Multi-Agent Reinforcement Learning with General Utilities via Decentralized Shadow Reward Actor-Critic

Junyu Zhang,[1] Amrit Singh Bedi,[2] Mengdi Wang,[3] Alec Koppel[4]

[1]Dept. of ISEM, National University of Singapore, Singapore

[2] CISD, US Army Research Laboratory, USA

[3] Dept. of EE, Center for Statistics and Machine Learning, Princeton University/Deepmind, Princeton, USA

[4] Supply Chain Optimization Technologies, Amazon, USA

{junyuz@nus.edu.sg, amrit0714@gmail.com, mengdiw@princeton.edu, aekoppel@amazon.com}

Abstract

We posit a new mechanism for cooperation in multi-agent reinforcement learning (MARL) based upon any nonlinear function of the team's long-term state-action occupancy measure, i.e., a general utility. This subsumes the cumulative return but also allows one to incorporate risk-sensitivity, exploration, and priors. We derive the Decentralized Shadow Reward Actor-Critic (DSAC) in which agents alternate between policy evaluation (critic), weighted averaging with neighbors (information mixing), and local gradient updates for their policy parameters (actor). DSAC augments the classic critic step by requiring agents to (i) estimate their local occupancy measure in order to (ii) estimate the derivative of the local utility with respect to their occupancy measure, i.e., the "shadow reward". DSAC converges to a stationary point in sublinear rate with high probability, depending on the amount of communications. Under proper conditions, we further establish the non-existence of spurious stationary points for this problem, that is, DSAC finds the globally optimal policy. Experiments demonstrate the merits of goals beyond the cumulative return in cooperative MARL.

Introduction

Reinforcement learning (RL) is a framework for directly estimating the parameters of a controller through repeated interaction with the environment (Sutton and Barto 2018), and has gained attention for its ability to alleviate the need for a physically exact model across a number of domains, such as robotic manipulation (Kober, Bagnell, and Peters 2013), web services (Zhao et al. 2018), and logistics (Feinberg 2016), and various games (Silver et al. 2016). In RL, an agent in a given state takes an action, and transitions to another according to a Markov transition density, whereby a reward informing the merit of the action is revealed by the environment. Mathematically, this setting may be encapsulated by a Markov Decision Process (MDP) (Puterman 2014), in which the one seeks to select actions to maximize the long-term accumulation of rewards.

In many domains, multiple agents interact in order to obtain favorable outcomes, as in finance (Lee, Zhang et al. 2002), social networks (Jaques et al. 2019), and games (Vinyals et al. 2019). In multi-agent RL (MARL) and more generally, stochastic games, a key question is the payoff structure (Shapley 1953; Başar and Olsder 1998). We focus on common payoffs among agents, i.e., the utility of the team is the sum of local utilities (Busoniu, Babuska, and De Schutter 2008), which contrasts with competitive settings where one agent's gain is another's loss, or combinations thereof (Littman 1994). Whereas typically cooperative MARL defines the global utility as the average over agents' local reward accumulations, here we define a *new mechanism for cooperation* that permits agents to incorporate risk-sensitivity (Huang and Kallenberg 1994; Borkar and Meyn 2002; Prashanth and Ghavamzadeh 2016), prior experience (Argall et al. 2009), or exploration (Hazan et al. 2019; Tarbouriech and Lazaric 2019). The usual common-payoff setting focuses on global cumulative return of rewards, which is a linear function of the the state-action occupancy measure. By contrast, the aforementioned decision-making goals define *nonlinear* functions of the state-action occupancy measure (Kallenberg 1994). Such functions, which we call *general utilities*, have recently yielded impressive performance in practice via prioritizing exploration (Mahajan et al. 2019; Gupta et al. 2020), risk-sensitivity (Mystery 2021), and prior experience (Le et al. 2017; Lee and Lee 2019). However, to the best of our knowledge, there exists few formal guarantees for algorithms designed to optimize general utilities in multi-agent settings.

This gap motivates us to put forth the first decentralized MARL scheme for general utilities, and establish its consistency and sample complexity. Our approach hinges upon first noting that the embarking point for most RL methodologies is the Policy Gradient Theorem (Williams 1992; Sutton et al. 2000) or Bellman's equation, both of which break down for general utilities. One potential path forward is a recent generalization of the PG Theorem for general utilities (Zhang et al. 2020b), which expresses the gradient as product of the partial derivative of the utility with respect to the occupancy measure, and the occupancy measure with respect to the policy. However, in the team setting, this later factor is a *global nonlinear function* of agents' policies, and hence does not permit decentralization. Thus, we define an agent's local occupancy measure as the joint occupancy measure of all agents' polices with all others' marginalized out, and its local general utility as any (not-necessarily concave) function of its marginal occupancy measure. The team objective, then, is the global aggregation of all local utilities.

From this definition, we derive a new variant of the Policy Gradient [cf. (6)] where each agent estimate its policy gradient based on local information and message passing with neighbors. This leads to a model-free algorithm, **D**ecentralized **S**hadow Reward **A**ctor-**C**ritic (DSAC), that generalizes multi-agent actor-critic (see (Konda and Borkar 1999; Konda and Tsitsiklis 2000)) beyond cumulative return (Zhang et al. 2018). Each agent's procedure follows four stages: (i) a marginalized occupancy measure estimation step used to evaluate the instantaneous gradient of the local utility with respect to the occupancy measure, which we dub the "shadow reward" (shadow reward computation); (ii) accumulate "shadow rewards" along a trajectory to estimate "shadow" critic parameters (critic); (iii) average critic parameters with those of its neighbors; and (iii) a stochastic policy gradient ascent step along trajectories (actor).

Contributions. Overall, our contributions are:

- present the first MARL formulation that permits broader goals than the cumulative return and specialization among agents' roles;
- derive a variant of multi-agent actor-critic to solve this problem that employs an occupancy measure estimation step to construct the gradient of the general utility with respect to the occupancy measure, which serves as a "shadow reward" for the critic step;
- for ϵ-stationarity with high probability, we respectively establish that DSAC requires $\mathcal{O}(1/\epsilon^{2.5})$ and $\mathcal{O}(1/\epsilon^{2})$ steps if agents exchange information once (Theorem 2) or multiple times per policy update (Corollary 2). Under proper assumptions, we further establish the convergence to the globally optimal policy under diminishing step-sizes (Corollary 1).
- provide experimental evaluation of this scheme for exploration maximization and safe navigation in cooperative settings (Lowe et al. 2017).

Problem Formulation

Consider a Markov decision process (MDP) over the finite state space $\mathcal{S}$ and a finite action space $\mathcal{A}$. For each state $s \in \mathcal{S}$, a transition to state $s' \in \mathcal{S}$ occurs when selecting action $a \in \mathcal{A}$ according to a conditional probability distribution $s' \sim \mathcal{P}(\cdot|a, s)$, for which we define the short-hand notation $P_a(s, s')$. Let ξ be the initial state distribution of the MDP, i.e., $s_0 \sim \xi$. We let $S := |\mathcal{S}|$ denote the number of states and $A := |\mathcal{A}|$ the number of actions. Consider policy optimization for maximizing general objectives that are nonlinear function of the *cumulative discounted state-action occupancy measure* under policy π, which contains the cumulative return as a special case (Zhang et al. 2020a,b):

$$\max_{\pi} R(\pi) := F(\lambda^{\pi}) \tag{1}$$

where F is a general (not necessarily concave) functional and λ^{π} is occupancy measure given by

$$\lambda^{\pi}(s, a) = \sum_{t=0}^{\infty} \gamma^t \cdot \mathbb{P}\Big(s^t = s, a^t = a \;\Big|\; \pi, s^0 \sim \xi\Big) \tag{2}$$

for $\forall a \in \mathcal{A}, \forall s \in \mathcal{S}$. For instance, often in applications one has access to demonstrations which can be used to learn a prior on the policy for ensuring baseline performance. Suppose $\bar{\lambda}$ is a prior state-action distribution obtained from demonstrations. One may seek to maintain baseline performance with respect to this prior via minimizing the Kullback-Liebler (KL) divergence between the normalized distribution $\hat{\lambda} = (1-\gamma)\lambda$ and the prior $\bar{\lambda}$ stated as $\rho(\lambda) = \mathrm{KL}\big((1-\gamma)\lambda || \bar{\lambda}\big)$. In behavioral cloning, action information is missing, in which case one may instead consider a variant with respect to only the state occupancy measure. Other forms for (1) are considered in Sec. 12.

In this work, we consider the decentralized version of the problem in (1), where the state space $\mathcal{S}$, the action space $\mathcal{A}$, the policy π, and the general utility F are decentralized among $N = |\mathcal{V}|$ distinct agents associated with an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertex set $\mathcal{V}$ and edge set $\mathcal{E}$. Each agent $i \in \mathcal{V}$ is associated with its own local incentives and actions, detailed as follows.

Space Decomposition. The global state space $\mathcal{S}$ is the product of N local spaces $\mathcal{S}_i$, i.e., $\mathcal{S} = \mathcal{S}_1 \times \mathcal{S}_2 \times \cdots \times \mathcal{S}_N$, meaning that for any $s \in \mathcal{S}$, we may write $s = (s_{(1)}, s_{(2)}, \cdots, s_{(N)})$ with $s_{(i)} \in \mathcal{S}_i, i \in \mathcal{V}$. Each agent has access to the global state s, as customary of joint-action learners training in a decentralized manner under full observability (Kar, Moura, and Poor 2013; Zhang et al. 2018; Lee et al. 2018; Wai et al. 2018; Qu et al. 2019; Doan, Maguluri, and Romberg 2019). Similarly, the global action space $\mathcal{A}$ is the product of N local spaces $\mathcal{A}_i$: $\mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2 \times \cdots \times \mathcal{A}_N$, meaning that for any $a \in \mathcal{A}$, we may write $a = (a_{(1)}, a_{(2)}, \cdots, a_{(N)})$ with $a_{(i)} \in \mathcal{A}_i, i \in \mathcal{V}$. Full observability means each agent i has access to global actions a concatenating all local ones.

Policy Factorization. The global policy $\pi(a|s)$ that maps global action a for a given global state s is defined as the product of local policies $\prod_{i=1}^{N} \pi^{(i)}(a_{(i)}|s)$, which prescribes statistical independence among agents' policies. For the parameterized policy $\pi_\theta(a|s)$ where $\theta \in \Theta$, we denote $\theta = (\theta_1, \theta_2, \cdots, \theta_N)$ as the parameter, so we can write $\pi_\theta(a|s) = \prod_{i \in \mathcal{V}} \pi_{\theta_i}^{(i)}(a_{(i)}|s)$, where the local policy of agent i is parameterized by θ_i. Since the global state is visible to all agents, the *local policy* is based on the observation of the *global state*. The parameters θ_i are kept private by agent i, meaning that agents must pass messages to become informed about others' incentives.

Local Cumulative State-Action Occupancy Measure. Similar to the global occupancy measure $\lambda^{\pi}(s, a)$ [cf. (2)], define the *local cumulative state-action occupancy measure*:

$$\lambda_{(i)}^{\pi}(s_{(i)}, a_{(i)}) = \sum_{t=0}^{\infty} \gamma^t \cdot \mathbb{P}\Big(s_{(i)}^t = s_{(i)}, a_{(i)}^t = a_{(i)} \;\Big|\; \pi, s^0 \sim \xi\Big) \tag{3}$$

for $\forall a_{(i)} \in \mathcal{A}_i, s_{(i)} \in \mathcal{S}_i$. This local occupancy measure is the marginalization of the global occupancy measure with respect to all others' measures than agent i, whose indices are denoted as $\{-i\} \subset \mathcal{V}$. Via marginalization, we write

$$\lambda_{(i)}^{\pi}(s_{(i)}, a_{(i)}) = \sum_{a \in \{a_{(i)}\} \times \mathcal{A}_{-i}} \; \sum_{s \in \{s_{(i)}\} \times \mathcal{S}_{-i}} \lambda^{\pi}(s, a) \tag{4}$$

Objective	Approach	Convergence
Cumulative Return	Value-Based [In caption]	✓
	Policy-Based (Zhang et al. 2018; Chen et al. 2018)	✓
Risk	(Mystery 2021)	✗
Exploration	(Mahajan et al. 2019; Gupta et al. 2020)	✗
Priors	(Le et al. 2017; Lee and Lee 2019)	✗
		This Work ✓

Table 1: Cumulative Returns, Risk-Sensitivity, Exploration, and the incorporation of Priors are common goals in multi-agent reinforcement learning, and subsumed by the general utilities considered here. Value-based approaches for the cumulative return include (Kar, Moura, and Poor 2013; Wai et al. 2018; Lee et al. 2018; Qu et al. 2019; Doan, Maguluri, and Romberg 2019). We focus on the setting when agents are **cooperative** and transition according to a common **global** dynamics model (Busoniu, Babuska, and De Schutter 2008). The state space model under consideration is most similar to (Zhang et al. 2018). We note that the technical settings of (Le et al. 2017; Lee and Lee 2019; Mahajan et al. 2019; Gupta et al. 2020; Mystery 2021) are different; their inclusion here is to spotlight their use of goals beyond cumulative return, which is given a conceptual underpinning for the first time in this work.

with $\mathcal{A}_{-i}$=$\Pi_{j\neq i}\mathcal{A}_j$ and $\mathcal{S}_{-i}$=$\Pi_{j\neq i}\mathcal{S}_j$. Note that (4) is a linear transform of λ^π in (2).

Local Utility. Let $S_i = |\mathcal{S}_i|$ denote the number of local states and $A_i := |\mathcal{A}_i|$ the number of local actions. For agent i, define the local utility function $F_i(\cdot) : \mathbb{R}^{S_i A_i} \mapsto \mathbb{R}$ as a function of $\lambda^\pi_{(i)}$, depends on θ_i when agent i follows policy π_{θ_i}. Then, define the global utility as the sum of local ones:

$$R(\pi_\theta) = F(\lambda^{\pi_\theta}) := \frac{1}{N}\sum_{i=1}^{N} F_i\big(\lambda^{\pi_\theta}_{(i)}\big). \tag{5}$$

Note that (5) is *not node-separable*, and local occupancy measures *depend on the global one* through (4). This means that the policy parameters θ_i of agent i depends on global policy π, and hence on global parameter $\theta = (\theta_1, \theta_2, \cdots, \theta_N)$. This is a key point of departure from standard multi-agent optimization (Nedic and Ozdaglar 2009). Next we shift to deriving a variant of actor-critic that is attuned to the multi-agent setting with general utilities (5).

Elements of MARL with General Utilities

This section develops an actor-critic type algorithm for MARL with general utilities (5). One challenge is that the occupancy measure, the policy parameters, and the utility are coupled. Specifically, the value function is not additive across trajectories, and hence invalidates RL approaches tailored to maximizing cumulative returns based upon either the Policy Gradient Theorem (Williams 1992; Sutton et al. 2000) or Bellman's equation (Puterman 2014). To address this issue, we employ a combination of the chain rule, an additional density estimation step, and the construction of a "shadow reward." We first define the shadow reward and value function as follows and then will proceed towards the proposed algorithm.

Shadow Rewards and Policy Evaluation

The general utility objective cannot be written as cumulative sum of returns. The nonlinearity invalidates the additivity, which is the origination of the definition of the conventional reward function and Q function, quantities that are central to approaches for maximizing cumulative-returns, via either dynamic programming (Puterman 2014) or policy search (Williams 1992; Sutton et al. 2000). To circumvent the need for additivity, we will introduce auxiliary variables, which we call shadow rewards and shadow Q functions.

Definition 1 (Shadow Reward and Shadow Q Function)**.** *The shadow reward $r_\pi : \mathcal{S}\times\mathcal{A}\mapsto\mathbb{R}$ of policy π w.r.t. general utility F is $r^\pi(s,a) := \frac{\partial F(\lambda^\pi)}{\partial\lambda(s,a)}$, with associated shadow Q function $Q^\pi_F(s,a) := \mathbb{E}\big[\sum_{t=0}^{+\infty}\gamma^t \cdot r^\pi(s^t,a^t)\,\big|\,s^0 = s, a^0 = a, \pi\big]$.*

To understand these definitions, consider linearizing (differentiating) general utility F with respect to λ^π. The linearized problem, via the chain rule, is equivalent to a MDP with cumulative return, with the shadow reward and Q function in place of the usual reward and Q functions:

$$\nabla_\theta F(\lambda^{\pi_\theta}) = \mathbb{E}\left[\sum_{t=0}^{+\infty}\gamma^t Q^\pi_F(s^t,a^t)\nabla_\theta\log\pi_\theta(a^t|s^t)\Big|s_0\sim\xi,\pi\right]. \tag{6}$$

This expression for the policy gradient illuminates the centrality of the shadow reward/value function for nonlinear functions of the occupancy measure (2), which motivates the generalized policy evaluation scheme we present next.

Policy Evaluation Criterion. We shift to how one may compute the Shadow Q-function from trajectory information, upon the basis of which we can estimate the parameters of a critic. To do so, we use function approximation to parameterize the high-dimensional shadow Q-function. One simple choice is linear function approximation. That is, given a set of feature vectors $\{\phi(s,a)\in\mathbb{R}^d : s\in\mathcal{S}, a\in\mathcal{A}\}$, we want to find some weight parameter $w\in\mathbb{R}^d$ so that

$$Q_w(s,a) := \langle\phi(s,a), w\rangle \quad \forall(s,a)\in\mathcal{S}\times\mathcal{A}. \tag{8}$$

In our algorithm, we will update a sequence of $\hat{w}$ to closely approximate the sequence of implicit shadow Q functions, as policy gets updated. In practice, the parametrization (8) needs not be linear. Indeed, experimentally, we consider Q defined by a multi-layer neural network.

Thus, the critic objective of policy π is defined as the mean-square-error w.r.t. shadow Q-function:

$$\ell(w;\pi) := \mathbb{E}\left[\sum_{t=0}^{\infty}\frac{\gamma^t}{2}\big(Q_w(s^t,a^t) - Q^\pi_F(s^t,a^t)\big)^2\Big|s^0\sim\xi,\pi\right]$$
$$= \frac{1}{2}\sum_{s,a}\lambda^\pi(s,a)\big(\phi(s,a)^\top w - Q^\pi_F(s,a)\big)^2. \tag{9}$$

Algorithm 1: Decentralized **S**hadow Reward **A**ctor-**C**ritic (DSAC)

Input: initial policy θ^0; actor step-sizes $\{\eta_\theta^k\}$; Batch sizes $\{B_k\}$; Episode lengths $\{H_k\}$; initial critic $W^0 := [w_1^0, w_2^0, ..., w_N^0] \in \mathbb{R}^d$ with $w_i^0 = w_j^0$, for all i, j; critic step-size $\{\eta_w^k\}$; mixing matrix $M \in \mathbb{R}_+^{N\times N}$; mixing round $m \geq 1$.

for *iteration* $k = 0, 1, 2, ...$ **do**

- Perform B_k Monte Carlo rollouts to obtain trajectories $\tau = \{s^0, a^0, \cdots, s^{H_k}, a^{H_k}\}$ with initial dist. ξ, policy π_{θ^k} collected as batch $\mathcal{B}_k$.
- **for** *agent* $i = 1, 2, ..., N$ **do**
  - Compute empirical local occupancy measure
  $$\hat{\lambda}_i^k = \frac{1}{B_k}\sum_{\tau\in\mathcal{B}_k}\sum_{t=0}^{H_k}\gamma^t\cdot\mathbf{e}\left(s_{(i)}^t, a_{(i)}^t\right). \quad (7)$$
  - Estimate shadow reward $\hat{r}_i^k = \nabla_{\lambda_i}F_i(\hat{\lambda}_i^k)$.
- **for** *agent* $i = 1, 2, ..., N$ **do**
  - With localized policy gradient estimate $G_{w_i}(\tau, r_i, w_i) = \sum_{t=0}^{H}\gamma^t\cdot(Q_{w_i}(s^t, a^t) - \hat{Q}_i^t)\cdot\nabla_{w_i}Q_{w_i}(s^t, a^t)$, compute
  $$\widehat{\Delta}_{w_i}^k = \frac{1}{B_k}\sum_{\tau\in\mathcal{B}_k}G_{w_i}(\tau, \hat{r}_i^k, w_i^k)\ , w_i^{k+1} = w_i^k - \eta_w^k\widehat{\Delta}_{w_i}^k.$$
- **for** $iter = 1, ..., m$ **do**
  - **for** *agent* $i = 1, 2, ..., N$ **do**
    - Exchange information with neighbours: $w_i^{k+1} = \sum_{\{j:(j,i)\in\mathcal{E}\}}M(j,i)\cdot w_i^{k+1}$.
- **for** *agent* $i = 1, 2, ..., N$ **do**
  - With $G_{\theta_i}(\tau, w_i) = \sum_{t=0}^{H}\gamma^t Q_{w_i}(s^t, a^t)\nabla_{\theta_i}\log\pi_{\theta_i}^{(i)}(a_{(i)}^t|s^t)$, update the policy:
  $$\widehat{\Delta}_{\theta_i}^k := \frac{1}{B_k}\sum_{\tau\in\mathcal{B}_k}G_{\theta_i}(\tau, w_i^{k+1})\ , \theta_i^{k+1} = \theta_i^k + \eta_\theta^k\widehat{\Delta}_{\theta_i}^k.$$

Via the definition of the occupancy measure λ^π [cf. (2)], the expectation may be substituted by weighting factors in the summand on the second line. We assume features $\{\phi(s,a)\}_{s\in\mathcal{S},a\in\mathcal{A}}$ are bounded (see (Zhang et al. 2021) for details). With the shadow reward and associated Q-function (Definition 1), the policy evaluation criterion (9), and its smoothness properties with respect to critic parameters w in place, we expand on their role in the multi-agent setting.

Multi-Agent Optimization for Critic Estimation

Setting aside the issue of policy parameter updates for now, we focus on estimating the global general utility. The shadow Q function and shadow reward (Definition 1) depend on global knowledge of all local utilities, which are unavailable as local incentives are local only. To mitigate this issue, we introduce their localized components, which together comprise the global shadow Q function and reward. Specifically, define the local shadow reward r_i^π for agent i:

$$r_i^\pi(s_{(i)}, a_{(i)}) := \frac{\partial F_i(\lambda_{(i)}^\pi)}{\partial\lambda_{(i)}(s_{(i)}, a_{(i)})}\ , \forall(s_{(i)}, a_{(i)}) \in \mathcal{S}_i \times \mathcal{A}_i. \quad (10)$$

Clearly, it holds that $r^\pi(s,a) = \frac{1}{N}\sum_{i=1}^N r_i^\pi(s_{(i)}, a_{(i)})$. Based on the local observation of the its own shadow reward, agent i may access its local shadow Q function $Q : \mathcal{S}\times\mathcal{A}\to\mathbb{R}$:

$$Q_i^\pi(s,a) := \mathbb{E}\left[\sum_{t=0}^{+\infty}\gamma^t\cdot r_i^\pi\left(s_{(i)}^t, a_{(i)}^t\right) \mid s^0 = s, a^0 = a, \pi\right], \quad (11)$$

for $\forall(s,a)\in\mathcal{S}\times\mathcal{A}$. Therefore, we also have $Q_F^\pi(s,a) = \frac{1}{N}\sum_{i=1}^N Q_i^\pi(s,a)$. Then, each agent i seeks to estimate common critic parameters w that well-represent its shadow Q function in the sense of minimizing the global mean-square error (9). By exploiting the aforementioned node-separability and introducing a localized critic parameter vector w_i associated to agent i, this may equivalently be expressed as a consensus optimization problem (Nedic and Ozdaglar 2009):

$$\min_{\{w_i\}_{i=1}^N}\frac{1}{N}\sum_{i=1}^N\ell_i(w_i;\pi)\ \text{ s.t. }\ w_i = w_j, (i,j)\in\mathcal{E}\ , \quad (12)$$

where the local policy evaluation criterion is defined as $\ell_i(w_i;\pi) := \mathbb{E}\left[\sum_{t=0}^\infty\frac{\gamma^t}{2}\left(Q_{w_i}(s^t,a^t) - Q_{F_i}^\pi(s^t,a^t)\right)^2\middle|s^0\sim\xi,\pi\right]$. This formulation allows agent i to evaluate its policy with respect to global utility (5) through the local criterion $\ell_i(w_i;\pi)$ as a surrogate for that which aggregates global information (9), when consensus over local parameters w_i is imposed. Next, we incorporate solutions to (12) into the critic step together with a policy parameter θ_i update along stochastic ascent directions via (6) for the actor to assemble DSAC.

Decentralized Shadow Reward Actor-Critic

Next, we put together these pieces to present **D**ecentralized **S**hadow Reward **A**ctor-**C**ritic (DSAC) as Algorithm 1. This scheme allows agents to keep their local utilities F_i, and policies π_{θ_i} with associated parameters θ_i private. The agents share a common function approximator for the shadow Q function. Further, they retain local copies w_i of the shadow critic parameters, which they communicate to neighbors according to the network structure defined by edge set $\mathcal{E}$ and mixing matrix M to be subsequently specified. Algorithm 1 proceeds in four stages: (i) density estimation step for to obtain the shadow reward; (ii) shadow critic updates; (iii) information mixing via weighted averaging; and (iv) actor updates. Each step is detailed in Algorithm 1.

Consistency and Sample Complexity

In this section, we study the finite sample performance of Algorithm 1. We show $\tilde{\mathcal{O}}(\epsilon^{-2.5})$ (Theorem 2) or $\tilde{\mathcal{O}}(\epsilon^{-2})$

(Corollary 2) sample complexities to obtain ϵ-stationary points of global utility, depending on the number of communications per step, akin to best known rates for non-concave expected maximization problems (Shapiro, Dentcheva, and Ruszczyński 2014). We also establish the nonexistance of spurious extrema for this setting, indicating the convergence to global optimality (Corollary 1). Before continuing, we present a few key technical conditions for the utility F, the policy π_θ, the mixing matrix M, and the critic approximation. The other assumptions are stated in Appendix B.2 of the supplementary material (see (Zhang et al. 2021) for details).

Assumption 1. *For utility F [cf. (5)], we assume for $\forall i$ that:*

(i). $F_i(\cdot)$ *is private to agent i.*

(ii). $\exists C_F > 0$ *s.t.* $\|\nabla_{\lambda_{(i)}} F_i(\lambda_{(i)})\|_\infty \le C_F$ *in a neighbourhood of the occupancy measure set.*

(iii). $\exists L_\lambda > 0$ *s.t.* $\|\nabla_{\lambda_{(i)}} F_i(\lambda_{(i)}) - \nabla_{\lambda_{(i)}} F_i(\lambda'_{(i)})\|_\infty \le L_\lambda \|\lambda_{(i)} - \lambda'_{(i)}\|$.

(iv). $\exists L_\theta > 0$ *s.t.* $F \circ \lambda(\cdot)$ *is* L_θ*-smooth.*

Assumption 2. *For π_θ and the occupancy measure λ^{π_θ}, we assume:*

(i). *The local policy* $\pi^{(i)}_{\theta_i}$ *is private to each agent i.*

(ii). $\exists\, C_\pi > 0$ *s.t. for each agent i, its score function is bounded:* $\|\nabla_{\theta_i} \log \pi^{(i)}_{\theta_i}(a_{(i)}|s)\| \le C_\pi$, *for* $\forall\, \theta$ *and* $\forall (s,a)$.

(iii). $\exists\, \ell_\theta > 0$ *s.t.* $\|\lambda^{\pi_\theta} - \lambda^{\pi_{\theta'}}\| \le \ell_\theta \|\theta - \theta'\|$.

Assumption 3. *The mixing matrix M is a doubly stochastic matrix satisfying:*

(i). $M \in \mathbb{S}^{N\times N}_+$, $M(i,j) > 0$ *iff.* $(i,j) \in \mathcal{E}$.

(ii). $M \cdot \mathbf{1}_N = \mathbf{1}_N$, *where* $\mathbf{1}_N \in \mathbb{R}^N$ *is an all-ones vector.*

(iii). *Let the eigenvalues of M be* $1 = \sigma_1(M) > \sigma_2(M) \ge \cdots \ge \sigma_N(M)$. *We define* $\rho := \max\{|\sigma_2(M)|, |\sigma_N(M)|\} < 1$.

Assumption 4. *For $\forall\theta$, define the optimal critic parameter* $w^*(\theta) := \operatorname{argmin}_w \frac{1}{N}\sum_{i=1}^N \ell_i(w;\pi_\theta)$. *We assume that* $\exists W > 0$ *s.t.* $E_\theta^2 = \sum_{i=1}^N \|\nabla_{\theta_i} F(\lambda^{\pi_\theta}) - \Delta_{\theta_i}\|^2 \le W$, *for* $\forall\theta$, *where* $\Delta_{\theta_i} := \mathbb{E}\big[\sum_{t=0}^{+\infty} \gamma^t \cdot Q_{w^*(\theta)}(s^t,a^t) \cdot \nabla_{\theta_i} \log \pi^{(i)}_{\theta_i}(a^t|s^t)\, \big| s^0 \sim \xi, \pi_\theta\big]$ *is the PG estimate under* $w^*(\theta)$.

Assumption 1 requires the boundedness and Lipschitz continuity of the gradient of the utility function. Assumption 2 ensures that the score function is bounded, and the occupancy measure is Lipschitz w.r.t. the policy parameters. These conditions are common to RL algorithms focusing on occupancy measures in recent years (Hazan et al. 2019; Zhang et al. 2020b), and are automatically satisfied by common policies such as the softmax. Assumption 3 holds for any undirected connected loop-free static graph (Chung and Graham 1997). Assumption 4 states that the feature misspecification error is uniformally upper bounded by W.

Next, we present a brief proof sketch with details provided in the appendices.

Step 1. We begin by a standard stochastic gradient ascent analysis (Lemma **??**), which yields:

$$F(\lambda^{\pi_{\theta^{k+1}}}) - F(\lambda^{\pi_{\theta^k}}) \ge \frac{\eta_\theta^k}{4} \left\|\nabla_\theta F(\lambda^{\pi_{\theta^k}})\right\|^2 - \frac{3\eta_\theta^k}{4} \sum_{i=1}^N \left\| \nabla_{\theta_i} F(\lambda^{\pi_{\theta^k}}) - \widehat{\Delta}^k_{\theta_i} \right\|^2. \tag{13}$$

Step 2. We provide high probability bounds for gradient estimation errors:

$$\sum_{i=1}^N \left\|\nabla_{w_i}\ell_i(w_i^k;\pi_{\theta^k}) - \widehat{\Delta}^k_{w_i}\right\|^2 \le \mathcal{O}(B_k^{-1}) \quad \text{and} \tag{14}$$

$$\sum_{i=1}^N \left\|\nabla_{\theta_i} F(\lambda^{\pi_{\theta^k}}) - \widehat{\Delta}^k_{\theta_i}\right\|^2 \le \mathcal{O}\Big(B_k^{-1} + \sum_{i=1}^N \|w_i^{k+1} - w_*^{k+1}\|^2\Big),$$

where $w^*_{k+1} := \operatorname{argmin}_w \frac{1}{N}\sum_{i=1}^N \ell_i(w;\pi_{\theta^k})$ is the ideal critic variable at policy π_{θ^k} (Lemma **??**).

Step 3. By analyzing the consensus error of the communication steps and the gradient descent step for the critic update, we bound critic fitting error term $\sum_{i=1}^N \|w_i^{k+1} - w_*^{k+1}\|^2$ with high probability:

$$\|\bar{w}^{k+1} - w_*^{k+1}\|^2 \le (1-c)\|\bar{w}^k - w_*^k\|^2 + \mathcal{O}(B_k^{-1}) + \mathcal{O}\Big(\sum_{i=1}^N \|w_i^k - \bar{w}^k\|^2\Big) + \text{other controllable noise}, \tag{15}$$

where $c \in (0,1)$ is some constant, $\bar{w}^k = \sum_{i=1}^N w_i^k/N$ is the average critic variable in the k-the iteration, and $\sum_{i=1}^N \|w_i^k - \bar{w}^k\|^2 \le \mathcal{O}\big(\big(\rho^m \sum_{k'=0}^k \eta_w^{k'} \rho^{m(k-k')}\big)^2\big)$. See Lemmas **??** - **??**.

Step 4. Next, we construct the following potential function with a carefully selected constant α as to enable the convergence analysis:

$$R_k := F(\lambda^{\pi_{\theta^k}}) - \alpha\|\bar{w}^k - w_*^k\|^2. \tag{16}$$

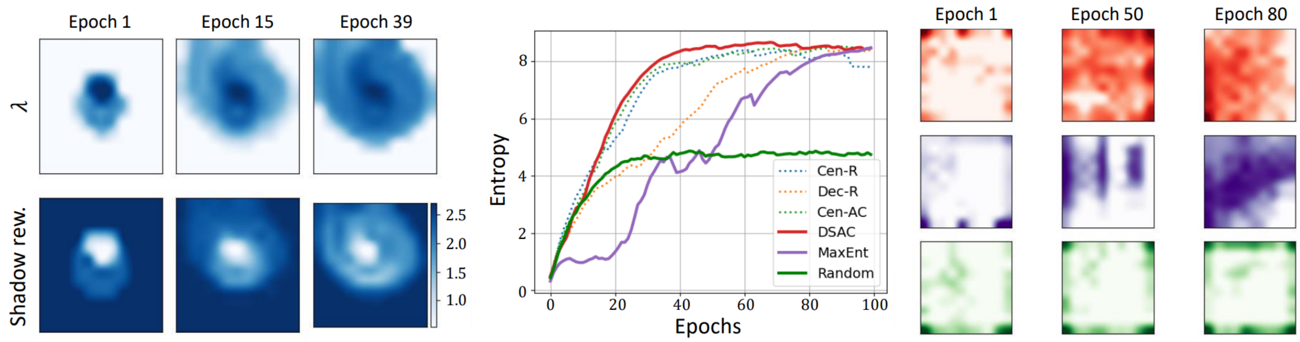
Taking the advantage of the contraction property of $\|\bar{w}^k - w_*^k\|^2$ and this specific potential function, we characterize algorithm performance in terms of optimization error, the feature mis-specification error, the stochastic PG approximation error, and the multi-agent consensus error. See Lemma **??**.

Combining the above steps and suitably specify the parameters, we have the final theorem.

Theorem 2. *Under Assumption* **??**, *1, 2,* **??** *and 3, with one communication round per iteration, i.e. $m = 1$, Algorithm 1 satisfies, under the following parameter selections:*
(i) For final iteration $T = \mathcal{O}(\epsilon^{-1.5})$, *trajectory lengths* $H_k \equiv \mathcal{O}(\log(1/\epsilon)/1-\gamma)$, $\delta_k \equiv \delta/(3N(T+1))$, $\delta \in (0,1)$, *batch sizes* $B_k \equiv \log(1/\delta_k)\epsilon^{-1}$, *constant step-sizes* $\eta_w = \mathcal{O}(\sqrt{\epsilon})$, $\eta_\theta = \min\Big\{\frac{(1-\gamma)\mu_w\eta_w}{C_w C_\phi C_\pi} \cdot \frac{1}{\max\{4\sqrt{3N}, 6\sqrt{10}\}}, \frac{1}{4L_\theta}\Big\} = \mathcal{O}(\sqrt{\epsilon})$, *then*

$$\frac{1}{T}\sum_{k=1}^T \|\nabla_\theta F(\lambda^{\pi_{\theta^k}})\|^2 \le \mathcal{O}(\epsilon + W)\,. \quad w.p. \quad 1-\delta$$

(ii) For unspecified final iteration T, we adaptively set: $\delta_k = \frac{2\delta}{N\pi^2(k+1)^2}$, $\delta \in (0,1)$, *trajectory lengths* $H_k = \mathcal{O}((1 -$



(a) Shadow reward (b) Entropy comparison (c) State space coverage

Figure 1: (a) Occupancy measure (first row) and shadow reward (second row) for *MountainCar* environment. Each subplot represents a heatmap for two dimentional state space. Observe that over the course of training the measure and shadow reward's coverage of the state and action spaces grows, as a consequence of selecting actions towards maximizing the entropy. (b) Comparison for exploration maximization in a *cooperative multiagent* environment, (c) Agent 1 marginalized occupancy measure. For the DSCA implementation, each agents needs to estimate only 100 dimensional marginalized occupancy measure while for the centralized counterpart MaxEnt, we need to estimate 10^4 dimensional occupancy measure making it slow in practice.

$\gamma)^{-1}\log(k+1))$, *batchsizes* $B_k=\log(1/\delta_k)(k+1)^{\frac{2}{3}}$, *and step-sizes* $\eta_\theta^k=\min\Big\{\frac{(1-\gamma)\mu_w\eta_w^{k+1}}{C_wC_\phi C_\pi}\cdot\frac{1}{\max\{4\sqrt{3N},6\sqrt{10}\}},\frac{1}{4L_\theta}\Big\}$, $\eta_w^k=\min\{(k+1)^{-\frac{1}{3}},L_w^{-1}\}$, *then*

$$\frac{\sum_{k=1}^T\eta_\theta^k\big\|\nabla_\theta F(\lambda^{\pi_{\theta^k}})\big\|^2}{\sum_{k=1}^T\eta_\theta^k}\leq\mathcal{O}\left(\frac{\log T}{T^{\frac{2}{3}}}+W\right),\quad w.p.\quad 1-\delta$$

In either case, Algorithm 1 requires $\tilde{\mathcal{O}}(\epsilon^{2.5})$ *samples to satisfy* $\frac{\sum_{k=1}^T\eta_\theta^k\|\nabla_\theta F(\lambda^{\pi_{\theta^k}})\|^2}{\sum_{k=1}^T\eta_\theta^k}\leq\mathcal{O}(\epsilon+W)$.

Next, we establish that for concave general utilities (1), there are no spurious stationary points.

Corollary 1 (Convergence to global optimality)**.** *Suppose F is concave, and the shadow Q function Q_F is realizable, i.e., $W=0$ in Assumption 4. For π_θ satisfying Assumption 1 of (Zhang et al. 2020b), every stationary point is a global optimizer. In Theorem 2(ii), if we further let $\bar{\theta}_T$ be the parameter randomly chosen from $\{\theta^k\}_{k=1}^T$ where $\bar{\theta}_T=\theta^k$ w.p. $\eta_\theta^k/(\sum_{k'=1}^T\eta_\theta^{k'})$, then $\lim_{T\to\infty}\mathbb{E}[\|\nabla_\theta F(\lambda^{\pi_{\bar{\theta}_T}})\|^2]=0$ w.p. $1-\delta$. Thus, Algorithm 1 converges to the set of global optimizers.*

Next we spotlight the role of the number of communication steps in the convergence rate.

Corollary 2 (Multiple-round communication)**.** *Suppose multiple-round communication is allowed, i.e., $m>1$. Under the same parameter selections as Theorem 2(i), while setting final iteration index $T=\epsilon^{-1}$, communication rounds $m=\mathcal{O}((1-\rho)^{-1}\log(\epsilon^{-1}))$, and the step-sizes* $\eta_\theta^k\equiv\min\Big\{\frac{(1-\gamma)\mu_w/L_w}{C_wC_\phi C_\pi}\cdot\frac{1}{\max\{4\sqrt{3N},6\sqrt{10}\}},\frac{1}{4L_\theta}\Big\}$, $\eta_w^k\equiv L_w^{-1}$, *then the total sample complexity is* $\mathcal{O}(\epsilon^{-2})$.

Namely, with additional communication rounds $m=\mathcal{O}((1-\rho)^{-1}\log(\epsilon^{-1}))$ per iteration, the convergence rate refines from $\mathcal{O}(\epsilon^{-2.5})$ to $\mathcal{O}(\epsilon^{-2})$. Next, we investigate the experimental merit of the proposed approach for giving rise to emergent teamwork among multiple agents.

Experimental Results

We experimentally investigate the merit of Algorithm 1 in the context of both single and multi-agent problems. The single-node case ($N=1$) bears investigation as the proposed scheme is a new way to solve RL problems with general utilities relative to (Zhang et al. 2020b). For this case, we consider the continuous *MountainCar* environment of OpenAI Gym (Brockman et al. 2016).

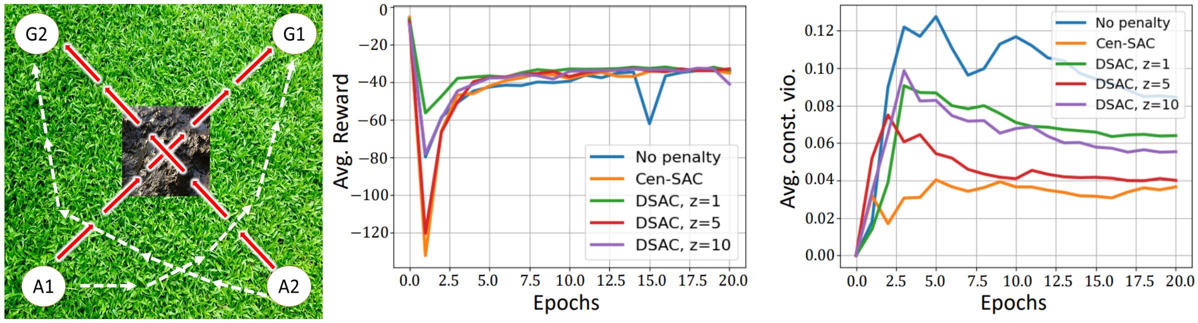
Concept of Shadow Reward

To understand the concept of shadow reward, we experiment with the single-agent setup. We consider the **exploration maximization** problem for the *MountainCar* environment in which the two dimensional continuous state space is divided into $[12,11]$ grid size. We run the proposed algorithm for 40 epochs and then plot the count based occupancy measure estimate in the first row of Fig. 1(a). In the figure, light color denotes lower value and dark color represent the higher values as shown in the colorbar. We see that as we go from epoch 1 to epoch 39, the algorithm yields occupancy measures that better cover the state space, which is achieved by the special structure of the "shadow reward" we define as a by-product of the general utility.

Multi-Agent Experiments

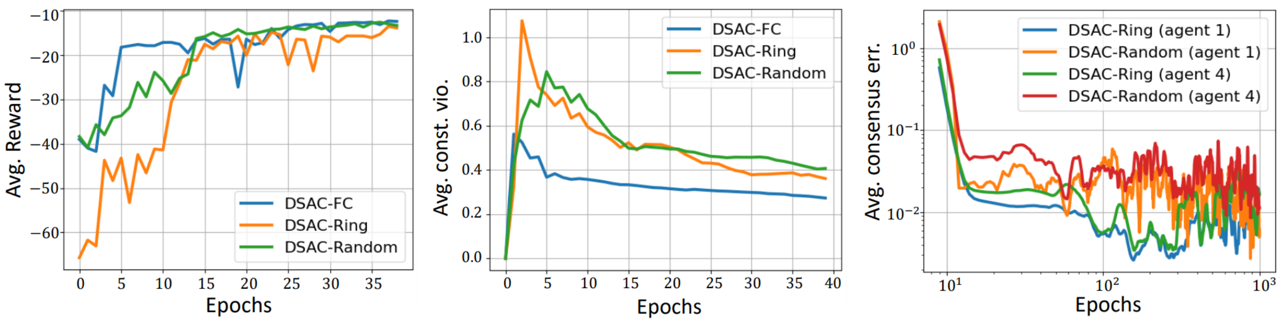
For multi-agent problems, we experiment with $N\geq 2$ agents moving in a two-dimensional continuous space associated with the problem of *Cooperative navigation* (Lowe et al. 2017).

Exploration Maximization. We consider a variant of the *cooperative navigation* multi-agent environment provided in (Lowe et al. 2017) for $N=2$ agents. The goal of maximum entropy exploration in the multi-agent setting is one in which all agents in the network seek to cover the unknown space, whereby their local utility is the entropy in (5) is given by $F_i(\lambda_{(i)}^\pi)=-\sum_{s(i)}\lambda_{(i)}^\pi(s_{(i)})\cdot\log(\lambda_{(i)}^\pi(s_{(i)}))$.



(a) World model (b) Average return (c) Average cost

Figure 2: (a) Two agent safe navigation with green safe and brown unsafe state space. The goal is to reach G1 and G2 safely from the starting positions A1 and A2, respectively. (b) Undiscounted average reward comparison and (c) average constraint violation comparison for different values of penalty parameter z. Observe that imposing constraints allows agents to avoid collision and the unsafe region, while effectively reaching the goals more often.



(a) Average return (b) Average cost (c) Consensus error

Figure 3: Safe navigation in a multi-agent cooperative environment with 4 agents and 4 landmarks. Note that the state space in this case would be 16 dimensional (location of agent and landmarks). We run this experiment for three different communication graphs among agents; *fully connected (FC) (all the agents are connected to each other)*, *ring* (all the agents are connected using ring topology), and *random* (where agents are randomly using Erdős-Rényi random graph model). (a) Running average of the reward return, (b) running average of the constraint violation, and (c) running average of the consensus error for agent 1 and agent 4 for *ring* and *random* network connectivity.

We compare DSAC against its corresponding centralized implementations (Cen-AC) or a variant that uses Monte-Carlo rollouts (Cen-R, Dec-R), as well as existing Max-Ent (Hazan et al. 2019) in Fig.1(b)-1(c). Observe that Max-Ent does not achieve comparable performance, and DSAC achieves comparable performance to its variants that require centralization. Fig. 1(c) visualizes the heatmap of the marginalized measure at agent 1 for DSAC (red) at different epochs as compared to MaxEnt (purple) and random baseline (green) – note the superior space coverage of DSAC (red).

Safe Cooperative Navigation. We consider a two agent cooperative environment from (Lowe et al. 2017) where each agent needs to reach its assigned goal while traversing only through the safe region as visualized in Fig.2(a). Agents receive a negative reward proportional to its distance from the landmark, and an additional negative reward of -1 if agents collide. Additionally, each agents receive a high cost of $c = 1$ if it passes through the unsafe region (middle of the state space) – see Fig. 2(a). We impose safety via the constraint for each agent $\langle \lambda_i^\pi, c\rangle \leq C$ where λ_i^π in the marginalized occupancy measure, and including the constraint as a quadratic penalty in a manner similar to (**??**) (see (Zhang et al. 2021) for further details). To solve this problem, we compare the performance of DSAC for various values of its penalty parameter z to its centralized variant, and a version of multi-agent actor-critic that only ignores the cost. Results for the average reward and constraint violation, respectively, are given in Fig. 2(b)-2(c). The decentralized DSAC achieves comparable performance to its centralized variant, and outperforms existing alternatives, yielding effective learned behaviors for navigation in team settings. Demonstrations for larger networks with different connectivities are in Figure 3.

1 Conclusions

We contributed a conceptual basis for defining agents' behavior in cooperative MARL beyond the cumulative return via nonlinear functions of their occupancy measure. This motivates defining "shadow rewards" and DSAC, whose critic employs shadow value functions and weighted averaging. Its consistency and sample complexity was rigorously established. Further, experiments illuminated the upsides of general utilities for teams. Future work includes improving communications and sample efficiencies, connections to meta-learning, and allowing information asymmetry.

References

Argall, B. D.; Chernova, S.; Veloso, M.; and Browning, B. 2009. A survey of robot learning from demonstration. *Robotics and autonomous systems*, 57(5): 469–483.

Bachrach, Y.; Everett, R.; Hughes, E.; Lazaridou, A.; Leibo, J. Z.; Lanctot, M.; Johanson, M.; Czarnecki, W. M.; and Graepel, T. 2020. Negotiating team formation using deep reinforcement learning. *Artificial Intelligence*, 288: 103356.

Başar, T.; and Olsder, G. J. 1998. *Dynamic noncooperative game theory*. SIAM.

Bernstein, D. S.; Givan, R.; Immerman, N.; and Zilberstein, S. 2002. The complexity of decentralized control of Markov decision processes. *Mathematics of operations research*, 27(4): 819–840.

Borkar, V. S.; and Meyn, S. P. 2002. Risk-sensitive optimal control for Markov decision processes with monotone cost. *Mathematics of Operations Research*, 27(1): 192–209.

Boyd, S.; Ghosh, A.; Prabhakar, B.; and Shah, D. 2006. Randomized gossip algorithms. *IEEE transactions on information theory*, 52(6): 2508–2530.

Boyd, S.; Parikh, N.; and Chu, E. 2011. *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc.

Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; and Zaremba, W. 2016. Openai gym. *arXiv preprint arXiv:1606.01540*.

Busoniu, L.; Babuska, R.; and De Schutter, B. 2008. A comprehensive survey of multiagent reinforcement learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(2): 156–172.

Chen, T.; Zhang, K.; Giannakis, G. B.; and Başar, T. 2018. Communication-efficient distributed reinforcement learning. *arXiv preprint arXiv:1812.03239*.

Chung, F. R.; and Graham, F. C. 1997. *Spectral graph theory*. 92. American Mathematical Soc.

Claus, C.; and Boutilier, C. 1998. The dynamics of reinforcement learning in cooperative multiagent systems.

Doan, T.; Maguluri, S.; and Romberg, J. 2019. Finite-Time Analysis of Distributed TD (0) with Linear Function Approximation on Multi-Agent Reinforcement Learning. In *International Conference on Machine Learning*, 1626–1635.

Eccles, T.; Bachrach, Y.; Lever, G.; Lazaridou, A.; and Graepel, T. 2019. Biases for emergent communication in multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, 13111–13121.

Feinberg, E. A. 2016. Optimality conditions for inventory control. In *Optimization Challenges in Complex, Networked and Risky Systems*, 14–45. INFORMS.

Foerster, J.; Assael, I. A.; De Freitas, N.; and Whiteson, S. 2016. Learning to communicate with deep multi-agent reinforcement learning. *Advances in neural information processing systems*, 29: 2137–2145.

Foerster, J.; Nardelli, N.; Farquhar, G.; Afouras, T.; Torr, P. H.; Kohli, P.; and Whiteson, S. 2017. Stabilising experience replay for deep multi-agent reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 1146–1155.

Gupta, T.; Mahajan, A.; Peng, B.; Böhmer, W.; and Whiteson, S. 2020. UneVEn: Universal Value Exploration for Multi-Agent Reinforcement Learning. *arXiv preprint arXiv:2010.02974*.

Hazan, E.; Kakade, S.; Singh, K.; and Van Soest, A. 2019. Provably efficient maximum entropy exploration. In *International Conference on Machine Learning*, 2681–2691. PMLR.

Huang, Y.; and Kallenberg, L. C. M. 1994. On Finding Optimal Policies for Markov Decision Chains: A Unifying Framework for Mean-Variance-Tradeoffs. *Mathematics of Operations Research*, 19(2): 434–448.

Jaques, N.; Lazaridou, A.; Hughes, E.; Gulcehre, C.; Ortega, P.; Strouse, D.; Leibo, J. Z.; and De Freitas, N. 2019. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. In *International Conference on Machine Learning*, 3040–3049. PMLR.

Kallenberg, L. C. M. 1994. Survey of linear programming for standard and nonstandard Markovian control problems. Part I: Theory. *Zeitschrift für Operations Research*, 40(1): 1–42.

Kar, S.; Moura, J. M.; and Poor, H. V. 2013. QD-Learning: A Collaborative Distributed Strategy for Multi-Agent Reinforcement Learning Through Consensus+ Innovations. *IEEE Transactions on Signal Processing*, 61(7): 1848–1862.

Kober, J.; Bagnell, J. A.; and Peters, J. 2013. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 32(11): 1238–1274.

Kohler, J. M.; and Lucchi, A. 2017. Sub-sampled cubic regularization for non-convex optimization. *arXiv preprint arXiv:1705.05933*.

Konda, V. R.; and Borkar, V. S. 1999. Actor-Critic–Type Learning Algorithms for Markov Decision Processes. *SIAM Journal on Control and Optimization*, 38(1): 94–123.

Konda, V. R.; and Tsitsiklis, J. N. 2000. Actor-critic algorithms. In *Advances in Neural Information Processing Systems*, 1008–1014.

Koppel, A.; Jakubiec, F. Y.; and Ribeiro, A. 2015. A saddle point algorithm for networked online convex optimization. *IEEE Transactions on Signal Processing*, 63(19): 5149–5164.

Krishnamurthy, V. 2016. *Partially observed Markov decision processes*. Cambridge University Press.

Le, H. M.; Yue, Y.; Carr, P.; and Lucey, P. 2017. Coordinated Multi-Agent Imitation Learning. *Proceedings of Machine Learning Research*, 70: 1995–2003.

Lee, D.; He, N.; Kamalaruban, P.; and Cevher, V. 2020. Optimization for Reinforcement Learning: From a single agent to cooperative agents. *IEEE Signal Processing Magazine*, 37(3): 123–135.

Lee, D.; Yoon, H.; Cichella, V.; and Hovakimyan, N. 2018. Stochastic primal-dual algorithm for distributed gradient temporal difference learning. *arXiv preprint arXiv:1805.07918*.

Lee, H.-R.; and Lee, T. 2019. Improved cooperative multi-agent reinforcement learning algorithm augmented by mixing demonstrations from centralized policy. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, 1089–1098.

Lee, J. W.; Zhang, B.-T.; et al. 2002. Stock Trading System Using Reinforcement Learning with Cooperative Agents. In *Proceedings of the Nineteenth International Conference on Machine Learning*, 451–458.

Littman, M. L. 1994. Markov games as a framework for multi-agent reinforcement learning. In *Machine learning proceedings 1994*, 157–163. Elsevier.

Lowe, R.; Wu, Y.; Tamar, A.; Harb, J.; Abbeel, P.; and Mordatch, I. 2017. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. *Neural Information Processing Systems (NIPS)*.

Mahajan, A.; and Mannan, M. 2016. Decentralized stochastic control. *Annals of Operations Research*, 241(1-2): 109–126.

Mahajan, A.; Rashid, T.; Samvelyan, M.; and Whiteson, S. 2019. Maven: Multi-agent variational exploration. In *Advances in Neural Information Processing Systems*, 7613–7624.

Mystery, M. 2021. RMIX: Risk-Sensitive Multi-Agent Reinforcement Learning. *Under Review at International Conference on Learning Representations*.

Nayyar, A.; Mahajan, A.; and Teneketzis, D. 2013. Decentralized stochastic control with partial history sharing: A common information approach. *IEEE Transactions on Automatic Control*, 58(7): 1644–1658.

Nedic, A.; and Ozdaglar, A. 2009. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1): 48–61.

Prashanth, L.; and Ghavamzadeh, M. 2016. Variance-constrained actor-critic algorithms for discounted and average reward MDPs. *Machine Learning*, 105(3): 367–417.

Puterman, M. L. 2014. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.

Qu, C.; Mannor, S.; Xu, H.; Qi, Y.; Song, L.; and Xiong, J. 2019. Value propagation for decentralized networked deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, 1184–1193.

Qu, G.; Wierman, A.; and Li, N. 2020. Scalable reinforcement learning of localized policies for multi-agent networked systems. In *Learning for Dynamics and Control*, 256–266. PMLR.

Rashid, T.; Samvelyan, M.; Schroeder, C.; Farquhar, G.; Foerster, J.; and Whiteson, S. 2018. QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning. In *International Conference on Machine Learning*, 4295–4304.

Shapiro, A.; Dentcheva, D.; and Ruszczyński, A. 2014. *Lectures on stochastic programming: modeling and theory*. SIAM.

Shapley, L. S. 1953. Stochastic games. *Proceedings of the national academy of sciences*, 39(10): 1095–1100.

Shi, W.; Ling, Q.; Wu, G.; and Yin, W. 2015. Extra: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25(2): 944–966.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. 2016. Mastering the game of Go with deep neural networks and tree search. *nature*, 529(7587): 484–489.

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

Sutton, R. S.; McAllester, D. A.; Singh, S. P.; and Mansour, Y. 2000. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*, 1057–1063.

Tanner, H. G.; Jadbabaie, A.; and Pappas, G. J. 2007. Flocking in fixed and switching networks. *IEEE Transactions on Automatic control*, 52(5): 863–868.

Tarbouriech, J.; and Lazaric, A. 2019. Active Exploration in Markov Decision Processes. In *The 22nd International Conference on Artificial Intelligence and Statistics*, 974–982.

Terelius, H.; Topcu, U.; and Murray, R. M. 2011. Decentralized multi-agent optimization via dual decomposition. *IFAC proceedings volumes*, 44(1): 11245–11251.

Vinyals, O.; Babuschkin, I.; Czarnecki, W. M.; Mathieu, M.; Dudzik, A.; Chung, J.; Choi, D. H.; Powell, R.; Ewalds, T.; Georgiev, P.; et al. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature*, 575(7782): 350–354.

Wai, H.-T.; Yang, Z.; Wang, Z.; and Hong, M. 2018. Multi-agent reinforcement learning via double averaging primal-dual optimization. In *Advances in Neural Information Processing Systems*, 9649–9660.

Williams, R. J. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4): 229–256.

Zhang, J.; Bedi, A. S.; Wang, M.; and Koppel, A. 2020a. Cautious Reinforcement Learning via Distributional Risk in the Dual Domain. *arXiv preprint arXiv:2002.12475*.

Zhang, J.; Bedi, A. S.; Wang, M.; and Koppel, A. 2021. MARL with General Utilities via Decentralized Shadow Reward Actor-Critic. *arXiv preprint arXiv:2106.00543*.

Zhang, J.; Koppel, A.; Bedi, A. S.; Szepesvari, C.; and Wang, M. 2020b. Variational policy gradient method for reinforcement learning with general utilities. *Advances in Neural Information Processing Systems*, 33.

Zhang, K.; Miehling, E.; and Başar, T. 2019. Online planning for decentralized stochastic control with partial history sharing. In *2019 American Control Conference (ACC)*, 3544–3550. IEEE.

Zhang, K.; Yang, Z.; Liu, H.; Zhang, T.; and Basar, T. 2018. Fully Decentralized Multi-Agent Reinforcement Learning with Networked Agents. In *International Conference on Machine Learning*, 5872–5881.

Zhao, X.; Xia, L.; Zhang, L.; Ding, Z.; Yin, D.; and Tang, J. 2018. Deep reinforcement learning for page-wise recommendations. In *Proceedings of the 12th ACM Conference on Recommender Systems*, 95–103.