Randomized Linear Programming for Tabular Average-Cost Multi-agent Reinforcement Learning

Invited paper for Session Titled "Reinforcement learning over networks"

Alec Koppel[†], Amrit Singh Bedi[†], Bhargav Ganguly[§], and Vaneet Aggarwal[§]

Abstract—**We focus on multi-agent reinforcement learning in tabular average-cost settings: a team of agents sequentially interacts with the environment and observes localized incentives. The setting we focus on is one in which the global reward is a sum of all local rewards, the joint policy factorizes into agents' marginals, and full observability. To date, exceptionally few global optimality guarantees exist for this simple setting, as most results, asymptotic or non-asymptotic, yield convergence to stationarity under parameterized settings for possibly large/continuous spaces. To strengthen performance guarantees in MARL, we focus on linear programming (LP) reformulations of RL for which stochastic primal-dual method has recently been shown to achieve *optimal sample complexity in the centralized tabular case*. We develop multi-agent LP extensions, whereby agents solve their local saddle point problems and then compose their variable estimates with weighted averaging steps to diffuse information between agents across time. We establish that the number of samples required to attain near-globally optimal solutions matches tight dependencies on the cardinality of the state and action spaces, and exhibits classical scalings with the size of the team in accordance with multi-agent optimization. Experiments then demonstrate the merits of this approach for cooperative navigation problems.**

I. Introduction

Multi-agent reinforcement learning (MARL), where a collection of agents repeatedly interact with their environment and are exposed to localized incentives, has gained traction in recent years for its ability to encapsulate numerous tasks involving sequential reasoning and coordination. For instance, autonomous vehicular networks [1], games [2], and various settings in econometrics [3], [4]. At the core of MARL is a Markov Decision Process (MDP) [5], which determines how agents, starting from one state, repeatedly select actions which trigger state transitions according to a Markov transition density, whereby instantaneous rewards are revealed by the environment. The goal of agents is to discern a policy associated with maximizing the cumulative return in the long-run, where the reward of the team decomposes into a node-separable sum of all localized rewards [6].

Defining the team reward in this way implies that agents seek to cooperate towards a common goal, which may be contrasted with competitive or mixed settings [7]. Due to the surge of interest in MARL in recent years, disparate possible technical settings have been considered, which span how one defines MDP transition dynamics; the observability of agents trajectories, the availability of computational resources at a centralized location, and the protocol by which agents exchange information. We consider the case that agents have global knowledge of the state and action (in contrast to partially observed settings [8], [9], which may necessitate pooling information at a centralized location as in centralized training decentralized execution (CTDE) [10]–[13]). We also hypothesize that the joint policy of the team factorizes into a product of individual marginal policies, which is referred to as *joint action learners* (JAL) [14], [15].

Our focus is on *decentralized training* of JAL, which means trajectory information is globally known, but agents' rewards and policy parameters are held locally private, which is a setting in common with numerous recent works that have developed multi-agent extensions of temporal difference (TD) learning [16], [17], Q-learning [18], value iteration [19], [20], and actor-critic [21], [22]. In these works, agents may communicate according to the connectivity of a possibly time-varying graph, which is intimately connected to multi-agent optimization.[1] Even for this conceptually clean setting, the complexity and convergence tradeoffs have only recently come into view. This is because in the aforementioned references, most stability guarantees are asymptotic only, apply only to sub-problems of the MARL problem such as policy evaluation (estimating the value function assuming a fixed policy [28], [29]), or due to implied non-convexity induced by policy parameterization, convergence stationarity is the most one may hope for [26], [30] – see [31] for further details.

For these reasons, we focus on linear programming reformulation of RL [32]–[34], for which stochastic primal-dual method has recently been shown to achieve *optimal sample complexity in the centralized tabular case* [35]. We develop multi-agent extensions of this framework, still in the tabular, whereby a weighted averaging step is employed in order to diffuse information between agents across time while optimizing their local utility [6], [36]. [2] Our contributions are then to:

[1] A separate but related body of works seek to estimate the communications architecture when agents' behavior is fixed using graph neural networks [23]–[25] or statistical tests for correlation between agents' local utilities [26], [27].

[2] One may more sharply enforce consensus via Lagrangian relaxation, .e.g, with primal-dual method [37], alternating direction method of multipliers (ADMM) [38], and dual reformulations [39]; however, we opt for a primal-only approach to enforcing consensus for simplicity and its compatibility with Perron-Frobenius theory [40].

[†]CISD, U.S. Army Research Laboratory, Adelphi, MD 20783

[§] School of Industrial Engineering, Purdue University, 315 N. Grant Street, West Lafayette, IN

- propose a new multi-agent variant of the dual LP formulation of reinforcement learning, where agents' decisions are defined by estimates of an average state-action occupancy measure and value vector, and consensus constraints are imposed on agents' localized estimates (Sec. II).
- owing to node-separability of the Lagrangian relaxation of the resulting optimization problem, we derive a decentralized model-free training mechanism based on a stochastic variant of primal-dual method that employs Kullback-Lieber (KL) divergence as its proximal term in the space of occupancy measures (Sec. III), together with local weighted averaging.
- establish that the number of samples required to attain near-globally optimal solutions matches tight dependencies on the cardinality of the state and action spaces [35], and exhibits classical scalings with the size of the team in prior theory [6].
- demonstrate the experimental merits of this approach for solving cooperative navigation problems.

II. Problem Setting

We consider the problem of reinforcement learning among multiple agents who share a globally observable state, but take actions and observe rewards both of which are distinctly local. Thus, agents must coordinate in order to maximize the team's cumulative return of rewards, which is a sum over all locally observed rewards. More specifically, we consider a time-varying network $\mathcal{G}^t = (\mathcal{V}, \mathcal{E}^t, W^t)$ of n agents $\mathcal{N} := \{1, 2, \dots, n\}$, where agent $i \in \mathcal{N}$ may communicate with its neighbors at a given time t, i.e., those nodes with which it shares an edge at time t, $(i,j) \in \mathcal{E}^t$, and no others; the weighting matrix $W^t{:=}[w_{ij}^t] \in \mathbb{R}^{n \times n}$, where $w_{ij}^t \geq 0$ and $w_{ij}^t = w_{ji}^t$ for alli,j,t, assigns weights to each edge (i,j) at given time t.

With the network structure clarified, we now detail how the states, actions, and rewards interconnect in the multi-agent setting under consideration. To be specific, at each time, each agent $i \in \mathcal{V}$ observes the current system state $s \in \mathcal{S}$ and synchronously takes an action $a_i \in \mathcal{A}_i$, resulting in a joint action $a := (a_1, \dots, a_n) \in \bigtimes_{i=1}^n \mathcal{A}_i$, where the system state space $\mathcal{S}$ and the constituent action spaces $\mathcal{A}_i$ are discrete finite sets. The state trajectories are Markovian, that is, upon execution of the joint action a, the state transitions to next state s' with probability $p_{s,s'}(a) := \mathbb{P}(s \mid s, a)$. We assume the joint action a is observed by all agents after execution which is needed to avoid the subtleties of partial observability. By taking the joint action a when in state s, each agent i receives an expected reward $r_{a,s}^i(s) \in [0,1]$, only known to the agent i. The system reward $r_{a,s}$ is defined as the sum of the agents' rewards over the network, $r_{a,s}{:=}\sum_{i=1}^n r_{a,s}^i$.

The goal of the team of the cooperative agents is the maximization of the *global* cumulative return defined as

$$\max_{\pi} J_\pi(s) := \lim_{T \to \infty} \quad \frac{1}{T} \mathbf{E}\left[\sum_{t=0}^{T-1} r_{a,s} \Big| s_0 = s\right] \tag{1}$$

where π denotes the joint policy of all agents, that is, a probability distribution over joint action-space given system state, $\pi : \mathcal{S} \times \mathcal{A} \to [0,1]$. The joint policy prescribes the probability that a joint action $a := (a_1, \dots, a_n)$ is taken by the collection of the agents when in system state s, which we assume factors into marginals of each individual agent's policy: $\pi(a|s) := \prod_{i=1}^N \pi_i(a_i|s)$. That is, the local policies are statistically independent, and are further denoted as $\pi_i(a_i|s)$ which define the probability of taking action a_i by agent i when in state s.

Our specific goal in this work is the design of policy optimization schemes to solve (1) such that each agent, upon the basis of its local action selections and local rewards and information exchange with its neighbors, as well as global state-action information, learns local policy parameters that result in the overall team attaining the optimal value (1). Moreover, we consider in the model-free setting, i.e., the dynamics of the environment (the transition probabilities and transitional rewards) are unknown to the agents, but a simulation oracle is available to the agents to generate state-action-reward tuples (s, a, r). Under the setting that the transition dynamics are Markovian and irreducible, the optimal policy satisfies the *average-cost Bellman equation* [41].

$$\lambda + v_s = \max_{a \in \mathcal{A}} \left\{ \sum_{s'} p_{s,s'}(a) r_{a,s} + \sum_{s'} p_{s,s'}(a) v_{s'} \right\} \text{ for all } s \in \mathcal{S} \tag{2}$$

We denote solutions to the Bellman's equation by pairs (λ^*, v^*), the search for which may be reformulated as the solution of the following linear program [41], [42]:

By substituting the definition of the global reward $r_{a,s}$ in terms of the local rewards $r_{a,s}^i$ into the dual linear program representation of (2), we obtain the following *multi-agent optimization* problem with the global variables $\mu_{a,s}$ corresponding to the joint policy π:

$$\max_{\mu \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{A}|}} \sum_{i=1}^n \sum_a \mu_a^T r_a^i \text{ s.t.} \begin{cases} \sum_a (I - P_a^T) \mu_a = 0 \text{ for all } s \\ \sum_{s,a} \mu(s,a) = 1, \;\; \mu_{a,s} \geq 0 \text{ for all } a, s \end{cases} \tag{3}$$

where I is an identity matrix of the appropriate size and $P_a \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{S}|}$ is the matrix whose (s, s')-th entry equals to $p_{s,s'}(a)$. For every feasible point of the above linear program $\mu = (\mu_a)_{a \in \mathcal{A}}$, the $\xi^\pi = (\xi_s^\pi)_{s \in \mathcal{S}}$ is the stationary state distribution where $\xi_s^\pi = \sum_a \mu_{a,s}$, and $\sum_{x,a} \mu(x,a) r_{a,s}$ corresponds to the average reward λ_π of policy π where $\pi(a|s) = \frac{\mu_{a,s}}{\xi_s^\pi}$. Moreover, $\mu_a \in \mathbb{R}^{|\mathcal{S}|}$ denotes the unnormalized occupancy measure over the state space $\mathcal{S}$ for each action $a \in \mathcal{A}$. Through normalization, one may recover the associated policy π for any feasible μ as $\xi_s^\pi = \sum_a \mu_{a,s}, \quad \pi(a|s) = \frac{\mu_{a,s}}{\sum_a \mu_{a,s}}, \quad \mu_{a,s} = \xi_s^\pi \pi(a|s)$ Then, an optimal joint policy π^* can be constructed by normalizing the occupancy measures associated with the solution to the above linear program. See [5] for details. $\pi^*(a|s) = \frac{\mu_{a,s}^*}{\sum_a \mu_{a,s}^*}$ This work develops a decentralized model-free algorithm to solve (1) upon the basis of Lagrangian relaxations of (3), detailed next.

III. Saddle Point Method

In this section, we reformulate the multi-agent LP of (3) as a saddle point problem by considering its Lagrangian relaxation We consider the Lagrangian relaxation of the preceding stochastic linear program after rescaling the first constraint in (3) by n, which yields following saddle point problem

$$\min_{v \in \mathcal{V}} \max_{\mu \in \mathcal{U}} \quad L(\mu, v) := \sum_{i=1}^{n} \sum_{a \in \mathcal{A}} \mu_a^T (n(P_a - I)v + r_a^i). \tag{4}$$

The optimal solution v^* and μ^* to the multi-agent linear program satisfy $v^* \in \mathcal{V}$ and $\mu^* \in \mathcal{U}$, where the search spaces for the value $\mathcal{V}$ and policy $\mathcal{U}$ are suitably defined to refine the limiting radius of convergence. Then, based upon applying stochastic primal-dual method to the aforementioned problem, one may derive the iterative updates in which agent i computes the weighted neighborhood average of the primal and the dual variables by taking a convex combination $\tilde{\mu}_i^t$ (resp. $\tilde{v}_i^t$) of its own estimate μ_i^t (resp. v_i^t) with the estimates received from its neighboring agents at time t as

$$\tilde{\mu}_i^t = \sum_{j=1}^{n} w_{ij}^t \mu_j^t , \qquad \tilde{v}_i^t = \sum_{j=1}^{n} w_{ij}^t v_j^t . \tag{5}$$

Then, each agent makes a gradient descent (respectively, ascent) step to minimize (respectively, maximize) its local component of the global Lagrangian function L_i, followed by a projection onto the constraint set $\mathcal{U}$ (respectively, $\mathcal{V}$). At every $t \geq 0$, each agent i generates new estimates μ_i^{t+1}, h_{t+1}^i according to the following update rules:

$$\mu_i^{t+\frac{1}{2}}(s,a) = \frac{\tilde{\mu}_i^t(s,a) \exp(\alpha \Delta_i^{t+1}(s,a))}{\sum_{s'} \sum_{a'} \tilde{\mu}_i^t(s,a) \exp(\alpha \Delta_i^{t+1}(s',a'))}$$

$$\mu_i^{t+1} = \operatorname*{argmin}_{\mu_i \in \mathcal{U}} D_{KL}(\mu_i \| \mu_i^{t+\frac{1}{2}}), \text{where} \tag{6}$$

$$v_i^{t+1} = \Pi_{\mathcal{V}}[\tilde{v}_i^t - \beta d_i^{t+1}] \tag{7}$$

where α and β are constant step sizes; $\Pi_{\mathcal{V}}$ is a Euclidean projection onto the set $\mathcal{V}$, and $d_i^t + 1$, Δ_t^{t+1} are the respective primal/dual gradients of the local Lagrangian with respect to μ_i / v_i. Note that the update on μ is mirror-descent with a Kullback-Leibler (KL) divergence over the unnormalized probability simplex and the gradient step on the variable v is a simple projected gradient descent. We assume algorithm initialization as $\mu_i = 0$ and $v_i = 0$ for all $i \in \mathcal{N}$.

Our main theoretical contributions associated with the project underlying this abstract submission is the non-asymptotic convergence analysis of the MARL scheme defined by (6) - (7). Specifically, we establish that this algorithm allows agents to converge to an ϵ-globally optimal policy in a number of samples (MDP queries) linear in the total number of state-action pairs, which is sublinear in the input size. Experimentally, we validate the proposed approach on tabular MARL approaches associated with grid worlds, namely, a cooperative navigation task.

References

[1] P. Wang, C.-Y. Chan, and A. de La Fortelle, "A reinforcement learning based approach for automated lane change maneuvers," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 1379–1384.

[2] O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev *et al.*, "Grandmaster level in starcraft ii using multi-agent reinforcement learning," *Nature*, vol. 575, no. 7782, pp. 350–354, 2019.

[3] G. Tesauro and J. O. Kephart, "Pricing in agent economies using multi-agent q-learning," *Autonomous agents and multi-agent systems*, vol. 5, no. 3, pp. 289–304, 2002.

[4] J. Lussange, I. Lazarevich, S. Bourgeois-Gironde, S. Palminteri, and B. Gutkin, "Modelling stock markets by multi-agent reinforcement learning," *Computational Economics*, vol. 57, no. 1, pp. 113–147, 2021.

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

[6] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 48–61, 2009.

[7] T. Başar and G. J. Olsder, *Dynamic noncooperative game theory*. SIAM, 1998.

[8] A. Mahajan and M. Mannan, "Decentralized stochastic control," *Annals of Operations Research*, vol. 241, no. 1-2, pp. 109–126, 2016.

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

[10] J. Foerster, I. A. Assael, N. De Freitas, and S. Whiteson, "Learning to communicate with deep multi-agent reinforcement learning," *in NeurIPS*, vol. 29, pp. 2137–2145, 2016.

[11] J. Leibo, V. Zambaldi, M. Lanctot, J. Marecki, and T. Graepel, "Multi-agent reinforcement learning in sequential social dilemmas," in *AAMAS*, vol. 16. ACM, 2017, pp. 464–473.

[12] J. Foerster, N. Nardelli, G. Farquhar, T. Afouras, P. H. Torr, P. Kohli, and S. Whiteson, "Stabilising experience replay for deep multi-agent reinforcement learning," in *Proceedings of the 34th in ICML-Volume 70*, 2017, pp. 1146–1155.

[13] T. Rashid, M. Samvelyan, C. Schroeder, G. Farquhar, J. Foerster, and S. Whiteson, "Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning," in *in ICML*, 2018, pp. 4295–4304.

[14] C. Claus and C. Boutilier, "The dynamics of reinforcement learning in cooperative multiagent systems," 1998.

[15] D. Lee, N. He, P. Kamalaruban, and V. Cevher, "Optimization for reinforcement learning: From a single agent to cooperative agents," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 123–135, 2020.

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

[17] T. Doan, S. Maguluri, and J. Romberg, "Finite-time analysis of distributed td (0) with linear function approximation on multi-agent reinforcement learning," in *in ICML*, 2019, pp. 1626–1635.

[18] S. Kar, J. M. Moura, and H. V. Poor, "Qd-learning: A collaborative distributed strategy for multi-agent reinforcement learning through consensus+ innovations," *IEEE Transactions on Signal Processing*, vol. 61, no. 7, pp. 1848–1862, 2013.

[19] H.-T. Wai, Z. Yang, Z. Wang, and M. Hong, "Multi-agent reinforcement learning via double averaging primal-dual optimization," in *in NeurIPS*, 2018, pp. 9649–9660.

[20] C. Qu, S. Mannor, H. Xu, Y. Qi, L. Song, and J. Xiong, "Value propagation for decentralized networked deep multi-agent reinforcement learning," in *in NeurIPS*, 2019, pp. 1184–1193.

[21] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," *Neural Information Processing Systems (NIPS)*, 2017.

[22] K. Zhang, Z. Yang, H. Liu, T. Zhang, and T. Basar, "Fully decentralized multi-agent reinforcement learning with networked agents," in *in ICML*, 2018, pp. 5872–5881.

[23] T. Eccles, Y. Bachrach, G. Lever, A. Lazaridou, and T. Graepel, "Biases for emergent communication in multi-agent reinforcement learning," in *in NeurIPS*, 2019, pp. 13 111–13 121.

[24] S. Ahilan and P. Dayan, "Correcting experience replay for multi-agent communication," *arXiv preprint arXiv:2010.01192*, 2020.

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

[26] G. Qu, A. Wierman, and N. Li, "Scalable reinforcement learning of localized policies for multi-agent networked systems," in *Learning for Dynamics and Control*. PMLR, 2020, pp. 256–266.

[27] Y. Lin, G. Qu, L. Huang, and A. Wierman, "Distributed reinforcement learning in multi-agent networked systems," *arXiv preprint arXiv:2006.06555*, 2020.

[28] X. Sha, J. Zhang, K. You, K. Zhang, and T. Başar, "Fully asynchronous policy evaluation in distributed reinforcement learning over networks," *arXiv preprint arXiv:2003.00433*, 2020.

[29] P. Heredia and S. Mou, "Finite-sample analysis of multi-agent policy evaluation with kernelized gradient temporal difference," in *2020 59th IEEE Conference on Decision and Control (CDC)*. IEEE, 2020, pp. 5647–5652.

[30] G. Qu, Y. Lin, A. Wierman, and N. Li, "Scalable multi-agent reinforcement learning for networked systems with average reward," *arXiv preprint arXiv:2006.06626*, 2020.

[31] K. Zhang, A. Koppel, H. Zhu, and T. Basar, "Global convergence of policy gradient methods to (almost) locally optimal policies," *SIAM Journal on Control and Optimization*, vol. 58, no. 6, pp. 3586–3612, 2020.

[32] L. C. M. Kallenberg, *Linear Programming and Finite Markovian Control Problems*. CWI Mathematisch Centrum, 1983.

[33] ——, "Survey of linear programming for standard and nonstandard Markovian control problems. Part I: Theory," *Zeitschrift für Operations Research*, vol. 40, no. 1, pp. 1–42, 1994.

[34] D. P. De Farias and B. Van Roy, "The linear programming approach to approximate dynamic programming," *Operations research*, vol. 51, no. 6, pp. 850–865, 2003.

[35] M. Wang, "Randomized linear programming solves the markov decision problem in nearly linear (sometimes sublinear) time," *Mathematics of Operations Research*, vol. 45, no. 2, pp. 517–546, 2020.

[36] J. Chen and A. H. Sayed, "Diffusion adaptation strategies for distributed optimization and learning over networks," *IEEE Transactions on Signal Processing*, vol. 60, no. 8, pp. 4289–4305, 2012.

[37] A. Koppel, F. Y. Jakubiec, and A. Ribeiro, "A saddle point algorithm for networked online convex optimization," *IEEE Transactions on Signal Processing*, vol. 63, no. 19, pp. 5149–5164, 2015.

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

[39] H. Terelius, U. Topcu, and R. M. Murray, "Decentralized multi-agent optimization via dual decomposition," *IFAC proceedings volumes*, vol. 44, no. 1, pp. 11 245–11 251, 2011.

[40] F. R. Chung and F. C. Graham, *Spectral graph theory*. American Mathematical Soc., 1997, no. 92.

[41] D. P. Bertsekas, D. P. Bertsekas, D. P. Bertsekas, and D. P. Bertsekas, *Dynamic programming and optimal control*. Athena scientific Belmont, MA, 1995, vol. 1, no. 2.

[42] D. P. De Farias and B. Van Roy, "The linear programming approach to approximate dynamic programming," *Operations Research*, 2003.