Peer Learning: Learning Complex Policies in Groups from Scratch via Action Recommendations

Cedric Derstroff[1,2], Mattia Cerrato[3], Jannis Brugger[1,2], Jan Peters[1,2,4,5], Stefan Kramer[3]
[1] Technische Universität Darmstadt
[2] Hessian Center for Artificial Intelligence (hessian.AI)
[3] Johannes Gutenberg-Universität Mainz
[4] German Research Center for AI (DFKI)
[5] Centre for Cognitive Science
{cedric.derstroff, jannis.brugger, jan.peters}@tu-darmstadt.de, mcerrato@uni-mainz.de, kramer@informatik.uni-mainz.de

Abstract

Peer learning is a novel high-level reinforcement learning framework for agents learning in groups. While standard reinforcement learning trains an individual agent in trial-and-error fashion, all on its own, peer learning addresses a related setting in which a group of agents, i.e., *peers*, learns to master a task simultaneously together from scratch. Peers are allowed to communicate only about their own states and actions recommended by others: "What would you do in my situation?". Our motivation is to study the learning behavior of these agents. We formalize the teacher selection process in the action advice setting as a multi-armed bandit problem and therefore highlight the need for exploration. Eventually, we analyze the learning behavior of the peers and observe their ability to rank the agents' performance within the study group and understand which agents give reliable advice. Further, we compare peer learning with single agent learning and a state-of-the-art action advice baseline. We show that peer learning is able to outperform single-agent learning and the baseline in several challenging discrete and continuous OpenAI Gym domains. Doing so, we also show that within such a framework complex policies from action recommendations beyond discrete action spaces can evolve.

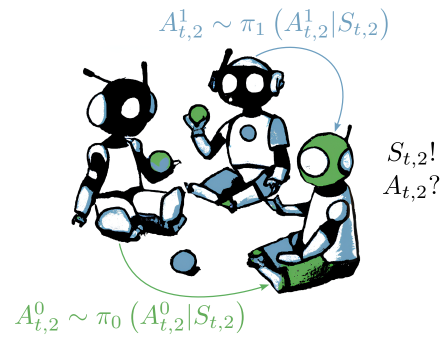

Figure 1: Peer learning: Agent 2 (right) asks for advice about which action to perform at the current step $S_{t,2}$ it finds itself in, and receives action suggestions $A^0_{t,2}$ and $A^1_{t,2}$ from agents 0 and 1, on the left and center, respectively. The agents operate in their own environment and on their own complex (motor) task, here: learning how to handle a ball.

1 Introduction

Learning by trial-and-error is one of the most influential theories of learning. Its roots go deeper than theories of *human* learning, in ethology and particularly in Morgan's observations on animal behavior (Thorpe et al. 1979). Edward Thorndike's "Law of Effect", for instance, posits that animals select responses (*actions*) to a given situation or stimulus (*state*) based on the satisfaction that followed in previous occurrences of the same situation and action (Thorndike 1898). The Law of Effect is the basis for the behaviorist view of biological learning as pioneered by Skinner (Skinner 1988), who introduced the term "reinforcement" to describe the conditioning effect of certain operants (responses, or *rewards*) on the behavior of lab mice. Computational investigations of behaviorism closely followed, from both Turing (Turing 1951) and Minsky, who explicitly referenced the work done by Skinner in studying animal behavior as an influence to early artificial intelligence (AI) (Minsky 1961).

Learning as trial-and-error under a reinforcement signal has since been the core idea of reinforcement learning (RL) as a subfield of AI and machine learning (ML), as discussed by Sutton and Barto (Sutton and Barto 2018). Another classic line of investigation in learning theory seeks to account for the *social* aspects of learning. This approach was pioneered by Rotter (Rotter 1954) and further developed by Bandura (Bandura and Walters 1977). Bandura's social learning theory augments trial-and-error, stimulus-response behaviorism with more cognitively informed accounts of learning. More specifically, Bandura posits that reinforcement may be *vicarious*, i.e., obtained through a *model* that is separate from the learning subject. Models may be live, verbal or symbolic: a live model demonstrates some behavior by acting it out, whereas a verbal model gives an instruction or suggestion. Lastly, symbolic models may be fictional characters presented through print or some other media. Beyond observation and blind imitation, learners in the social learning theory framework are afforded *motivation*: the capability to observe the reinforcement signal obtained by the model and reason about whether they should actually imitate it.

In social psychology, considerable attention has been given to the phenomena of social facilitation and social inhibition, defined as the phenomenon of improved or worsened task performance in the presence of other individuals (Geen 1991; Guerin 1983; Zajonc, Heingartner, and Herman 1969). Moreover, the effectiveness of collaborative learning as a complementary or alternative approach to teacher-centered instruction remains a subject of debate. In this domain, perhaps the most well-known investigation is due to Whitman and Fife (Whitman and Fife 1988), who described and developed "peer-to-peer teaching" strategies in higher education; recent results, both survey-based (Rybczynski and Schussler 2011) and in controlled studies (Jain and Kapoor 2015), have shown that the more competent agents rarely benefit from group collaboration.

Inspired by social learning theory and social psychology, we introduce *peer learning* (see Fig. 1), a learning scheme that enables knowledge exchange and social learning in groups of RL agents. Our setup investigates the reward trajectories of agents which *train together* but *perform alone*, similar to students engaging in collaborative or cooperative learning, e.g., a study group, who might still have to pass an individual exam. This is a common behavior in human learning. Humans rarely learn a task alone and from scratch. Although infants start by trial-and-error learning, they will observe their parents (Waismeyer and Meltzoff 2017) and as they grow older, at some point, they will learn from experts such as teachers or coaches. In contrast, for intelligent agents, the learning is often modelled through pure RL, hence, the agent is isolated in its learning setup (Sutton and Barto 2018). Consequently, similar to social learning theory, we introduce *peer agents* as capable of learning either from pure stimulus-response or from *vicarious reinforcement*. We model vicarious reinforcement by allowing agents to observe other peers' suggestions for the situation (state) they are in. Different from "blind" imitation learning approaches, e.g., (Ross, Gordon, and Bagnell 2011), agents are afforded motivation and may decide whether to trust their peers' advice in the future. Motivation is modeled via a *trust mechanism*, implying that at each learning step each agent needs to decide whether to follow the advice they have been given based on their previous experiences (and rewards) following the advice given by the peer/model. We formalize the trust mechanism as a non-stationary, multi-armed bandit problem. We find that groups of agents are able to isolate malicious, adversarial peers, but that honest, low-performing agents can still improve the performance of other learners.

Compared to previous agent-to-agent communication strategies (e.g., (Nunes and Oliveira 2002; Da Silva, Glatt, and Costa 2017)), we are interested in cooperative, *tabula rasa* learning without expert advice—we do not assume the presence of a teacher who, by definition, gives trustworthy advice—, and our peers start out as complete novices in all the considered tasks. Furthermore, we test our framework's performance in learning motor skills via the MuJoCo control suite (Todorov, Erez, and Tassa 2012), whereas agent-to-agent communication strategies have been so far limited to discrete action spaces (Cheng and Li 2021; Da Silva et al. 2020b; Ilhan, Gow, and Perez-Liebana 2021; Omidshafiei et al. 2019). On this regard, our results stand in contrast to studies in social psychology (Bond and Titus 1983) and collaborative learning (Rybczynski and Schussler 2011; Jain and Kapoor 2015): the presence of multiple agents is beneficial both in learning complex motor skills and navigation in simpler discrete environments.

Our contribution may be summarized as follows:

- We introduce *peer learning*, an *action advice framework* for reinforcement learning (RL) in groups of agents (*peers*).
- It works for *discrete* as well as *continuous action spaces*,
- with almost every *off-policy RL algorithm*,
- featuring a *trust mechanism* that is able to identify the performance of peers within the group well to (i) boost performance over single agent training and the baseline and (ii) prevent poisoning attacks from malicious, adversarial agents.

2 Related Work

Imitation learning and behavior cloning have a relatively long tradition in ML—especially in supervised learning but also RL—with the first approaches appearing in the 90s (Bain and Sammut 1995; Ng and Russell 2000). In these setups an agent learns from demonstration, which is provided by an expert—be it another software agent or a human. This avenue of research has generated a surge of recent interest, with many approaches studying the problem of inverse RL (Arora and Doshi 2021), i.e., learning a reward function from examples. Our approach differs in that we seek to model the social aspect of learning and allow for communication between concurrently learning agents, displaying motivation in the sense of social learning theory (see Section 1), which is closely related to what is referred to as trust in the advice exchange community.

In multi-agent RL (MARL) (Zhang, Yang, and Başar 2021), several agents act in the same instance of an environment and need to compete or cooperate to solve the task at hand. While some recent approaches in MARL have investigated inter-agent communication (Zhu, Dastani, and Wang 2022), our approach focuses instead on separate instances of the same single-agent environment. The objective of our investigation is to understand whether communication is intrinsically beneficial to the learning of complex, continuous-valued actions in single-agent learning. Therefore, our setup is more akin to social learning theory and collaborative learning (Jain and Kapoor 2015; Rybczynski and Schussler 2011): We allow students to give each other action recommendations during the learning process, whereas the testing (inference) is done on an individual basis.

More closely related to this work are proposals for *advice exchange* in RL. The first approaches in this area are due to the work of Nunes and Oliveira (Nunes and Oliveira 2002, 2003) and rely on tabular methods to learn strategies in grid-world environments. In the advice exchange framework, as originally developed, learners act in a single multi-agent environment where the optimal policy requires collaboration at testing time. Advice is defined as the action the adviser would perform if it found itself in the same state as the advisee—from now on called *action advice*. While some algorithmic proposal to learn which agents give trustworthy advice is already present in this early work (Nunes and Oliveira 2002), the authors do not consider scenarios in which adversarial, bad-faith agents may be part of the study group. Notably, the recommendation to perform further work in the space of motivation and trust is also present in more recent work in advice exchange (Bignold et al. 2021; Omidshafiei et al. 2019; Daniluk and Emami 2020; Da Silva et al. 2020b), but is actually undertaken only in a minority of methods (Ndousse et al. 2021; Cheng, Kolobov, and Agarwal 2020). Another limitation of existing approaches in advice exchange is that they might require fully-trained teacher agents—also called experts or oracles—, either during training (Da Silva et al. 2020a; Da Silva, Glatt, and Costa 2017; Cheng, Kolobov, and Agarwal 2020; Ilhan, Gow, and Perez-Liebana 2021) or even at test time (Ndousse et al. 2021). Finally, to the best of our knowledge, previous advice exchange proposals have focused on discrete action spaces and relatively simple navigation-based skills. In contrast, our approach is i) to design motivation and trust mechanisms to be employed by ii) non-expert agents which learn concurrently, starting from scratch iii) on continuous action spaces representing fine-grained motor skills. In the interest of enabling a comparison to work performed in advice exchange, we compare our methodology to the more recent methods which do not explicitly require fully-trained agents (Omidshafiei et al. 2019; Torrey and Taylor 2013).

3 Peer Learning

Motivation. Peer learning introduces a training paradigm rooted in pure reinforcement learning (RL) while also incorporating social learning principles. Our framework, represented in Fig. 1, is analogous to a study group where numerous peers undertake simultaneous training. Each agent engages with *its own* environment. Even though all environments mirror each other, they exist independently, each embodying the same Markov Decision Process (MDP). Agents may exhibit social behavior by sharing action recommendations, which allows the learning process to be analyzed through the lens of social learning theory (Bandura and Walters 1977). In peer learning, each agent operates independently: Peers do not operate cooperatively, and environments remain detached, contrasting with Multi-Agent RL (Omidshafiei et al. 2019) and prior advice exchange proposals (Ndousse et al. 2021). The rationale behind exploring this arrangement is to discern whether independent agents exhibit vicarious reinforcement potential—the ability to evaluate peers' actions and determine whether it would be beneficial to imitate them.

In practical terms, we suggest that this framework is suitable for shared, cross-institutional training of RL agents. Under a "peer learning" configuration, various autonomous organizations would train their RL agents to execute an identical task, such as exploration in adverse conditions or autonomous driving. In this context, details regarding agent design (action space, reward) and world representation (state space) could feasibly be negotiated in advance; nevertheless, the task should continue to be performed independently following successful training.

Definition. Peer learning can be characterized by the tuple $\mathcal{P} = (\mathcal{N}, \mathcal{M}_i)$, in which $\mathcal{N}$ signifies a set of agents $0, \ldots, n-1$ and $\mathcal{M}_i$ symbolizes a collection of distinct yet equivalent MDPs. Each MDP is represented as a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T}, \mu, \gamma)$. In the context of this MDP, at every step t, the agent surveys the state $S_t \in \mathcal{S}$, with S_0 drawn from the initial state distribution μ, and undertakes the action $A_t \in \mathcal{A}$ in line with its policy $\pi_i(A_t|S_t)$. Subsequently, the environment transitions from state S_t to $S_{t+1} \sim \mathcal{T}(S_t, A_t) = \Pr(S_{t+1}|S_t, A_t)$ and produces the immediate reward $R_{t+1} = \mathcal{R}(S_t, A_t, S_{t+1}) \in \mathbb{R}$, which is observed by the agent. Peer learning can then be defined as the concurrent training of $n = |\mathcal{N}|$ RL agents, all participating in identical versions of a singular MDP $\mathcal{M}$. Every agent $i \in \mathcal{N}$ aims to amplify its own discounted long-term reward $G_{t,i}$:

$$G_{t,i} = \mathbb{E}_{\pi_i}\left[\sum_{k=t+1}^{T} \gamma^{k-t-1} R_{k,i}\right], \tag{1}$$

where $\gamma \in [0, 1)$ denotes the discount factor. This return can be estimated by the value function $V_{\pi_i}(S_{t,i})$, computing the anticipated return from state $S_{t,i}$ while adhering to policy π_i. Furthermore, we can define the Q-function $Q_{\pi_i}(S_{t,i}, A_{t,i})$ as the expected return following the action $A_{t,i}$ in state $S_{t,i}$ and implementing the policy π_i thereafter.

While this formalization is not the sole way to define peer learning, it does offer a direct and intuitive interpretation of the collaborative learning task at hand. An alternative definition, which we only sketch in the following due to space constraints, is to characterize peer learning as a decentralized MDP, which is additionally state-factored, transition-independent, and reward-independent. We refer the reader to the seminal work by Becker et al. (Becker et al. 2003) for formal definitions of these properties.

Agent communication. The n agents in a peer learning setup are able to communicate in a constrained manner. Each agent $i \in \mathcal{N}$ solicits action advice $A^j_{t,i} \sim \pi_j(A^j_{t,i}|S_{t,i})$ from all other agents j ($j \in \mathcal{N}$) based on the state $S_{t,i}$ it currently occupies and the policies of all agents. Agents then decide from the compilation of all proposed actions: $A^0_{t,i}, \ldots, A^{n-1}_{t,i}$, including their own, in accordance with a distinct *peer* policy $\Pi_i(A^0_{t,i}, \ldots, A^{n-1}_{t,i})$. Therefore, in peer learning, each agent has the ability to produce actions—and action recommendations—through its own policy π_i and select the most beneficial one through the peer policy Π_i. A detailed depiction of peer learning with three agents is presented in Fig. 2.

While the primary goal of peer learning still aligns with the conventional RL aim of maximizing the discounted long-term reward of the MDP, we can also frame a similar goal for the sub-task of requesting and choosing action advice from peers. Here, the objective is for each agent to learn about its peer's proficiency level, that is, estimating the quality of their advice. This task is essentially a *multi-armed bandit problem* (Slivkins 2019) , where the goal is to maximize the reward and thus identifying the "best" bandit. Similarly, in

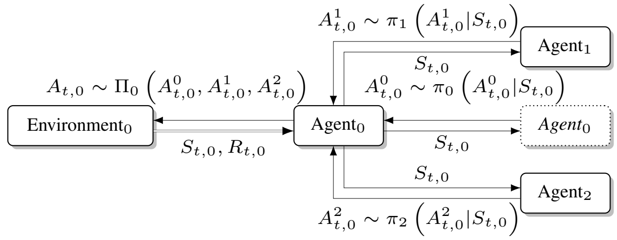

Figure 2: An intricate depiction of the advice exchange in peer learning with three agents. Agent 0 operates within its environment and requests advice in state $S_{t,0}$ from all peers in the group, including itself (outlined in dotted frame). In return, it receives suggested actions $A^0_{t,0}, \ldots, A^2_{t,0}$ and then selects an action from those suggestions according to its peer policy Π_0.

peer learning, each agent needs to identify the most proficient peer from whom to seek advice. Unlike the standard multi-armed bandit problem, in peer learning, we are dealing with a non-stationary multi-armed bandit problem. Therefore, more recent rewards should be given more weight than long-past ones (Sutton and Barto 2018).

Framing the problem of taking advice in this manner underscores the *need for exploration* in terms of whom to ask for advice or whose advice to heed, respectively. Surprisingly, this critical aspect is often overlooked in most previous work and has recently been mentioned as one critical avenue for further research in advice exchange techniques (Da Silva et al. 2020b). Consequently, we propose our approach as follows.

During the training process, agent i must decide on one action recommendation $A^j_{t,i}$ to be actually performed. This decision is made via weighted sampling with weights v^j_i using a Boltzmann (or Gibbs) distribution

$$\begin{aligned} A_{t,i} &\sim \Pi_i(A^0_{t,i}, \ldots, A^{n-1}_{t,i}) \\ &= \Pr(A_{t,i} = A^j_{t,i}|S_{t,i}) = \frac{e^{v^j_i/\tau_m}}{\sum_k e^{v^k_i/\tau_m}}, \qquad (2) \\ &\text{with } A^j_{t,i} \sim \pi_j\left(A^j_{t,i}|S_{t,i}\right), \\ &\text{and } \tau_m = \tau_0\ e^{-\lambda m}, \end{aligned}$$

where $i, j, k \in \mathcal{N}$, m represents the current epoch, τ is the temperature parameter, and λ is the temperature decay. For $\tau_0 = 1$ and $\lambda = 0$, the Boltzmann distribution[1] equates to the softmax function. Hence, we can also interpret it as a weighted softmax function. The weight v^j_i has a critical function here, as it represents the level of trust, or motivation, that agent i has in its peer j. As previously mentioned, our framework makes it possible to learn this weight adaptively at training time. In the following, we formalize three different techniques to do so.

Critic: Advice Evaluation via Q-function. The simplest evaluation technique we propose is to employ agent i's Q-function to evaluate the *advice* received, independently of which agent j has sent it. Specifically, for each agent i and every other agent j the advice weight is defined as

$$v^j_i = Q_i(S_{t,i}, A^j_{t,i}) \qquad (3)$$

We dub this approach **Critic** as the Q-function is commonly referred to as the critic in actor-critic RL algorithms. One limitation of this approach is that Q_i could fail in recognizing valuable advice. This is especially problematic at the beginning of training—agents could reject advice from an expert simply because their own Q-function is currently underfit. Furthermore, this technique only evaluates the advice, and not the advice-giver. Thus, it could be difficult for the peer group to isolate bad-faith actors who participate in the training with adversarial objectives. In the following, we propose two other motivation mechanisms that seek to solve these issues.

Trust Values: Local Evaluation of Peers. In this mechanism, we seek to compute v^j_i based on the quality of the advice that agent j gave to its peer i throughout the training process. At a basic level, this may be computed by simply keeping track of the immediate rewards incurred by i when following advice given by j. Starting from this straightforward concept, we extend the experience replay buffer (Mnih et al. 2015) so that peers may keep track of whose advice they previously followed and how beneficial it was. Specifically, experience replay is usually defined as a tuple $< S_t; A_t; R_{t+1}; S_{t+1} >$. In this work, we propose to extend it to the tuple $< S_{t,i}; A^j_{t,i}; R^j_{t+1,i}; S^j_{t+1,i}; j >$ where, with a slight abuse of notation, we use the variable $R^j_{t+1,i}$ to denote the immediate reward that agent i obtained by following the advice of agent j at step $t + 1$. We then initialize v^j_i randomly and update it throughout training by bootstrapping $R^j_{t+1,i}$ with agent i's Q-function, as follows:

$$v^j_i \leftarrow (1 - \alpha)v^j_i + \alpha\omega^j_i \qquad (4)$$

$$\omega^j_i = R^j_{t+1,i} + \gamma Q_i(S^j_{t+1,i}, \pi_i(S^j_{t+1,i})), \qquad (5)$$

where α is a "trust learning rate" separate from the base learning algorithm's learning rate. In our experiments, we keep this fixed at 0.99.

Agent Values: Global Leaderboard of Peers. To introduce this last motivation technique, we start from the observation that sharing information between peers about who is giving good advice may be beneficial to quickly learn to identify experts or bad-faith actors. To this end, we introduce the idea of maintaining a global experience buffer $<S_{t,i}; A^j_{t,i}; R^j_{t+1,i}; S^j_{t+1,i}; j>$ and computing a global weight v^j to be substituted in for v^j_i in Eq. (2) for all i:

$$v^j := v^j_0 = v^j_1 = \cdots = v^j_{n-1} \qquad (6)$$

$$v^j \leftarrow (1 - \alpha)v^j + \alpha\omega^j_i, \quad \forall i \in \mathcal{N} \qquad (7)$$

Learning the Advantage of Following Advice. While the techniques introduced so far all are able to effectively model the motivation of trust that each agent has in following the advice given by other peers, we find that it is beneficial for

[1]We replace the negative energy $-E$, that is commonly used—especially in physics—, by our positive weights v.

the agents to compare the benefit of participating in peer learning with the reward that they would obtain by learning independently. As a result, we also train agents which update their trust and agent values *with advantage*, that is, by computing v_i^j as the difference between the quality of the advice received and the action they themselves would have taken if isolated. In practice, this means changing Eq. (5) to

$$\omega_i^j = R_{t+1,i}^j + \gamma Q_i(S_{t+1,i}^j, \pi_i(S_{t+1,i}^j)) - Q_i(S_{t,i}, \pi_i(S_{t,i})), \tag{8}$$

where the difference to Eq. (5) is in the last subtracted term.

4 Experiments

Our experimentation seeks to understand the capabilities of peer learning in two different contexts: *learning complex policies*, i.e., in the continuous state and action spaces provided by the MuJoCo control suite; furthermore, we elaborate on the effect of *motivation*, or trust mechanisms, introduced in the previous section. Our findings can be summarized as follows:

- Vicarious reinforcement via peer learning is able to improve performance in the MuJoCo control suite when compared with single-agent learning. We discuss these results further in Section 4.2. To the best of our knowledge, our contribution marks the first time in which techniques based on the concept of exchanging advice have been proven successful in such environments.
- In Section 4.3, we show how the motivation mechanisms introduced in Section 3 may be employed to find and "isolate" bad-faith peers. We investigate this effect in a room-like environment where agents need to navigate to a certain position to obtain the reward. We find that our agents are able to ignore advice coming from non-trustworthy peers, even when the short-term rewards for following bad advice are not worse. This situation has been previously called "poisoning attack" in the literature (Cheng and Li 2021).

We consider the following learning algorithms and agents:

Peer Learning agent A learning agent in the newly introduced peer learning setting with advantage which gives and receives advice through action recommendations for specific environment states within its peer group.

Single agent A normal reinforcement learning single agent which trains alone in an MDP without advice exchange.

LeCTR agent (Omidshafiei et al. 2019) A state-of-the-art action advising agent which learns a task-level execution policy and a teaching policy for action advising its peer.

Early Advising agent An agent which only exchanges action advice in the early phase of training until the teaching budget exhausts. Like, e.g., used by (Omidshafiei et al. 2019) and (Torrey and Taylor 2013).

Adversarial agent An adversarial agent as defined by (Cheng and Li 2021) with the objective to minimize its peers' long-term reward without changing the immediate reward.

Non-training novice A randomly initialized agent which does not update its policy and/or Q-function.

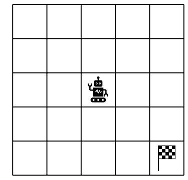

Figure 3: Our `Room` environment of size 5×5. The agent (robot) starts in the center of a square grid-world, where the goal position is a random position at the border (flag). It can choose among the four actions: right, left, up and down.

Peer Learning (Random advice) An agent similar to the peer learning agent but choosing randomly among all the suggested actions.

4.1 Implementation

Our Python code can be found on GitHub[2] and works with several off-policy RL algorithms that make use of a Q-function—especially but not limited to actor-critic methods. For most of our experiments, i.e., in the continuous action space settings, we used the state-of-the-art Soft-Actor Critic (SAC) algorithm (Haarnoja et al. 2018) as basis for our setup. For our experiments with discrete action spaces, we chose the widely used vanilla Deep Q-networks (DQN)(Mnih et al. 2013, 2015) algorithm. Further information on our code and reproducibility can be found in the appendix of the ArXiv version of this paper (Derstroff et al. 2023).

Being among the small group of approaches that work for continuous action spaces, we used several MuJoCo (Todorov, Erez, and Tassa 2012) OpenAI Gym environments (Brockman et al. 2016), i.e., `HalfCheetah-v4`, `Walker2d-v4`, `Ant-v4` and `Hopper-v4`. Apart from the MuJoCo environments, for comparison to our baseline and poisoning attacks (Cheng and Li 2021), we also use our room grid-world environment displayed in Fig. 3. In this environment, the agents starts in the middle of a squared grid and its task is to reach a specific goal position, which is chosen randomly among the border position. The agent observes the position of the goal next to its own position. The exploration in this task is more complex than in the usual grid-world tasks and satisfies all requirements for our baseline. In Fig. 3, we display the environment with a small grid size of 5×5 for visualization purposes. In our experiments, we use a bigger grid sizes of 27×27 and 21×21 creating the right amount of challenge for the agent.

4.2 Learning Complex Policies from Scratch

We test our approach in the MuJoCo control suite, comparing it against independent agents without communication and the commonly employed baseline Early Advising (Torrey and Taylor 2013). In this setting, we use a group size of 4 peers[3]. These results can be found in Table 1, which displays the average reward over the learning process—which

[2]https://github.com/kramerlab/PeerLearning

[3]Although we only use a fixed number of agents during our experiments, in theory peer learning can deal with a changing number

Task	Peer Learning	RL Single Agent	Early Advice	Peer Learning (Random Advice)	LeCTR
HalfCheetah-v4	$\mathbf{9014 \pm 715}$	$\mathbf{9270 \pm 247}$	7553 ± 1284	$\mathbf{9129 \pm 504}$	-
Walker2d-v4	$\mathbf{2160 \pm 346}$	1558 ± 489	248 ± 245	1970 ± 526	-
Ant-v4	2459 ± 667	$\mathbf{2694 \pm 660}$	-2222 ± 114	2109 ± 632	-
Hopper-v4	2123 ± 963	2059 ± 666	549 ± 381	$\mathbf{2483 \pm 87}$	-
Room-v21	$\mathbf{72 \pm 8}$	46 ± 16	12 ± 4	68 ± 10	15 ± 3
Room-v27	58 ± 11	32 ± 13	8 ± 5	$\mathbf{69 \pm 11}$	5 ± 2

Table 1: Comparison of learning speed and final performance expressed as average reward over time ($\pm$ the standard deviation). Values within 5 percent of the maximum are printed bold. For all experiments, we used 10 random seeds except for the Room-v27 where we used 15.

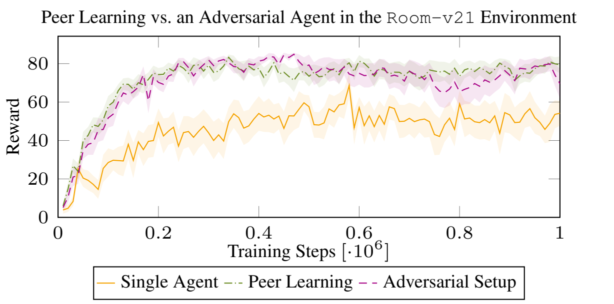

Figure 4: The reliability mechanisms in our approach protect the learning against the influence of a malicious agent. Peer learning cannot be harmed by an adversarial agent and still outperforms single agent learning.

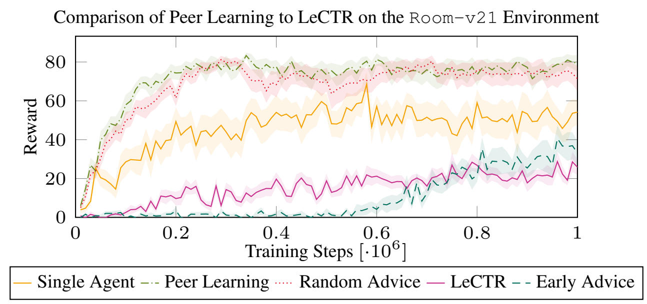

Figure 5: In a comparison, our approach (*Peer Learning*) outperforms the *LeCTR* approach (Omidshafiei et al. 2019) and a solo RL agent (*Single Agent*) on average over 10 random seeds on the Room environment of size 21×21. The shaded area marks the standard error of the mean.

can be seen as a more interpretable equivalent to the area under the learning curve—as a measure of learning speed. We observe that peer learning (with advantage) gives stronger results compared to the baselines in 3 out of 4 environments.

In `Ant-v4`, employing peer learning accelerates training: however, the benefits of agent communication seem to decrease over time. We conclude that vicarious reinforcement is a strong learning strategy in RL when learning complex motor skills, in contrast to early results on the topic (Bond and Titus 1983).

4.3 The Role of Trust and Motivation

As discussed in Section 3, we proposed different mechanisms to deal with the motivation/trust problem—i.e., deciding whether to follow advice and which advice is most promising. To verify the positive effect of each of our proposed mechanisms and to find a best combination, we show the results of our combination study in Table 2. We have averaged results over multiple MuJoCo environments and normalized them between 0 and 100. We also use the average normalized cross-environment reward over time as measure for the learning speed. We observe any of our proposed methods boosts performance over single agent learning. When using a combination of at least two of our proposals, the performance increases again by a small margin. Nonetheless, our study did not yield a single best combination that significantly outperforms all the other combinations in all environments. We can, however, notice that a combination of all three ranks in the leading group, especially when using advantage. Hence, we used advantage and an equally weighted combination[4] of all three approaches, i.e., critic, trust and agent values, for our further experiments.

When employing this setup, we find that peers are able to identify bad actors and ignore their advice: We train a group of peers on a grid-like environment (see Fig. 3) where agents only obtain rewards when they position themselves in a certain randomly sampled location at the borders of the rooms, a "goal position". In this setting, we include one adversarial agent which always gives out the suggestion for another agent to distance itself from the goal. We note that the short-term reward for following this advice is aligned with the short-term reward for following the theoretically optimal advice, as rewards are only obtained when reaching the goal position exactly; this follows the design of a "poisoning attack" for advice exchange as proposed by Cheng et al. (Cheng and Li 2021). Results for this experiment can be found in Fig. 4. Here, it is apparent that employing our suggestion selection mechanisms increases performance substantially. More generally, this simpler room environment enables us for a broader comparison with other methods in advice exchange which only support discrete state, and more importantly, action spaces. We limit our comparison to methods which do not explicitly require pre-trained experts: LeCTR (Omidshafiei et al. 2019) and Early Advising (Torrey and Taylor 2013). It is discernible that our methodology

of peers when the maximum number of peers is known beforehand.

[4]For equal weighting, we normalize each term to $[-1, +1]$.

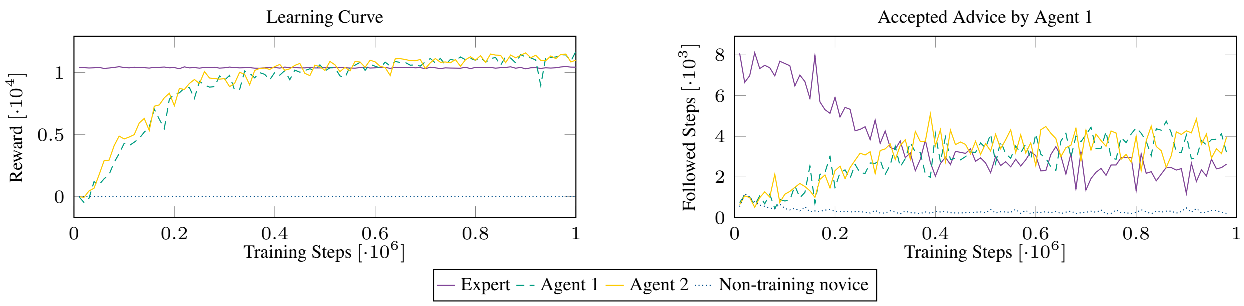

Figure 6: In a study group with one domain expert and a non-training novice, the two peer learning agents with advantage learn very well which of its peers gives good advice and whom to avoid. They also learn to switch to another peer if the best performing agent gets overtaken.

Adv	ACT	AC	AT	CT	C	A	T	RL	Random
No	66.2	64.9	**68.2**	65.7	65.4	66.4	66.4	59.7	65.0
Yes	**67.1**	**67.1**	65.8	64.7		64.3	62.6		

Table 2: Averaged over the 4 MuJoCo environments, it is evident that peer learning of any kind boosts training significantly compared to the single agent (RL), slightly against the peer learning with random advice agent (Random) and that several combinations of the agent values (A), critic (C) and trust (T) lead to great performance. The maxima are printed bold. For this ablation study, we used 5 random seeds.

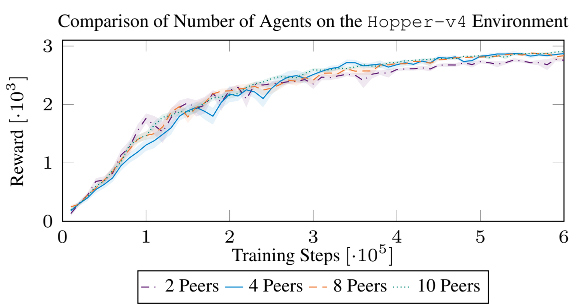

Figure 7: In peer learning, the performance slightly increases with the number of agents in the setting and does not peak or saturate for 2 agents. Averaged over 10 random seeds.

is able to outperform these two methodologies in the environment we considered, even over multiple room dimensions (Fig. 5, Table 1). We also see that training the suggestion selection mechanisms with advantage (see Eq. (8)) allows peers to identify experts: Figure 6 shows the number of times the learning agents accepted advice from its peers side-to-side with the learning curve. The group of learners here was formed by 2 training novices, 1 non-training "expert" and 1 non-training novice. We obtain non-training agents by straightforwardly setting their learning rate to 0: a non-training novice will give arbitrarily bad advice throughout training. Figure 6 shows that the 2 training novices, i.e., peers, are able to identify that one agent is giving out particularly bad advice. It also emphasizes that they are quick to recognize when the "expert" is being overtaken.

4.4 Number of Agents

The literature suggests that a setup with 2 or 3 agents or experts, respectively, yields the best results, concluding that the overhead of learning which advice to follow hinders learning when the group becomes too big (Cheng, Kolobov, and Agarwal 2020; Kretchmar 2003). In Fig. 7, we show that our results do not align with that hypothesis and that in peer learning the performance increases with the number of agents. Although the increase is not significant and further experiments on the limit have to be pursued, it is clearly visible that only 2 simultaneously learning agents do not outperform bigger groups of size 4 to 10.

5 Conclusion and Future Work

We have introduced a novel high-level framework called peer learning that enables social learning in groups and is compatible with many off-policy RL algorithms. In our experiments, we have shown that the introduced peer learning framework was able to boost RL and outperform a single agent (Section 4.2). Incorporating trust mechanisms that take into account the quality of other agents allows for a stronger focus on the good peers in the group. Our setup is therefore able to avoid poisoning attacks from adversarial agents. Furthermore, peers whose skills are comparatively more developed than the rest of their group may also rely on motivation and trust to reject untrustworthy advice (Section 4.3). To the best of our knowledge, we are the first to identify this trust problem as a non-stationary multi-armed bandit problem, and to successfully employ such a mechanism in learning complex continuous policies. Our current proposal, while valuable, does not define a singular method for implementing trust and motivation. As a result, it falls short of achieving a comprehensive understanding of social learning theory in RL. The evidence we gathered suggests that, while employing trust can be beneficial (Table 2) and is critical when not every peer is a trusted partner (Fig. 4), there is no single superior choice among our three proposals. This warrants future investigation, especially in setups with more than 10 concurrently learning agents (Section 4.4). While our conceptualization of peer learning is embarrassingly parallel, our current implementation is not. As a consequence, this work does not discuss the properties of peer learning in setups with, e.g., tens of agents and leaves it to future work.

Ethical Statement

No known ethical concern is implied by the presented work.

Acknowledgements

This research project was partly funded by the Hessian Ministry of Science and the Arts (HMWK) within the projects “The Third Wave of Artificial Intelligence - 3AI” and hessian.AI. This work was partly funded by the RMU by means of the DeCoDeML project. Parts of this research were conducted using the supercomputer MOGON 2 and/or advisory services offered by Johannes Gutenberg-University Mainz (hpc.uni-mainz.de), which is a member of the AHRP (Alliance for High Performance Computing in Rhineland-Palatinate, www.ahrp.info) and the Gauss Alliance e.V. The authors gratefully acknowledge the computing time granted on the supercomputer MOGON 2 at Johannes Gutenberg-University Mainz (hpc.uni-mainz.de). We further acknowledge the use of DALL·E 2 (Open AI, https://openai.com/dall-e-2) to create parts of Fig. 1.

References

Arora, S.; and Doshi, P. 2021. A survey of inverse reinforcement learning: Challenges, methods and progress. *Artificial Intelligence*, 297: 103500.

Bain, M.; and Sammut, C. 1995. A Framework for Behavioural Cloning. In *Machine Intelligence 15*, 103–129.

Bandura, A.; and Walters, R. H. 1977. *Social learning theory*, volume 1. Englewood cliffs Prentice Hall.

Becker, R.; Zilberstein, S.; Lesser, V.; and Goldman, C. V. 2003. Transition-independent decentralized Markov decision processes. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, 41–48.

Bignold, A.; Cruz, F.; Taylor, M.; Brys, T.; Dazeley, R.; Vamplew, P.; and Foale, C. 2021. A conceptual framework for externally-influenced agents: an assisted reinforcement learning review. *Journal of Ambient Intelligence and Humanized Computing*.

Bond, C. F.; and Titus, L. J. 1983. Social facilitation: A meta-analysis of 241 studies. *Psychological bulletin*, 94(2): 265.

Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; and Zaremba, W. 2016. OpenAI Gym. arXiv:1606.01540.

Cheng, C.-A.; Kolobov, A.; and Agarwal, A. 2020. Policy improvement via imitation of multiple oracles. *Advances in Neural Information Processing Systems*, 33: 5587–5598.

Cheng, Z.; and Li, Z. 2021. Poisoning Attack for Inter-agent Transfer Learning. In *Security and Privacy in Communication Networks: 17th EAI International Conference, SecureComm 2021, Virtual Event, September 6–9, 2021, Proceedings, Part II 17*, 394–404. Springer.

Da Silva, F. L.; Glatt, R.; and Costa, A. H. R. 2017. Simultaneously learning and advising in multiagent reinforcement learning. In *Proceedings of the 16th conference on autonomous agents and multiagent systems*, 1100–1108.

Da Silva, F. L.; Hernandez-Leal, P.; Kartal, B.; and Taylor, M. E. 2020a. Uncertainty-Aware Action Advising for Deep Reinforcement Learning Agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 5792–5799.

Da Silva, F. L.; Warnell, G.; Costa, A. H. R.; and Stone, P. 2020b. Agents teaching agents: a survey on inter-agent transfer learning. *Autonomous Agents and Multi-Agent Systems*, 34(1): 1–17.

Daniluk, S.; and Emami, R. 2020. An Advice Mechanism for Heterogeneous robot Teams. *International Journal of Robotics and Automation*, 35(1).

Derstroff, C.; Cerrato, M.; Brugger, J.; Peters, J.; and Kramer, S. 2023. Peer Learning: Learning Complex Policies in Groups from Scratch via Action Recommendations. arXiv:2312.09950.

Geen, R. G. 1991. Social motivation. *Annual review of psychology*, 42(1): 377–399.

Guerin, B. 1983. Social facilitation and social monitoring: A test of three models. *British Journal of Social Psychology*, 22(3): 203–214.

Haarnoja, T.; Zhou, A.; Abbeel, P.; and Levine, S. 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, 1861–1870. PMLR.

Ilhan, E.; Gow, J.; and Perez-Liebana, D. 2021. Action advising with advice imitation in deep reinforcement learning. arXiv:2104.08441.

Jain, T.; and Kapoor, M. 2015. The Impact of Study Groups and Roommates on Academic Performance. *The Review of Economics and Statistics*, 97(1): 44–54.

Kretchmar, R. M. 2003. Reinforcement learning algorithms for homogenous multi-agent systems. In *Workshop on Agent and Swarm Programming*, volume 94.

Minsky, M. 1961. Steps toward artificial intelligence. *Proceedings of the IRE*, 49(1): 8–30.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; and Riedmiller, M. 2013. Playing atari with deep reinforcement learning. arXiv:1312.5602.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G.; et al. 2015. Human-level control through deep reinforcement learning. *Nature*, 518: 529–533.

Ndousse, K. K.; Eck, D.; Levine, S.; and Jaques, N. 2021. Emergent social learning via multi-agent reinforcement learning. In *International Conference on Machine Learning*, 7991–8004. PMLR.

Ng, A. Y.; and Russell, S. 2000. Algorithms for Inverse Reinforcement Learning. In *Proceedings of the 17th International Conference on Machine Learning*, volume 1, 663–670.

Nunes, L.; and Oliveira, E. 2002. On learning by exchanging advice. arXiv:cs/0203010.

Nunes, L.; and Oliveira, E. 2003. Exchanging advice and learning to trust. In *Cooperative Information Agents VII: 7th International Workshop, CIA 2003, Helsinki, Finland, August 27-29, 2003. Proceedings 7*, 250–265. Springer.

Omidshafiei, S.; Kim, D.-K.; Liu, M.; Tesauro, G.; Riemer, M.; Amato, C.; Campbell, M.; and How, J. P. 2019. Learning to teach in cooperative multiagent reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 6128–6136.

Ross, S.; Gordon, G.; and Bagnell, D. 2011. A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning. In Gordon, G.; Dunson, D.; and Dudík, M., eds., *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, 627–635. Fort Lauderdale, FL, USA: PMLR.

Rotter, J. B. 1954. *Social learning and clinical psychology.* Prentice-Hall, Inc.

Rybczynski, S. M.; and Schussler, E. E. 2011. Student use of out-of-class study groups in an introductory undergraduate biology course. *CBE—Life Sciences Education*, 10(1): 74–82.

Skinner, B. F. 1988. *The selection of behavior: The operant behaviorism of BF Skinner: Comments and consequences*. CUP Archive.

Slivkins, A. 2019. Introduction to Multi-Armed Bandits. *Foundations and Trends® in Machine Learning*, 12(1–2): 1–286.

Sutton, R. S.; and Barto, A. G. 2018. *Reinforcement learning: An introduction*. MIT press.

Thorndike, E. L. 1898. Animal intelligence: An experimental study of the associative processes in animals. *The Psychological Review: Monograph Supplements*, 2(4): i.

Thorpe, W. H.; et al. 1979. *Origins and rise of ethology*. Heinemann Educational Books.

Todorov, E.; Erez, T.; and Tassa, Y. 2012. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 5026–5033. IEEE.

Torrey, L.; and Taylor, M. 2013. Teaching on a budget: Agents advising agents in reinforcement learning. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, 1053–1060.

Turing, A. M. 1951. Intelligent machinery, a heretical theory. *The Turing digital archive (http://www.turingarchive.org), contents of AMT/B/4*.

Waismeyer, A.; and Meltzoff, A. 2017. Learning to make things happen: Infants' observational learning of social and physical causal events. *Journal of Experimental Child Psychology*, 162: 58–71.

Whitman, N. A.; and Fife, J. D. 1988. *Peer Teaching: To Teach Is To Learn Twice. ASHE-ERIC Higher Education Report No. 4, 1988.* ERIC.

Zajonc, R. B.; Heingartner, A.; and Herman, E. M. 1969. Social enhancement and impairment of performance in the cockroach. *Journal of Personality and Social Psychology*, 13(2): 83.

Zhang, K.; Yang, Z.; and Başar, T. 2021. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handbook of reinforcement learning and control*, 321–384.

Zhu, C.; Dastani, M.; and Wang, S. 2022. A survey of multi-agent reinforcement learning with communication. arXiv:2203.08975.