Evolving Intrinsic Motivations for Altruistic Behavior

Jane X. Wang, Edward Hughes, Chrisantha Fernando, Wojciech M. Czarnecki,
Edgar A. Duéñez-Guzmán, & Joel Z. Leibo
DeepMind
London
{wangjane,edwardhughes,chrisantha,lejlot,duenez,jzl}@google.com

ABSTRACT

Multi-agent cooperation is an important feature of the natural world. Many tasks involve individual incentives that are misaligned with the common good, yet a wide range of organisms from bacteria to insects and humans are able to overcome their differences and collaborate. Therefore, the emergence of cooperative behavior amongst self-interested individuals is an important question for the fields of multi-agent reinforcement learning (MARL) and evolutionary theory. Here, we study a particular class of multi-agent problems called intertemporal social dilemmas (ISDs), where the conflict between the individual and the group is particularly sharp. By combining MARL with appropriately structured natural selection, we demonstrate that individual inductive biases for cooperation can be learned in a model-free way. To achieve this, we introduce an innovative modular architecture for deep reinforcement learning agents which supports multi-level selection. We present results in two challenging environments, and interpret these in the context of cultural and ecological evolution.

KEYWORDS

multi-agent; evolution; altruism; social dilemmas

ACM Reference Format:
Jane X. Wang, Edward Hughes, Chrisantha Fernando, Wojciech M. Czarnecki, Edgar A. Duéñez-Guzmán, & Joel Z. Leibo. 2019. Evolving Intrinsic Motivations for Altruistic Behavior. In *Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), Montreal, Canada, May 13–17, 2019,* IFAAMAS, 10 pages.

1 INTRODUCTION

Nature shows a substantial amount of cooperation at all scales, from microscopic interactions of genomes and bacteria to species-wide societies of insects and humans [39]. This is in spite of natural selection pushing for short-term individual selfish interests [9]. In its purest form, altruism can be favored by selection when cooperating individuals preferentially interact with other cooperators, thus realising the rewards of cooperation without being exploited by defectors [10, 14, 22, 23, 51]. However, many other possibilities exist, including kin selection, reciprocity and group selection [43, 44, 54, 56, 57, 60].

Lately the emergence of cooperation among self-interested agents has become an important topic in multi-agent deep reinforcement learning (MARL). [35] and [28] formalize the problem domain as an *intertemporal social dilemma* (ISD), which generalizes matrix game social dilemmas to Markov settings. Social dilemmas are characterized by a trade-off between collective welfare and individual utility. As predicted by evolutionary theory, self-interested reinforcement-learning agents are typically unable to achieve the collectively optimal outcome, converging instead to defecting strategies [35, 48]. The goal is to find multi-agent training regimes in which individuals resolve social dilemmas, i.e., cooperation emerges. Previous work has found several solutions, belonging to three broad categories: 1) opponent modelling [16, 34], 2) long-term planning using perfect knowledge of the game's rules [36, 49] and 3) a specific intrinsic motivation function drawn from behavioral economics [28]. These hand-crafted approaches run at odds with more recent end-to-end model-free learning algorithms, which have been shown to have a greater ability to generalize (e.g. [11]). We propose that evolution can be applied to remove the hand-crafting of intrinsic motivation, similar to other applications of evolution in deep learning.

Evolution has been used to optimize single-agent hyperparameters [30], implement black-box optimization [59], and to evolve neuroarchitectures [41, 55], regularization [4], loss functions [27, 29], behavioral diversity [7], and entire reward functions [52, 53]. These principles tend to be driven by single-agent search and optimization or competitive multi-agent tasks. Therefore there is no guarantee of success when applying them in the ISD setting. More closely related to our domain are evolutionary simulations of predator-prey dynamics [61], which used enforced subpopulations to evolve populations of neurons which are sampled to form the hidden layer of a neural network.[1]

To address the specific challenges of ISDs, the system we propose distinguishes between optimization processes that unfold over two distinct time-scales: (1) the fast time-scale of learning and (2) the slow time-scale of evolution [similar to 26]. In the former, individual agents repeatedly participate in an intertemporal social dilemma using a fixed intrinsic motivation. In the latter, that motivation is itself subject to natural selection in a population. We model this intrinsic motivation as an additional additive term in the reward of each agent [6]. We implement the intrinsic reward function as a two-layer fully-connected feed-forward neural network, whose weights define the genotype for evolution. We propose that evolution can help mitigate this intertemporal dilemma by bridging between these two timescales via an intrinsic reward function.

Evolutionary theory predicts that evolving individual intrinsic reward weights across a population who interact uniformly at random does not lead to altruistic behavior [1]. Thus, to achieve our goal, we must structure the evolutionary dynamics [43]. We first implement a "Greenbeard" strategy [10, 31] in which agents choose

Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), N. Agmon, M. E. Taylor, E. Elkind, M. Veloso (eds.), May 13–17, 2019, Montreal, Canada.

[1] See also [50] and [47] for reviews of other evolutionary approaches to cooperative multi-agent problems.

interaction partners based on an honest, real-time signal of cooperativeness. We term this process *assortative matchmaking*. Although there is ecological evidence of assortative matchmaking [33], it cannot explain cooperation in all taxa [17, 18, 25]. Moreover it isn't a general method for multi-agent reinforcement learning, since honest signals of cooperativeness are not normally observable in the ISD models typically studied in deep reinforcement learning.

To address the limitations of the assortative matchmaking approach, we introduce an alternative modular training scheme loosely inspired by ideas from the theory of multi-level (group) selection [25, 60], which we term *shared reward network* evolution. Here, agents are composed of two neural network modules: a policy network and a reward network. On the fast timescale of reinforcement learning, the policy network is trained using the modified rewards specified by the reward network. On the slow timescale of evolution, the policy network and reward network modules evolve separately from one another. In each episode every agent has a distinct policy network but the same reward network. As before, the fitness for the policy network is the individual's reward. In contrast, the fitness for the reward network is the collective return for the entire group of co-players. In terms of multi-level selection theory, the policy networks are the lower level units of evolution and the reward networks are the higher level units. Evolving the two modules separately in this manner prevents evolved reward networks from overfitting to specific policies. This evolutionary paradigm not only resolves difficult ISDs without handcrafting but also points to a potential mechanism for the evolutionary origin of social inductive biases.

The paper is structured as follows. In Section 2, we define our problem domain, and describe in detail our agent architecture and training methods. In Section 3, we present results from our experiments and further analyses of agent policies. Finally in Section 4, we discuss interpretations of our model as well as make suggestions for future work.

2 METHODS

We varied and explored different combinations of parameters, namely: (1) environments {Harvest, Cleanup}, (2) reward network features {prospective, retrospective}, (3) matchmaking {random, assortative}, and (4) reward network evolution {individual, shared, none}. We describe these in the following sections.

2.1 Intertemporal social dilemmas

In this paper, we consider Markov games [37] within a MARL setting. Specifically we study intertemporal social dilemmas [28, 35], defined as games in which individually selfish actions produce individual benefit on short timescales but have negative impacts on the group over a longer time horizon. This conflict between the two timescales characterizes the intertemporal nature of these games. The tension between individual and group-level rationality identifies them as social dilemmas (e.g. the famous Prisoner's Dilemma).

We consider two dilemmas, each implemented as a partially observable Markov game on a 2D grid (see Figure 1), with $N = 5$ players playing at a time. In the *Cleanup* game, agents tried to collect apples (reward +1) that spawned in a field at a rate inversely related to the cleanliness of a geographically separate aquifer. Over time, this aquifer filled up with waste, lowering the respawn rate of apples linearly, until a critical point past which no apples could spawn. Episodes were initialized with no apples present and zero spawning, thus necessitating cleaning. The dilemma occurred because in order for apples to spawn, agents must leave the apple field and clean, which conferred no reward. However if all agents declined to clean (defect), then no rewards would be received by any. In the *Harvest* game, again agents collected rewarding apples. The apple spawn rate at a particular point on the map depended on the number of nearby apples, falling to zero once there were no apples in a certain radius. There is a dilemma between the short-term individual temptation to harvest all the apples quickly and the consequential rapid depletion of apples, leading to a lower total yield for the group in the long-term.

All episodes last 1000 steps, and the total size of the playable area is 25×18 for Cleanup and 38×16 for Harvest. Games are partially observable in that agents can only observe via a 15×15 RGB window, centered on their current location. The action space consists of moving left, right, up, and down, rotating left and right, and the ability to tag each other. This action has a reward cost of 1 to use, and causes the player tagged to lose 50 reward points, thus allowing for the possibility of punishing free-riders [21, 45]. The Cleanup game has an additional action for cleaning waste.

2.2 Modeling social preferences as intrinsic motivations

In our model, there are three components to the reward that enter into agents' loss functions (1) total reward, which is used for the policy loss, (2) extrinsic reward, which is used for the extrinsic value function loss and (3) intrinsic reward, which is used for the intrinsic value function loss.

The *total reward* for player i is the sum of the extrinsic reward and an intrinsic reward as follows:

$$r_i(s_i, a_i) = r_i^E(s_i, a_i) + u_i(\mathbf{f}_i)\,. \tag{1}$$

The *extrinsic reward* $r_i^E(s, a)$ is the environment reward obtained by player i when it takes action a_i from state s_i, sometimes also written with a time index t. The *intrinsic reward* $u(\mathbf{f})$ is an aggregate social preference across features $\mathbf{f}$ and is calculated according to the formula,

$$u_i(\mathbf{f}_i|\boldsymbol{\theta}) = \mathbf{v}^{\mathrm{T}}\sigma\left(\mathbf{W}^{\mathrm{T}}\mathbf{f}_i + \mathbf{b}\right)\,, \tag{2}$$

where σ is the ReLU activation function, and $\boldsymbol{\theta} = \{\mathbf{W}, \mathbf{v}, \mathbf{b}\}$ are the parameters of a 2-layer neural network with 2 hidden nodes. These parameters are evolved based on fitness (see Section 2.3). The elements of $\mathbf{v} = (v_1, v_2)$ approximately correspond to a linear combination of the coefficients related to advantageous and disadvantageous inequity aversion mentioned in [28], which were found via grid search in this previous work, but are here evolved.

The feature vector $\mathbf{f}_i$ is a player-specific vector quantity that agents can transform into intrinsic reward via their reward network. It's composed of features f_{ij} derived from all players [2], so that each

[2]Note that we use both i and j to index over the players, but i makes reference to the player *receiving* the intrinsic reward, while j indexes the players *sending* the features over which the intrinsic reward of player i is defined.

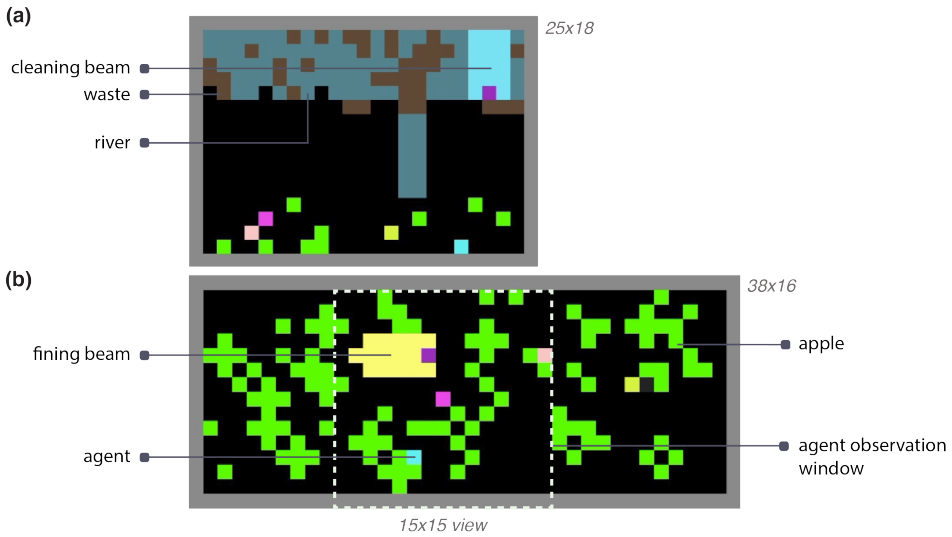

Figure 1: Screenshots from (a) the Cleanup game, (b) the Harvest game. The size of the agent-centered observation window is shown in (b). The same size observation was used in all experiments.

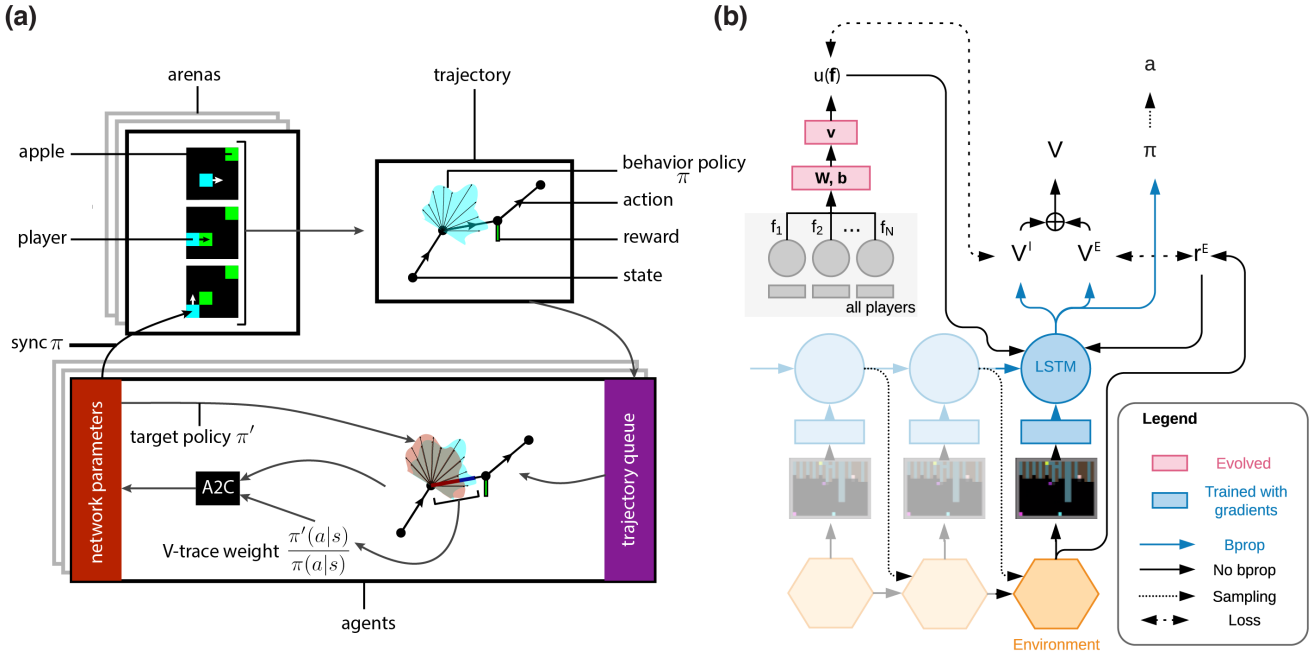

Figure 2: (a) Agent A_j adjusts policy $\pi_j(s, a|\phi)$ using off-policy importance weighted actor-critic (V-Trace) [11] by sampling from a queue with (possibly stale) trajectories recorded from 500 actors acting in parallel arenas. (b) The architecture (shown only for 1 agent) includes a visual encoder (1-layer convolutional neural net with 6 3x3 filters, stride 1, followed by two fully-connected layers with 32 units each), intrinsic and extrinsic value heads (V^I and V^E), a policy head π, and a long-short term memory (LSTM, with 128 hidden units), which takes last intrinsic and extrinsic rewards ($u(\mathbf{f})$ and r^E) and last action as input. The reward network weights are evolved based on total episode return.

player has access to the same set of features, with the exception that its own feature is demarcated specially (by always occupying the first element of the vector). The features themselves are a function of recently received or expected future (extrinsic) reward for each agent. In Markov games the rewards received by different players may not be aligned in time. Thus, any model of social preferences should not be overly influenced by the precise temporal alignment of different players' rewards. Intuitively, they ought to depend on comparing temporally averaged reward estimates between players,

Algorithm 1 Training Pseudocode - Shared Reward Network

Require: $\mathcal{P}$: pop. of policy networks and hyper-parameters $\{\{\phi_1, h_1\}, \ldots, \{\phi_N, h_N\}\}$
Require: $\mathcal{R}$: pop. of reward networks $\{\theta_1, \ldots, \theta_N\}$, where $\theta = \{\mathbf{W}, \mathbf{v}, \mathbf{b}\}$
Require: $\mathcal{S}$: procedure to sample from $\mathcal{P}$ and $\mathcal{R}$ and return 5 players
Require: $\mathcal{U}$: procedure to update/evolve weights given a population of fitness-scored individuals
Require: $\mathcal{F}$: procedure to calculate and assign fitness given the sampled players

while not done **do**
  $p_{1:5} = p_1, \ldots, p_5 \leftarrow \mathcal{S}(\mathcal{P}, \mathcal{R})$ ▷ Sample 5 players
  env $\sim p(\mathcal{T})$ ▷ Sample from distribution p over environments $\mathcal{T}$
  $e^0_{1:5} \leftarrow 0$ ▷ Initialize temporally decayed reward
  for $t := 1$ **to** T **do** ▷ Run for $T = 1000$ steps
    $\tau^t = (r^{E,t}_{1:5}, V^{E,t}_{1:5}, o^t_{1:5}, a^t_{1:5}) \leftarrow \mathbf{env}(p_{1:5})$ ▷ Play players in environment, collect outputs in a trajectory $\tau = \{\tau^1, \ldots, \tau^T\}$
    for $j := 1$ **to** 5 **do** ▷ Calculate feature vectors for players
      if retrospective then
        $f_j = e^t_j \leftarrow \eta\, e^{t-1}_j + r^{E,t}_j$ ▷ Decayed extrinsic reward
      else if prospective then
        $f_j \leftarrow V^{E,t}_j$ ▷ Value estimate from extrinsic value head
    for $i := 1$ **to** 5 **do** ▷ Calculate intrinsic rewards for players
      $\mathbf{f}_i \leftarrow \mathbf{reorder}(\mathbf{f}_i, i)$
      $u^t_i \leftarrow \mathbf{v}^{\mathrm{T}} \sigma\left(\mathbf{W}^{\mathrm{T}} \mathbf{f}_i + \mathbf{b}\right)$ ▷ Calculate intrinsic reward
  for $i := 1$ **to** 5 **do**
    $\phi_i \leftarrow \mathbf{RL}(\phi_i, \tau, u_i, h_i)$ ▷ RL training for each ϕ
    $F_{\phi_k}, F_{\theta_k} \leftarrow \mathcal{F}(\tau)$ ▷ Calculate smoothed fitnesses associated with each reward and policy network sampled in this episode
  for $(\phi_k, h_k) \in \mathcal{P}$ **do**
    if available_to_evolve(ϕ_k, h_k) **then** ▷ If burn-in period has passed, update population based on smoothed fitness of individuals
      $(\phi_k, h_k) \leftarrow \mathcal{U}(\mathcal{P}, F_{\phi_k})$
  for $\theta_k \in \mathcal{R}$ **do**
    if available_to_evolve(θ_k) **then**
      $\theta_k \leftarrow \mathcal{U}(\mathcal{R}, F_{\theta_k})$

rather than instantaneous values. Therefore, we considered two different ways of temporally aggregating the rewards.

The *retrospective* method derives intrinsic reward from whether an agent judges that other agents have been actually (extrinsically) rewarded in the recent past. The *prospective* variant derives intrinsic reward from whether other agents are expecting to be (extrinsically) rewarded in the near future.[3] For the retrospective variant, $f_{ij} = e^t_j$, where the temporally decayed reward e^t_j for the agents $j = 1, \ldots, N$ are updated at each timestep t according to

$$e^t_j = \eta\, e^{t-1}_j + r^{E,t}_j\,, \tag{3}$$

and $\eta = 0.975$. The prospective variant uses the value estimates V^E_j (see Figure 2b) for f_{ij} and has a stop-gradient before the reward network module so that gradients don't flow back into other agents (as in for example DIAL from [15]).

[3]Our terms prospective and retrospective map onto the terms intentional and consequentialist respectively as used by [36, 49].

2.3 Architecture and Training

We used the same training framework as in [29], which performs distributed asynchronous training in multi-agent environments, including population-based training (PBT) [30]. We trained a population of $N = 50$ agents[4] with policies $\{\pi_i\}$, from which we sampled 5 players in order to populate each of 500 arenas (where *arena* is an instantiation of a single episode of the environment) running in parallel. Within each arena, an episode of the environment was played with the sampled agents, before resampling new ones. Agents were sampled using one of two matchmaking processes (described in more detail below). Episode trajectories lasted 1000 steps and were written to queues for learning, from which weights were updated using V-Trace (Figure 2a).

The set of weights evolved included learning rate, entropy cost weight, and reward network weights θ[5]. The parameters of the policy network ϕ were inherited in a Lamarckian fashion as in [30]. Furthermore, we allowed agents to observe their last actions $a_{i,t-1}$, last intrinsic rewards $(u_{i,t-1}(\mathbf{f}_i))$, and last extrinsic rewards $(r^E_{i,t-1}(s_i, a_i))$ as input to the LSTM in the agent's neural network.

The objective function was identical to that presented in [11] and comprised three components: (1) the value function gradient, (2) policy gradient, and (3) entropy regularization, weighted according to hyperparameters baseline cost and entropy cost (see Figure 2b).

Evolution was based on a fitness measure calculated as a moving average of total episode return, which was a sum of apples collected minus penalties due to tagging, smoothed as follows:

$$F^n_i = (1 - \nu) F^{n-1}_i + \nu R^n_i\,, \tag{4}$$

where $\nu = 0.001$ and $R^n_i = \sum_t r^{E,t}_i$ is the total episode return obtained on episode n by agent i (or reward network i in the case of the shared reward network evolution (see Section 2.5 for details).

Training was done via joint optimization of network parameters via SGD and hyperparameters/reward network parameters via evolution in the standard PBT setup. Gradient updates were applied for every trajectory up to a maximum length of 100 steps, using a batch size of 32. Optimization was via RMSProp with epsilon=10^{-5}, momentum=0, decay rate=0.99, and an RL discount factor of 0.99. The baseline cost weight (see Mnih et al. [42]) was fixed at 0.25, and the entropy cost was sampled from LogUniform(2×10^{-4},0.01) and evolved throughout training using PBT. The learning rates were all initially set to 4×10^{-4} and then allowed to evolve.

PBT uses evolution (specifically genetic algorithms) to search over a space of hyperparameters rather than manually tuning or performing a random search, resulting in an adaptive schedule of hyperparameters and joint optimization with network parameters learned through gradient descent [30].

There was a mutation rate of 0.1 when evolving hyperparameters, using multiplicative perturbations of ±20% for entropy cost and learning rate, and additive perturbation of ±0.1 for reward network parameters. We implemented a burn-in period for evolution of 4×10^6 agent steps, to allow network parameters and hyperparameters

[4]Similar to as in [11], we distinguish between an "agent" which acts in the environment according to some policy, and a "learner" which updates the parameters of a policy. In principle, a single agent's policy may depend on parameters updated by several separate learners.

[5]We can imagine that the reward weights are simply another set of optimization hyperparameters since they enter into the loss.

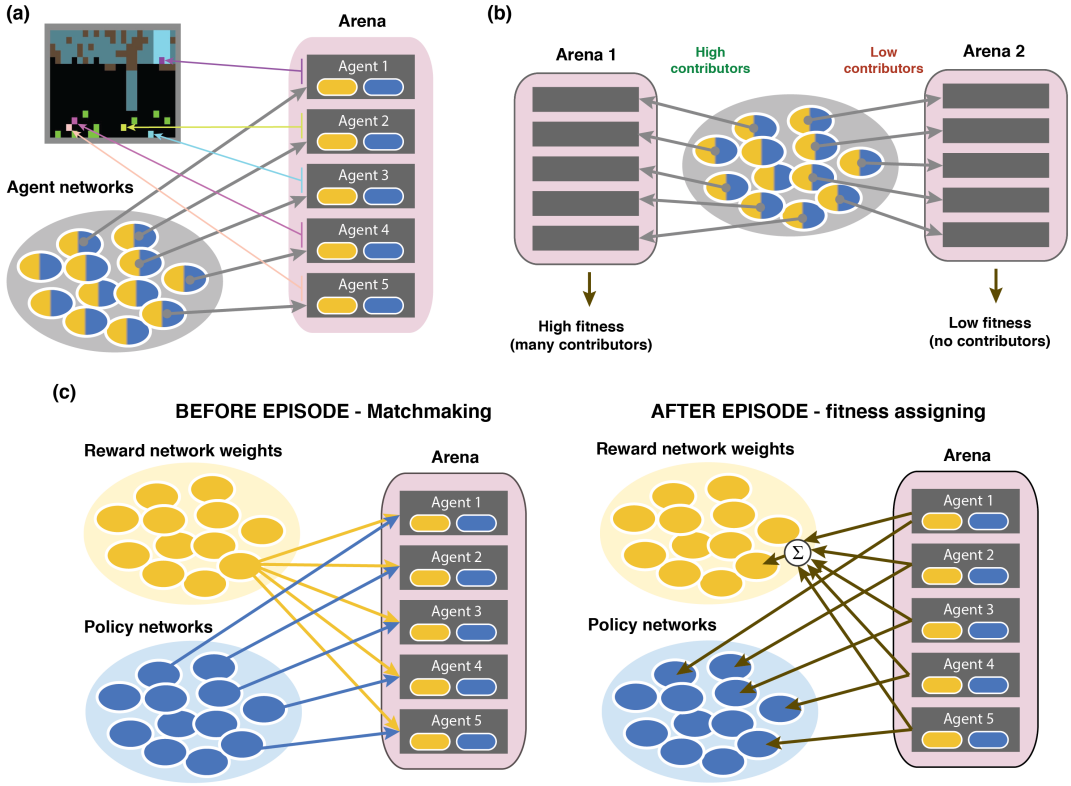

Figure 3: (a) Agents assigned and evolved with individual reward networks. (b) Assortative matchmaking, which preferentially plays cooperators with other cooperators and defectors with other defectors. (c) A single reward network is sampled from the population and assigned to all players, while 5 policy networks are sampled and assigned to the 5 players individually. After the episode, policy networks evolve according to individual player returns, while reward networks evolve according to aggregate returns over all players.

to be used in enough episodes for an accurate assessment of fitness before evolution.

2.4 Random vs. assortative matchmaking

Matches were determined according to two methods: (1) random matchmaking and (2) assortative matchmaking. Random matchmaking simply selected uniformly at random from the pool of agents to populate the game, while cooperative matchmaking first ranked agents within the pool according to a metric of recent cooperativeness, and then grouped agents such that players of similar rank played with each other. This ensured that highly cooperative agents played only with other cooperative agents, while defecting agents played only with other defectors. For Cleanup, cooperativeness was calculated based on the amount of steps in the last episode the agent chose to clean. For Harvest, it was calculated based on the difference between the the agent's return and the mean return of all players, so that having less return than average yielded a high cooperativeness ranking. Cooperative metric-based matchmaking was only done with either individual reward networks or no reward networks (Figure 3b). We did not use cooperative metric-based matchmaking for our multi-level selection model, since these are theoretically separate approaches.

2.5 Individual vs. shared reward networks

Building on previous work that evolved either the intrinsic reward [29] or the entire loss function [27], we considered the reward network weights to be hyperparameters that could be evolved in parallel with the policy parameters (Figure 3a). Distinct from these methods, we separately evolved the reward network within its own population, thereby allowing different modules of the agent to compete only with like components. This allowed for independent exploration of hyperparameters via separate credit assignment of fitness, and thus considerably more of the hyperparameter landscape could be explored compared with using only a single pool. In addition, reward networks could be randomly assigned to any policy network, and so were forced to generalize to a wide range of policies. In a given episode, 5 separate policy networks were paired with the same reward network, which we term a *shared reward network*. In line with [30], the fitness determining the copying of policy network weights and evolution of optimization-related hyperparameters (entropy cost and learning rate) were based on individual agent return. By contrast, the reward network parameters were evolved according to fitness based on total episode return across the group of co-players (Figure 3c).

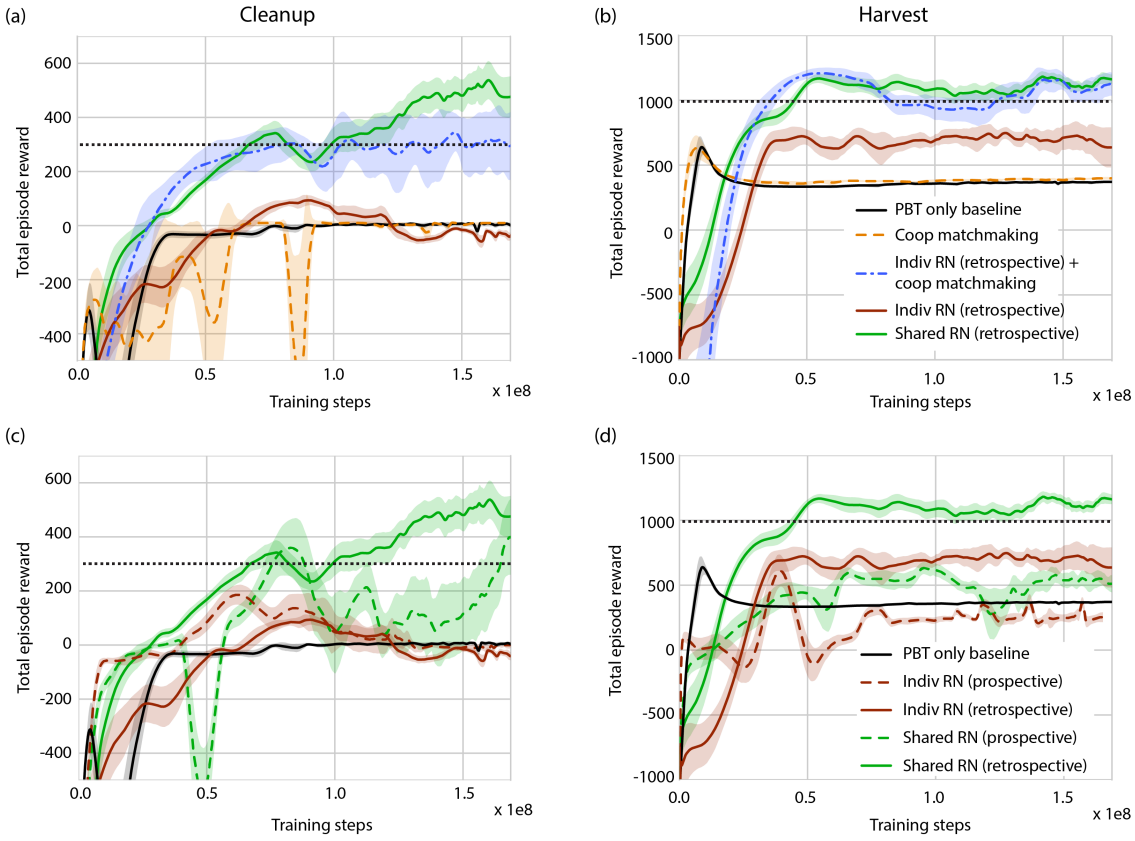

Figure 4: Total episode rewards, aggregated over players. (a), (b): Comparing retrospective (backward-looking) reward evolution with assortative matchmaking and PBT-only baseline in (a) Cleanup and (b) Harvest. (c), (d): Comparing prospective (forward-looking) with retrospective (backward-looking) reward evolution in (c) Cleanup and (d) Harvest. The black dotted line indicates performance from [28]. The shaded region shows standard error of the mean, taken over the population of agents.

This contribution is distinct from previous work which evolved intrinsic rewards [e.g. 29] because (1) we evolve over social features rather than a remapping of environmental events, and (2) reward network evolution is motivated by dealing with the inherent tension in ISDs, rather than merely providing a denser reward signal. In this sense it's more akin to evolving a form of communication for social cooperation, rather than learning reward-shaping in a sparse-reward environment. We allow for multiple agents to share the same components, and as we shall see, in a social setting, this winds up being critical. Shared reward networks provide a biologically principled method that mixes group fitness on a long timescale and individual reward on a short timescale. This contrasts with hand-crafted means of aggregation, as in previous work [5, 38].

3 RESULTS

As shown in Figure 4, PBT without using an intrinsic reward network performs poorly on both games, where it asymptotes to 0 total episode reward in Cleanup and 400 for Harvest (the number of apples gained if all agents collect as quickly as they can).

Figures 4a-b compare random and assortative matchmaking with PBT and reward networks using retrospective social features. When using random matchmaking, individual reward network agents perform no better than PBT at Cleanup, and only moderately better at Harvest. Hence there is little benefit to adding reward networks over social features if players have separate networks, as these tend to be evolved selfishly. The assortative matchmaking experiments used either no reward network ($u(\mathbf{f}) = 0$) or individual reward networks. Without a reward network, performance was the same as the PBT baseline. With individual reward networks, performance was very high, indicating that both conditioning the internal rewards on social features and a preference for cooperative agents to play together were key to resolving the dilemma. On the other hand, shared reward network agents perform as well as assortative matchmaking and the handcrafted inequity aversion intrinsic reward from [28], even using random matchmaking. This implies that agents didn't necessarily need to have immediate access to honest signals of other agents' cooperativeness to resolve the dilemma; it was enough to simply have the same intrinsic reward function, evolved according to collective episode return. Videos comparing performance of the PBT baseline with the retrospective variant of shared reward network evolution can be found at https://youtu.be/medBBLLM4c0 and https://youtu.be/yTjrlH3Ms9U.

Figures 4(c) and (d) compare the retrospective and prospective variants of reward network evolution. The prospective variant, although better than PBT when using a shared reward network, generally results in worse performance and more instability. This is likely because the prospective variant depends on agents learning good value estimates before the reward networks become useful, whereas the retrospective variant only depends on environmentally provided reward and thus does not suffer from this issue. Interestingly, we observed that the prospective variant does achieve very high performance if gradients are allowed to pass between agents via the value estimates V_j^E (data not shown); however, this constitutes centralized learning, albeit with decentralized execution (see [15]). Such approaches are promising but less consistent with biologically plausible mechanisms of multi-agent learning which are of interest here and so were not pursued.

We next plot various social outcome metrics in order to better capture the complexities of agent behavior (see Figure 5). Equality is calculated as $\mathbb{E}(1 - G(\mathbf{R}))$, where $G(\mathbf{R})$ is the Gini coefficient over individual returns. Figure 5b demonstrates that, in Harvest, having the prospective version of reward networks tends to lead to lower equality, while the retrospective variant has very high equality. Equality in Cleanup is more unstable throughout training, since it's not necessarily optimal, but tends to be lower overall than for Harvest, even when performance is high, indicating that equality might be harder to achieve in different games. Tagging measures the average number of times a player fined another player throughout the episode. The middle panel of Figure 5b shows that there is a higher propensity for tagging in Harvest when using either a prospective reward network or an individual reward network, compared to the retrospective shared reward network. This explains the performance shown in Figure 4, as being tagged results in a very high negative reward. Tagging in the Cleanup task is overall much lower than in Harvest. Sustainability measures the average time step on which agents received positive reward, averaged over the episode and over agents. We see in the bottom panel of 5b that having no reward network results in players collecting apples extremely quickly in Harvest, compared with much more sustainable behavior with reward networks. In Cleanup, the sustainability metric is not meaningful and so this was not plotted.

Finally, we can directly examine the weights of the final retrospective shared reward networks which were best at resolving the ISDs. Interestingly, the final weights evolved in the second layer suggest that resolving each game might require a different set of social preferences. In Cleanup, one of the final layer weights v_2 evolved to be close to 0, whereas in Harvest, v_1 and v_2 evolved to be of large magnitude but opposite sign. We can see a similar pattern with the biases $\mathbf{b}$. We interpret this to mean that Cleanup required a less complex reward network: it was enough to simply find other agents' being rewarded as intrinsically rewarding. In Harvest, however, a more complex reward function was perhaps needed in order to ensure that other agents were not over-exploiting the apples. We found that the first layer weights $\mathbf{W}$ tended to take on arbitrary (but positive) values. This is because of random matchmaking: co-players were randomly selected and thus there was little evolutionary pressure to specialize these weights.

4 DISCUSSION

Real environments don't provide scalar reward signals to learn from. Instead, organisms have developed various internal drives based on either primary or secondary goals [2]. Here we examined intrinsic rewards based on features derived from other agents in the environment, in order to establish whether such social signals could enable the evolution of altruism to solve intertemporal social dilemmas. In accord with evolutionary theory [1, 43], we found that naïvely implementing natural selection via genetic algorithms did not lead to the emergence of cooperation. Furthermore, assortative matchmaking was sufficient to generate cooperative behavior in cases where honest signals were available. Finally, we proposed a new multi-level evolutionary paradigm based on shared reward networks that achieves cooperation in more general situations.

We demonstrated that the reward network weights evolved differently for Cleanup versus Harvest, indicating that the two tasks necessitate different forms of social cooperation for optimal performance. This highlights the advantage of evolving rather than hand-crafting the weighting between individual reward and group reward, as optimal weightings cannot necessarily be anticipated for all environments. Evolving such weightings thus constitutes a form of meta-learning, wherein an entire learning system, including intrinsic reward functions, is optimized for fast learning [13, 53]. Here we have extended these ideas to the multi-agent domain.

Why does evolving intrinsic social preferences promote cooperation? Firstly, evolution ameliorates the intertemporal choice problem by distilling the long timescale of collective fitness into the short timescale of individual reinforcement learning, thereby improving credit assignment between selfish acts and their temporally displaced negative group outcomes [28]. Secondly, it mitigates the social dilemma itself by allowing evolution to expose social signals that correlate with, for example, an agent's current level of selfishness. Such information powers a range of mechanisms for achieving mutual cooperation like competitive altruism [24], other-regarding preferences [8], and inequity aversion [12]. In accord, laboratory experiments show that humans cooperate more readily when they can communicate [32, 46].

The shared reward network evolution model was inspired by multi-level selection; yet it does not correspond to the prototypical case of that theory since its lower level units of evolution (the policy networks) are constantly swapping which higher level unit (reward network) they are paired with. Nevertheless, there are a variety of ways in which we see this form of modularity arise in nature. For example, free-living microorganisms occasionally form multicellular structures to solve a higher order adaptive problem, like slime mold forming a spore-producing stalk for dispersal [58], and many prokaryotes can incorporate plasmids (modules) found in their environment or received from other individuals as functional parts of their genome, thereby achieving cooperation in social dilemmas [20, 40]. Alternatively, in humans a reward network may represent a shared "cultural norm", with its fitness based on cultural information accumulated from the groups in which it holds sway. In this way, the spread of norms can occur independently of the success of individual agents [3].

Note that in this work, we have assumed that agents have perfect knowledge of other agents' rewards, while in real-world systems

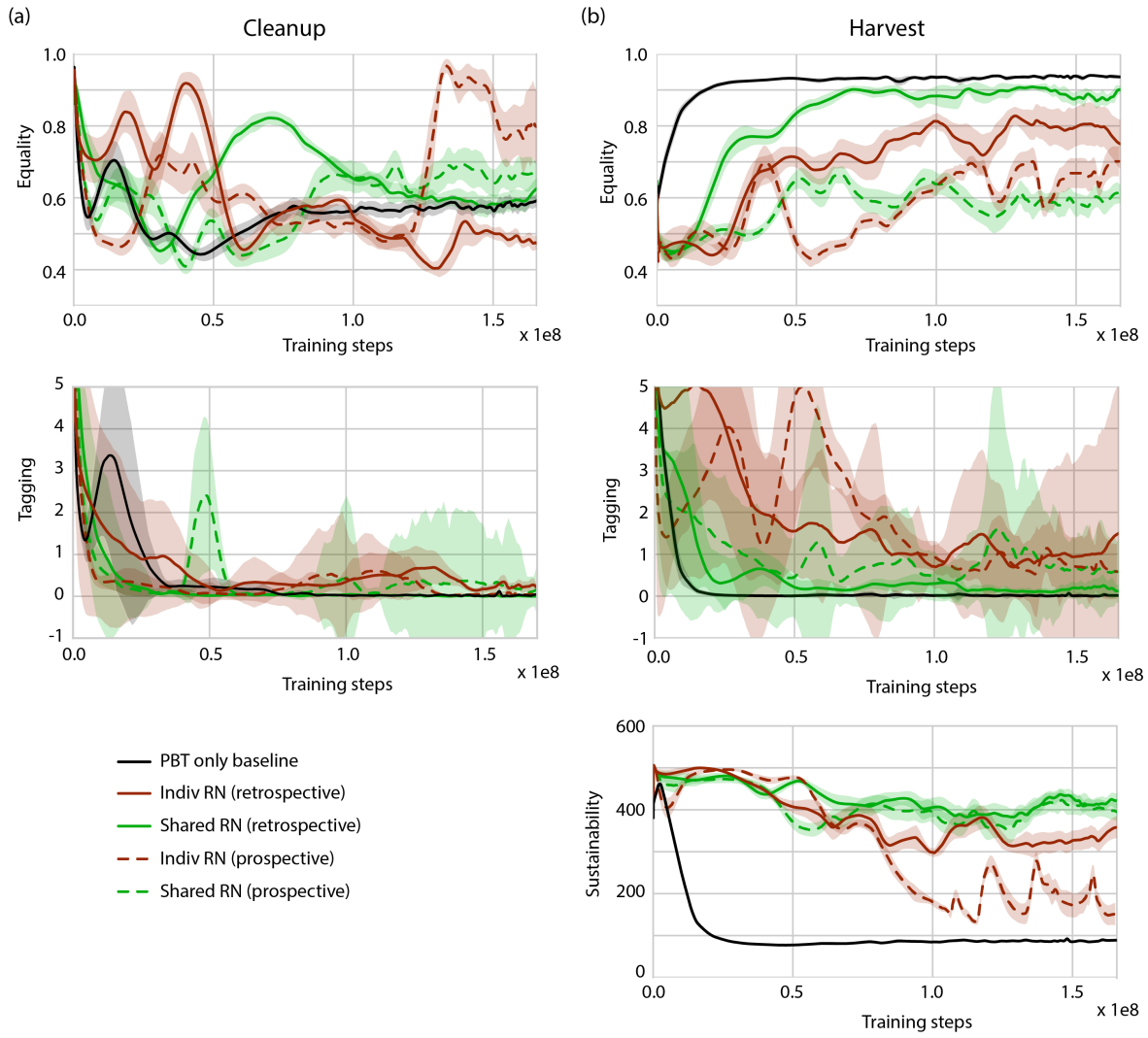

Figure 5: Social outcome metrics for (a) Cleanup and (b) Harvest. *Top:* equality, *middle:* total amount of tagging, *bottom:* sustainability. The shaded region shows the standard error of the mean.

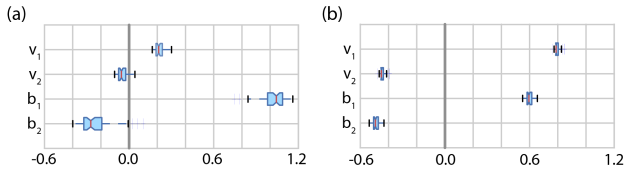

Figure 6: Distribution of layer 2 weights and biases of evolved retrospective shared reward network at 1.5×10^8 training steps for (a) Cleanup, and (b) Harvest.

this is not typically the case. This assumption was made in order to disentangle the effects of cultural evolution from the quality of the signals being evolved over. Natural next steps include adding partial observability or noise to this signal (to make it more analogous to, for instance, a smile/frown or other locally observable social signals), identifiability across episodes, or even deception.

The approach outlined here opens avenues for investigating alternative evolutionary mechanisms for the emergence of cooperation, such as kin selection [19] and reciprocity [56]. It would be interesting to see whether these lead to different weights in a reward network, potentially hinting at the evolutionary origins of different social biases. Along these lines, one might consider studying an emergent version of the assortative matchmaking model along the lines suggested by [25], adding further generality and power to our setup. Finally, it would be fascinating to determine how an evolutionary approach can be combined with multi-agent communication to produce that most paradoxical of cooperative behaviors: cheap talk.

ACKNOWLEDGMENTS

We would like to thank Simon Osindero, Iain Dunning, Andrea Tacchetti, and many DeepMind colleagues for valuable discussions and feedback, as well as code development and support.

REFERENCES

[1] Robert Axelrod and William D. Hamilton. 1981. The Evolution of Cooperation. *Science* 211, 4489 (1981), 1390–1396. http://www.jstor.org/stable/1685895
[2] Gianluca Baldassarre and Marco Mirolli. 2013. *Intrinsically Motivated Learning in Natural and Artificial Systems*. 1–458 pages.
[3] Robert Boyd and Peter J. Richerson. 2009. Culture and the evolution of human cooperation. *Philosophical Transactions of the Royal Society of London B: Biological Sciences* 364, 1533 (2009), 3281–3288. https://doi.org/10.1098/rstb.2009.0134 arXiv:http://rstb.royalsocietypublishing.org/content/364/1533/3281.full.pdf
[4] ZSH Chan, HW Ngan, AB Rad, and TK Ho. 2002. Alleviating'overfitting'via genetically-regularised neural network. *Electronics Letters* 38, 15 (2002), 1.
[5] Yu-Han Chang, Tracey Ho, and Leslie P Kaelbling. 2004. All learning is local: Multi-agent learning in global reward games. In *Advances in neural information processing systems*. 807–814.
[6] Nuttapong Chentanez, Andrew G. Barto, and Satinder P. Singh. 2005. Intrinsically Motivated Reinforcement Learning. In *Advances in Neural Information Processing Systems 17*, L. K. Saul, Y. Weiss, and L. Bottou (Eds.). MIT Press, 1281–1288. http://papers.nips.cc/paper/2552-intrinsically-motivated-reinforcement-learning.pdf
[7] Edoardo Conti, Vashisht Madhavan, Felipe Petroski Such, Joel Lehman, Kenneth O Stanley, and Jeff Clune. 2017. Improving exploration in evolution strategies for deep reinforcement learning via a population of novelty-seeking agents. *arXiv preprint arXiv:1712.06560* (2017).
[8] David J Cooper and John H Kagel. 2016. Other-regarding preferences. *The handbook of experimental economics* 2 (2016), 217.
[9] Charles Darwin. 1859. *On the Origin of Species by Means of Natural Selection*. Murray, London. or the Preservation of Favored Races in the Struggle for Life.
[10] Richard Dawkins. 1976. The selfish gene Oxford university press. *New York* (1976).
[11] Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Volodymir Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, et al. 2018. IMPALA: Scalable distributed Deep-RL with importance weighted actor-learner architectures. *arXiv preprint arXiv:1802.01561* (2018).
[12] Ernst Fehr and Klaus M. Schmidt. 1999. A Theory of Fairness, Competition, and Cooperation*. *The Quarterly Journal of Economics* 114, 3 (1999), 817–868. https://doi.org/10.1162/003355399556151
[13] Chrisantha Thomas Fernando, Jakub Sygnowski, Simon Osindero, Jane Wang, Tom Schaul, Denis Teplyashin, Pablo Sprechmann, Alexander Pritzel, and Andrei A Rusu. 2018. Meta Learning by the Baldwin Effect. *arXiv preprint arXiv:1806.07917* (2018).
[14] Jeffrey A Fletcher and Michael Doebeli. 2009. A simple and general explanation for the evolution of altruism. *Proceedings of the Royal Society of London B: Biological Sciences* 276, 1654 (2009), 13–19.
[15] Jakob N. Foerster, Yannis M. Assael, Nando de Freitas, and Shimon Whiteson. 2016. Learning to Communicate with Deep Multi-Agent Reinforcement Learning. *CoRR* abs/1605.06676 (2016). arXiv:1605.06676 http://arxiv.org/abs/1605.06676
[16] Jakob N. Foerster, Richard Y. Chen, Maruan Al-Shedivat, Shimon Whiteson, Pieter Abbeel, and Igor Mordatch. 2017. Learning with Opponent-Learning Awareness. *CoRR* abs/1709.04326 (2017). arXiv:1709.04326 http://arxiv.org/abs/1709.04326
[17] Andy Gardner and Stuart A West. 2010. Greenbeards. *Evolution: International Journal of Organic Evolution* 64, 1 (2010), 25–38.
[18] Alan Grafen et al. 1990. Do animals really recognize kin?. *Animal Behaviour* 39, 1 (1990), 42–54.
[19] Ashleigh Griffin and Stuart West. 2002. Kin selection: Fact and fiction. 17 (01 2002), 15–21.
[20] Ashleigh S Griffin, Stuart A West, and Angus Buckling. 2004. Cooperation and competition in pathogenic bacteria. *Nature* 430, 7003 (2004), 1024.
[21] Özgür Gürerk, Bernd Irlenbusch, and Bettina Rockenbach. 2006. The competitive advantage of sanctioning institutions. *Science* 312, 5770 (2006), 108–111.
[22] W.D. Hamilton. 1964. The Genetical Evolution of Social Behaviour. I. *Journal of Theoretical Biology* 7, 1 (July 1964), 1–16.
[23] William D Hamilton. 1964. The genetical evolution of social behaviour. II. *Journal of theoretical biology* 7, 1 (1964), 17–52.
[24] Charlie L Hardy and Mark Van Vugt. 2006. Nice guys finish first: The competitive altruism hypothesis. *Personality and Social Psychology Bulletin* 32, 10 (2006), 1402–1413.
[25] Joseph Henrich. 2004. Cultural group selection, coevolutionary processes and large-scale cooperation. *Journal of Economic Behavior & Organization* 53, 1 (2004), 85–88.
[26] Geoffrey E Hinton and Steven J Nowlan. 1987. How learning can guide evolution. *Complex systems* 1, 3 (1987), 495–502.
[27] Rein Houthooft, Richard Y. Chen, Phillip Isola, Bradly C. Stadie, Filip Wolski, Jonathan Ho, and Pieter Abbeel. 2018. Evolved Policy Gradients. *CoRR* abs/1802.04821 (2018). arXiv:1802.04821 http://arxiv.org/abs/1802.04821
[28] Edward Hughes, Joel Z Leibo, Matthew G Phillips, Karl Tuyls, Edgar A Duéñez-Guzmán, Antonio García Castañeda, Iain Dunning, Tina Zhu, Kevin R McKee, Raphael Koster, et al. 2018. Inequity aversion improves cooperation in intertemporal social dilemmas. In *Advances in neural information processing systems (NIPS)*. Montreal, Canada.
[29] Max Jaderberg, Wojciech M Czarnecki, Iain Dunning, Luke Marris, Guy Lever, Antonio Garcia Castaneda, Charles Beattie, Neil C Rabinowitz, Ari S Morcos, Avraham Ruderman, et al. 2018. Human-level performance in first-person multiplayer games with population-based deep reinforcement learning. *arXiv preprint arXiv:1807.01281* (2018).
[30] Max Jaderberg, Valentin Dalibard, Simon Osindero, Wojciech M Czarnecki, Jeff Donahue, Ali Razavi, Oriol Vinyals, Tim Green, Iain Dunning, Karen Simonyan, et al. 2017. Population based training of neural networks. *arXiv preprint arXiv:1711.09846* (2017).
[31] Vincent Jansen and Minus van Baalen. 2006. Altruism through beard chromodynamics. 440 (04 2006), 663–6.
[32] Marco A Janssen, Robert Holahan, Allen Lee, and Elinor Ostrom. 2010. Lab experiments for the study of social-ecological systems. *Science* 328, 5978 (2010), 613–617.
[33] Laurent Keller and Kenneth G. Ross. 1998. Selfish genes: A green beard in the red fire ant. 394 (08 1998), 573–575.
[34] Max Kleiman-Weiner, Mark K. Ho, Joseph L. Austerweil, Michael L. Littman, and Joshua B. Tenenbaum. 2016. Coordinate to cooperate or compete: Abstract goals and joint intentions in social interaction. In *CogSci*.
[35] Joel Z Leibo, Vinicius Zambaldi, Marc Lanctot, Janusz Marecki, and Thore Graepel. 2017. Multi-agent reinforcement learning in sequential social dilemmas. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 464–473.
[36] Adam Lerer and Alexander Peysakhovich. 2017. Maintaining cooperation in complex social dilemmas using deep reinforcement learning. *CoRR* abs/1707.01068 (2017). arXiv:1707.01068 http://arxiv.org/abs/1707.01068
[37] Michael L Littman. 1994. Markov games as a framework for multi-agent reinforcement learning. In *Machine Learning Proceedings 1994*. Elsevier, 157–163.
[38] Maja J Mataric. 1994. Learning to behave socially. In *Third international conference on simulation of adaptive behavior*, Vol. 617. 453–462.
[39] John Maynard Smith and Eors Szathmary. 1997. *The major transitions in evolution*. Oxford University Press.
[40] Sorcha E McGinty, Daniel J Rankin, and Sam P Brown. 2011. Horizontal gene transfer and the evolution of bacterial cooperation. *Evolution: International Journal of Organic Evolution* 65, 1 (2011), 21–32.
[41] Geoffrey F Miller, Peter M Todd, and Shailesh U Hegde. 1989. Designing Neural Networks using Genetic Algorithms.. In *ICGA*, Vol. 89. 379–384.
[42] Volodymyr Mnih, Adrià Puigdomènech Badia, Mehdi Mirza, Alex Graves, Timothy P. Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. 2016. Asynchronous Methods for Deep Reinforcement Learning. *CoRR* abs/1602.01783 (2016). arXiv:1602.01783 http://arxiv.org/abs/1602.01783
[43] Martin A. Nowak. 2006. Five Rules for the Evolution of Cooperation. *Science* 314, 5805 (2006), 1560–1563. https://doi.org/10.1126/science.1133755 arXiv:http://science.sciencemag.org/content/314/5805/1560.full.pdf
[44] Martin A Nowak and Karl Sigmund. 2005. Evolution of indirect reciprocity. *Nature* 437, 7063 (2005), 1291.
[45] Pamela Oliver. 1980. Rewards and punishments as selective incentives for collective action: theoretical investigations. *American journal of sociology* 85, 6 (1980), 1356–1375.
[46] Elinor Ostrom, James Walker, and Roy Gardner. 1992. Covenants with and without a sword: Self-governance is possible. *American political science Review* 86, 2 (1992), 404–417.
[47] Liviu Panait and Sean Luke. 2005. Cooperative multi-agent learning: The state of the art. *Autonomous agents and multi-agent systems* 11, 3 (2005), 387–434.
[48] Julien Pérolat, Joel Z. Leibo, Vinícius Flores Zambaldi, Charles Beattie, Karl Tuyls, and Thore Graepel. 2017. A multi-agent reinforcement learning model of common-pool resource appropriation. *CoRR* abs/1707.06600 (2017). arXiv:1707.06600 http://arxiv.org/abs/1707.06600
[49] Alexander Peysakhovich and Adam Lerer. 2018. Consequentialist conditional cooperation in social dilemmas with imperfect information. In *International Conference on Learning Representations*. https://openreview.net/forum?id=BkabRiQpb
[50] Mitchell A Potter and Kenneth A De Jong. 2000. Cooperative coevolution: An architecture for evolving coadapted subcomponents. *Evolutionary computation* 8, 1 (2000), 1–29.
[51] Francisco C Santos, Jorge M Pacheco, and Tom Lenaerts. 2006. Cooperation prevails when individuals adjust their social ties. *PLoS computational biology* 2, 10 (2006), e140.
[52] Satinder Singh, Richard L Lewis, and Andrew G Barto. 2009. Where do rewards come from. In *Proceedings of the annual conference of the cognitive science society*. 2601–2606.

[53] Satinder Singh, Richard L Lewis, Andrew G Barto, and Jonathan Sorg. 2010. Intrinsically motivated reinforcement learning: An evolutionary perspective. *IEEE Transactions on Autonomous Mental Development* 2, 2 (2010), 70–82.
[54] J Maynard Smith. 1964. Group selection and kin selection. *Nature* 201, 4924 (1964), 1145.
[55] Kenneth O Stanley and Risto Miikkulainen. 2002. Evolving neural networks through augmenting topologies. *Evolutionary computation* 10, 2 (2002), 99–127.
[56] Robert L Trivers. 1971. The evolution of reciprocal altruism. *The Quarterly review of biology* 46, 1 (1971), 35–57.
[57] Francisco Úbeda and Edgar A Duéñez-Guzmán. 2011. Power and corruption. *Evolution: International Journal of Organic Evolution* 65, 4 (2011), 1127–1139.
[58] Stuart A West, Ashleigh S Griffin, Andy Gardner, and Stephen P Diggle. 2006. Social evolution theory for microorganisms. *Nature Reviews Microbiology* 4, 8 (2006), 597.
[59] Daan Wierstra, Tom Schaul, Jan Peters, and Juergen Schmidhuber. 2008. Natural evolution strategies. In *Evolutionary Computation, 2008. CEC 2008.(IEEE World Congress on Computational Intelligence). IEEE Congress on.* IEEE, 3381–3387.
[60] David Sloan Wilson. 1975. A theory of group selection. *Proceedings of the national academy of sciences* 72, 1 (1975), 143–146.
[61] Chern Han Yong and Risto Miikkulainen. 2001. Cooperative coevolution of multi-agent systems. *University of Texas at Austin, Austin, TX* (2001).