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

Alec Koppel*, Amrit Singh Bedi**, Bhargav Ganguly†, and Vaneet Aggarwal†

*Amazon[1]

**Computational and Information Sciences Directorate

U.S. Army Research Laboratory

† Purdue University

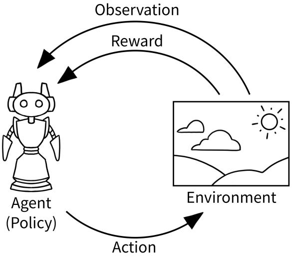
Asilomar Conference on Signals, Systems, and Computers, Moterey, CA

Oct 31- Nov 3, 2021

[1] Work done while being at U.S. Army Research Laboratory, Adelphi, MD, USA.

Reinforcement Learning

- Reinforcement learning: data-driven control



2

→ Recent successes:

⇒ AlphaGo[3]

⇒ Bipedal walker on terrain[4]

⇒ Personalized web services[5]

2 https://towardsdatascience.com/multi-agent-deep-reinforcement-learning-in-15-lines-of-code-using-pettingzoo-e0b963c0820b

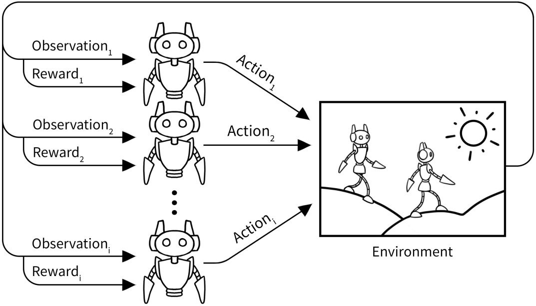
3 Silver, D. et al., Mastering the game of Go without human knowledge. Nature 550, 354359 (2017).

4 Heess, N. et al., Learning continuous control policies by stochastic value gradients. In NeurIPS, 2015.

5 Theocharous, G., "Ad recommendation systems for life-time value optimization." In ICWWW, pp. 1305-1310. 2015.

Multi-Agent Reinforcement Learning (MARL)

- Reinforcement learning: Multi-Agent Settings N : number of agents



6

$\rightarrow$ Different Settings[7]:

$\Rightarrow$ Cooperative $\rightarrow$ common payoffs – our focus

$\Rightarrow$ Competitive $\rightarrow$ contrasting payoffs

$\Rightarrow$ Mixed

[6] https://towardsdatascience.com/multi-agent-deep-reinforcement-learning-in-15-lines-of-code-using-pettingzoo-e0b963c0820b

[7] Zhang, Kaiqing et al., "Multi-agent reinforcement learning: A selective overview of theories and algorithms." arXiv:1911.10635 (2019).

Mathematical Model

- Markov decision process (MDP) $(\mathcal{S}, \mathcal{A}, \mathbb{P}, R, \gamma)$[8]
  - $\Rightarrow$ State space $\mathcal{S}$, action space $\mathcal{A} := \mathcal{A}_1 \times \mathcal{A}_2 \times \cdots \times \mathcal{A}_n$
  - $\Rightarrow$ Markov transition kernel $\mathbb{P}(s' \,|\, s, a) : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$
  - $\Rightarrow$ Reward $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$

$\rightarrow$ Stochastic policy $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$, i.e., $a_t \sim \pi(\cdot \,|\, s_t)$

$\rightarrow$ Average reward setting value function:

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

$\rightarrow$ Goal: find $\{a_t = \pi(s_t)\}$ to maximize $V_\pi(s)$

- $\Rightarrow$ Define action-state value (Q) function $Q_\pi(s,a) = \mathbb{E}[V_\pi(s) \,|\, a_0 = a]$

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

Context

- Centralized solutions are available in literature
- Our focus is decentralized training of joint action learners
- Different multi-agent extensions are available
  - $\Rightarrow$ TD learning based methods[9]
  - $\Rightarrow$ Q learning based methods [10]
  - $\Rightarrow$ Value iteration based [11]
  - $\Rightarrow$ and Actor-Critic methods [12]
- Limitations:
  - $\Rightarrow$ All of these works are for discounted settings
  - $\Rightarrow$ Most of the works have only asymptotic guarantees
  - $\Rightarrow$ Parametrization $\rightarrow$ non-convex, stationary guarantees only
  - $\Rightarrow$ No Sample Complexity results in MARL settings for average reward
  - $\Rightarrow$ No decentralized solution available in MARL for average reward

---

[9] Donghwan Lee et al', Stochastic primal-dual algorithm for distributed gradient temporal difference learning. arXiv preprint, 2018
Thinh Doan et al', Finite-time analysis of distributed td (0) with linear function approximation on MARL, ICML, pages 1626 1635, 2019.

[10] Soummya Kar et al, Qd-learning: A collaborative distributed strategy for MARL through consensus+ innovations, IEEE TSP, 2013.

[11] Hoi-To Wai et al', Multi-agent reinforcement learning via double averaging primal-dual optimization. In in NeurIPS, pages 96499660, 2018.

[12] Kaiqing Zhang et al., Fully decentralized multiagent reinforcement learning with networked agents, in ICML, pages 58725881, 2018.

Problem Formulation

- Average-cost Bellman equation

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

- Linear reformulation by [DeFarias2003]:

$$\begin{aligned} \max_{\mu\in\mathbb{R}^{|\mathcal{S}|\times|\mathcal{A}|}} \quad & \sum_{a\in\mathcal{A}} \mu(a)^T r(a) \\ \text{subject to:} & \begin{cases} \sum_{a\in\mathcal{A}} (I - P_a^T)\mu_a = 0, \quad \text{for all } s \\ \sum_{s\in\mathcal{S}, a\in\mathcal{A}} \mu(s,a) = 1 \\ \mu_{a,s} \geq 0 \quad \text{for all } a, s \end{cases} \end{aligned} \tag{0.1}$$

How to solve in a decentralized manner ?

Primal-Dual Based Algorithms

- Lagrangian function

$$\min_{v\in\mathcal{V}} \max_{\mu\in\mathcal{U}} \quad L(\mu, v) := \sum_{i=1}^{n} \sum_{a\in\mathcal{A}} \mu(a)^T[(P_a - I)v + r^i(a)]. \tag{0.2}$$

where

$$\mathcal{V} = \left\{ v \in \mathbb{R}^{|\mathcal{S}|} \middle| \quad \|v\|_\infty \ \leq 2t_{mix} \right\} \ , \tag{0.3}$$

$$\mathcal{U} = \left\{ \mu = (\mu_a)_{a\in\mathcal{A}} \middle| \mathbf{e}^T\mu = 1, \mu \geq 0, \sum_{a\in\mathcal{A}} \mu(a) \geq \frac{1}{\sqrt{\tau}|\mathcal{S}|}\mathbf{e} \right\}, \tag{0.4}$$

- We propose to use DGD style updates for μ and v

Proposed Decentralized Updates

- Consensus step:

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

- Local updates:

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

$$\text{where} \quad \mu_i^{t+\frac{1}{2}}(s,a) = \frac{\widetilde{\mu}_i^t(s,a)\exp(\Delta_i^{t+1}(s,a))}{\sum_{s'}\sum_{a'}\widetilde{\mu}_i^t(s,a)\exp(\Delta_i^{t+1}(s',a'))}$$

$$v_i^{t+1} = \Pi_{\mathcal{V}}[\widetilde{v}_i^t + \alpha(e_s - e_{s'})],$$

$\Rightarrow$ where $\Delta_i^{t+1} = \beta \frac{v_i^t(s') - v_i^t(s) + r_i^t(s,s',a) - M}{\widetilde{\mu}_i^t(s,a)} . \mathbf{e}_{s,a}$

Randomized Multi-agent Primal-dual (RMAPD) Algorithm

- **Input:** $\epsilon > 0$, $\mathcal{S}$, $\mathcal{A}$, t^*_{mix}, τ
- **For** each iteration $t = 0, 1, 2, \cdots$
- **For** each agent i in parallel do
- **Observe** the system state s
- **Execute** action $a_i \sim \pi_i(\cdot|s)$; observe $a = (a_1, \ldots, a_N)$
- **Observe** the local reward $r^i_{s,s'}(a)$
- **Send** primal and dual variables (μ^t_i, v^t_i) to neighbors $j \in n_i$, receive (μ^t_j, v^t_j) from neighbors
- **Perform** the consensus update
- **Perform** the primal and dual variable local updates

Theoretical Guarantees

Divided into three steps:

- Step 1: Bound the consensus error
- Step 2: Bound the duality gap
- Step 3: From duality gap to primal average reward

Step 1: Bound the Consensus Error

Let us define

$$\overline{\mu}^t = \frac{1}{n}\sum_{i=1}^{n} \mu_i^t, \qquad \overline{v}^t = \frac{1}{n}\sum_{i=1}^{n} v_i^t$$

$\Rightarrow$ Under some regulatory conditions, it holds that

- (Dual variable) For constant step size α, for all $i \in V$ and $t \geq 0$, we have

$$\mathbb{E}\left[\|\mathbf{v}^t - \left(\frac{1}{n}\mathbf{e}\mathbf{e}^T \otimes I_{|\mathcal{S}|}\right)\mathbf{v}^t\| \mid \mathcal{F}_t\right] \leq \mathcal{O}(\sqrt{n}\alpha)\left[1 + \frac{\Gamma(1-\rho^{t-1})}{1-\rho}\right]$$

$\Rightarrow$ where $\mathbf{v}^t = [[v_1^t]^T; \cdots; [v_n^t]^T] \in \mathbb{R}^{n|\mathcal{S}|}$ stacks v_i^t
$\Rightarrow$ $\left(\frac{1}{n}\mathbf{e}\mathbf{e}^T \otimes I_{|\mathcal{S}|}\right)\mathbf{v}^t$ stacks $\overline{v}^t$

- (Primal variable) For constant step size β, for all $i \in V$ and $t \geq 0$

$$\mathbb{E}\left[\|\boldsymbol{\mu}^t - \left(\frac{1}{n}\mathbf{e}\mathbf{e}^T \otimes I_{|\mathcal{S}||\mathcal{A}|}\right)\boldsymbol{\mu}^t\| \mid \mathcal{F}_t\right] \leq \mathcal{O}(\sqrt{n}\beta)\left[1 + \frac{\Gamma(1-\rho^{t-1})}{1-\rho}\right],$$

Step 2: Bound the Duality Gap

- Two interesting assumptions (unique to this analysis)

$\Rightarrow$ Ergodic Decision Process: $\exists \tau > 1$ such that

$$\frac{1}{\sqrt{\tau}|\mathcal{S}|}\mathbf{e} \leq \xi^{\pi} \leq \frac{\sqrt{\tau}}{|\mathcal{S}|}\mathbf{e}$$

where ξ^{π} is the stationary distribution under policy π

$\Rightarrow$ Fast-Mixing Markov Chain: MDP is t_{mix}-mixing in the sense that

$$t_{mix} \geq \max_{\pi} \min \left\{ t \geq 1 \middle| \quad \|(P^{\pi})^{t}(s,.) - \xi^{\pi}\|_{TV} \leq \frac{1}{4}, \text{for all } s \in \mathcal{S} \right\}$$

Step 2: Bound the Duality Gap

- Consider the Lyapunov function $\mathcal{E}_t$ given by

$$\mathcal{E}_t := \frac{1}{n}\sum_{i=1}^{n} D_{KL}(\mu^* \| \mu_i^t) + \frac{1}{2|\mathcal{S}|t_{mix}^2}\left\|\overline{v}^t - v^*\right\|^2$$

$\Rightarrow$ Novel multi-agent extension[13]

$\Rightarrow$ Tracks both complementary slackness and consensus error

- We prove the decrement lemma for $\mathcal{E}_t$ as

$$\begin{aligned}\mathbb{E}\left[\mathcal{E}_{t+1} \mid \mathcal{F}_t\right] \leq & \mathcal{E}_t - \beta\left[\lambda^* + \sum_{a\in A}\left[\overline{\mu}^t(a)^T[(I-P_a)v^* + r_a]\right]\right] \\ & + \beta^2\widetilde{\mathcal{O}}\left(n|\mathcal{S}||\mathcal{A}|t_{mix}^2\right) \\ & + \frac{\beta}{n}\sum_{i=1}^{n}\sum_{a\in A}\left[\left(\overline{v}^t - v_i^t\right)^T\left((I-P_a)^T(\widetilde{\mu}_i^t(a) - \mu^*(a))\right)\right] \\ & + \frac{\beta}{n}\sum_{i=1}^{n}\sum_{a\in A}\left[\left(\widetilde{\mu}_i^t(a) - \overline{\mu}^t(a)\right)^T[(P_a - I)v^* + r_a]\right]\end{aligned}$$

[13] Wang, Mengdi. "Randomized linear programming solves the Markov decision problem in nearly linear (sometimes sublinear) time."

Step 2: Bound the Duality Gap

- Consider the Lyapunov function $\mathcal{E}_t$ given by

$$\mathcal{E}_t := \frac{1}{n}\sum_{i=1}^{n} D_{KL}(\mu^* \| \mu_i^t) + \frac{1}{2|\mathcal{S}|t_{mix}^2}\left\|\overline{v}^t - v^*\right\|^2$$

$\Rightarrow$ Novel multi-agent extension[13]

$\Rightarrow$ Tracks both complementary slackness and consensus error

- We prove the decrement lemma for $\mathcal{E}_t$ as

$$\begin{aligned}\mathbb{E}\left[\mathcal{E}_{t+1} \mid \mathcal{F}_t\right] \leq & \mathcal{E}_t - \beta\left[\lambda^* + \sum_{a\in A}\left[\overline{\mu}^t(a)^T[(I-P_a)v^* + r_a]\right]\right] \\ & + \beta^2\widetilde{\mathcal{O}}\left(n|\mathcal{S}||\mathcal{A}|t_{mix}^2\right) \\ & + \frac{\beta}{n}\sum_{i=1}^{n}\sum_{a\in A}\left[\left(\overline{v}^t - v_i^t\right)^T\left((I-P_a)^T(\widetilde{\mu}_i^t(a) - \mu^*(a))\right)\right] \\ & + \frac{\beta}{n}\sum_{i=1}^{n}\sum_{a\in A}\left[\left(\widetilde{\mu}_i^t(a) - \overline{\mu}^t(a)\right)^T\left[(P_a - I)v^* + r_a\right]\right]\end{aligned}$$

[13] Wang, Mengdi. "Randomized linear programming solves the Markov decision problem in nearly linear (sometimes sublinear) time."

Step 2: Bound the Duality Gap

- Consider the Lyapunov function $\mathcal{E}_t$ given by

$$\mathcal{E}_t := \frac{1}{n}\sum_{i=1}^{n} D_{KL}(\mu^* \| \mu_i^t) + \frac{1}{2|\mathcal{S}|t_{mix}^2} \left\| \overline{v}^t - v^* \right\|^2$$

- We prove the decrement lemma for $\mathcal{E}_t$ as

$$\begin{aligned}
\mathbb{E}\left[\mathcal{E}_{t+1} \mid \mathcal{F}_t\right] \leq & \mathcal{E}_t - \beta \left[\lambda^* + \sum_{a \in A} \left[\overline{\mu}^t(a)^T [(I - P_a)v^* + r_a] \right] \right] \\
& + \beta^2 \widetilde{\mathcal{O}}\left(n|\mathcal{S}||\mathcal{A}|t_{mix}^2 \right) \\
& + \frac{\beta}{n}\sum_{i=1}^{n}\sum_{a \in A} \left[\left(\overline{v}^t - v_i^t \right)^T \left((I - P_a)^T (\widetilde{\mu}_i^t(a) - \mu^*(a)) \right) \right] \\
& + \frac{\beta}{n}\sum_{i=1}^{n}\sum_{a \in A} \left[\left(\widetilde{\mu}_i^t(a) - \overline{\mu}^t(a) \right)^T \left[(P_a - I)v^* + r_a \right] \right]
\end{aligned}$$

Step 2: Bound the Duality Gap

- Duality Gap:

Theorem

For $\overline{\mu}^t = \frac{1}{n}\sum_{i=1}^n \mu_i^t$*, after* T *number of iterations, with the step size selection* $\beta = \widetilde{\mathcal{O}}\left(\sqrt{\frac{\mathcal{E}_0}{n|\mathcal{S}|^{1.5}|\mathcal{A}|t_{mix}^2 D(\Gamma,\rho)T}}\right)$*, it holds that*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\sum_{a\in A}\left[[v^* - P_a v^* + r_a]^T\overline{\mu}^t(a)\right]\right] + \lambda^* \leq \widetilde{\mathcal{O}}\left(t_{mix}\sqrt{\frac{n\mathcal{E}_0|\mathcal{S}|^{1.5}|\mathcal{A}|D(\Gamma,\rho)}{T}}\right)$$

$\Rightarrow$ n is the number of agents

$\Rightarrow$ $D(\Gamma,\rho) := \left[\frac{1+\Gamma}{1-\rho}\right]$ where $\Gamma = \left(1 - \frac{w}{4n^2}\right)^{-2}$ and $\rho = \left(1 - \frac{w}{4n^2}\right)^{1/B}$

$\Rightarrow$ B is the strong-connectivity parameter

$\Rightarrow$ w is the lower bound on weights w_{ij} for $j \in \mathcal{N}_i$

Step 3: From Duality Gap to Primal Average Reward

- Average reward result:

Lemma

By selecting $T = \Omega\left(\tau^2 t_{mix}^2 \frac{n\mathcal{E}_0|\mathcal{S}|^{1.5}|\mathcal{A}|D(\Gamma,\rho)}{\epsilon^2}\right)$, *the proposed algorithm output a policy* $\hat{\pi}=\frac{1}{T}\sum_{t=1}^{T}\overline{\pi}^t$ *such that* $\lambda^* - \epsilon \leq \lambda_{\hat{\pi}}$ *with probability* $2/3$. *Hence, the algorithm outputs an* ϵ *optimal policy with probability* $2/3$.

- Sample Complexity Result:

Theorem

Under some regularity conditions, the proposed algorithm draws

$$T = \Omega\left(\tau^2 t_{mix}^2 \frac{n\mathcal{E}_0|\mathcal{S}|^{1.5}|\mathcal{A}|D(\Gamma,\rho)}{\epsilon^2}\log\frac{1}{\delta}\right)$$

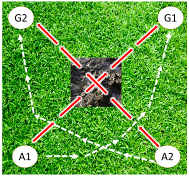
state transitions to output an approximate policy $\widetilde{\pi}$ *such that* $\lambda_{\widetilde{\pi}} \geq \lambda^* - \epsilon$ *with probability* $\log\left(\frac{1}{\delta}\right)$ *at least.*

Meta-RMAPD Algorithm

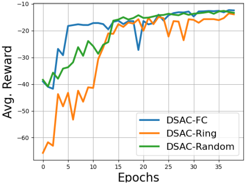
- **Input:** $\epsilon > 0$, $\mathcal{S}$, $\mathcal{A}$, t^*_{mix}, τ
- Run the RMAPD for K number of iterations with precision $\frac{\epsilon}{3}$ and denote the output as $\overline{\pi}^{(1)}, \overline{\pi}^{(2)}, \cdots, \overline{\pi}^{(K)}$.
- For each output policy $\overline{\pi}^{(k)}$, conduct the approximate value evaluation for L time steps and obtain $\overline{Y}^{(k)}$ which is approximate value evaluation with precision level $\epsilon/3$ and probability $\frac{\delta}{2K}$.
- **Output** $\widetilde{\pi} = \overline{\pi}^{(k^*)}$ such that $k^* = \mathrm{argmax}_k \, \overline{Y}^{(k)}$.

Experiments: Multi-Agent

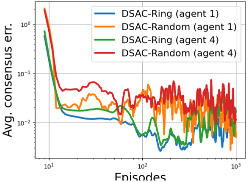
- Setup details:
  - $\Rightarrow$ 4 agents, Cooperative navigation
  - $\Rightarrow$ Goal is to reach destination safely without colliding
  - $\Rightarrow$ We run on different graphs, Fully connected, ring, and random



(a) 2 agent analog env.



(b) Average return



(c) Consensus error

$\rightarrow$ Main takeaway:

- $\Rightarrow$ Proposed Algorithm works well across variety of network topologies

Conclusions

- We consider the multi-agent RL problem with full observability
- Developed the first fully decentralized algorithm to solve the problem
- First PAC type guarantees for MARL in average reward case

Future Directions:

- Extensive simulation results for the proposed techniques
- Consider the state space approximation for scalability
- Develop the communication efficient version

Thanks