MULTI-AGENT REINFORCEMENT LEARNING IN PARTIALLY OBSERVABLE ENVIRONMENTS USING SOCIAL LEARNING

Ainur Zhaikhan and Ali H. Sayed

School of Engineering, École Polytechnique Fédérale de Lausanne

ABSTRACT

This work employs a social learning strategy to estimate the global state in a partially observable multi-agent reinforcement learning (MARL) setting. We prove that the proposed methodology can achieve results within an ε-neighborhood of the solution for a fully observable setting, provided that a sufficient number of social learning updates are performed. We illustrate the results through computer simulations.

Index Terms— social learning, partial observability, multi-agent system, reinforcement learning.

1. INTRODUCTION

A fundamental challenge in multi-agent reinforcement learning is the issue of partial observability. Traditional extensions of methodologies used for single-agent reinforcement learning (RL) to the multi-agent scenario generally assume full observability of the state variable. They also require knowledge of this information by all agents. Moreover, most multi-agent reinforcement learning (MARL) techniques in current practice are designed by assuming centralized training and decentralized execution, such as MADDPG [1], QMIX [2], MAVEN [3], or Independent Learning (IL) [4]. All these methods also impose some assumptions that tend to be restrictive for many complex scenarios. Therefore, developing decentralized RL algorithms that can operate reliably under Partially Observable Markov Decision Processes (Dec-POMDP) would be ideal for MARL applications, except that this objective is known to be NEXP-hard [5] and lacks a formal solution. Our approach relies on infusing a decentralized MARL implementation with elements of social learning to enable agents to learn the unobservable state through social interactions.

Specifically, in this work, we extend the multi-agent off-policy actor-critic (MAOPAC) algorithm [6], originally designed for fully observable environments, to handle partially observable settings. Motivated by the approach followed in [7], we leverage *social learning* strategies [8, 9, 10] to estimate the global state in a *fully decentralized* manner. Under these strategies, agents estimate belief vectors using local observations and then iteratively diffuse these estimates to their immediate neighbors. In comparison, some existing works on fully decentralized solutions tend to rely on neural network implementations, which can be complex and challenging to analyze. We adopt instead a social learning framework and, unlike many existing solutions, the proposed method does not necessitate transition models for state estimation.

The proposed method is supported by theoretical guarantees. We derive conditions for estimating the global state, ensuring that the ultimate error in the policy parameter is bounded by ε. Through empirical evaluation and analysis, we demonstrate the effectiveness and robustness of our approach in addressing the challenges posed by partially observable multi-agent environments. Additionally, we benchmark our method against the zeroth order policy optimization (ZOPO) approach.

2. MODEL SETTING

Our setting can be modeled through a Decentralized Partial Observable Markov decision process (Dec-POMDP) denoted by the tuple $(\mathcal{K}, \mathcal{A}_\ell, \{\mathcal{O}_\ell\}_{\ell=1}^K, \mathcal{S}, \{r_\ell\}_{\ell=1}^K, \mathcal{P}, \{\mathcal{L}_{\ell=1}\}_{\ell=1}^K)$, where $\mathcal{K} \triangleq \{1, 2, ..., K\}$ is a set of agents, $\{\mathcal{A}_\ell\}_{\ell=1}^K$ is the set of possible actions for agent ℓ, $\mathcal{O}$ is a continuous set of observations, and $\mathcal{S}$ is the set of states. Moreover, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ is a transition model where the value $\mathcal{P}(s'|s, a)$ denotes the probability of transition to state $s' \in \mathcal{S}$ from state $s \in \mathcal{S}$ after taking joint action $a \triangleq \{a_\ell \in \mathcal{A}_\ell\}_{\ell=1}^K$. The likelihood function $\mathcal{L}_\ell(\xi|s)$ denotes the probability of observing $\xi \in \mathcal{O}_\ell$ when the true global state is $s \in \mathcal{S}$. We assume that agents communicate according to some fixed graph.

Let $r_{\ell,n} \triangleq r_\ell(s_n, a_{\ell,n})$ denote the individual reward achieved by agent ℓ at time n if at global state $s_n \in \mathcal{S}$ it chooses action $a_{\ell,n} \in \mathcal{A}$. We denote the reward upper-bound by $R_{\max}$ and define the average reward at time n by

$$\bar{r}_n = \frac{1}{K} \sum_{\ell=1}^{K} r_{\ell,n} \tag{1}$$

Let $b_\ell : S \times A \to [0, 1]$ denote the individual policy of agent ℓ. The individual target policies are approximated by local parameterized functions $\pi_\ell(\cdot; \theta_\ell)$, where $\theta_\ell \in \mathbb{R}^d$.

Off-policy learning relies on corrections, which are im-

Emails: ainur.zhaikan@epfl.ch, ali.sayed@epfl.ch

plemented using the *importance sampling ratio*, defined as:

$$\rho_n \triangleq \prod_{\ell=1}^{K} \rho_{\ell,n} = \prod_{\ell=1}^{K} \frac{\pi_{\ell,n}}{b_\ell} \tag{2}$$

In the MAOPAC algorithm [6], the importance ratio is estimated in a decentralized manner at each actor-critic update by iteratively diffusing individual importance ratios $\rho_{\ell,n}$ until consensus. We also adopt this assumption in our work.

3. MULTI-AGENT OFF-POLICY ACTOR-CRITIC FOR DEC-POMDP

In our study, we extend MAOPAC by considering scenarios where the global state is not fully observed. We refer to the proposed extended algorithm as MAOPAC-dec-POMDP. It is based on the repeated alternation of two main learning phases: 1) traditional MAOPAC and 2) state estimation, as detailed in the following subsections. The complete listing of our method is presented in Algorithm 1.

3.1. Multi-agent off-policy actor-critic learning

The MAOPAC algorithm belongs to the family of policy gradient methods designed to concurrently learn the optimal policies (actors) and the corresponding state values (critics). In our study, we assume that each agent has its own estimate for state values, approximated using the following linear function:

$$v_{\omega_{k,n}} \approx \mu_{k,n}^T \omega_{k,n} \tag{3}$$

where $\mu_{k,n}$ is a state feature vector and $\omega_{k,n}$ is a parameter vector, also referred to as *the critic parameter*, estimated by agent k at time n. Target policies are approximated with some functions $\pi(\mu_{k,n}; \theta_{k,n})$ parametrized by $\theta_{k,n}$, which will be referred to as *the actor parameters*.

As shown in steps (4)-(8) of Algorithm 1, the updates of $\theta_{k,n}$ and $\omega_{k,n}$ are similar to those of traditional actor-critic algorithms with the exception of several correction factors: $\rho_{k,n}$, $e_{k,n}$ and $M_{k,n}^{\theta}$. The importance sampling ratio $\rho_{k,n}$ is essential for correcting the discrepancies inherent in off-policy learning methodologies. The variables $e_{k,n}$ and $M_{k,n}^{\theta}$ are motivated by *emphatic temporal difference learning* (ETD), which was proposed in [11, 12, 6] to address instabilities due to off-policy learning under function approximations. The step (4) involves intermediate calculations necessary for computing $M_{k,n}^{\theta}$.

3.2. State estimation

At each iteration of the critic and actor updates, agents perform state estimation using the social learning strategy described by (10)-(11). Social learning is a form of group learning that identifies the most suitable hypothesis from a set $\mathcal{S}$, best explaining the given observations $\xi_{k,i} \in \mathcal{O}_k$. In this work, we use the social learning technique for state estimation from [13, 14]. At each iteration of MAOPAC learning, we implement an internal loop for learning the global state. At iteration n of actor-critic updates, all agents receive a set of individual observations $\{\xi_{k,t}^n\}_{t=0}^T$, which are dependent on the current unknown global state of nature, i.e., s_n. Using their individual likelihood functions $\mathcal{L}_k(\xi_{k,t}^n|s)$, the s-th elements of the individual belief vectors $\mu_{k,n}$ are updated as follows:

1. Repeat (10) and (11) for $t = 0, 1, ..., T$, $\forall s \in \mathcal{S}$ and $\forall k \in \mathcal{K}$:

   Adapt:

   $$\psi_{k,t}(s) = \frac{\mathcal{L}_k\left(\xi_{k,t}^n|s\right)\widetilde{\mu}_{k,t-1}(s)}{\sum_{s'\in\mathcal{S}}\mathcal{L}_k\left(\xi_{k,t}^n|s'\right)\widetilde{\mu}_{k,t-1}(s')} \tag{10}$$

   Combine:

   $$\widetilde{\mu}_{k,t}(s) = \frac{\prod_{\ell\in\mathcal{N}_k}\left[\psi_{\ell,t}(s)\right]^{c_{\ell k}}}{\sum_{s'\in\Theta}\prod_{\ell\in\mathcal{N}_k}\left[\psi_{\ell,t}(s)\right]^{c_{\ell k}}} \tag{11}$$

2. Assign $\mu_{k,n} = \widetilde{\mu}_{k,T}$

Algorithm 1: MAOPAC-dec-POMDP

Initialize parameters: $\lambda \in (0,1)$, $\zeta \in (0,1)$, $\gamma \in (0,1)$, $\omega_{k,0}(s,a)$, $\rho_{k,0}$, $\widetilde{\mu}_{k,0} = \frac{1}{S}$, $\eta_{k,0} = \frac{1}{S}$, $F_{k,0} = 0$;

for *n=0,1,2....* **do**

  if *n=0* **then**

    Each agent k receives observations $\{\boldsymbol{\xi}_{k,t}^0\}_{t=0}^T$;

    Each agent k estimates $\mu_{k,0}$ using (10)-(11);

  end

  Each agent k takes action $a_{k,n} \sim b_k$;

  Each agent k receives $r_{k,n}$ and $\{\xi_{k,t}^{n+1}\}_{t=0}^T$;

  Each agent k estimates $\eta_{k,n}$;

  Each agent k estimates $\rho_{k,n}$ as given in [6] ;

  for *each agent k* **do**

    Update $M_{k,n}^{\theta}$ and $e_{k,n}$ as given in [6] (4)

    $\delta_{k,n} = r_{k,n} + \gamma\omega_{k,n}^T\eta_{k,n} - \omega_{k,n}^T\mu_{k,n}$ (5)

    $\Psi_{k,n} = \frac{\nabla_\theta \pi\left(a_{k,n} \mid \mu_{k,n}\right)}{\pi\left(a_{k,n} \mid \mu_{k,n}\right)}$ (6)

    $\widetilde{\omega}_{k,n} = \omega_{k,n} + \beta_n\rho_{k,n}\delta_{k,n}e_{k,n}$ (7)

    $\theta_{k,n+1} = \theta_{k,n} + \beta_n\rho_{k,n}M_{k,n}^{\theta}\delta_{k,n}\Psi_{k,n}$ (8)

  end

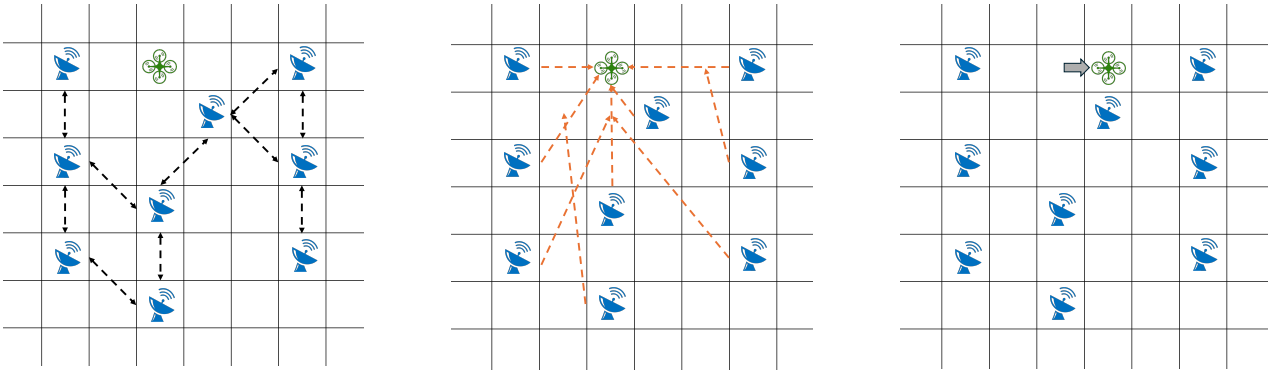
  for *each agent k* **do**

    $\omega_{k,n+1} = \sum_{\ell\in\mathcal{N}_k} c_{\ell,k}\widetilde{\omega}_{\ell,n}$ (9)

  end

  Reset: $\mu_{k,n} = \eta_{k,n}$, $\widetilde{\eta}_{k,0}(s) = \frac{1}{S}$;

end



(a) Parameter exchanges (b) Action selection (c) State transition

Fig. 1. Illustration of the agents/target framework: (a) shows a phase where agents exchange parameters according to the communication graph: black arrows demonstrate communication links; (b) illustrates a phase where agents, based on their individual policies, choose an action, i.e., select the possible location of the target (cells indicated by the orange arrows); (c) demonstrates the transition of the target to another state (cell) as a result of the agents' actions.

where $c_{\ell,k}$ are the entries of a combination matrix C satisfying Assumption 1 below, and the notation $\mathcal{N}_k$ denotes the neighbors of agent k. Actor-critic learning requires knowledge of the current and next states. We estimate the belief vector for the next state $\eta_{k,n}$, in the same manner as $\mu_{k,n}$, but using the observations for the next state, $\xi_{k,t}^{n+1^T}{}_{t=0}$.

As demonstrated in [9], under Assumptions 1–3 (listed below), repeated application of the updates in (10) and (11) allows agents to almost surely learn the true current state s_n. The convergence rate of the algorithm is discussed in [8]. Generally, the belief vectors converge at an exponential rate, which depends on the second largest eigenvalue of C [10]. Therefore, in Section 4, we analyze the maximum allowable state estimation error to ensure that, by the end of the MARL run, the estimation of the policy parameter $\theta_{k,n}$ is ε-optimal.

Assumption 1 (Combination matrix). *The combination matrix C assigns non-negative weights to neighboring agents and is assumed to be doubly-stochastic.*

Assumption 2 (Strong connectivity). *The underlying graph topology is assumed to be strongly connected.*

Assumption 3 (Likelihood function). *For all agents $k \in \mathcal{K}$ and all states $s \in \mathcal{S}$, the KL-divergence between the true model $f_k(\xi_k)$ and the likelihood function $\mathcal{L}_k(\xi_k|s)$ is finite:*

$$\mathrm{D}_k\left(f \mid\mid \mathcal{L}_k\right) \triangleq \mathbb{E}_{f_k} \log \frac{f_k(\xi)}{L_k(\xi \mid s)} < \infty \tag{12}$$

■

4. THEORETICAL GUARANTEES

For analysis we compare the performance of MAOPAC when the global state is fully observed against when it is only partially observed, as is the case in this paper. In essence, we compare the original algorithm MAOPAC and the proposed MAOPAC-dec-POMDP. Let $\widehat{\mu}_{k,n}$, $\widehat{\rho}_{k,n}$, and $\widehat{\theta}_{k,n}$ denote the full-observation counterparts of $\mu_{k,n}$, $\rho_{k,n}$, and $\theta_{k,n}$, respectively. These variables undergo the same update processes as their counterparts, with the distinction that they are privileged with knowledge of the true global state. We introduce the error variables

$$\begin{aligned} &\Delta\mu_{k,n} \triangleq \widehat{\mu}_{k,n} - \mu_{k,n},\ \Delta\rho_{k,n} \triangleq \widehat{\rho}_{k,n} - \rho_{k,n}, \\ &\Delta\theta_{k,n} \triangleq \widehat{\theta}_{k,n} - \theta_{k,n} \end{aligned} \tag{13}$$

In the analysis of the proposed method, we establish the following upper bounds:

$$\|\omega_{k,n}\| \leq \sum_{i=0}^{n-1} \frac{\beta_i \Omega^{n-i} R_{\max} B_e \|\omega_{0,\max}\|}{b_\epsilon} \triangleq B_n^\omega \tag{14}$$

$$|\delta_{k,n}| \leq R_{\max} + (1+\gamma) B_n^\omega \triangleq B_n^\delta \tag{15}$$

The proofs are omitted due to space limitations.

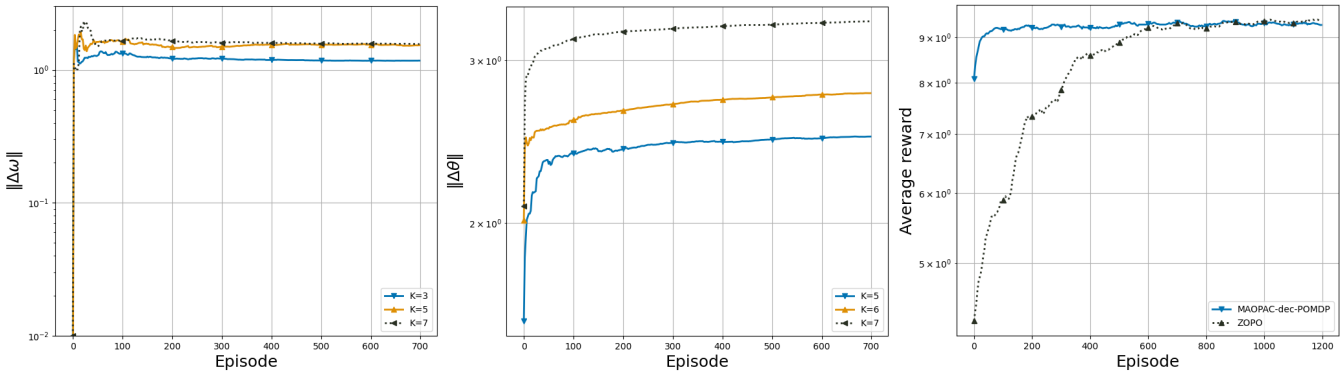
In our analysis, we determine conditions under which the actor parameter under partial observability, $\theta_{k,n}$, can get ε-close to the optimal result of MAOPAC. This finding is formally stated in Theorem 1, whose proof is omitted for brevity.

Theorem 1 (ϵ-optimality). *Let $\|\cdot\|$ denote the Euclidean norm of a vector. Then, under Assumptions 1-3 and Assumption 6.1.4. from [6], for all agents k and $b_\epsilon > \gamma$, $\Delta\theta_{k,n}$ is ε-bounded at time n if $\forall j \leq n$:*

$$\|\Delta\mu_{\ell,j})\| \leq \min(\widetilde{B}_1, \widetilde{B}_2) \tag{16}$$

and

$$\|\Delta\rho_{\ell,j}\| \leq \min(\widetilde{D}_1, \widetilde{D}_2, \widetilde{D}_3) \tag{17}$$



(a) Critic error (b) Actor error (c) MAOPAC-dec-POMDP vs ZOPO

Fig. 2. Comparison between MAOPAC and the proposed MAOPAC-dec-POMDP: (a) shows the difference in critic values computed by MAOPAC and MAOPAC-dec-POMDP for different numbers of agents (b) shows the difference in actor values computed by MAOPAC and MAOPAC-dec-POMDP for different numbers of agents (c) compares the proposed MAOPAC-dec-POMDP and ZOPO in terms of cumulative average reward

where γ is a reward discount factor,

$$\widetilde{B}_1 \triangleq \frac{\varepsilon\alpha_1}{\Phi_n\beta_j B_j^{\omega}\Omega^{n-j}}, \quad \widetilde{B}_2 \triangleq \frac{\varepsilon\alpha_2}{\Phi_n|M_{k,j}|B_n^{\delta}(\gamma\lambda)^{-j}} \tag{18}$$

$$\widetilde{D}_1 \triangleq \frac{\varepsilon\alpha_3}{\Phi_n\beta_j B_j^{\delta}\Omega^{n-j}}, \quad \widetilde{D}_2 \triangleq \frac{\varepsilon\alpha_4\,(b_\epsilon/\gamma)^{-j}}{\Phi_n F_{k,j} B_n^{\delta}\Omega^{n}}, \tag{19}$$

$$\widetilde{D}_3 \triangleq \frac{\varepsilon\alpha_5\,(b_\epsilon/\gamma)^{-j}}{\beta_n B_n^{\delta} F_{k,j}}, \quad \Phi_n \triangleq 4\beta_0(1+\gamma)\pi^2 B_M^{\theta} n^3, \tag{20}$$

$$\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5 < \infty \tag{21}$$

■

Note that over infinite time, $n \to \infty$, the bounds in (16) and (17) converge to 0, which is a reasonable outcome. State estimation occurs at every iteration of the actor-critic updates. As a result, each new state estimation error $\Delta\mu_{k,n}$ contributes to the overall error of the actor parameter $\Delta\theta_{k,n}$. Therefore, the convergence of the actor error $\Delta\theta_{k,n}$ necessitates the convergence of $\Delta\mu_{k,n}$ to 0 as n approaches infinity. Next, note that the bounds in (16) and (17) can be computed in real time, highlighting the practical importance of these bounds. Specifically, an agent undergoing an actor-critic update at time j needs to determine the maximum allowable state estimation error and use it to adjust the runtime of the inner state-estimation loop.

5. EXPERIMENTAL RESULTS AND DISCUSSIONS

For experiments we use the grid-based scenario from [7]. It involves K agents (representing radars) with fixed location and one object with changing location over some grid (see Figure 1). The states of the underlying POMDP are the cells of the grid. The objective for all agents is to correctly detect the location of the moving object. The actions by agents correspond to which cell in the grid to hit. Agents are rewarded based on the accuracy of their selections relative to the actual location of the object.

As shown in figures 2(a) and 2(b), both critic and actor values for MAOPAC-POMDP closely align with that of MAOPAC. Convergence of MAOPAC to the optimal policy under full observability has been proven in [6]. Hence, it follows that MAOPAC-POMDP can attain an ε-optimal solution.

ZOPO is an extension of Monte-Carlo-based policy gradient approaches such as the REINFORCE algorithm. ZOPO is inherently simple to implement and can be useful when the gradient of a function is not available. However, it tends to exhibit slow convergence and high noise levels, as confirmed in Figure 2(d).

6. CONCLUSION

This paper proposes a multi-agent off-policy actor-critic algorithm for partially observable environments. The key innovation lies in estimating the global state through social learning to guarantee $b_\epsilon-$boundedness of estimation error. The performance of the resulting algorithm is illustrated by comparing against state-of-the-art solutions.

7. ACKNOWLEDGEMENT

The authors wish to thank Ph.D. student Mert Kayaalp for useful discussions on the topic of this paper. To improve grammar and presentation, parts of this manuscript have been revised with ChatGPT 3.5.

8. REFERENCES

[1] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," in *Proc. Advances in Neural Information Processing Systems*, 2017, p. 6382–6393.

[2] T. Rashid, M. Samvelyan, C. S. D. Witt, G. Farquhar, J. Foerster, and S. Whiteson, "Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning," in *Proc. International Conference on Machine Learning*, 2018, pp. 4295–4304.

[3] A. Mahajan, T. Rashid, M. Samvelyan, and S. Whiteson, "Maven: Multi-agent variational exploration," in *Proc. Neural Information Processing Systems*, 2019, pp. 7613 – 7624.

[4] C. S. D. Witt, T. Gupta, D. Makoviichuk, V. Makoviychuk, P. H. S. Torr, M. Sun, and S. Whiteson, "Is independent learning all you need in the starcraft multi-agent challenge?" 2020. [Online]. Available: https://arxiv.org/abs/2011.09533

[5] D. S. Bernstein, R. Givan, N. Immerman, and S. Zilberstein, "The complexity of decentralized control of markov decision processes," *Mathematics of Operations Research*, vol. 27, no. 4, pp. 819–840, 2002.

[6] W. Suttle, Z. Yang, K. Zhang, Z. Wang, T. Başar, and J. Liu, "A multi-agent off-policy actor-critic algorithm for distributed reinforcement learning," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 1549–1554, 2020, 21st IFAC World Congress.

[7] M. Kayaalp, F. Ghadieh, and A. H. Sayed, "Policy evaluation in decentralized pomdps with belief sharing," *IEEE Open Journal of Control Systems*, vol. 2, pp. 125–145, 2023.

[8] A. Lalitha, T. Javidi, and A. D. Sarwate, "Social learning and distributed hypothesis testing," *IEEE Transactions on Information Theory*, vol. 64, no. 9, pp. 6161–6179, 2018.

[9] M. DeGroot, *Optimal Statistical Decisions*. Wiley, 2005.

[10] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge University Press, 1985.

[11] H. Yu, "On convergence of emphatic temporal-difference learning," in *Proc. Conference on Learning Theory*, vol. 40, 2015, pp. 1724–1751.

[12] R. S. Sutton, A. R. Mahmood, and M. White, "An emphatic approach to the problem of off-policy temporal-difference learning," *J. Mach. Learn. Res.*, vol. 17, no. 1, p. 2603–2631, 2016.

[13] V. Bordignon, S. Vlaski, V. Matta, and A. H. Sayed, "Learning from heterogeneous data based on social interactions over graphs," *IEEE Transactions on Information Theory*, vol. 69, no. 5, pp. 3347–3371, 2023.

[14] V. Bordignon, V. Matta, and A. H. Sayed, "Adaptive social learning," *IEEE Transactions on Information Theory*, vol. 67, no. 9, pp. 6053–6081, 2021.