promote cooperation?

To answer these questions, in this work-in progress paper we integrate insights from evolutionary game theory to define a partially observable Markov decision process (POMDP) [5, 61]. We define an environment containing a population of adaptive agents (representing humans or artificial agents) who update their strategies based on social learning. These agents repeatedly pair to play donation games, deciding to cooperate (**C**) or defect (**D**) based on

*The authors contributed equally to this work.

Proc.@ of the Adaptive and Learning Agents Workshop (ALA 2026), Aydeniz, Delgrange, Mohammedalamen, Yang (eds.), May 25 – 26, 2026, Paphos, Cyprus, https://alaworkshop2026.github.io/. 2026.

reputations assigned following the present social norm. The reinforcement learning (RL) agent acts as a centralized observer and norm governor, defining the social norm used to assign reputations based on a limited batch of observed interactions, and is rewarded solely based on the realized cooperation levels within the population. Our environment is illustrated in Figure 1. We provide a mathematical analysis of this environment, linking the number of observations to the entropy of the agent's belief state regarding the population's strategy. Our results demonstrate that: 1) An RL agent can successfully learn policies that sustain high levels of cooperation; and 2) These policies are highly sensitive to observational limits and are fundamentally constrained by the information they convey. We show that reduced observations lead to large regions of uncertainty and noise, requiring the agent to adopt conservative norm-setting strategies. We also study different learning regimes, connecting experience collection during learning with policy quality. This framework contributes to the intersection of RL and social dynamics, offering a scalable method to extract cooperative policies for multi-agent systems under realistic observation constraints. Importantly, our method can be easily applied to specific contexts, such as defining social norms in human or MARL systems [15] or content recommendation and moderation [38, 48].

RELATED WORKS

Indirect reciprocity and social norms Indirect reciprocity is a well-established mechanism underlying human cooperation, supported by extensive experimental [47] and evolutionary game-theoretical [35] evidence. Central to **IR** are social norms, which determine how individuals assign reputations based on observed actions and contextual information, thereby largely shaping cooperative outcomes [33]. Their effectiveness depends on their contextual complexity [51], but also on what social norms previously existed [62], and their robustness to errors in strategy execution or reputation assessment [13, 17]. More recent work highlights the importance of information constraints: limited gossip or partial observability can cause mismatched reputations, which lead to unjustified punishment, disagreement accumulation, and cooperation breakdown [21, 43]. These results emphasize that cooperation under **IR** is not only norm-dependent, but also fundamentally constrained by information availability.

In the context of artificial agents, non-explicit social norms may also emerge naturally through communication [41] or behaviors [10]. Different LLMs also exhibit a wide range of social norms, many of which are inconsistent and incapable of promoting cooperation [42]. As such, there is a growing interest in the implementation and selection of concrete social norms to moderate reputation systems that foster cooperation among learning agents, which typically relied on internally defined social norms [3, 46] or decentralized emergent norms [44, 45].

Normative governance and influence The capacity of artificial agents to influence human actions [14, 36], beliefs [8], and norms [16, 19, 60] is well-documented. These agents participate in society in various capacities as mediators, advisors, partners, and role models [23], exerting distinct influence dynamics at both the individual and societal levels. Even when not interacting directly, the systemic reach of AI allows agents to influence the spread of information and behavior through network-level interventions [48]. By altering the topology of interactions and the visibility of actions, these systems effectively shape the social norms, observability, and reputation dynamics that are fundamental to the stability of cooperation. The emerging behaviors and norms of these systems have been referred to as "machine culture" [7], including their unique spreading dynamics by artificial agents. Importantly, these influences also extend to systems of artificial agents [15], posing challenges to cooperation.

Learning social norms and social influence Despite the recognized capacity of artificial agents to influence social norms, learning to leverage this influence in socially beneficial ways remains largely understudied [9, 54]. Existing work primarily focuses on structural or strategic interventions rather than the explicit governance of norms. For instance, notable work by McKee et al. [28] applied deep learning models trained through simulations to rewrite interaction networks among humans playing cooperation games. Even when trained on networks of artificial agents, their model learned to identify agents based on their behaviors, clustering cooperators and isolating defectors. RL has also been applied to promote cooperation by learning social norms and behavior influence in human–AI settings through human-generated data [18, 26], and among societies of artificial agents [58]. However, to our knowledge, no prior work has applied learning algorithms to sustain cooperation in multi-agent systems through the selection of explicit social norms in the context of indirect reciprocity, or considered the challenges of limited strategic observability.

THE INDIRECT RECIPROCITY ENVIRONMENT

We consider a population of Z adaptive agents, representing any learning or evolving population (e.g. humans, artificial agents) [11]. In our model, agents repeatedly take part in donation games, playing any of the three canonical roles (Donor, Recipient, and Observer). In any game, two agents are randomly chosen as donor and recipient, and the donor has the option to cooperate (**C**), providing a benefit b to the recipient at a cost c to itself, where $b > c$, or to defect (**D**), providing no benefit to the recipient, at no cost. All other agents observe these interactions and assign a reputation (either good, **G**, or bad, **B**) to the donor following a common social norm. At every timestep, the learning agent observes a finite set of interactions and selects the social norm the population will use for the next interactions. We first describe the population's evolutionary dynamics under a static social norm, followed by the transitions between norms, which define the environment on which the learning agent will act. Finally, we formalize the environment as a POMDP [20].

Population dynamics

At any time, each adaptive agent adopts one strategy, which dictates when to cooperate or defect depending on the reputation of the receiver. A strategy is defined as a tuple $s = (s_G, s_B)$, where s_G and s_B are the probability of cooperating with a **G** and **B** individual, respectively. We define three strategies: *ALLC* $(1, 1)$, which always cooperates; *ALLD* $(0, 0)$, which always defects; and *DISC* $(1, 0)$, which only cooperates with **G** individuals, defecting otherwise. We

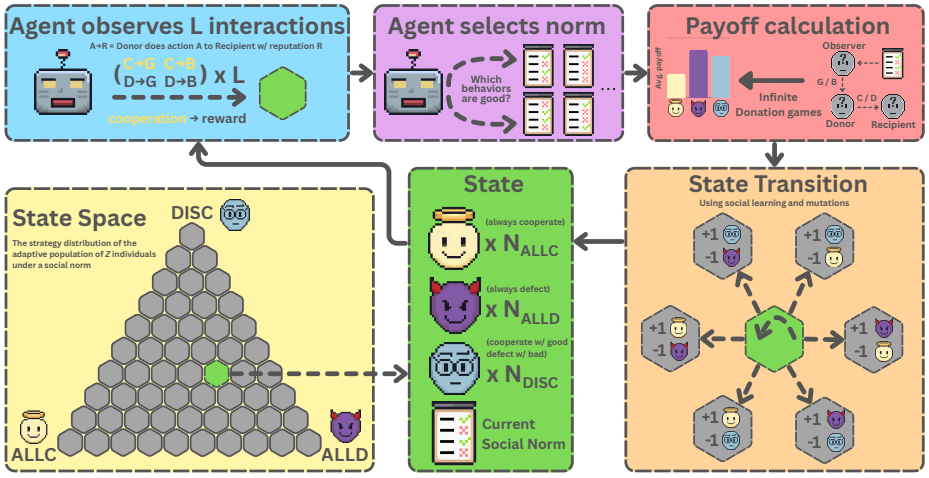


Figure 1: Overview of the environment. Each state is composed of a number of agents using each strategy under a specific social norm. At a given state and timestep, the agent will observe L interactions sampled from the state, and will be rewarded by the fraction of cooperation the strategy state produces. The agent then selects which social norm should be adopted. A social norm defines which of the possible interactions (Cooperate or Defect with a Good or Bad recipient) are considered good or bad. Given the new social norm, the payoffs of all strategies are calculated and an evolutionary step of social learning or mutation may occur, transitioning the population to a new state. The total state space is represented by a triangular simplex for each possible norm, where the corners represent monomorphic states where the population uses a single strategy.

also consider execution errors: a choice to cooperate may result in an unwanted defection with a probability e_e [12].

Reputation Dynamics. Agents update their reputations following a social norm. We consider second-order social norms [29, 51], which account for the donor's action and the recipient's reputation to determine the donor's new reputation. These norms are encoded as a 4-bit tuple $d = (d_{G,C}, d_{G,D}, d_{B,C}, d_{B,D}), d \in [0,1]^4 = \mathcal{D}$, representing the probability of assigning a good reputation in each of the four possible observation scenarios (e.g., $d_{B,C}$ denotes the probability of assigning **G** to a donor cooperating with a recipient perceived as **B**). This yields 16 second-order norms, including norms well-known to support cooperation [33, 34, 56], such as Image Score [31], $d = (1,0,1,0)$, where cooperation is always good and defection always bad, and Stern Judging [37], $d = (1,0,0,1)$, where cooperation with good recipients and defection against bad recipients is good, and other actions are bad. Assessment errors are also included: with probability e_a, an agent will assign the opposite reputation indicated by the social norm.

Under a social norm d, the probability that an observer assigns **G** to a donor using action $Y \in \{C, D\}$ facing a receiver with reputation $X \in \{G, B\}$, considering errors, is given by $P_{X,Y}$ [22]:

$$P_{X,Y} = \begin{cases} d_{X,C}(\epsilon - e_a) + d_{X,D}(1 - \epsilon - e_a) + e_a & \text{if } Y = C \\ d_{X,D}(1 - 2e_a) + e_a & \text{if } Y = D, \end{cases} \tag{1}$$

where $\epsilon = (1 - e_e)(1 - e_a) + e_e e_a$ is the probability that both errors occur or neither occurs.

Reputation dynamics are coupled to the population's strategy distribution, with the payoff of each strategy depending on the current reputation distribution. We assume that reputations evolve faster than strategies [17, 50], so that for any strategy state $n = (n_{ALLC}, n_{ALLD}, n_{DISC})$, with n_s representing the number of agents using strategy s, and $n_{ALLC} + n_{ALLD} + n_{DISC} = Z$, reputation dynamics are considered to have converged.

In a well-mixed population at a strategy state n under a social norm d, the probability that an agent using strategy $s \in S = \{ALLC, ALLD, DISC\}$ is considered good, $r_s(d,n)$, is given by the following set of ordinary differential equations [39]:

$$\frac{\partial r_s(d,n,t)}{\partial t} = g_s(d,n,t) - r_s(d,n,t), \quad s \in S \tag{2}$$

where $g_s(d,n,t)$ is the probability that an agent using a social norm d will assign a good reputation to an agent using strategy s, at time step t, given the distribution of strategies n. These are strategy dependent, and given by [21]:

$$\begin{aligned} g_{ALL\lambda}(d,n,t) &= r(d,n,t)P_{G,\lambda}(d) + \bar{r}(d,n,t)P_{B,\lambda}(d) \\ g_{DISC}(d,n,t) &= \tilde{q}^g P_{G,C}(d) + \tilde{q}^d \left(P_{G,D}(d) + P_{B,C}(d)\right) + \tilde{q}^b P_{B,D}(d) \end{aligned} \tag{3}$$

where $r(d,n,t) = \sum_{s \in S}(n_s/Z) \cdot r_s(d,n,t)$ denotes the average reputation in the population, $\lambda \in \{C, D\}$, $\bar{r}(d,n,t) = 1 - r(d,n,t)$, and $\tilde{q}^g$ and $\tilde{q}^b$ represent the probabilities that two agents agree that a focal agent is perceived as good or bad, respectively. Conversely, $\tilde{q}^d$ denotes the probability that the two agents disagree in their assessment. Before any gossip occurs, when reputations are entirely private [59], these probabilities are given by:

$$\begin{aligned} q^g &= \sum_{s \in S} \frac{n_s}{Z} r_s(d,n,t)^2, \qquad q^b = \sum_{s \in S} \frac{n_s}{Z} \bar{r}_s(d,n,t)^2, \\ q^d &= \sum_{s \in S} \frac{n_s}{Z} r_s(d,n,t)\bar{r}_s(d,n,t). \end{aligned} \tag{4}$$

We consider k rounds of gossip, where at each round a randomly selected agent adopts the reputations assigned by another agent [21]. To account for population size, this is normalized as the gossip duration $\mathcal{T} = k/Z$. At $\mathcal{T} = 0$, reputations remain fully private, and disagreement is maximized; conversely, as $\mathcal{T} \to \infty$, reputations become public and coordinated. After gossip, the agreement and disagreement probabilities are given by $\tilde{q}^g = q^g + q^d \cdot (1 - e^{-\mathcal{T}})$, $\tilde{q}^b = q^b + q^d \cdot (1 - e^{-\mathcal{T}})$ and $\tilde{q}^d = q^d \cdot e^{-\mathcal{T}}$.

Strategy Adoption Dynamics. Strategy adoption is modeled via a birth-death process, incorporating mutations (probability γ of switching to a random strategy, akin to exploration) and social learning, modeled through the *pairwise comparison rule* [57]. In the latter, the probability that an agent using strategy s imitates an agent using strategy s' is given by $P_{s\to s'}(d,n) = (1 + e^{-\beta \Delta F_{s,s'}})^{-1}$, where $\Delta F_{s,s'} = \bar{F}_{s'} - \bar{F}_s$ is the average fitness difference, and β is the selection strength. The limit $\beta \to +\infty$ yields near-deterministic evolution, while $\beta \to 0$ approximates random drift.

The average fitness of a strategy combines the benefit b received as a recipient and the cost c of donating: $F_s(d,n) = bR_s(d,n) - cD_s(d,n)$, where $R_s(d,n)$ is the probability of receiving,

$$R_s(d,n) = (1-e_e)\left(\frac{n_{ALLC}}{Z} + \frac{n_{DISC}}{Z} r_s(d,n)\right), \tag{5}$$

and $D_s(d,n)$ the probability of donating,

$$D_s(d,n) = (1-e_e)\Big(\, r(d,n)\, s_G + \big(1 - r(d,n)\big)\, s_B \Big). \tag{6}$$

Given the computed average fitness for each strategy state, we construct a Markov chain encompassing all possible strategy states to analyze the evolution of strategies over time under a given social norm [49]. This state space is defined as $\mathcal{M} = \{n = (n_i, n_j, n_k) \mid n_i + n_j + n_k = Z \wedge n_i, n_j, n_k \in [0, Z]\}$, comprising $|\mathcal{M}| = \binom{Z+2}{2}$ states. For any pair of states that differ by the strategy of a single agent, the transition probability is the likelihood of either a mutation or imitation occurring that switches that agent's strategy from s to s':

$$M_{s\to s'}(d,n) = (1-\gamma)\frac{n_s}{Z}\frac{n_{s'}}{Z-1} P_{s\to s'}(d,n) + \gamma \frac{n_s}{2Z}. \tag{7}$$

Each entry $M_{a,b}(d)$ in the transition matrix of the strategy Markov chain $M(d)$, representing the probability of moving from state n^a to state n^b, is given by

$$M_{a,b}(d) = \begin{cases} M_{s\to s'}(d, n^a) & \text{if } n^b_s = n^a_s - 1 \\ & \wedge n^b_{s'} = n^a_{s'} + 1 \\ & \wedge\ n^b_{s''} = n^a_{s''} \quad , \\ 1 - \sum_{s \neq s'} M_{s\to s'}(d, n^a) & \text{if } n^b = n^a \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

where $s, s', s'' \in S$ and $s \neq s' \wedge s \neq s'' \wedge s' \neq s''$.

Defining the POMDP

We next define the state space navigable by the learning agent, as well as the transition probabilities between states, and each possible observation. In our model, the learning agent repeatedly observes interactions and selects the social norm to be used by the population in the next interactions. Crucially, we assume that after selecting the social norm, one step of the strategy Markov chain is taken (see Strategy Adoption Dynamics section), where the reputation dynamics converge and imitation or mutation might take place for a single randomly selected individual. After this step, the learning agent receives the next observation and selects another action.

We consider a state space $\mathcal{S} = \{(\mathbf{s}_d, \mathbf{s}_n) \mid \mathbf{s}_d \in \mathcal{D}, \mathbf{s}_n \in \mathcal{M}\}$, where at each state the population is using a social norm $\mathbf{s}_d$ and is in a strategy state $\mathbf{s}_n$ (alternatively, states can hold reputation distributions instead of $\mathbf{s}_d$). At each state, the learning agent observes $L \in \mathbb{N}_{>0}$ interactions sampled from the state. A higher L corresponds to lower noise and lower uncertainty regarding the actual strategy distribution; the limit of $L = 1$ represents a setting where the learning agent only observes a single interaction before acting, maximizing noise. As such, the set of observations is

$$\Omega_L = \left\{ \frac{\mathbf{k}}{L} \middle| \mathbf{k} \in \mathbb{Z}^4_{\geq 0}, \sum_o \mathbf{k}_o = L \right\}, \tag{9}$$

where $\mathbf{k}_o, o \in \mathcal{O} = \{CG, DG, CB, DB\}$, corresponds to the fraction of times observation o was seen at that timestep.

In a state $\mathbf{s}$, the probability that an agent donates is given by $D(d,n) = \sum_{s\in S} \frac{n_s}{Z} D_s(d,n)$. The probability of observing each donor's action considering the receiver's reputation is given by

$$\begin{aligned} O_{G,C}(d,n) &= D(d,n) \cdot r(d,n), \\ O_{G,D}(d,n) &= (1 - D(d,n)) \cdot r(d,n), \\ O_{B,C}(d,n) &= D(d,n) \cdot \bar{r}(d,n), \\ O_{B,D}(d,n) &= (1 - D(d,n)) \cdot \bar{r}(d,n). \end{aligned} \tag{10}$$

As such, the probability of observing a set l of interactions after L observations is given by the multinomial probability mass function $\mathbf{O}(\mathbf{s}, l) = L! \prod_{o\in O} \frac{O_o(\mathbf{s}_d, \mathbf{s}_n)^{l_o}}{(l_o L)!}$.

After an observation, the learning agent selects the next social norm $a \in \mathcal{D}$. After this action, the payoffs of each strategy change, and a timestep of the Markov chain associated with the new social norm takes place. The probability of a next state s' is given by $T(\mathbf{s}, a, \mathbf{s}') = M_{\mathbf{s}_n, \mathbf{s}'_n}(a)$, and the reward is given by the actual level of cooperation in that state, $R(\mathbf{s}, a, \mathbf{s}') = \sum_{s\in S} D_s(\mathbf{s}'_d, \mathbf{s}'_n) \frac{\mathbf{s}'_{n_s}}{Z}$.

We model the system as a POMDP $\mathcal{P} = (\mathcal{S}, \mathcal{A}, T, R, \Omega, \mathbf{O})$, where the state space is $\mathcal{S} = \mathcal{D} \times \mathcal{M}$ and the action space is $\mathcal{A} = \mathcal{D}$. Transitions depend only on the action and strategy state, $T(\mathbf{s}, a, \mathbf{s}') = M_{\mathbf{s}_n, \mathbf{s}'_n}(a)$ and,

$$R(\mathbf{s}, a, \mathbf{s}') = \sum_{s\in S} D_s(\mathbf{s}'_d, \mathbf{s}'_n) \frac{\mathbf{s}'_{n_s}}{Z}. \tag{11}$$

Observations lie in $\Omega_L = \{\frac{k}{L} \mid k \in \mathbb{Z}^4_{\geq 0}, \sum_o k_o = L\}$ and are distributed as

$$\mathbf{O}(\mathbf{s}, l) = L! \prod_{o\in O} \frac{O_o(d,n)^{l_o L}}{(l_o L)!}. \tag{12}$$

NUMERICAL PROPERTIES OF THE ENVIRONMENT

The learning agent cannot observe the strategies used in the population it aims to guide. Instead, the agent relies on a finite set of interaction observations. This poses a significant challenge to selecting the actions optimally, as social norms (i.e., actions) that may be highly effective in a given strategic distribution, such as

one with many unconditional defectors, may be ineffective in other scenarios, such as one with many conditional cooperators. To quantify the impact of limited observability, we quantify how a finite set of L observations leads to a belief distribution over the state space.

For a candidate state $s' \in \mathcal{M}$, we define the belief weight as $\tilde{\mathbf{b}}(s') = \mathbb{E}_{l\sim \mathbf{O}(n,\cdot)}[\mathbf{O}(s', l)]$, where the expectation is approximated by Monte Carlo sampling and the resulting belief distribution is normalizated as $\mathbf{b}(s') = \frac{\tilde{\mathbf{b}}(s')}{\sum_{s''\in\mathcal{M}} \tilde{\mathbf{b}}(s'')}$. This belief does not correspond to a Bayesian posterior for a realized observation, but rather to the expected likelihood of each state under observations generated from the true state. Given the belief at each state s, we measure its overall uncertainty using the Shannon Entropy [25], $H(\mathbf{b}) = -\sum_{s\in\mathcal{M}} \mathbf{b}(s) \log \mathbf{b}(s)$. We also measure the expected structural dispersion of the belief distribution using Rao's Quadratic Entropy [6] $Q(\mathbf{b}) = \sum_{s\in\mathcal{M}} \sum_{s'\in\mathcal{M}} \mathbf{b}(s)\, \mathbf{b}(s')\, \mathrm{dist}(s, s')$, where $\mathrm{dist}(s, s') = \frac{1}{2}\sum_{k=1}^{K} |s_k - s'_k|$ is half of the Manhattan distance, corresponding to the minimum number of agents that must change strategy for state s to match state s'.

In Figure 2, we report these entropy metrics as a function of the number of interaction observations L, evaluated at each strategy state and averaged across social norms (results per norm are provided in Appendix B). On average, entropy decreases as L increases, indicating that additional observations concentrate the belief distribution over fewer and more similar states. At the same time, increasing L amplifies the heterogeneity of entropy across states: under noisy observations, beliefs tend to spread over many distant states, whereas with more precise observations the informativeness of observations becomes highly state-dependent, yielding varying levels of uncertainty across strategy states.

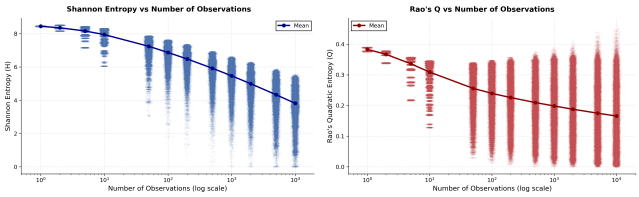


Figure 2: State entropy metrics at varying number of interaction observations. Left: Shannon Entropy. Right: Rao's Quadratic Entropy. Each point corresponds to an entropy measure in a strategy state and social norm pair. We observe that the average entropy decreases in both metrics as the number of observations increases, while also increasing the difference in entropy between states. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

PRELIMINARY EMPIRICAL RESULTS

We trained a variety of PPO-based learning agents under different observation counts L and training regimes. A significant challenge in this environment is its gradual dynamics: agents often become trapped near the ALLD monomorphic state for extended periods. This generates batches of homogeneous experience, which can lead to catastrophic forgetting (see Appendix D.1). To mitigate this and encourage experience diversity, we investigated four training regimes: One Worker No Resets (OWNR), where a single agent collects experience continuously, One Worker with Resets (OWR), where a single agent will reset to a random strategy state after some timesteps, Parallel Workers No Resets (PWNR), where multiple instances of the agent collect experience parallel, and Parallel Workers with Resets (PWR). For each setting, we conducted a parameter search and trained agents for 2 million timesteps of experience (see Appendix C for implementation details and Appendix D for learning curves). In this environment, the immediate reward (cooperation at a state) is a noisy proxy for policy quality, as it does not reflect the policy's capacity to promote cooperation across the state space. As such, we employed an exhaustive evaluation procedure: the policies of each setting were tested across all initial strategy states for 2000 timesteps and we report the average cooperation level measured during the last 1000 steps.

Figure 3 summarizes the evaluation results per learning regime and number of observation (L). We observe a general trend where increasing L results in higher cooperation, though this increase is sometimes non-monotonic. This reflects the added challenges of additional observations, discussed in the numerical analysis: more observations can increase entropy in a small subset of states, increasing the importance of selecting the correct norm at each timestep and thereby increasing learning difficulty. While all regimes reach non-trivial cooperation, methods utilizing parallel workers achieve higher average performance. A comparison with baselines employing fixed or randomly selected norms is presented and discussed in Appendix 6. To clarify the impact of initial conditions, Figure 4 presents the average cooperation level obtained starting from each state for the OWNR and PWR regimes (see Appendix D.3 for the remaining learning regimes). Our results show that increasing L significantly expands the region of strategic states from which the agent can guide the population towards cooperation: At $L = 1$, the agent is only able to sustain cooperation from a narrow set of initial states, while at $L = 1000$ this region is larger and with higher overall cooperation. Nevertheless, a region composed primarily of ALLC and ALLD agents sees no improvement, suggesting a norm-independent tendency towards the ALLD equilibrium.

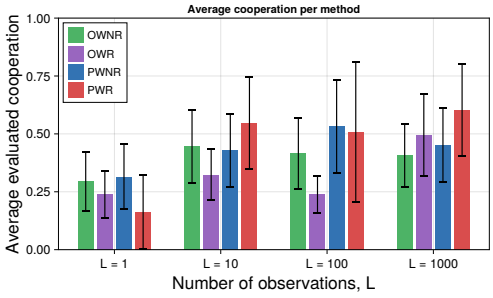


Figure 3: Evaluated cooperation for each training regime for different number of observations (L). We observe that, relative to $L = 1$, higher number of observations result in higher average cooperation. Training regimes based on multiple parallel workers also benefit more from an increased number of observations. Black bars indicate a single standard deviation. Environment parameters are identical to Figure 2.

Finally, to understand how observations shape the resulting policies, Figure 5 shows the most common norm employed by trained agents in each state, for the OWNR and PWR regimes (see Appendix D.4 for the remaining learning regimes). We observe that under low L, agents tend to apply the same norms across broad, undifferentiated regions of the state space, indicating a limited ability to distinguish between strategy states. As L increases, policies becomes more granular and deploy norms that are better suited to local strategy distributions. Furthermore, the policies apply more norms known to sustain cooperation [33, 34], implying higher quality decisions that increase the resulting cooperation.

Overall, our results suggest that training effective normative governance agents requires multiple samples of interactions to ensure awareness of the strategic landscape, although at a cost of increased learning complexity. This problem also benefits from training procedures that ensure experience diversity.

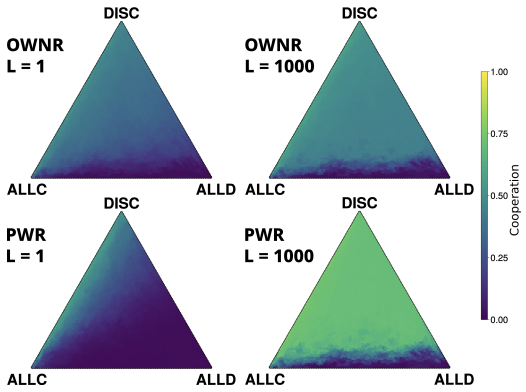


Figure 4: Average cooperation obtained after starting in each strategy state for the one worker no reset (OWNR) and parallel workers with resets (PWR) training regimes, for different numbers of observations ($L = 1$, $L = 1000$). We observe that more observations result in a larger strategy state region from where it is possible to guide the population towards cooperation, with higher overall levels of cooperation. Environment parameters are identical to Figure 2.

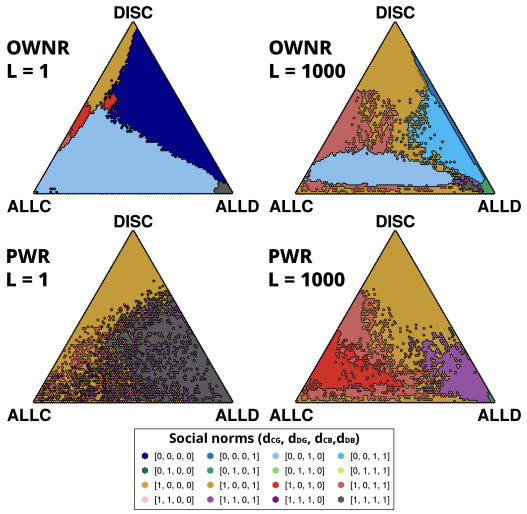


Figure 5: Average policy per state under the one worker no reset (OWNR) and parallel workers with resets (PWR) training regimes, for different number of observations ($L = 1$,$L = 1000$). Each color represents the most used social norm in the trained policies at that state. We observe that, under a low L, the same norms are used in broader regions of the state space, indicating less ability to distinguish between strategy states. Environment parameters are identical to Figure 2.

CONCLUSION

We have formalized a reinforcement learning environment where an agent influences social norms to promote cooperation in multi-agent systems under indirect reciprocity. Grounded in evolutionary game theory [5], this framework addresses the challenge of selecting effective social norms under realistic constraints of limited strategic observability of an adapting population. Our initial numerical and empirical results demonstrate that without information about the population's strategy distribution, learning optimal social norms becomes challenging. This limitation can be mitigated with repeated observations. However, added observations also lead to more complex learning dynamics, as observations become more sensitive to the selected norms. Despite these challenges, our initial results show that standard reinforcement learning regimes can sustain high levels of cooperation in an adaptive population, particularly when using learning regimes that diversity the collected learning experience. By studying learned policies across varying levels of observability, we show that the number of observations plays a crucial role in determining the selected social norms, with less observations resulting in less targeted and adaptive social norms.

These results highlight the critical role of understanding intentions and strategies in prosocial AI. Furthermore, the integration of evolutionary game theoretical models in reinforcement learning remains novel and underexplored. We offer a versatile framework for domains where artificial agents may govern or influence social norms, such as artificial multi-agent systems, as well as human-AI hybrid systems such as content recommendation and moderation [38, 48]. Future work should aim to provide new mechanisms for conveying and inferring agent intentions to AI systems beyond simple interaction outcomes, including techniques such as recurrent learning architectures [52]. Expanding on the complexity of the environment itself is also relevant, by allowing more strategies (e.g. Paradoxical discriminator) and studying the effects of population size, errors and gossip. Another promising direction is the introduction of multiple independent influencers [15], which presents new challenges in coordination and multi-agent alignment.

ACKNOWLEDGMENTS

A.S.P thanks the ELLIS Unit Amsterdam for funding. R.C. and F.P.S acknowledge funding from NWO: This publication is part of the project 'Reputation as a new route to cooperation in multi-agent reinforcement learning' with file number OCENW.M.22.322 of the research programme Open Competition Domain Science which is (partly) financed by the Dutch Research Council (NWO).

REFERENCES

[1] Zeynep Akata, Dan Balliet, Maarten De Rijke, Frank Dignum, and et al. 2020. A Research Agenda for Hybrid Intelligence: Augmenting Human Intellect With Collaborative, Adaptive, Responsible, and Explainable Artificial Intelligence. *Computer* 53, 8 (Aug. 2020), 18–28.

[2] Richard Alexander. 2017. *The Biology of Moral Systems.* Routledge.

[3] Nicolas Anastassacos, Julian Garcia, Stephen Hailes, Mirco Musolesi, et al. 2021. Cooperation and Reputation Dynamics with Reinforcement Learning. In *Proceedings of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021).* 115–123.

[4] Hui Bai, Jan Voelkel, Johannes Eichstaedt, and Robb Willer. 2025. Artificial Intelligence Can Persuade Humans on Political Issues. *Nature Communications* 16 (2025). https://doi.org/10.1038/s41467-025-61345-5

[5] Wolfram Barfuss, Jessica Flack, Chaitanya S Gokhale, Lewis Hammond, Christian Hilbe, Edward Hughes, Joel Z Leibo, Tom Lenaerts, Naomi Leonard, Simon Levin, et al. 2025. Collective cooperative intelligence. *Proceedings of the National Academy of Sciences* 122, 25 (2025), e2319948121.

[6] Zoltán Botta-Dukát. 2005. Rao's quadratic entropy as a measure of functional diversity based on multiple traits. *Journal of vegetation science* 16, 5 (2005), 533–540.

[7] Levin Brinkmann, Fabian Baumann, Jean-François Bonnefon, Maxime Derex, Thomas F. Müller, Anne-Marie Nussberger, Agnieszka Czaplicka, Alberto Acerbi, Thomas L. Griffiths, Joseph Henrich, Joel Z. Leibo, Richard McElreath, Pierre-Yves Oudeyer, Jonathan Stray, and Iyad Rahwan. 2023. Machine culture. *Nature Human Behaviour* (Nov. 2023).

[8] Jason W Burton, Ezequiel Lopez-Lopez, Shahar Hechtlinger, Zoe Rahwan, Samuel Aeschbach, Michiel A Bakker, Joshua A Becker, Aleks Berditchevskaia, Julian Berger, Levin Brinkmann, et al. 2024. How large language models can reshape collective intelligence. *Nature human behaviour* 8, 9 (2024), 1643–1655.

[9] Micah Carroll, Alan Chan, Henry Ashton, and David Krueger. 2023. Characterizing manipulation from AI systems. In *Proceedings of the 3rd ACM Conference on Equity and Access in Algorithms, Mechanisms, and Optimization.* 1–13.

[10] Yali Du, Joel Z. Leibo, Usman Islam, Richard Willis, and Peter Sunehag. 2023. A Review of Cooperation in Multi-agent Learning. arXiv:2312.05162 [cs].

[11] Emilio Ferrara, Onur Varol, Clayton Davis, Filippo Menczer, and Alessandro Flammini. 2016. The rise of social bots. *Commun. ACM* 59, 7 (2016), 96–104.

[12] Michael A Fishman. 2003. Indirect reciprocity among imperfect individuals. *Journal of Theoretical Biology* 225, 3 (2003), 285–292.

[13] Yuma Fujimoto and Hisashi Ohtsuki. 2023. Evolutionary stability of cooperation in indirect reciprocity under noisy and private assessment. *Proceedings of the National Academy of Sciences* 120, 20 (2023), e2300544120.

[14] Hao Guo, Chen Shen, Shuyue Hu, Junliang Xing, Pin Tao, Yuanchun Shi, and Zhen Wang. 2023. Facilitating cooperation in human-agent hybrid populations through autonomous agents. *Iscience* 26, 11 (2023).

[15] Lewis Hammond, Alan Chan, Jesse Clifton, Jason Hoelscher-Obermaier, Akbir Khan, Euan McLean, Chandler Smith, Wolfram Barfuss, Jakob Foerster, Tomáš Gavenčiak, et al. 2025. Multi-agent risks from advanced ai. *arXiv preprint arXiv:2502.14143* (2025).

[16] César A Hidalgo, Diana Orghian, Jordi Albo Canals, Filipa De Almeida, and Natalia Martin. 2021. *How Humans Judge Machines.* MIT Press.

[17] Christian Hilbe, Laura Schmid, Josef Tkadlec, Krishnendu Chatterjee, and Martin A Nowak. 2018. Indirect reciprocity with private, noisy, and incomplete information. *Proceedings of the National Academy of Sciences* 115, 48 (2018), 12241–12246.

[18] Joey Hong, Sergey Levine, and Anca Dragan. 2023. Learning to influence human behavior with offline reinforcement learning. arXiv.

[19] Fatimah Ishowo-Oloko, Jean-François Bonnefon, Zakariyah Soroye, Jacob Crandall, Iyad Rahwan, and Talal Rahwan. 2019. Behavioural evidence for a transparency–efficiency tradeoff in human–machine cooperation. *Nature Machine Intelligence* 1, 11 (2019), 517–521.

[20] Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. 1998. Planning and acting in partially observable stochastic domains. *Artificial intelligence* 101, 1-2 (1998), 99–134.

[21] Mari Kawakatsu, Taylor A Kessinger, and Joshua B Plotkin. 2024. A mechanistic model of gossip, reputations, and cooperation. *Proceedings of the National Academy of Sciences* 121, 20 (2024), e2400689121.

[22] Taylor A Kessinger, Corina E Tarnita, and Joshua B Plotkin. 2023. Evolution of norms for judging social behavior. *Proceedings of the National Academy of Sciences* 120, 24 (2023), e2219480120.

[23] Nils Köbis, Jean-François Bonnefon, and Iyad Rahwan. 2021. Bad machines corrupt good morals. *Nature Human Behaviour* 5, 6 (2021), 679–685.

[24] Sebastian Krügel, Andreas Ostermaier, and Matthias Uhl. 2023. ChatGPT's inconsistent moral advice influences users' judgment. *Scientific Reports* 13, 1 (2023), 4569.

[25] Annick Lesne. 2014. Shannon entropy: a rigorous notion at the crossroads between probability, information theory, dynamical systems and statistical physics. *Mathematical Structures in Computer Science* 24, 3 (2014), e240311.

[26] Yancheng Liang, Daphne Chen, Abhishek Gupta, Simon S Du, and Natasha Jaques. 2024. Learning to cooperate with humans using generative agents. *Advances in Neural Information Processing Systems* 37 (2024), 60061–60087.

[27] Chris Lu, Jakub Kuba, Alistair Letcher, Luke Metz, Christian Schroeder de Witt, and Jakob Foerster. 2022. Discovered policy optimisation. *Advances in Neural Information Processing Systems* 35 (2022), 16455–16468.

[28] Kevin R McKee, Andrea Tacchetti, Michiel A Bakker, Jan Balaguer, Lucy Campbell-Gillingham, Richard Everett, and Matthew Botvinick. 2023. Scaffolding cooperation in human groups with deep reinforcement learning. *Nature Human Behaviour* 7, 10 (2023), 1787–1796.

[29] Sebastián Michel-Mata, Mari Kawakatsu, Joseph Sartini, Taylor A Kessinger, Joshua B Plotkin, and Corina E Tarnita. 2024. The evolution of private reputations in information-abundant landscapes. *Nature* (2024), 1–7.

[30] Martin A Nowak. 2006. Five rules for the evolution of cooperation. *Science* 314, 5805 (2006), 1560–1563.

[31] Martin A Nowak and Karl Sigmund. 1998. Evolution of indirect reciprocity by image scoring. *Nature* 393, 6685 (1998), 573–577.

[32] Martin A Nowak and Karl Sigmund. 2005. Evolution of indirect reciprocity. *Nature* 437, 7063 (2005), 1291–1298.

[33] Hisashi Ohtsuki and Yoh Iwasa. 2006. The leading eight: social norms that can maintain cooperation by indirect reciprocity. *Journal of Theoretical Biology* 239, 4 (2006), 435–444.

[34] Hisashi Ohtsuki and Yoh Iwasa. 2007. Global analyses of evolutionary dynamics and exhaustive search for social norms that maintain cooperation by reputation. *Journal of Theoretical Biology* 244, 3 (2007), 518–531.

[35] Isamu Okada. 2020. A Review of Theoretical Studies on Indirect Reciprocity. *Games* 11, 3 (July 2020), 27.

[36] Raquel Oliveira, Patrícia Arriaga, Fernando P. Santos, Samuel Mascarenhas, and Ana Paiva. 2021. Towards prosocial design: A scoping review of the use of robots and virtual agents to trigger prosocial behaviour. *Computers in Human Behavior* 114 (Jan. 2021), 106547.

[37] Jorge M Pacheco, Francisco C Santos, and Fabio AC C Chalub. 2006. Stern-judging: A simple, successful norm which promotes cooperation under indirect reciprocity. *PLoS Computational Biology* 2, 12 (2006), e178.

[38] Dino Pedreschi, Luca Pappalardo, Emanuele Ferragina, Ricardo Baeza-Yates, Albert-László Barabási, Frank Dignum, Virginia Dignum, Tina Eliassi-Rad, Fosca Giannotti, János Kertész, et al. 2025. Human-AI coevolution. *Artificial Intelligence* 339 (2025), 104244.

[39] Cedric Perret, Marcus Krellner, and The Anh Han. 2021. The evolution of moral rules in a model of indirect reciprocity with private assessment. *Scientific Reports* 11, 1 (2021), 23581.

[40] Jinghua Piao, Zhihong Lu, Chen Gao, Fengli Xu, Fernando P Santos, Yong Li, and James Evans. 2025. Emergence of human-like polarization among large language model agents. *arXiv preprint arXiv:2501.05171* (2025).

[41] Giorgio Piatti, Zhijing Jin, Max Kleiman-Weiner, Bernhard Schölkopf, Mrinmaya Sachan, and Rada Mihalcea. 2024. Cooperate or collapse: Emergence of sustainable cooperation in a society of llm agents. *Advances in Neural Information Processing Systems (NeurIPS 2024)* (2024).

[42] Alexandre S Pires, Laurens Samson, Sennay Ghebreab, and Fernando P Santos. 2025. How large language models judge and influence human cooperation. *arXiv preprint arXiv:2507.00088* (2025).

[43] Arunas L Radzvilavicius, Alexander J Stewart, and Joshua B Plotkin. 2019. Evolution of empathetic moral evaluation. *Elife* 8 (2019), e44269.

[44] Siyue Ren, Wanli Fu, Xinkun Zou, Chen Shen, Yi Cai, Chen Chu, Zhen Wang, and Shuyue Hu. 2025. Beyond the Tragedy of the Commons: Building A Reputation System for Generative Multi-agent Systems. *arXiv preprint arXiv:2505.05029* (2025).

[45] Tianyu Ren, Xuan Yao, Yang Li, and Xiao-Jun Zeng. 2025. Bottom-Up Reputation Promotes Cooperation with Multi-Agent Reinforcement Learning. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems.* 1745–1754.

[46] Tianyu Ren and Xiao-Jun Zeng. 2023. Reputation-based interaction promotes cooperation with reinforcement learning. *IEEE Transactions on Evolutionary Computation* 28, 4 (2023), 1177–1188.

[47] Angelo Romano, Ali Seyhun Saral, and Junhui Wu. 2022. Direct and indirect reciprocity among individuals and groups. *Current opinion in psychology* 43 (2022), 254–259.

[48] Fernando P Santos. 2024. Prosocial dynamics in multiagent systems. *AI Magazine* (2024).

[49] Fernando P. Santos, Samuel Mascarenhas, Francisco C. Santos, Filipa Correia, Samuel Gomes, and Ana Paiva. 2020. Picky losers and carefree winners prevail in collective risk dilemmas with partner selection. *Autonomous Agents and Multi-Agent Systems* 34, 2 (Oct. 2020), 40.

[50] Fernando P Santos, Francisco C Santos, and Jorge M Pacheco. 2016. Social norms of cooperation in small-scale societies. *PLoS Computational Biology* 12 (2016), e1004709.

[51] Fernando P Santos, Francisco C Santos, and Jorge M Pacheco. 2018. Social norm complexity and past reputations in the evolution of cooperation. *Nature* 555,

7695 (2018), 242–245.

[52] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[53] Yonadav Shavit, Sandhini Agarwal, Miles Brundage, Steven Adler, Cullen O'Keefe, Rosie Campbell, Teddy Lee, Pamela Mishkin, Tyna Eloundou, Alan Hickey, et al. 2023. Practices for governing agentic AI systems. *Research Paper, OpenAI, December* (2023).

[54] Hua Shen, Tiffany Knearem, Reshmi Ghosh, Kenan Alkiek, Kundan Krishna, Yachuan Liu, Ziqiao Ma, Savvas Petridis, Yi-Hao Peng, Li Qiwei, et al. 2024. Towards bidirectional human-ai alignment: A systematic review for clarifications, framework, and future directions. *arXiv preprint arXiv:2406.09264* 2406 (2024), 1–56.

[55] Karl Sigmund. 2010. *The Calculus of Selfishness.* Princeton University Press.

[56] Nobuyuki Takahashi and Rie Mashima. 2006. The importance of subjectivity in perceptual errors on the emergence of indirect reciprocity. *Journal of Theoretical Biology* 243, 3 (2006), 418–436.

[57] Arne Traulsen, Martin A Nowak, and Jorge M Pacheco. 2006. Stochastic dynamics of invasion and fixation. *Physical Review E* 74, 1 (2006), 011909.

[58] Mycal Tucker, Yilun Zhou, and Julie Shah. 2020. Adversarially guided self-play for adopting social conventions. *arXiv preprint arXiv:2001.05994* (2020).

[59] Satoshi Uchida. 2010. Effect of private information on indirect reciprocity. *Physical Review E* 82, 3 (2010), 036111.

[60] Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, Zac Kenton, Sasha Brown, Will Hawkins, Tom Stepleton, Courtney Biles, Abeba Birhane, Julia Haas, Laura Rimell, Lisa Anne Hendricks, William Isaac, Sean Legassick, Geoffrey Irving, and Iason Gabriel. 2021. Ethical and social risks of harm from Language Models. arXiv:2112.04359 [cs.CL] https://arxiv.org/abs/2112.04359

[61] Kai Xie and Attila Szolnoki. 2026. Reinforcement learning in evolutionary game theory: A brief review of recent developments. *Appl. Math. Comput.* 510 (2026), 129685.

[62] Hitoshi Yamamoto, Isamu Okada, Satoshi Uchida, and Tatsuya Sasaki. 2017. A norm knockout method on indirect reciprocity to reveal indispensable norms. *Scientific reports* 7, 1 (2017), 44146.

A FIXED-NORM BASELINES

We next present the cooperation evaluation results when the norm-setting agent does not learn, and instead selects either a fixed norm, or randomly selects a social norm to use at each timestep. We test two fixed social norms: Image Score ($d = [1, 0, 1, 0]$), a first-order norm where cooperation is always good and defection is always bad, and Stern-Judging ($d = [1, 0, 0, 1]$), a second-order norm where only cooperating with a good individual or defecting with a bad individual is considered good. For these evaluations, we follow the evaluation protocol described in the main text, averaging the cooperation level obtained by starting the agent across the population strategy space.

Figure 6 presents the cooperation level of each baseline, showing that a randomly selected social norm achieves substantially lower cooperation than the tested fixed social norms. Comparing these results with our learning agent, we observe that $L = 1$ makes the problem of promoting cooperation particularly hard, resulting in cooperation comparable to a randomly selected norm. Similarly, even under a high number of observations, learning cooperative social norms is still difficult and the resulting cooperation is not as good as a cooperative fixed social norm.

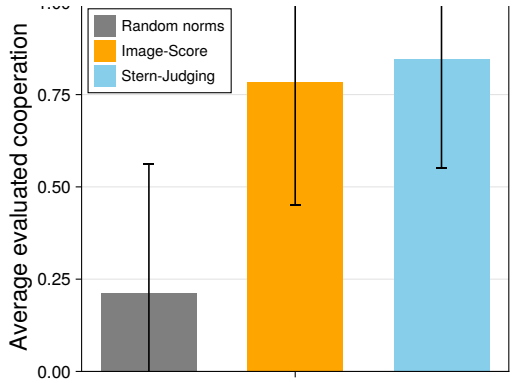
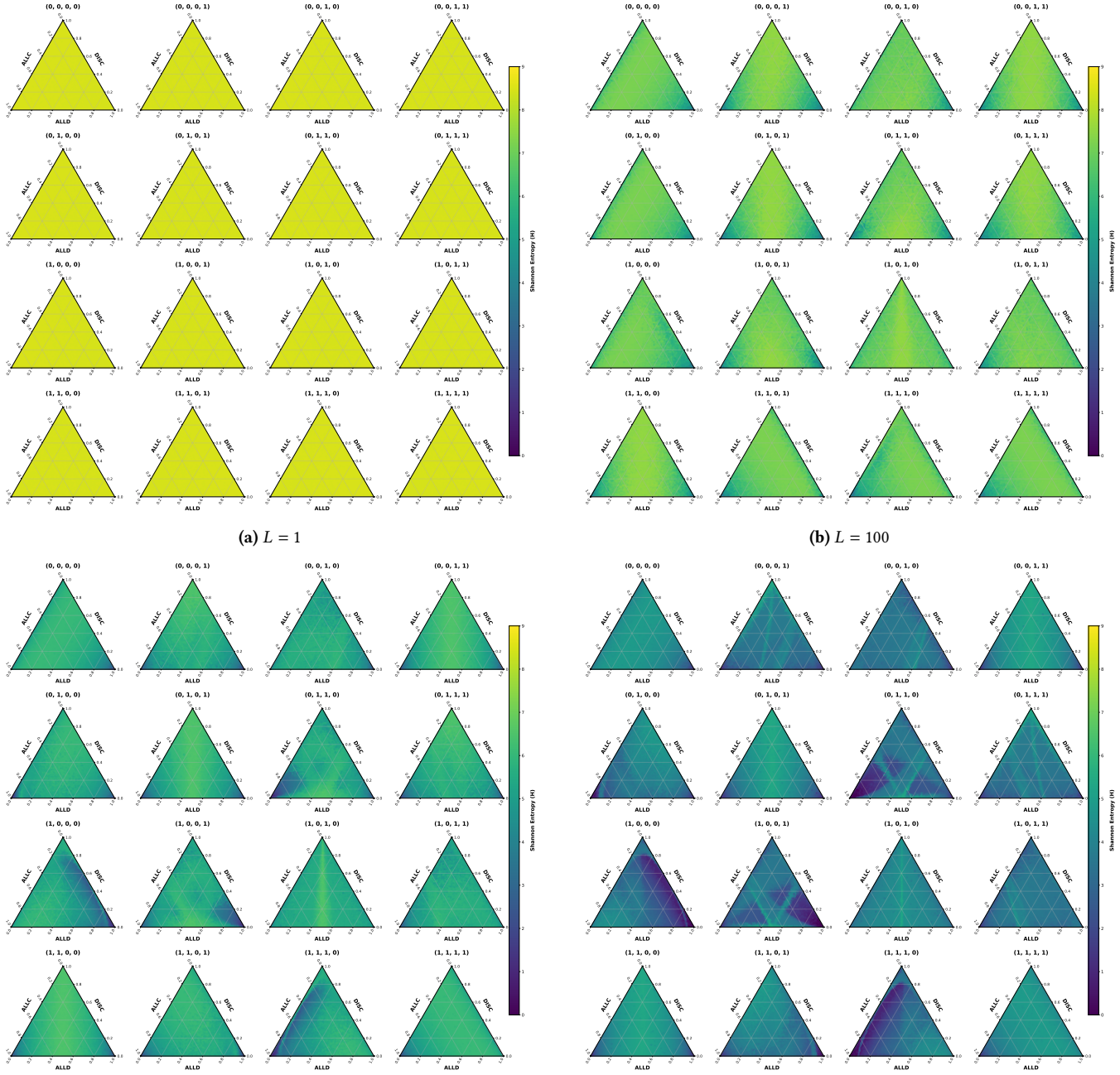


Figure 6: Evaluated cooperation for a randomly selected social norm and two fixed norms: Image Score ($d = [1, 0, 1, 0]$) and Stern-Judging ($d = [1, 0, 0, 1]$). We observe that cooperation under a randomly selected social norm is particularly low, while fixed norms achieve moderately high cooperation. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$.

B BELIEF ENTROPY MEASURES

In the main text, we presented both the Shannon entropy and Rao's Quadratic entropy measures across all pairs of strategy states and social norms. We next expand on this analysis by presenting these results split by what social norm was used in each strategy state. These are presented in Figure 7 for Shannon entropy and Figure 8 for Rao's Quadratic entropy. We observe that an increase in the number of interaction observations (L) generally results in a significant reduction in both entropy metrics. This suggests that increased observations generate less distant and more condensed belief distributions over the current strategy state. The entropy metrics are highly norm specific, with patterns of regions of high entropy becoming more apparent as the number of observations increases. This suggests that the previously used norm plays a crucial role in dictating which strategy states become easier to identify in the next timestep.



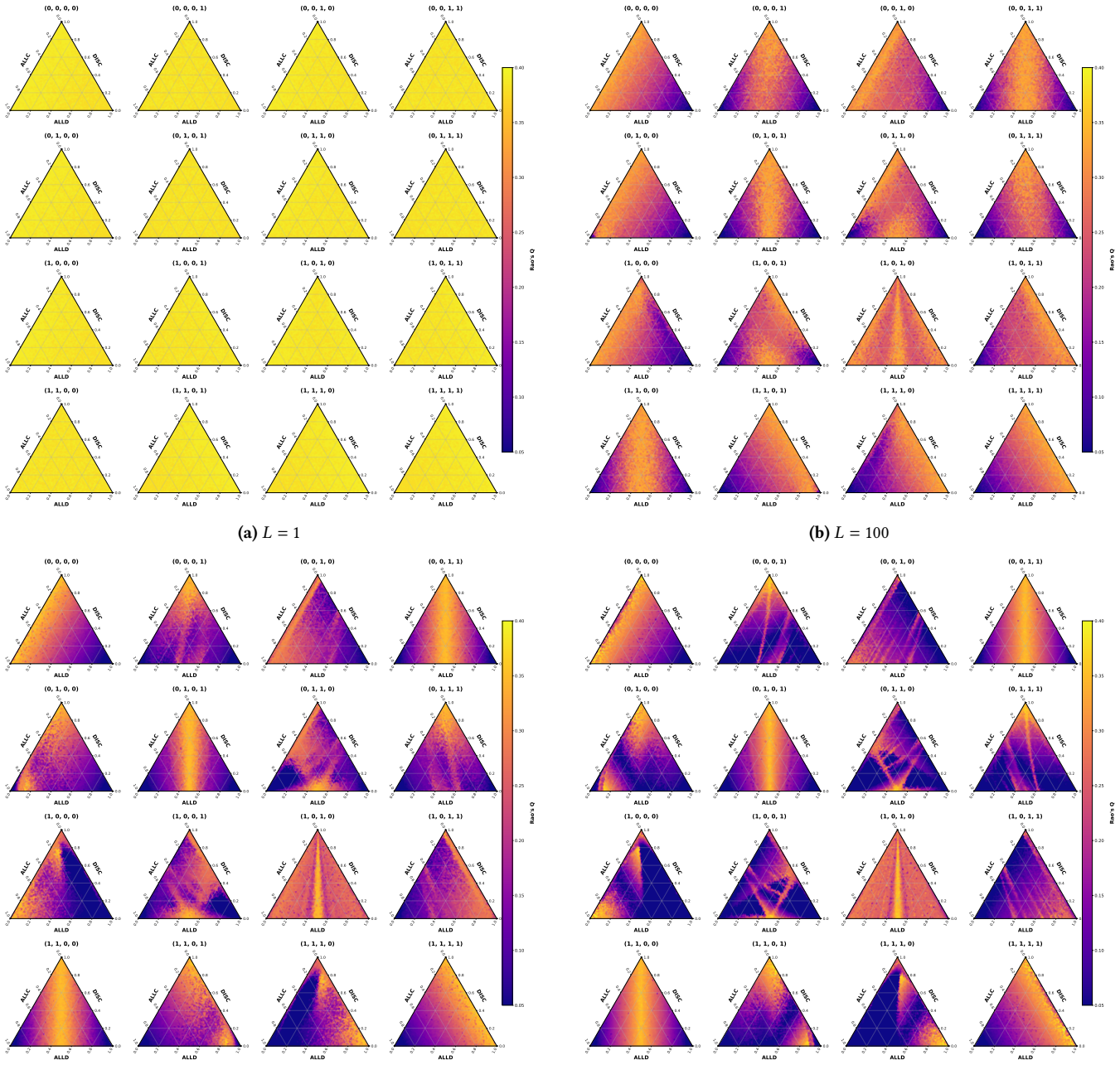
(a) $L = 1$

(b) $L = 100$

(c) $L = 1000$

(d) $L = 10000$

Figure 7: Shannon entropy metric at each state under each social norm, at varying numbers of interaction observations (L). Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.



(a) $L = 1$ **(b)** $L = 100$

(c) $L = 1000$ **(d)** $L = 10000$

Figure 8: Rao's Quadratic entropy metric at each state under each social norm, at a varying number of interaction observations (L). Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

C IMPLEMENTATION OF EACH LEARNING REGIME

Our implementation is based on the standard implementation of PPO in PureJaxRL [27], which we extend for a total of 4 variants of the learning algorithm exploring different numbers of parallel workers and the reset logic of the environment. The 4 variants are introduced together with a discussion of their parameters, the parameter selection method and a discussion of the impact of the different learning parameters in our environment. In addition to performance on the learning problem, the different training regimes are motivated in terms of their feasibility for practical applications given the assumptions they make about the environment (eg, Assume periodic resets or multiple

workers in parallel). Table 1 reports the design choices and parameters of our implementation. After introducing the different schemes, the hyperparameter search results are presented together with a brief discussion on the most influential hyperparameters.

Table 1: PPO Implementation Details

Network Architecture	
Shared trunk	3×256 units, tanh activation
Actor head	Linear layer $\rightarrow$ 16 logits (softmax)
Critic head	Linear layer $\rightarrow$ scalar value
Initialization	Orthogonal: $\sqrt{2}$ (hidden), 0.01 (actor), 1.0 (critic)
Fixed Hyperparameters	
Optimizer	Adam with constant LR (no scheduling)
Max Gradient Norm	0.5
Value loss coefficient	$c_v = 1.0$
Training Duration	2×10^6 environment steps
Optimized Hyperparameters (Grid Search)	
Learning rate	$\alpha \in \{0.001, 0.01, 0.05\}$
Batch size	$\in \{128, 256\}$
Rollout length	$T \in \{512, 1024\}$
Update epochs	$K \in \{10, 20\}$
PPO clip range	$\epsilon \in \{0.2, 0.4\}$
Entropy coefficient	$c_e \in \{0.01, 0.05, 0.1\}$
Discount factor	$\gamma \in \{0.95, 0.99\}$
GAE lambda	$\lambda \in \{0.95, 0.99\}$
Reset frequency	$\in \{1, 5, 10\}$ rollouts (OER/MER only)
Parallel Environments	$\in \{8, 32\}$ (MENR/MER only)
Environment Parameters	
Number of observations	$Z \in \{1, 10, 100, 1000\}$
Implementation	
Framework	JAX + Flax (JIT compilation)

C.1 One worker no resets (OWNR)

This training regime assumes the environment cannot be reset during deployment: the initial state of each training episode follows directly from the final state of the previous episode. Furthermore, experience is collected by a single worker. This setup reflects many practical applications in a social system, where multiple environmental states cannot be explored simultaneously and the environment cannot be reverted to previous states.

C.2 One worker with resets (OWR)

While maintaining a single worker, this training approach assumes an episodic environment that is occasionally reset to a new state. We introduce a training parameter: Reset Frequency, which controls how many rollout-update cycles the agent completes before the environment resets to a randomly sampled initial state. Given the environment dynamics, the agent often spends extended periods in a limited portion of the state space (In particular near monomorphic states, where most of the population adopts ALLD or DISC), developing a policy that overfits to that region. This leads to catastrophic forgetting, where the network shifts its representational capacity to focus predominantly on these frequently visited states. By occasionally resetting the environment to a randomly sampled state, we force the agent to explore different portions of the state space, exposing it to a broader range of experiences and breaking the cycle of repeated updates on the same subset of data.

C.3 Parallel workers no resets (PWNR) and parallel workers with resets (PMWR)

These training regimes introduce several parallel workers, each collecting experience in separate copies of the environment instantiated simultaneously. Experiences from all workers are then combined to perform each update step. We introduce the parameter Number of Environments, which specifies how many parallel workers are instantiated. The initial states for different workers are sampled randomly, creating a more diverse pool of experiences for computing parameter updates. By drawing from this diverse pool, we avoid over-prioritizing certain portions of the state space during updates. However, a potential issue is that all parallel workers might converge to similar states over time (eg, all collapsing to ALLD). To address this, we introduce a variant with resets similar to the OER approach, where all workers are reset to randomly sampled states after a specified number of rollout-update cycles determined by the Reset Frequency hyperparameter.

C.4 Parameter Selection

Parameter selection was guided by a non-exhaustive grid search over a representative set of hyperparameters. The goal was to identify trends and understand the impact of key parameters, rather than optimize each configuration to high precision. We sought to identify representative parameters that highlight the strengths of each training approach, enabling meaningful comparison between them.

We created a parameter grid as specified in Table 1. For each parameter combination, we ran 40 independent trials with different random seeds: $\{1, 2, 3, \ldots, 40\}$. During each trial, we collected training metrics (episode rewards, policy and value function losses, and policy entropy) and periodic evaluations (at quarter intervals of the training run to track global behavior, as reward alone can be an unreliable signal). To measure the performance of a given parameter combination we average the results of the final evaluation (as described in the main text) over the 40 random seeds. Finally, given the number of observations radically changes the structure of the learning task, the process is repeated keeping everything fixed but varying the number of observations sampled in the environment for $L = \{1, 10, 100, 1000\}$.

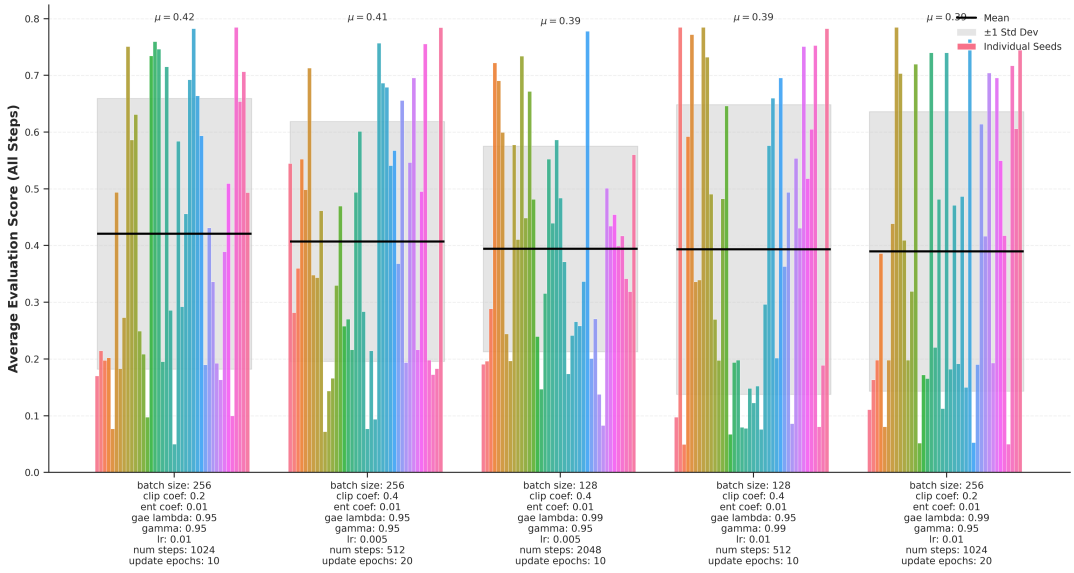


Figure 9: Best performing parameter configurations for One Worker No Resets, when $L = 1$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

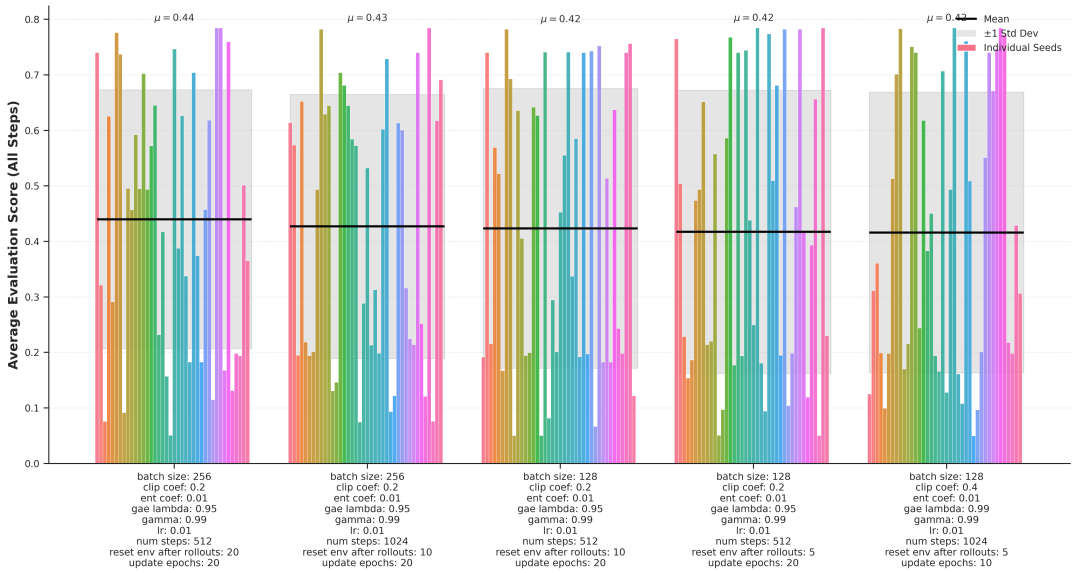


Figure 10: Best performing parameter configurations for One Worker with Resets, when $L = 1$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

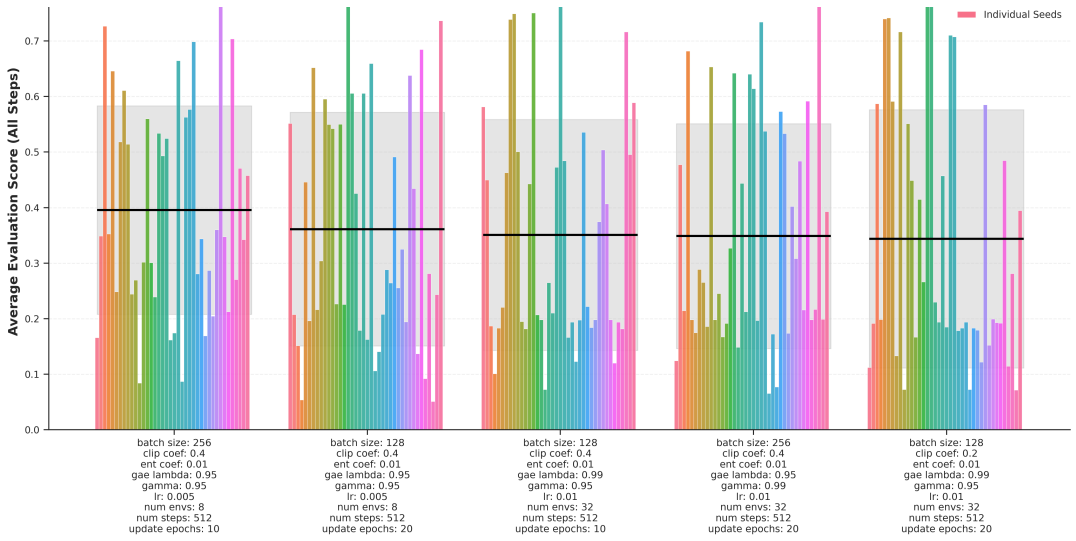


Figure 11: Best performing parameter configurations for Multiple Workers no Resets, when $L = 1$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

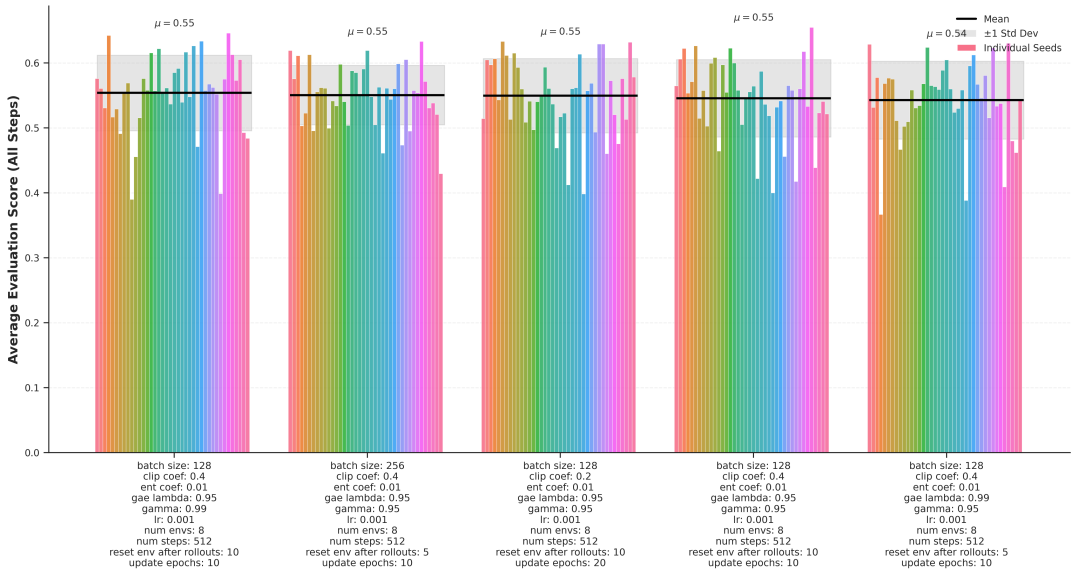


Figure 12: Best performing parameter configurations for Multiple Workers with Resets, when $L = 1$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

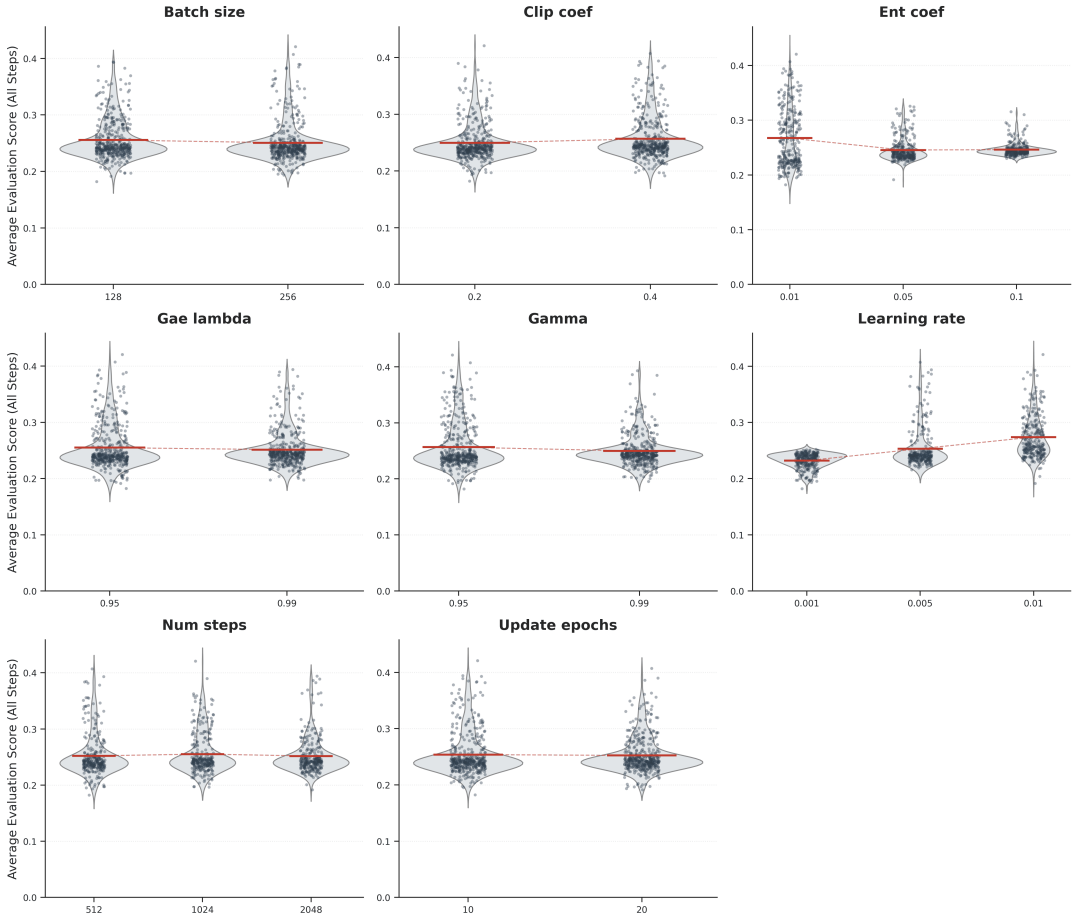


Figure 13: Evaluation distribution for the parameter search using One Worker No Resets, when $L = 1$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

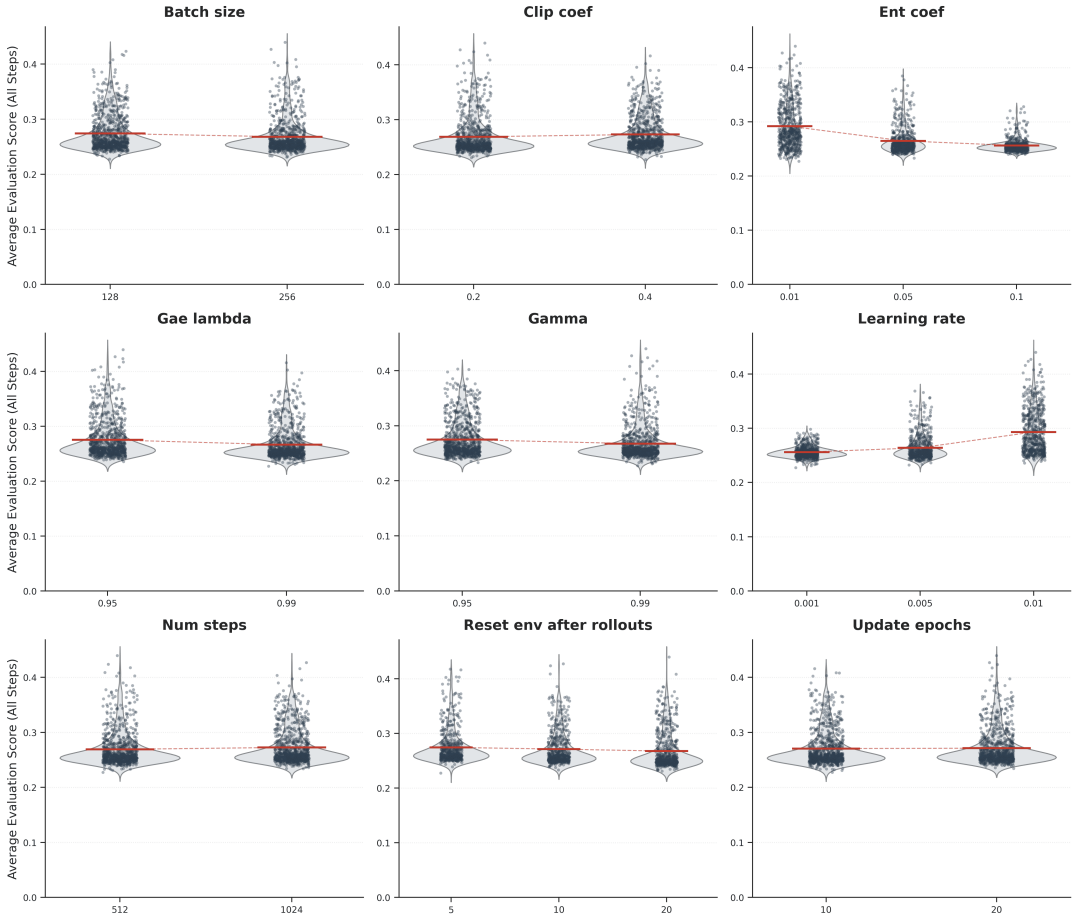


Figure 14: Evaluation distribution for the parameter search using One Worker with Resets, when $L = 1$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

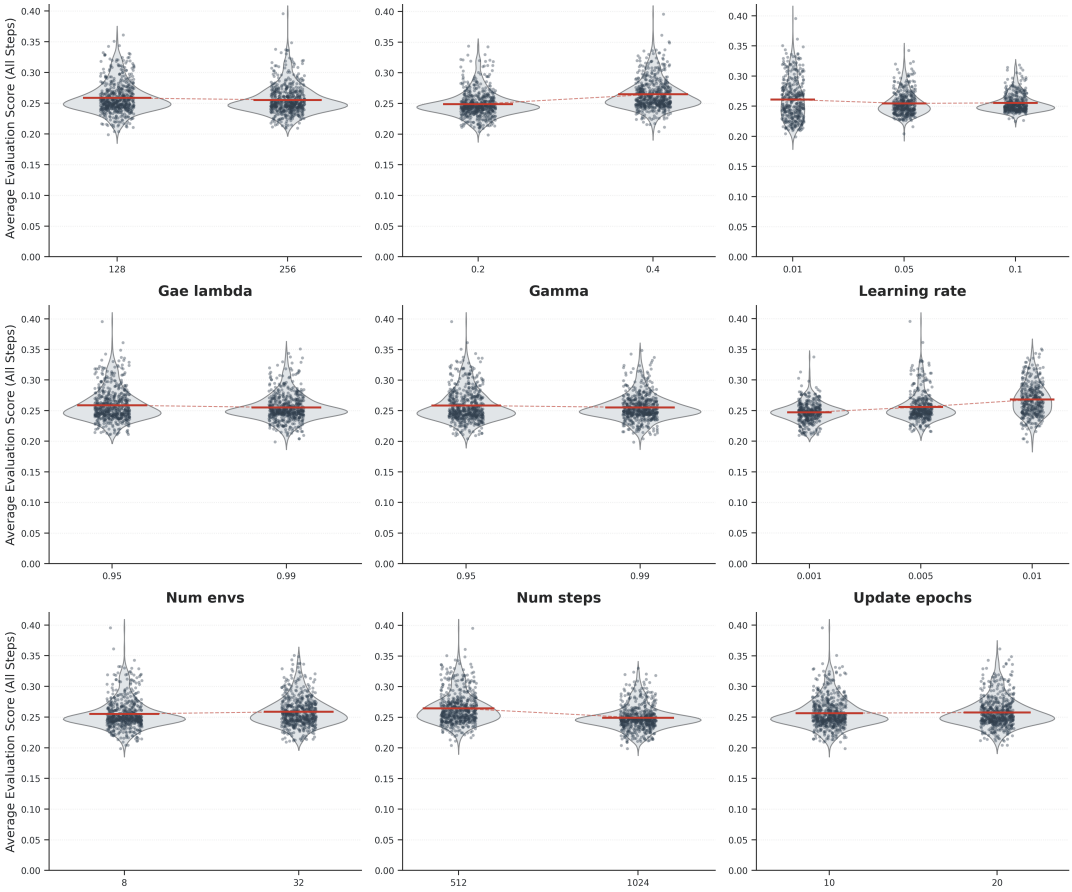


Figure 15: Evaluation distribution for the parameter search using Multiple Workers no Resets, when $L = 1$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

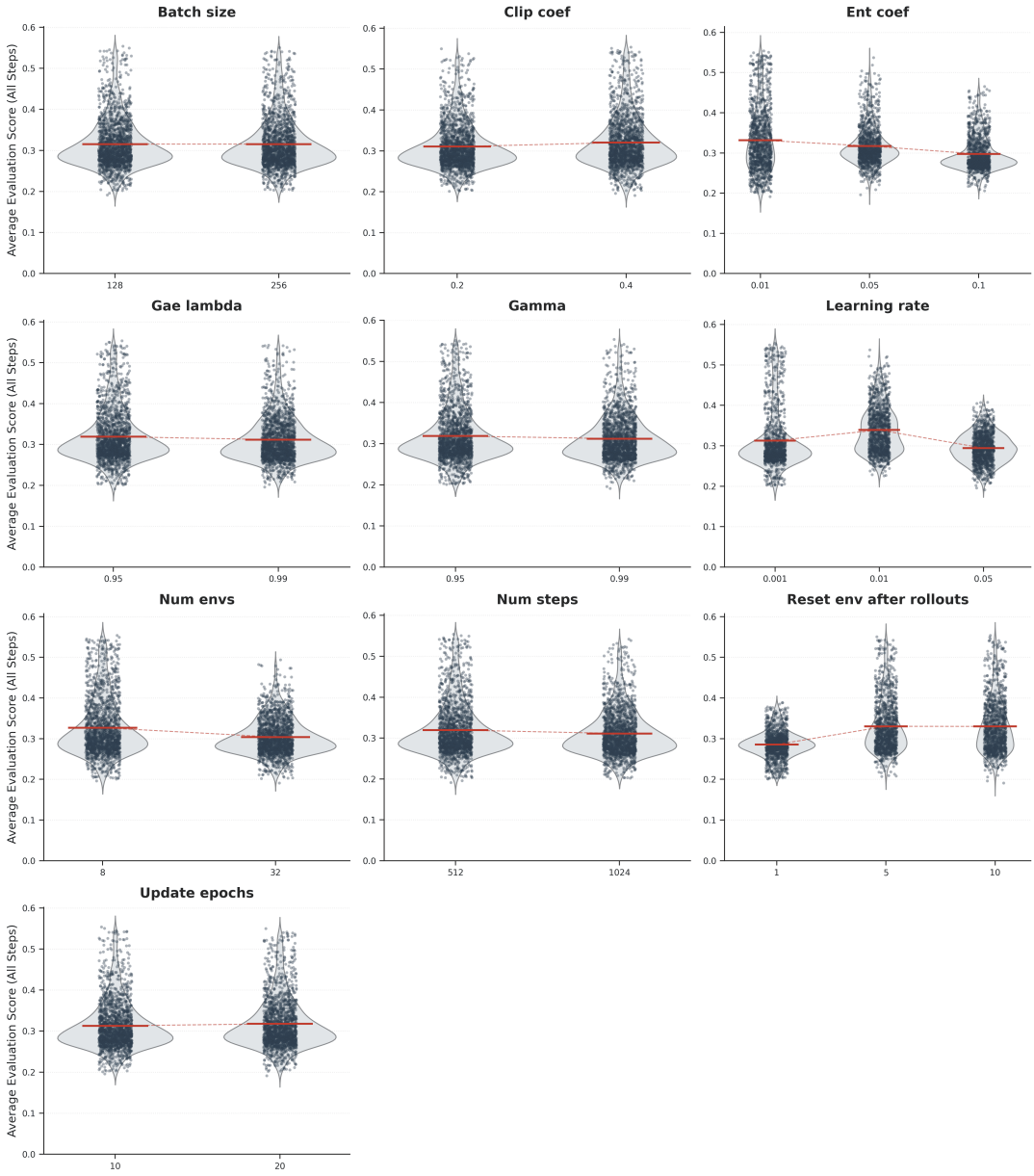


Figure 16: Evaluation distribution for the parameter search using Multiple Workers with Resets, when $L = 1$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

C.4.1 Hyperparameter Sweep Results: $L = 1$.

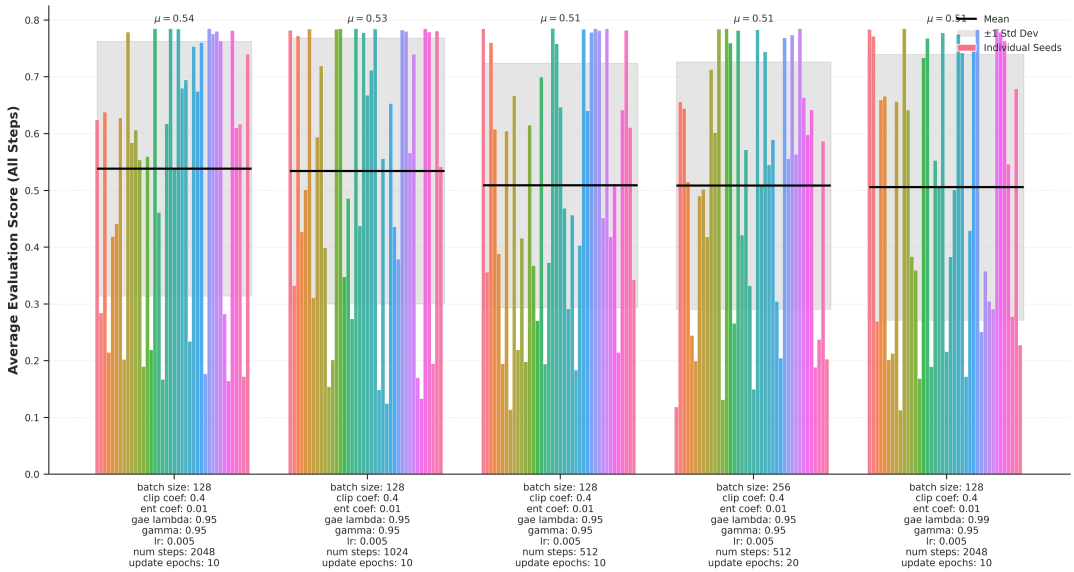


Figure 17: Best performing parameter configurations for One Worker No Resets, when $L = 10$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

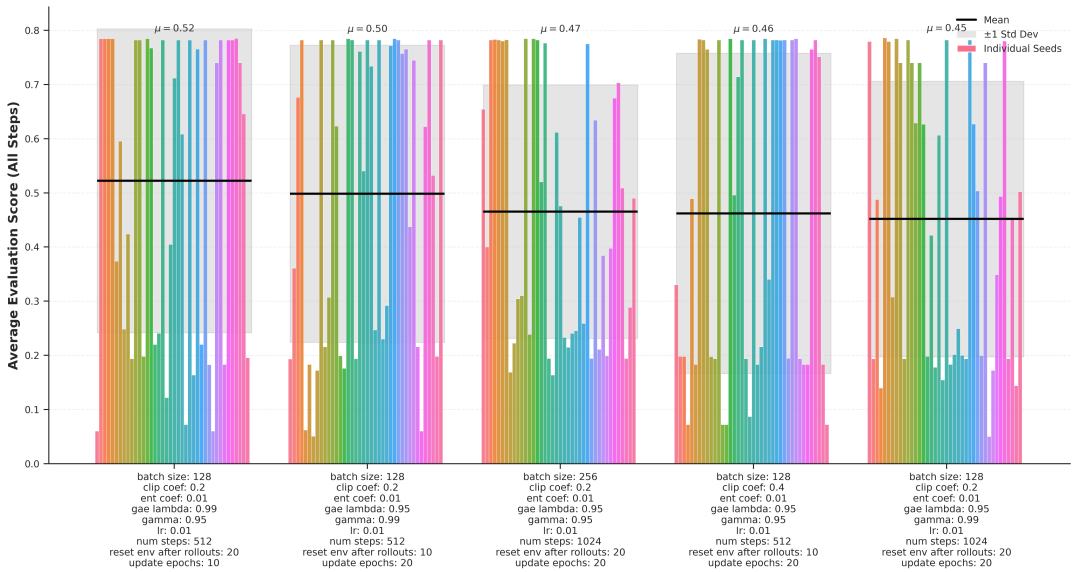


Figure 18: Best performing parameter configurations for One Worker with Resets, when $L = 10$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

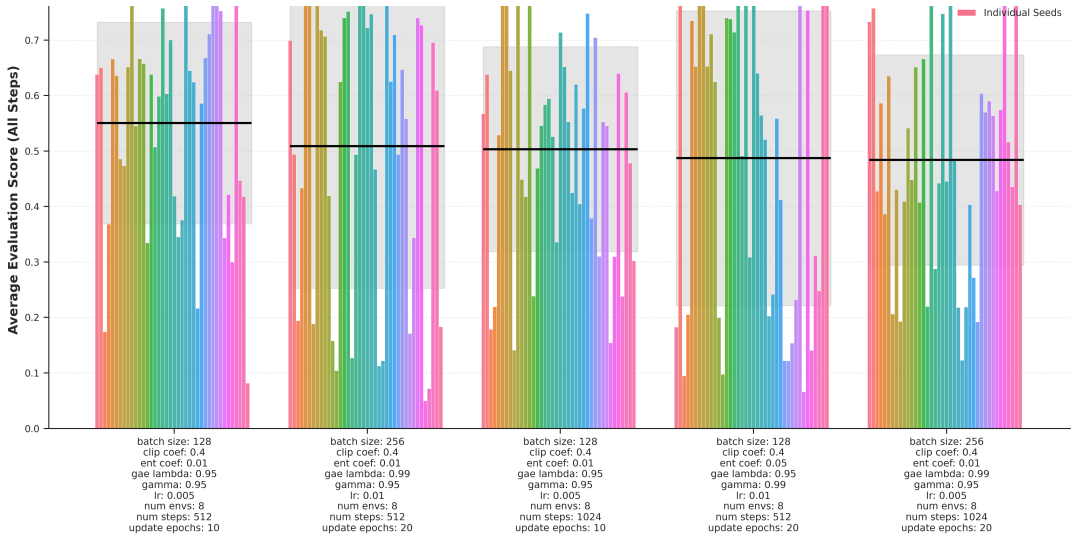


Figure 19: Best performing parameter configurations for Multiple Workers no Resets, when $L = 10$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

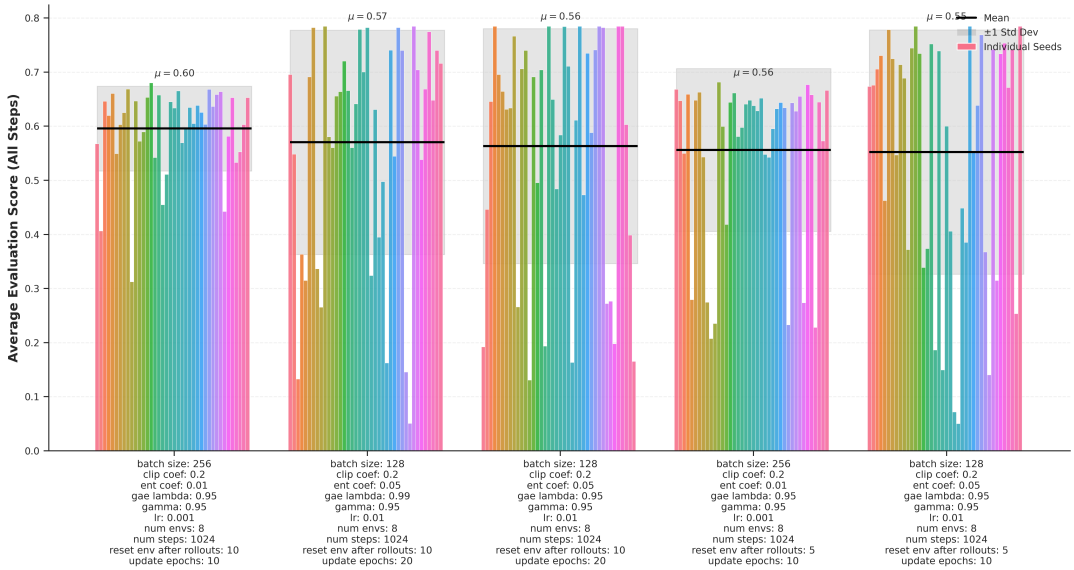


Figure 20: Best performing parameter configurations for Multiple Workers with Resets, when $L = 10$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

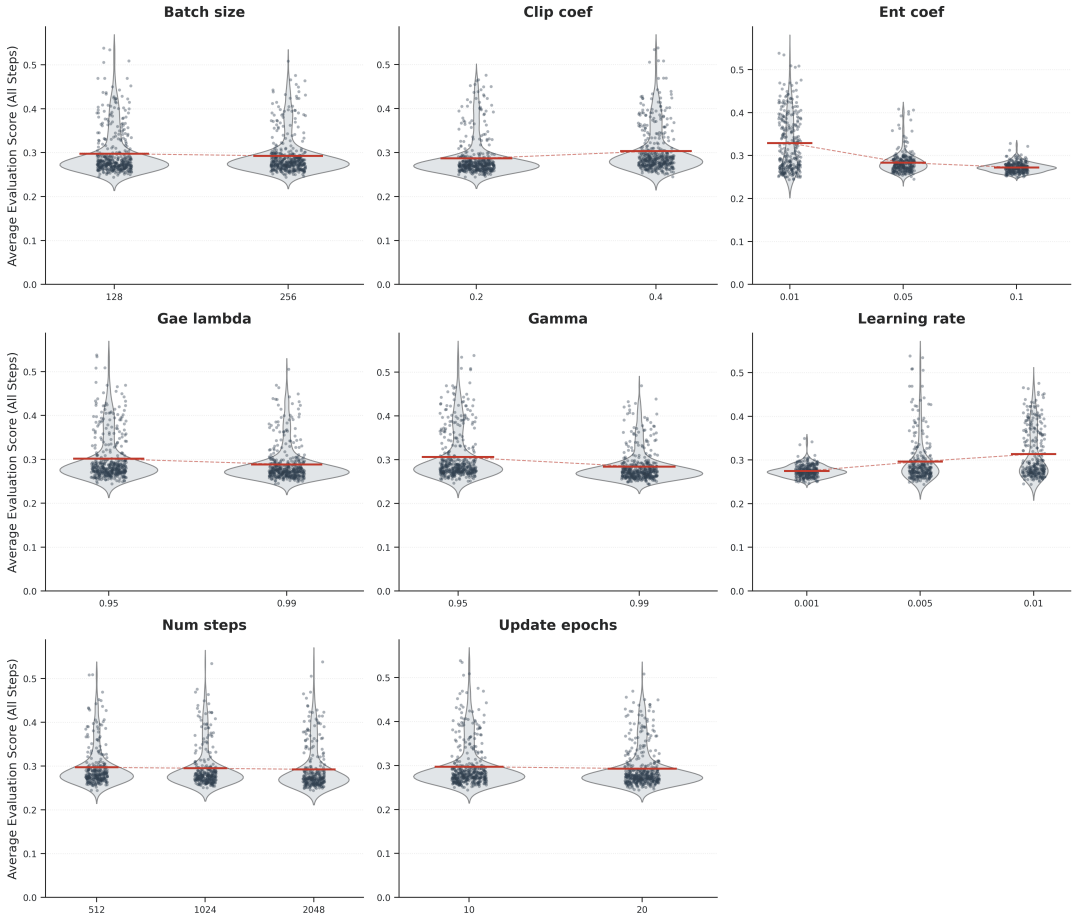


Figure 21: Evaluation distribution for the parameter search using One Worker No Resets, when $L = 10$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

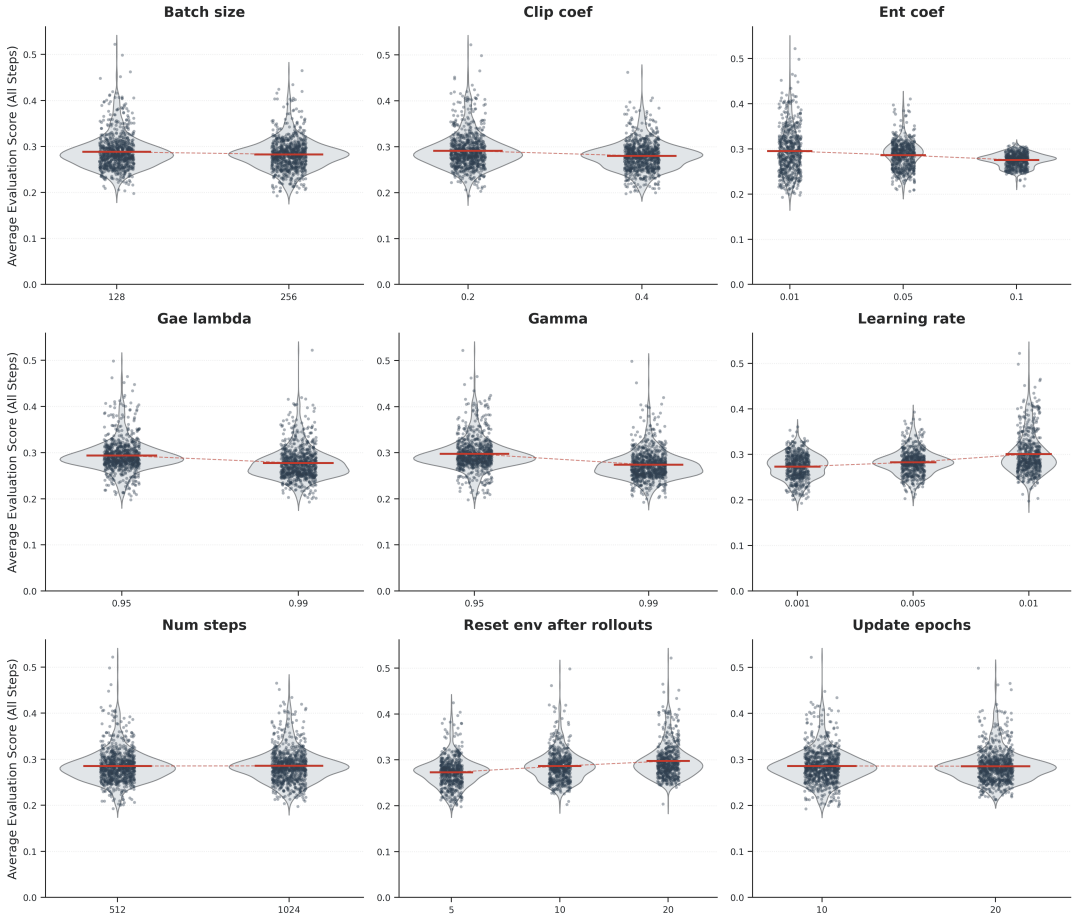


Figure 22: Evaluation distribution for the parameter search using One Worker with Resets, when $L = 10$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

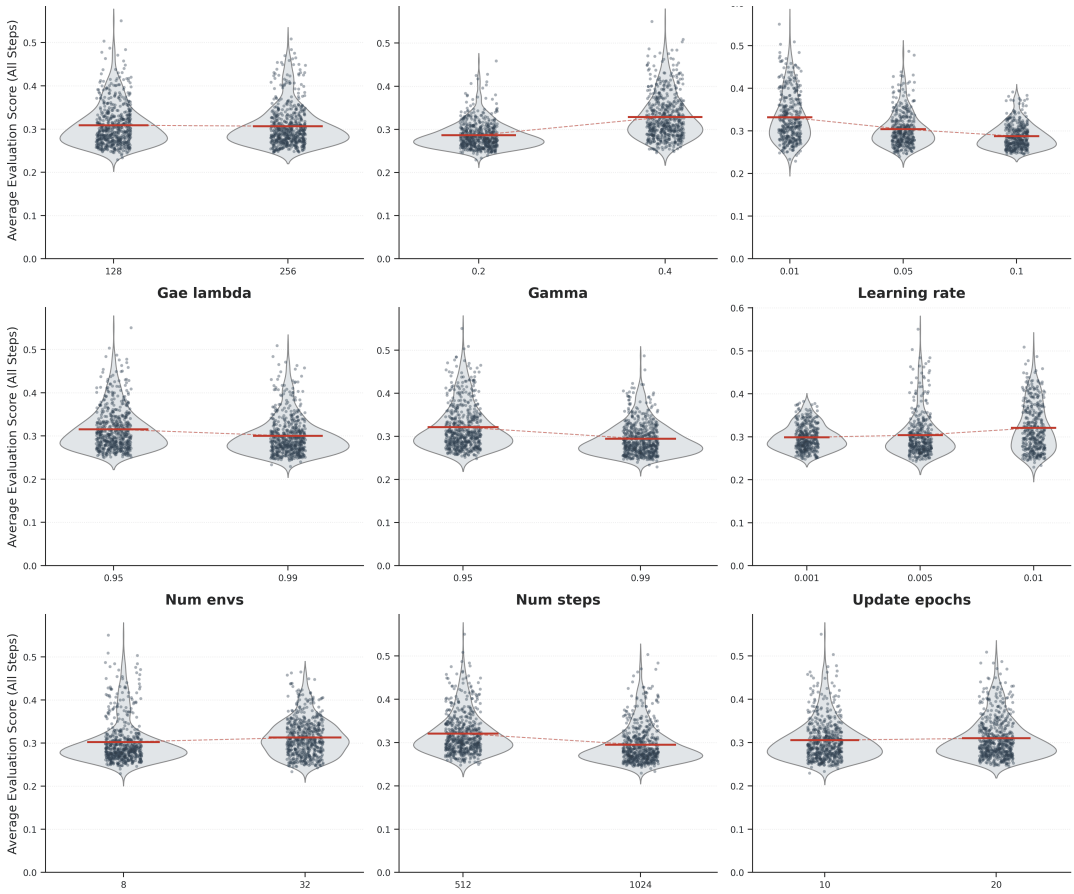


Figure 23: Evaluation distribution for the parameter search using Multiple Workers no Resets, when $L = 10$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

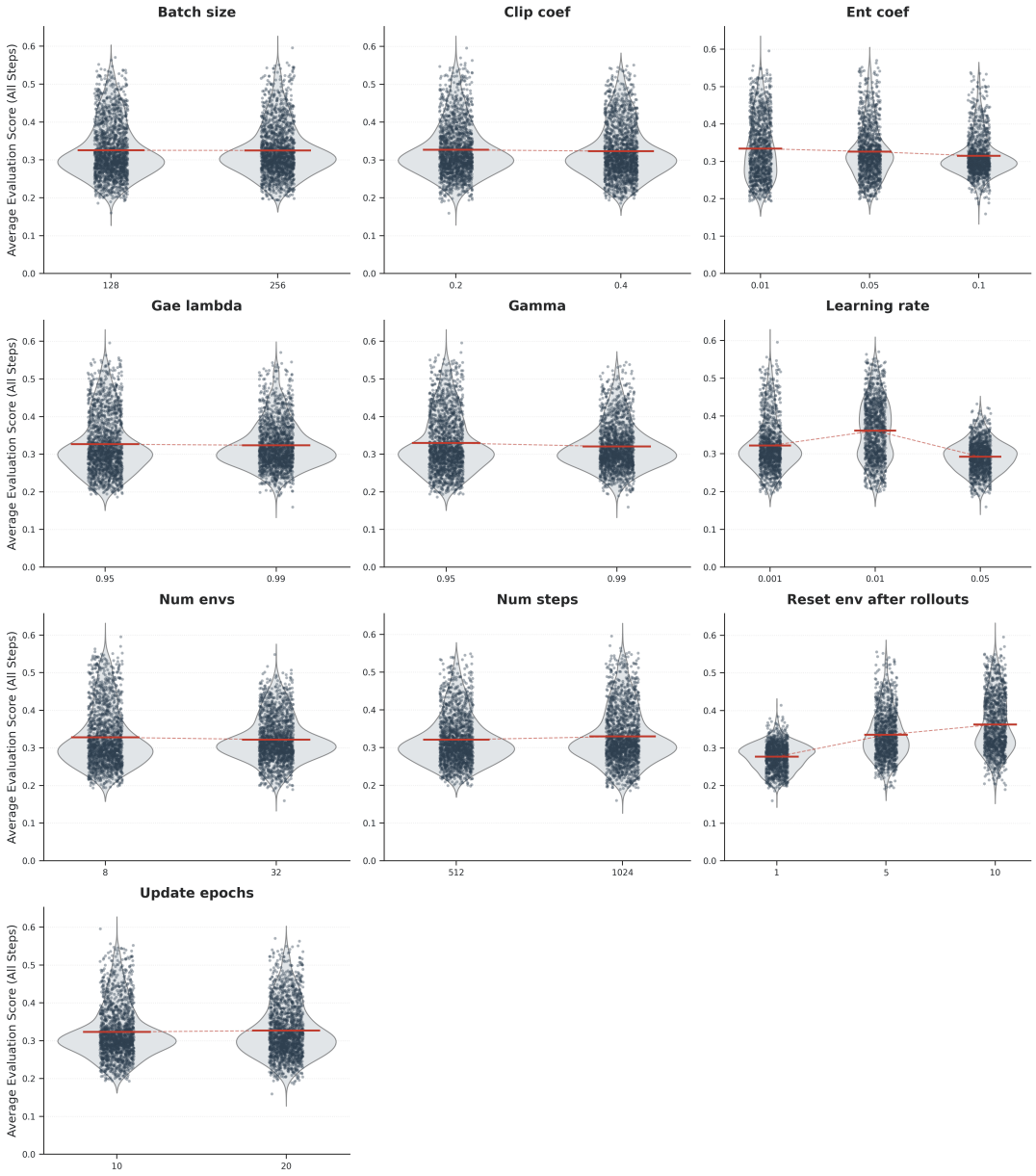


Figure 24: Evaluation distribution for the parameter search using Multiple Workers with Resets, when $L = 10$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

C.4.2 Hyperparameter Sweep Results: $L = 10$.

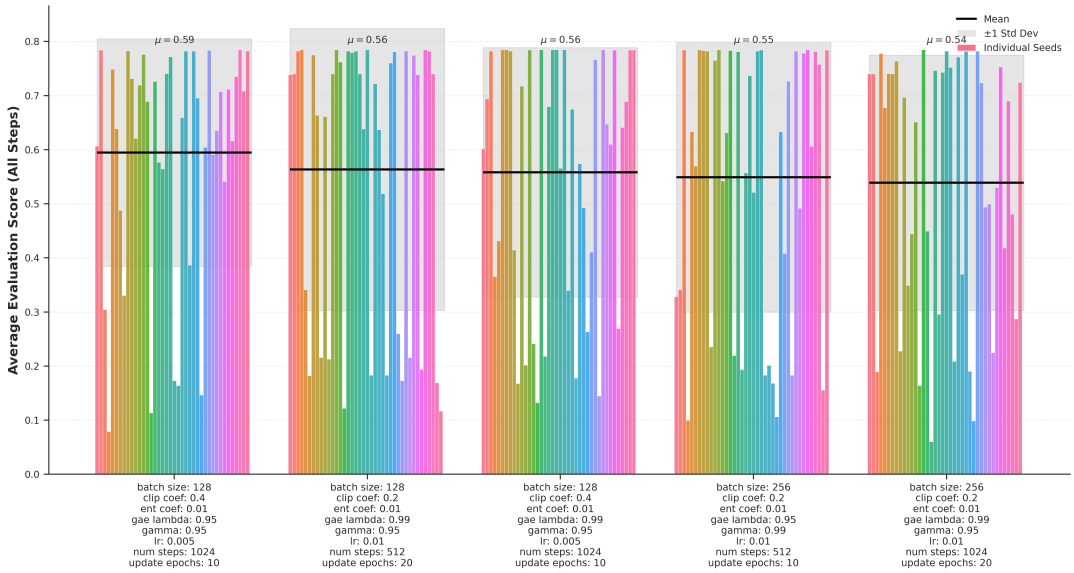


Figure 25: Best performing parameter configurations for One Worker No Resets, when $L = 100$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

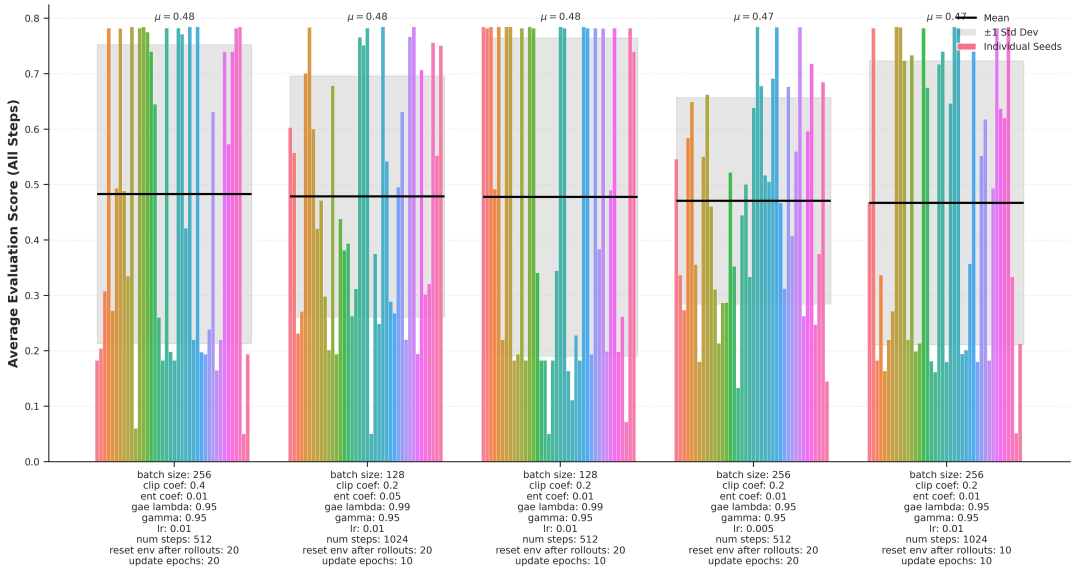


Figure 26: Best performing parameter configurations for One Worker with Resets, when $L = 100$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

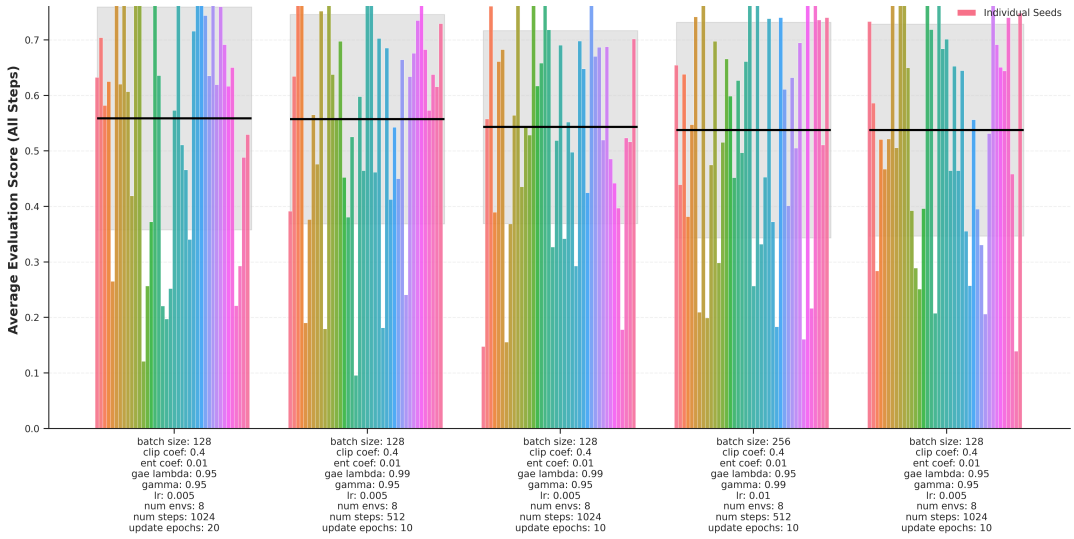


Figure 27: Best performing parameter configurations for Multiple Workers no Resets, when $L = 100$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

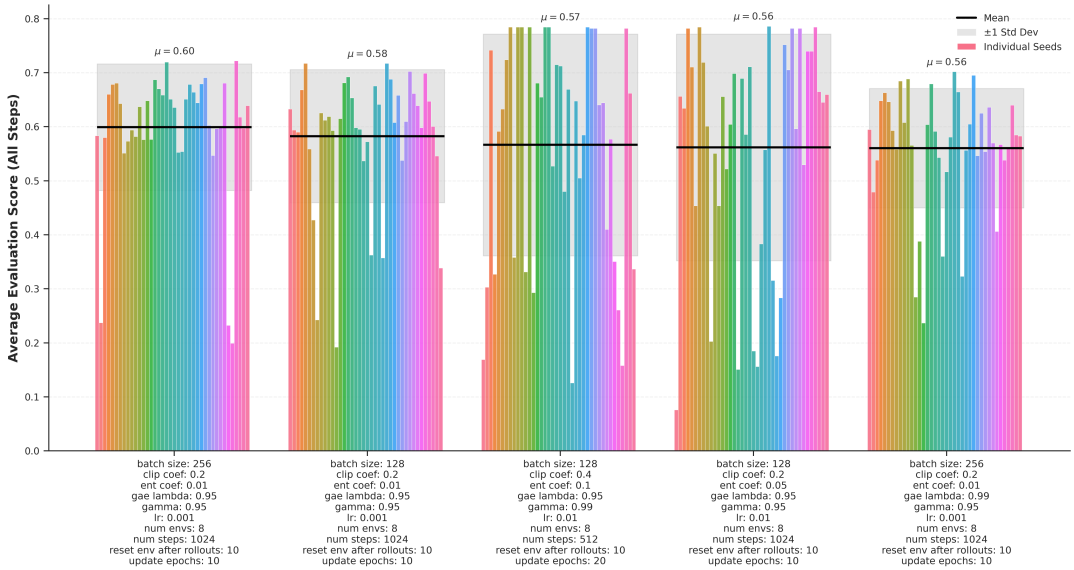


Figure 28: Best performing parameter configurations for Multiple Workers with Resets, when $L = 100$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

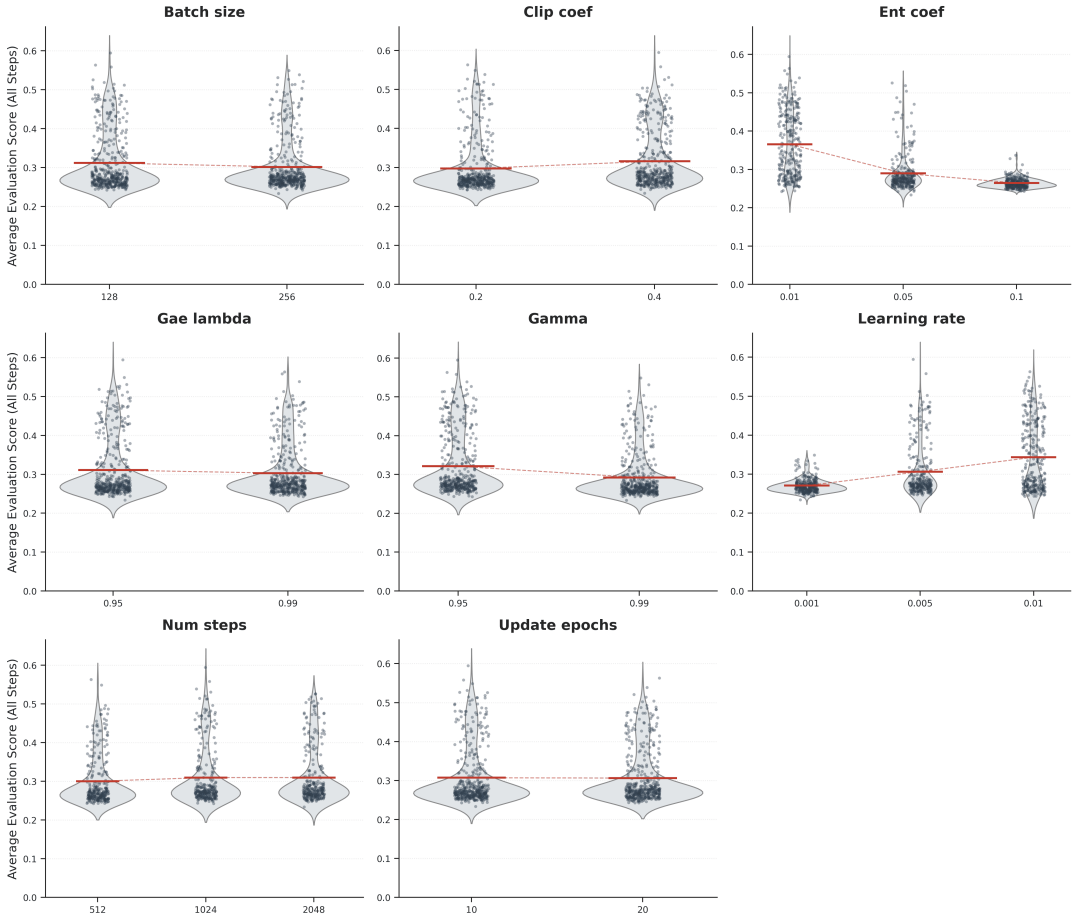


Figure 29: Evaluation distribution for the parameter search using One Worker No Resets, when $L = 100$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

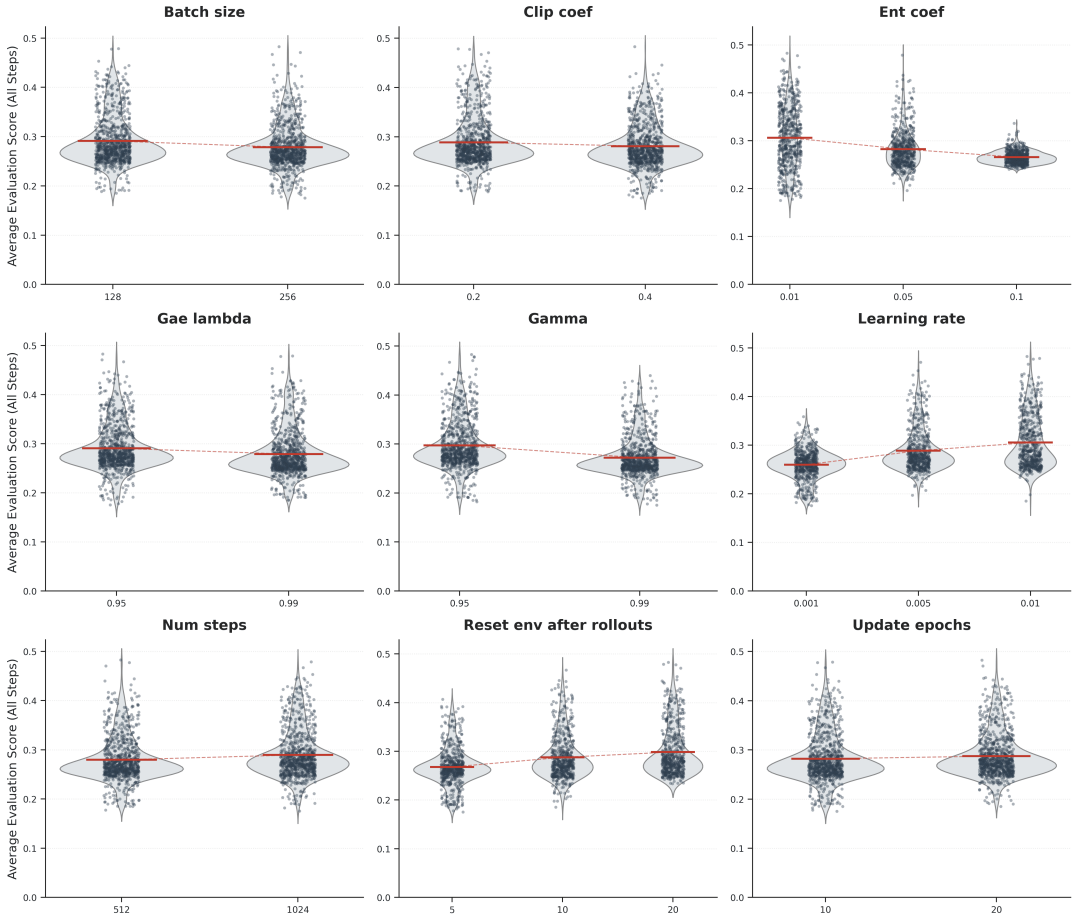


Figure 30: Evaluation distribution for the parameter search using One Worker with Resets, when $L = 100$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

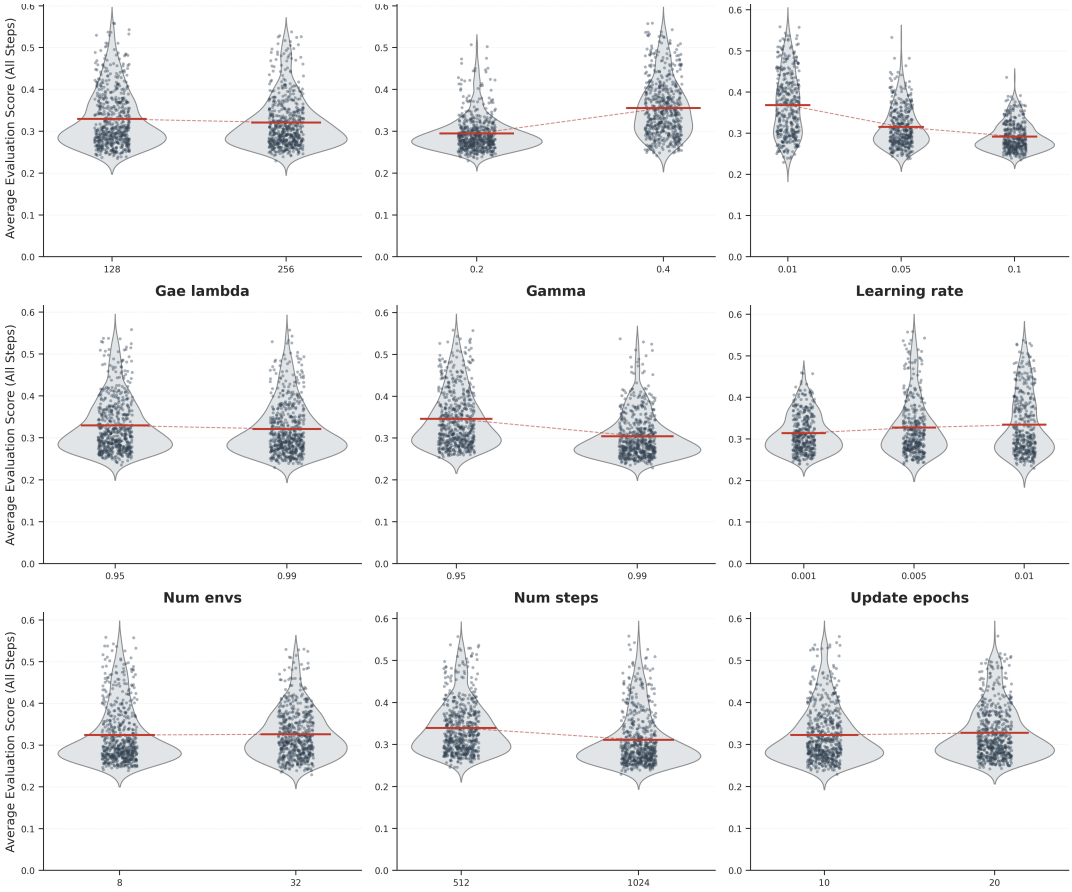


Figure 31: Evaluation distribution for the parameter search using Multiple Workers no Resets, when $L = 100$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

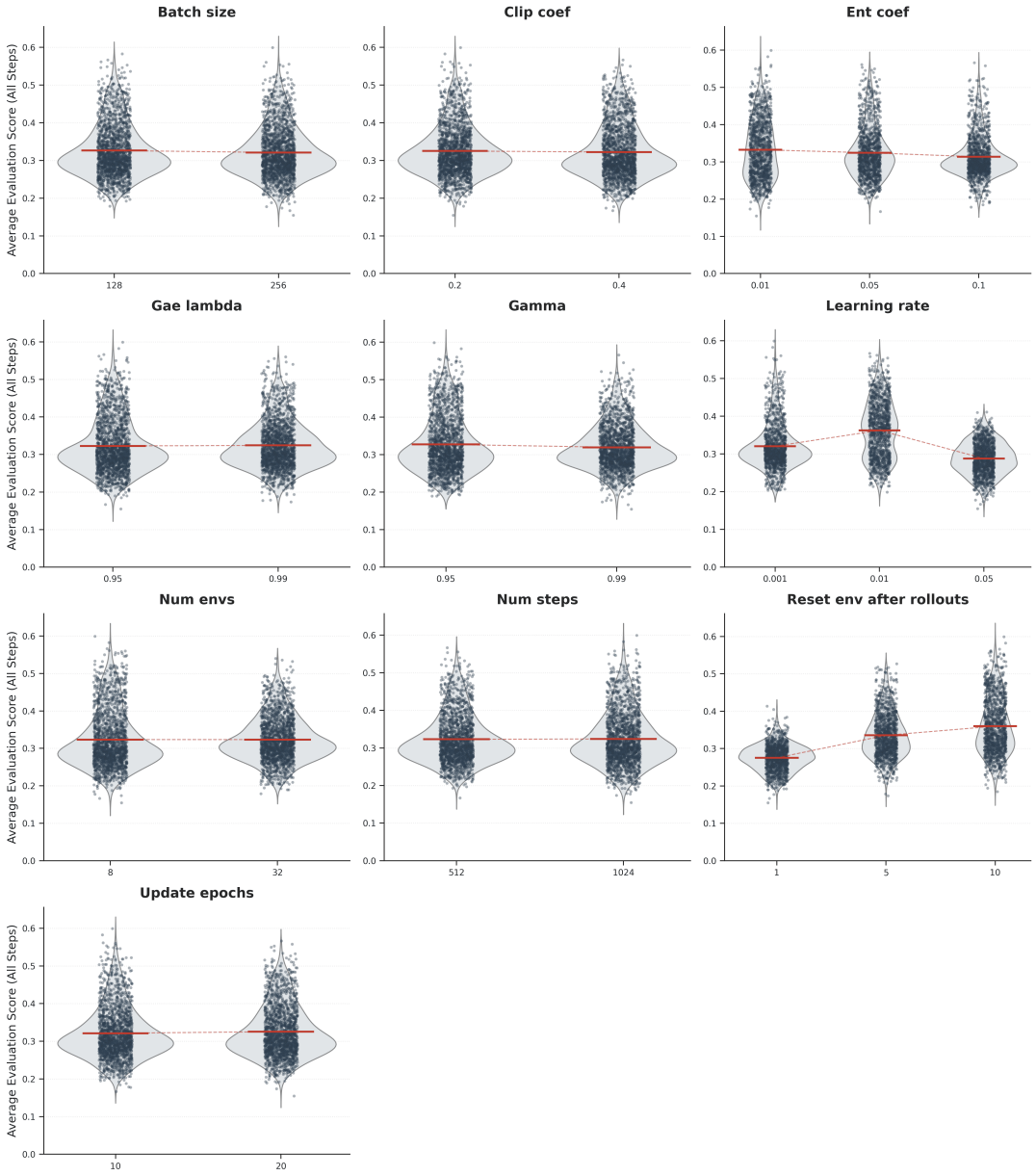


Figure 32: Evaluation distribution for the parameter search using Multiple Workers with Resets, when $L = 100$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

C.4.3 Hyperparameter Sweep Results: $L = 100$.

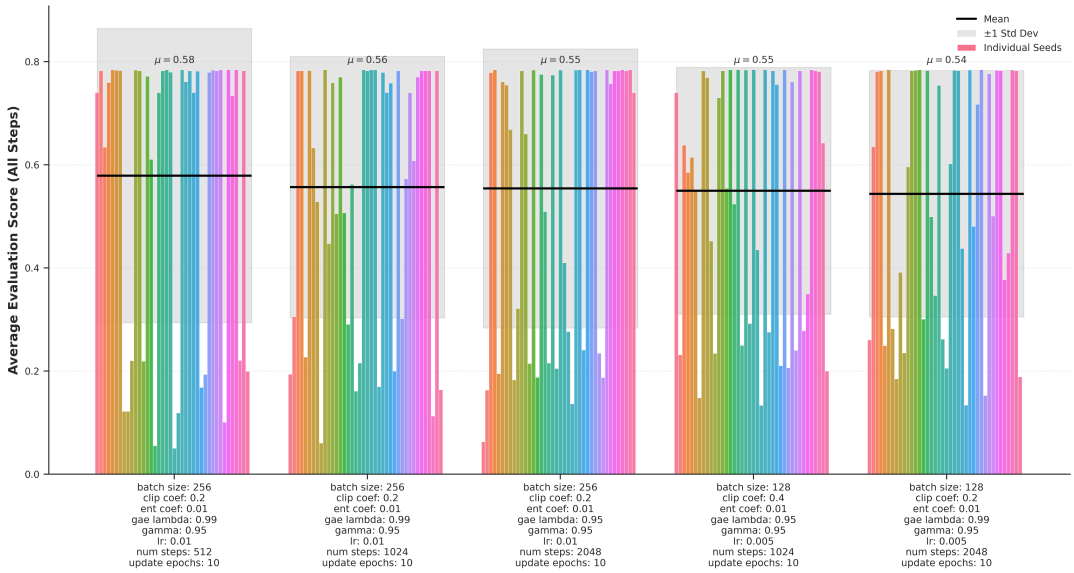


Figure 33: Best performing parameter configurations for One Worker No Resets, when $L = 1000$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

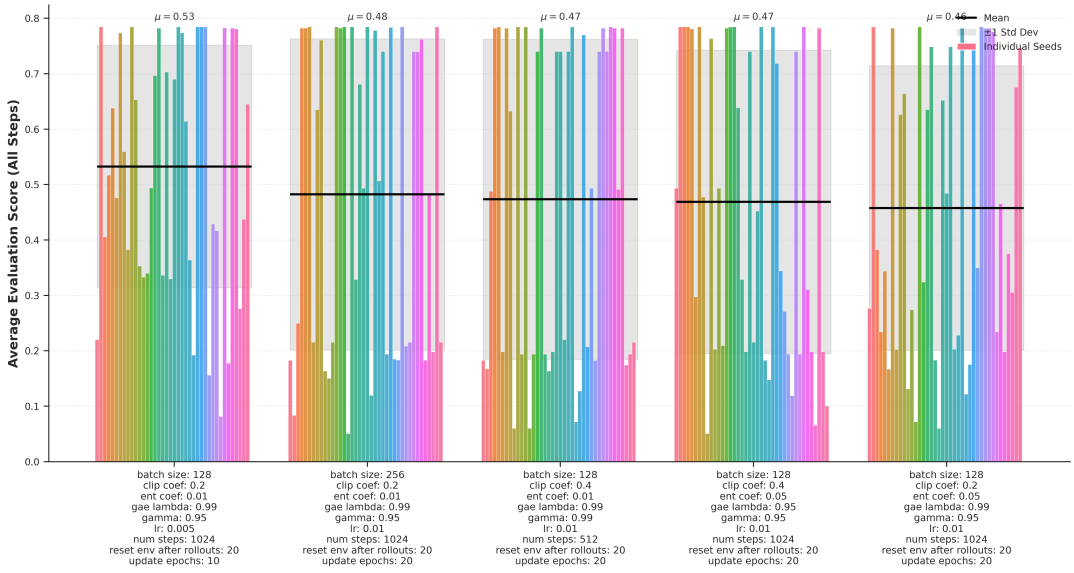


Figure 34: Best performing parameter configurations for One Worker with Resets, when $L = 1000$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

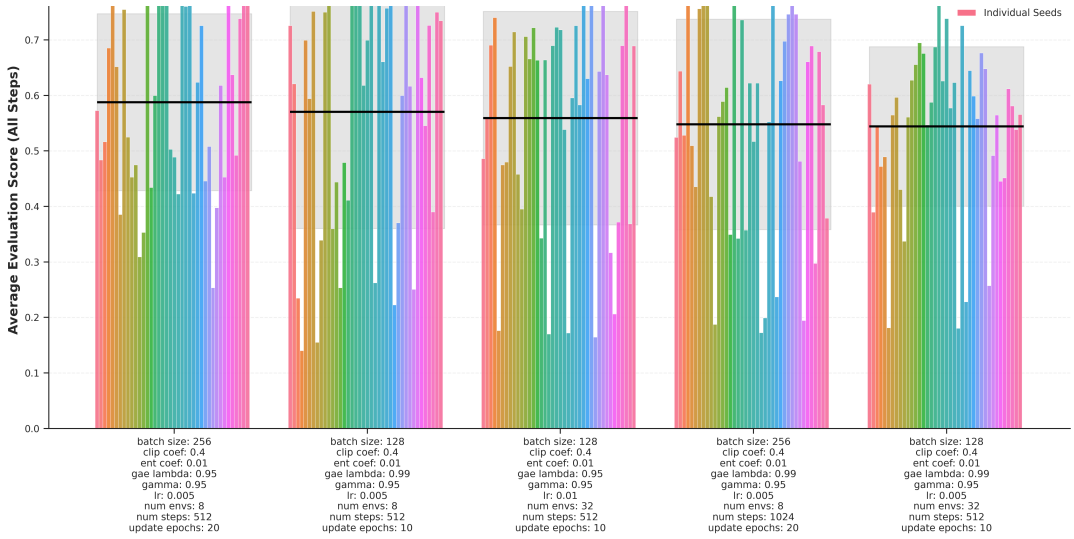


Figure 35: Best performing parameter configurations for Multiple Workers no Resets, when $L = 1000$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

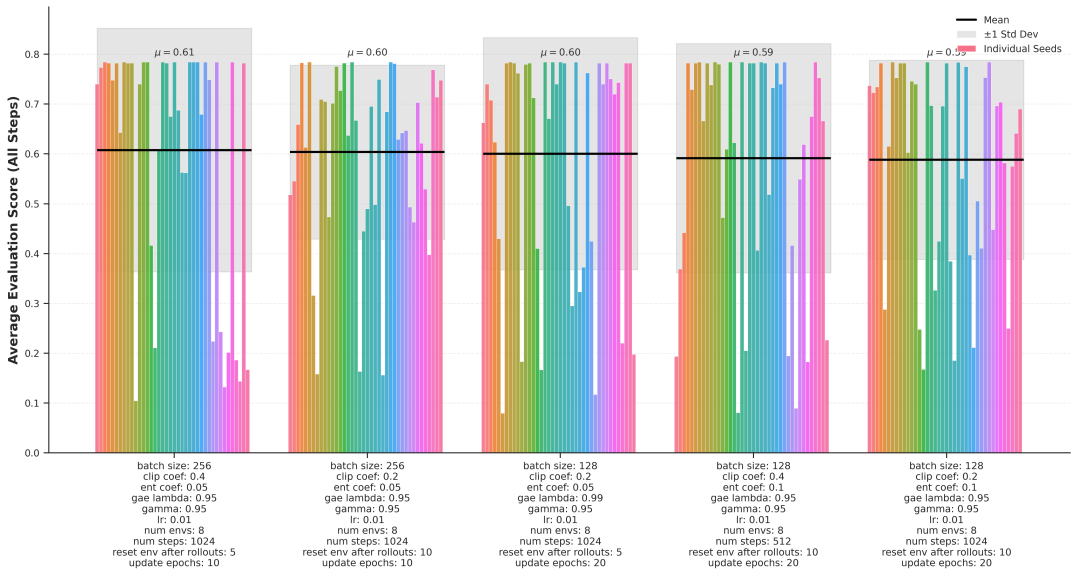


Figure 36: Best performing parameter configurations for Multiple Workers with Resets, when $L = 1000$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

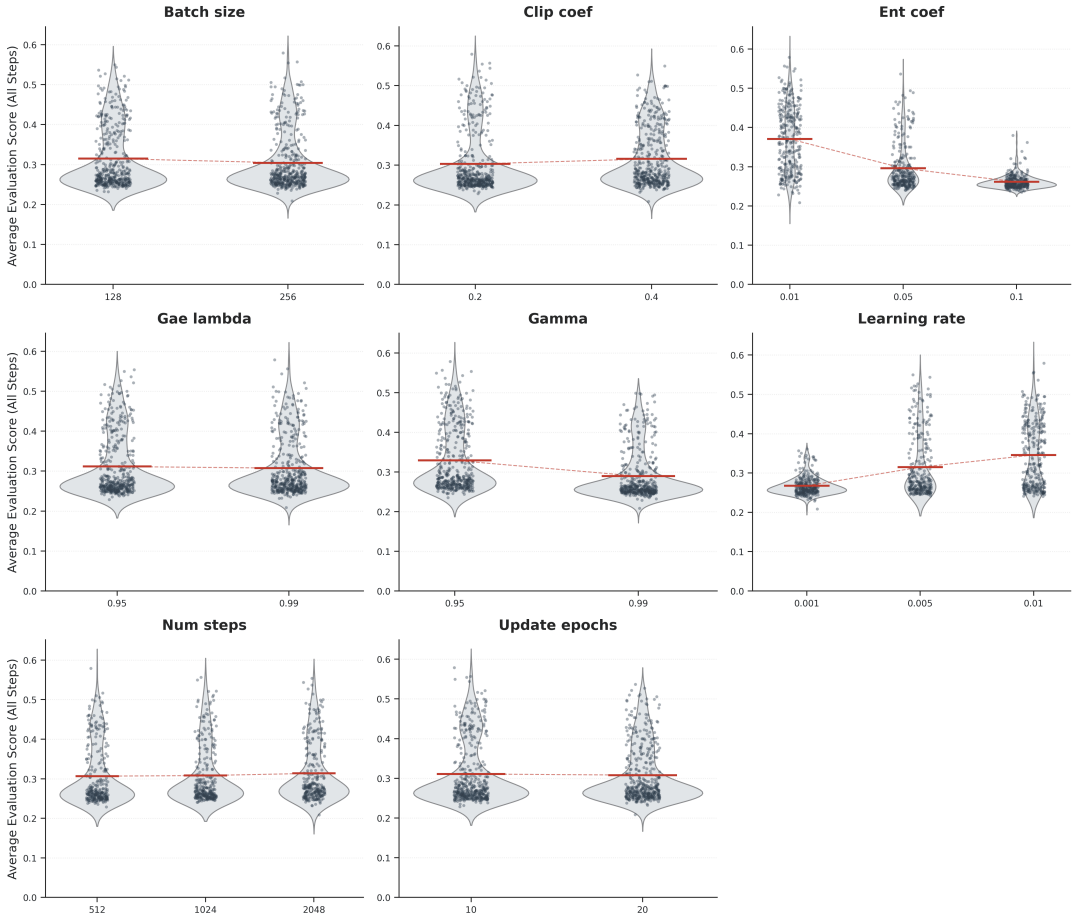


Figure 37: Evaluation distribution for the parameter search using One Worker No Resets, when $L = 1000$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

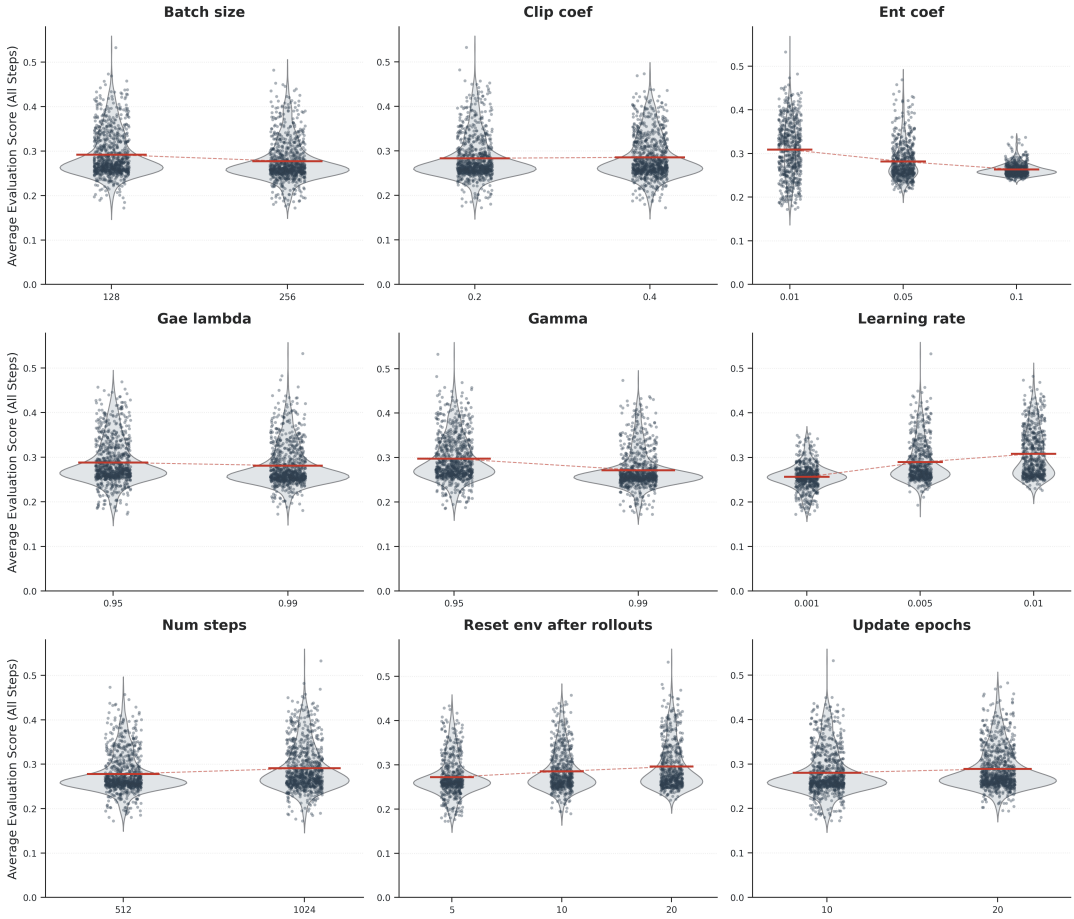


Figure 38: Evaluation distribution for the parameter search using One Worker with Resets, when $L = 1000$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

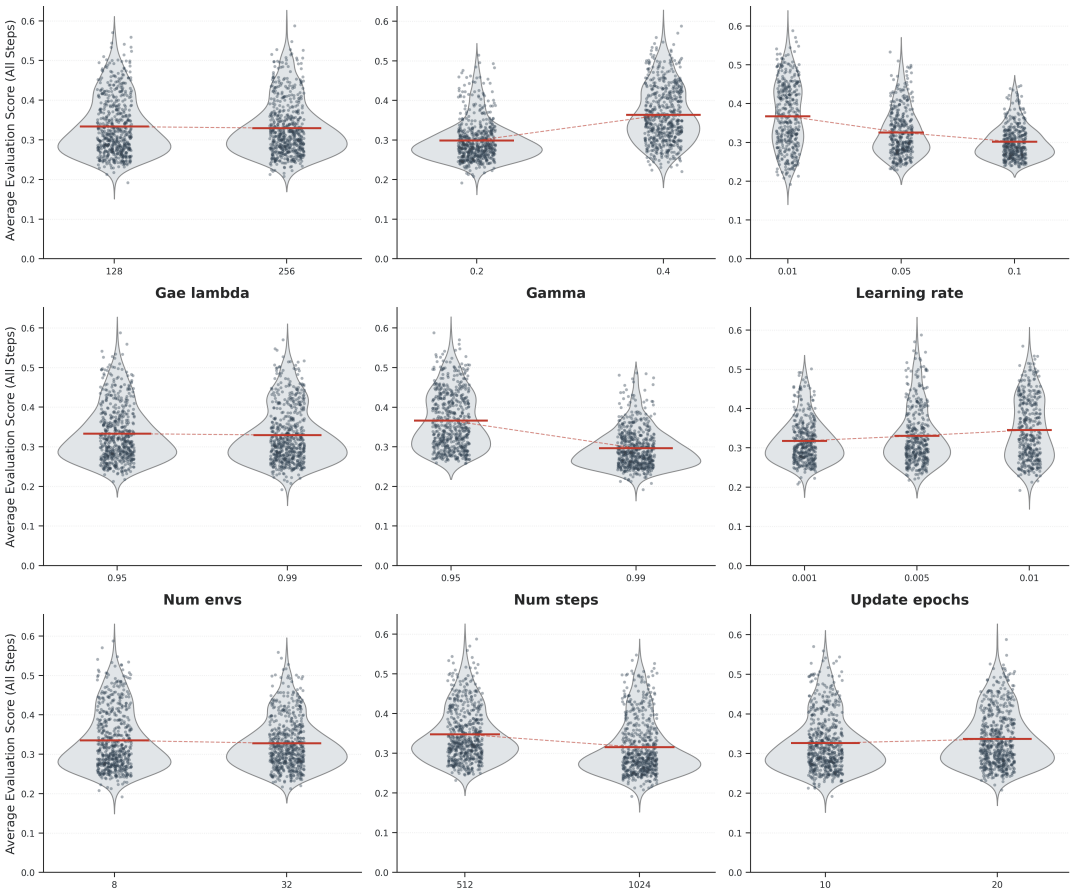


Figure 39: Evaluation distribution for the parameter search using Multiple Workers no Resets, when $L = 1000$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

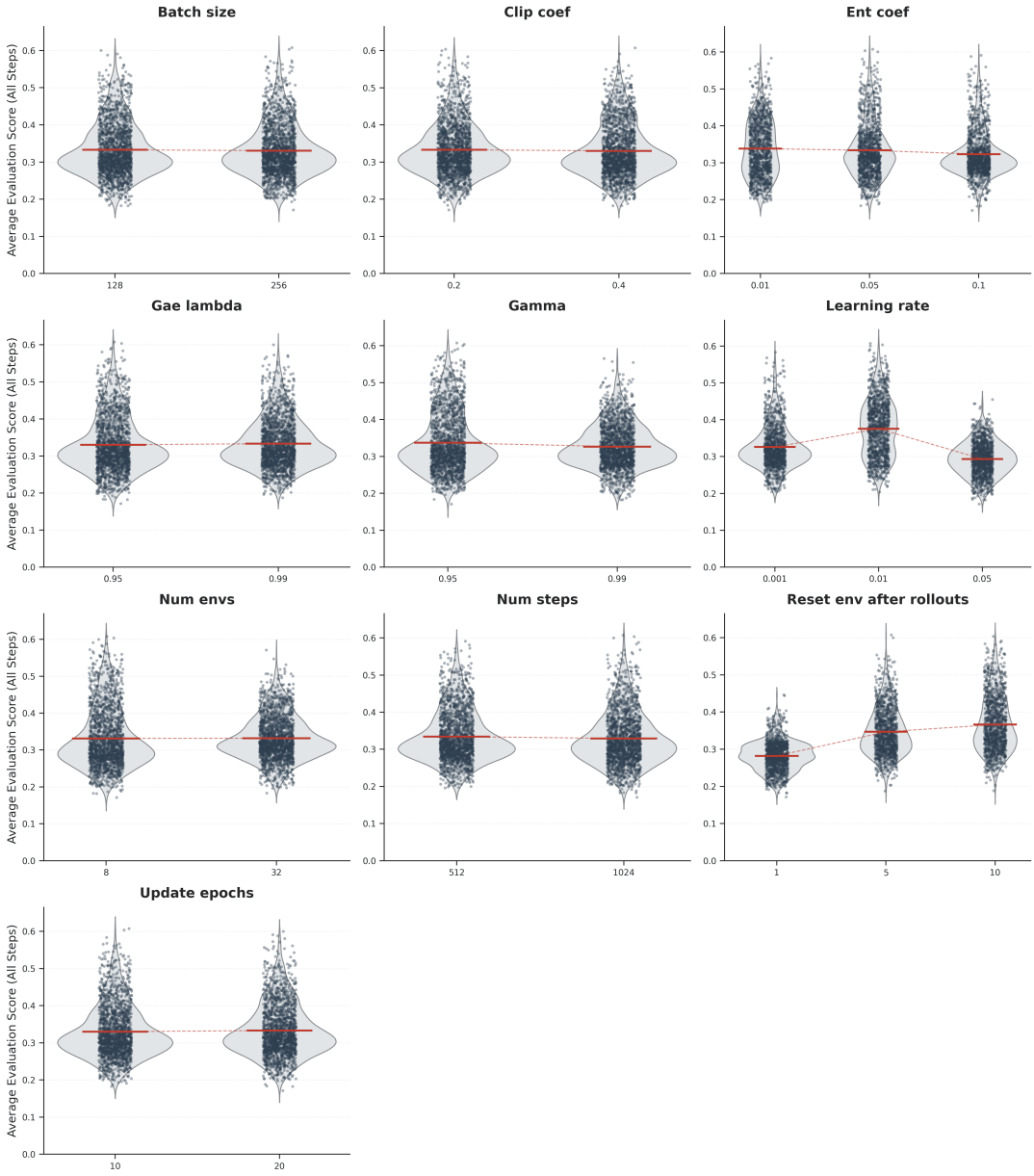


Figure 40: Evaluation distribution for the parameter search using Multiple Workers with Resets, when $L = 1000$. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

C.4.4 Hyperparameter Sweep Results: $L = 1000$.

D COMPLETE EMPIRICAL RESULTS

We next discuss the our empirical results from training agents under different learning regimes (Section D.2), and their resulting evaluated cooperation level (Section D.3) and policy (Section D.4) across different number of observations.

D.1 Catastrophic Forgetting

The environment is characterized by gradual dynamics, where transitions inside the simplex slow down significantly near the monomorphic ALLD and DISC states. Unlike other regions, these states can only be escaped through mutation rather than adaptation. Additionally, the

surrounding states point may have a high imitation rate back towards the monomorphic state, resulting in the policy spending large amounts of time trapped in these regions. This behavior can be observed in the reward curves in Figure 41. This leads to a sequence of training updates where experience batches contain only a small subset of states. In turn, the network shifts most of its representational capacity to focus on this limited subset, compromising its performance on the rest of the state space. Eventually, this causes the policy to collapse to the ALLD monomorphic state, from which it is largely unable to recover thereafter.

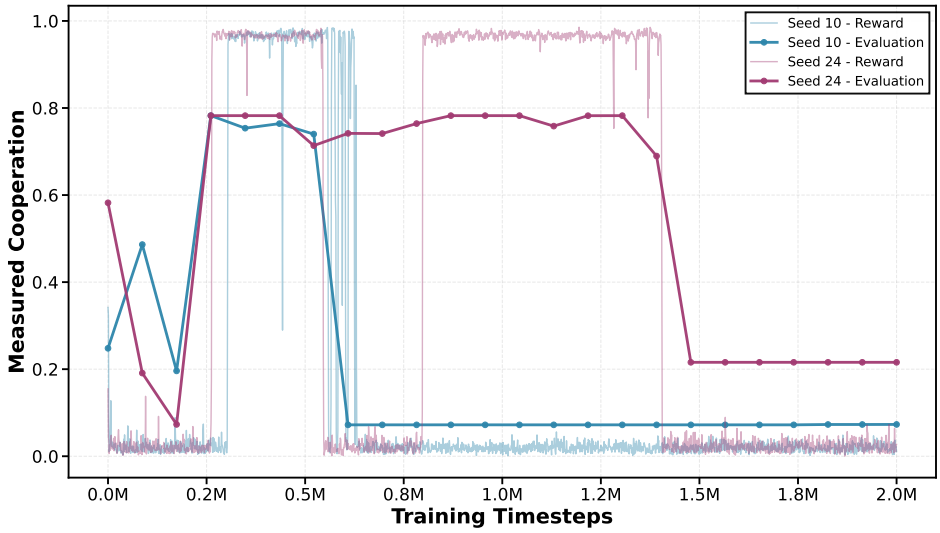


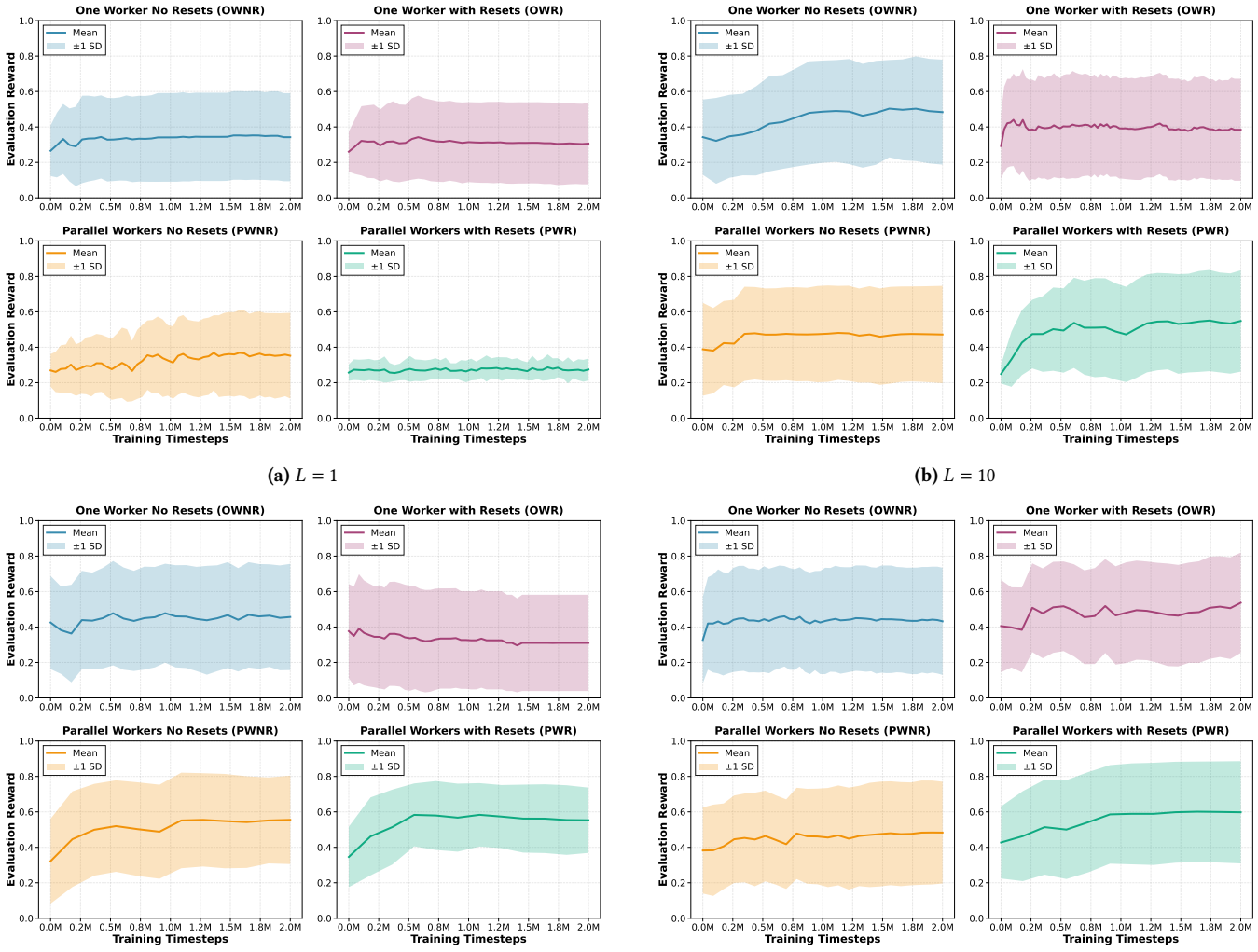
Figure 41: Learning curves showing reward and evaluation for two training runs with One Worker and No Resets (OWNR) using seeds: $\{10, 2\}$. The policies show catastrophic forgetting whereby updating with several batches of homogeneous experiences can lead the policy to collapse as seen in both curves here.

.

D.2 Evaluation Learning Curves for Different Numbers of Observations

Each training regime with different observation counts $L \in \{1, 10, 100, 1000\}$ was trained across 50 random seeds, with evaluations taken at regular intervals during training. The learning curves showing average evaluation reward and standard deviation are presented in Figure 42.

We observe that for $L > 1$, there is generally a noticeable increase in the rate of cooperation achieved. However, we also observe an interesting phenomenon in the broadening of the standard deviation across seeds. As more information is provided through additional observations, agents are able to form increasingly fine-grained beliefs over the underlying environment state. While this generally reduces the entropy of the belief distribution over states, it can paradoxically complicate decision-making by concentrating probability mass around states that are more dissimilar from each other, as evidenced by the increase in Rao's Q divergence (see Appendix B). This concentration can happen across decision boundaries and can potentially mislead agents. We hypothesize that which actions are learned in response to these refined observations become critical in determining whether the learned policy leads to success or failure. The increased variance across seeds suggests that with richer observations, the learning process becomes more sensitive to initialization and early experiences, as agents must learn to navigate finer distinctions between states that may require opposing actions.



(a) $L = 1$ **(b)** $L = 10$

(c) $L = 100$ **(d)** $L = 1000$

Figure 42: Learning curves for the different training regimes under different numbers of environment observations (L).

D.3 Frequency of cooperation per strategy state across training regimes and number of observations

The evaluated level of cooperation of each training regime after obtaining the optimal parameters is given by the average cooperation obtained in the last 1000 timesteps of a total of 2000 timesteps by the trained policy of 5 seeds (1,2,3,4,5) at all possible strategy states with a randomly selected initial social norm. We found this to be a more reliable metric for a policie's performance than the environmental reward as it measures outcomes under the policy for all possible initial states. As we can see in figure D.1, Evaluation Reward and Environment reward can diverge.

In Figure 43, we present these results for the different training regimes, across a different number of observations. We conclude that all policies have a substantial increase in performance by having more than one observation, leading to a greater strategy state region where the agent is able to guide the population towards cooperation. Interestingly, the effect of observations is not the same across learning regimes, suggesting an optimal finite value of observations per training method. Throughout all training regimes and number of observations, we observe the existence of two separate regions: one from where it is possible to guide the population towards a cooperative state, and one where the population always goes towards defection. We see that the latter region, composed primarily of ALLD and ALLC agents, is largely independent of the norm that is used, and as such it is present throughout all training regimes and number of observations. This suggests a limit in the obtainable average cooperation level when starting from all possible strategy states.

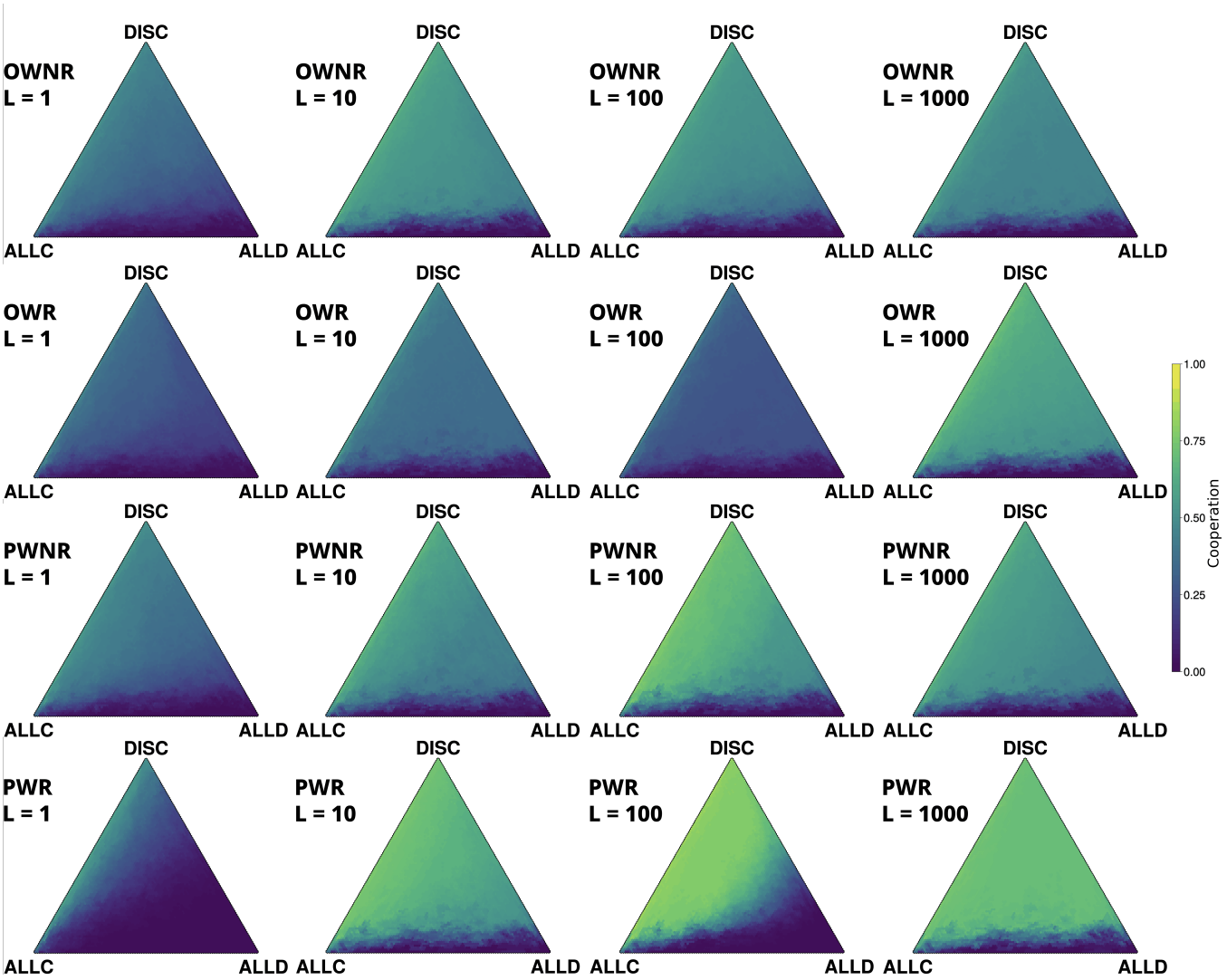


Figure 43: Average cooperation obtained after starting in each strategy state for all training regimes, for different numbers of observations ($L = 1$, $L = 10$, $L = 100$,$L = 1000$). We observe that more observations tends to result in a larger strategy state region from where it is possible to guide the population towards cooperation, with higher overall levels of cooperation. Despite this, not all learning regimes benefit equally from added observations. Environment parameters: $Z = 100$, $b = 10$, $c = 1$,$\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.

D.4 Average policy per state across training regimes and number of observations

The average policy of each training regime (after obtaining the optimal parameters) is given by the most commonly used social norm in each strategy state, obtained after a total of 2000 timesteps by the trained policy of 5 seeds (1,2,3,4,5) initiated at all possible strategy states with a randomly selected initial social norm. In Figure 44, we present these results for the different training regimes, across a different number of observations. We observe that the trained policies are highly sensitive to the training regime and the number of observations. Nevertheless, more observations lead to a generally more targeted usage of each norm. That is, as the number of observations increases, the strategy state region where a norm is used becomes less noisy and covers a more complex shape of the strategy state. These results align with the entropy metrics discussed in Appendix B, where more observations lead to a better capacity to distinguish the current strategy state.

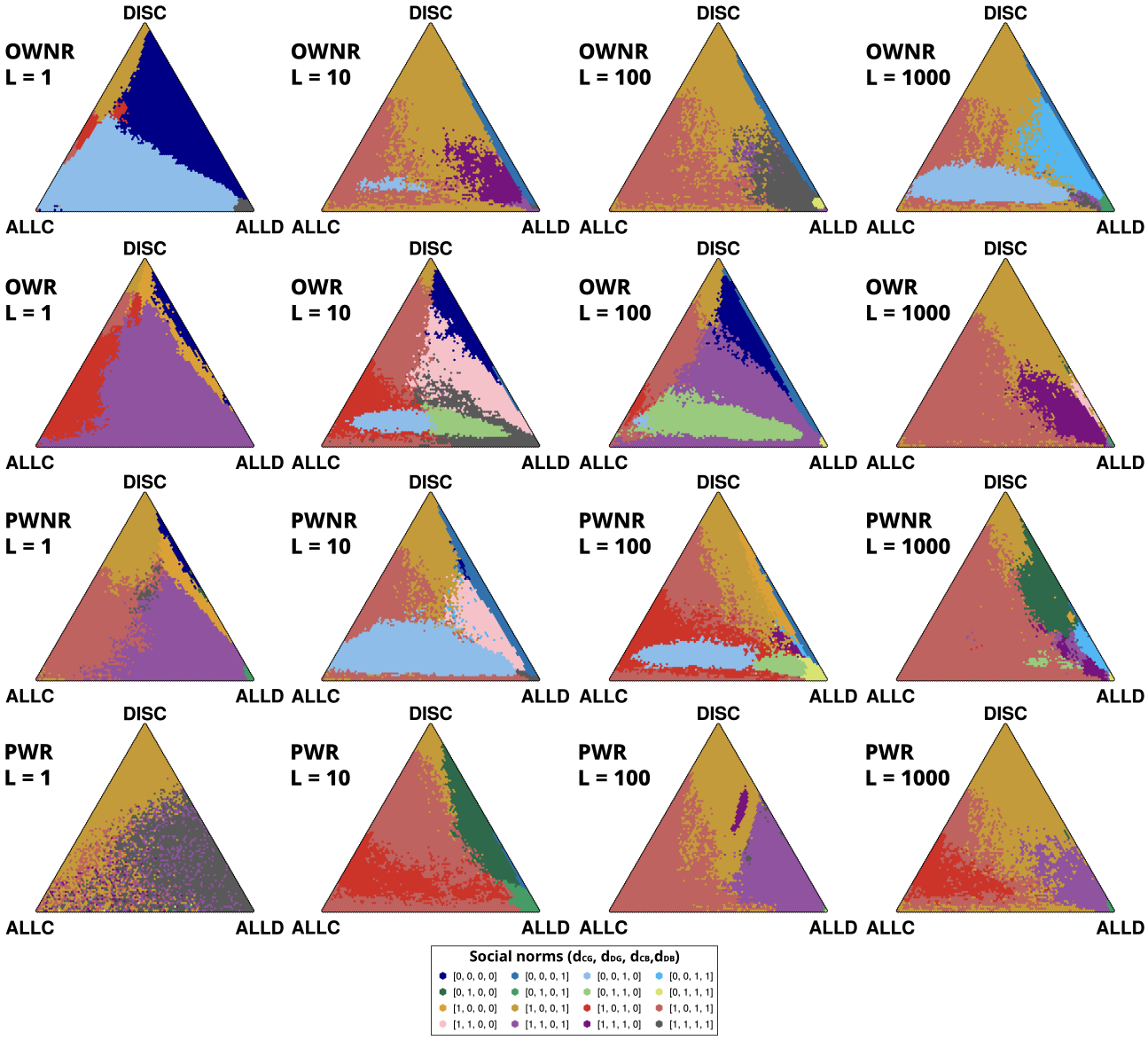


Figure 44: Average policy per state under all training regimes, for different numbers of observations ($L = 1$, $L = 10$, $L = 100$,$L = 1000$). Each color represents the social norm most commonly used in the trained policies at that state. We observe that the resulting policies are highly sensitive to the training regime and the number of observations. In general, under a lower number of observations, the same norms are used in broader regions of the state space, indicating less ability to distinguish between strategy states. Environment parameters: $Z = 100$, $b = 10$, $c = 1$, $\beta = 1.0$, $e_e = e_a = 0.01$, $\mathcal{T} = \infty$.