Finite-Time Analysis of Distributed TD(0) with Linear Function Approximation for Multi-Agent Reinforcement Learning

Thinh T. Doan [1 2] **Siva Theja Maguluri** [1] **Justin Romberg** [2]

Abstract

We study the policy evaluation problem in multi-agent reinforcement learning. In this problem, a group of agents work cooperatively to evaluate the value function for the global discounted accumulative reward problem, which is composed of local rewards observed by the agents. Over a series of time steps, the agents act, get rewarded, update their local estimate of the value function, then communicate with their neighbors. The local update at each agent can be interpreted as a distributed consensus-based variant of the popular temporal difference learning algorithm TD(0). While distributed reinforcement learning algorithms have been presented in the literature, almost nothing is known about their convergence rate. Our main contribution is providing a finite-time analysis for the convergence of the distributed TD(0) algorithm. We do this when the communication network between the agents is time-varying in general. We obtain an explicit upper bound on the rate of convergence of this algorithm as a function of the network topology and the discount factor. Our results mirror what we would expect from using distributed stochastic gradient descent for solving convex optimization problems.

1. Introduction

Reinforcement learning (RL) offers a general paradigm for learning optimal policies in stochastic control problems based on simulation (Sutton & Barto, 1998; Bertsekas & Tsitsiklis, 1999; Szepesvari, 2010). In this context, an agent seeks to find an optimal policy through interacting with the environment, often modeled as a Markov Decision Process (MDP), with the goal of optimizing its long-term future reward (or cost). During the last few years, RL has been recognized as a crucial solution for solving many challenging practical problems, such as, autonomous driving (Chen et al., 2015), robotics (Gu et al., 2017), helicopter flight (Abbeel et al., 2007), board games (Silver et al., 2016), and power networks (Kar et al., 2013).

A central problem in RL is to estimate the accumulative reward (value function) for a given stationary policy of an MDP, often referred to as the policy evaluation problem. This problem arises as a subproblem in the important policy iteration method in RL (Sutton & Barto, 1998; Bertsekas & Tsitsiklis, 1999). Perhaps, the most popular method for solving this problem is temporal-difference learning ($\mathsf{TD}(\lambda)$), originally proposed by Sutton (Sutton, 1988) and analyzed explicitly for various scenarios in (Dayan, 1992; Gurvits et al., 1994; Pineda, 1997; Tsitsiklis & Roy, 1997; 1999). This method approximates the long-term future cost as a function of current state, and depends on a scalar $\lambda \in [0, 1]$ that controls a trade-off between the accuracy of the approximation and the susceptibility to simulation noise. In this paper, we focus on the special case $\lambda = 0$, i.e., the $\mathsf{TD}(0)$ algorithm with function approximation, which has received a great success for solving many complicated problems involving a very large state space (Tesauro, 1995; Mnih et al., 2015; Silver et al., 2016). This algorithm has a straightforward implementation, and can be executed incrementally as observation are made.

Our interest in this paper is to study the policy evaluation problem in multi-agent reinforcement learning (MARL), where a group of agents operate in an environment. We are motivated by broad applications of the multi-agent paradigm within engineering, for example, mobile sensor networks (Cortes et al., 2004; Ogren et al., 2004), cell networks (Bennis et al., 2013), and power networks (Kar et al., 2013). In this context, each agent takes its own action based on the current state, and consequently a new state is determined. Moreover, the agents receive different local rewards, which are the functions of their current state, their new state, and their action. We assume that each agent only knows its own local reward. Their goal is to cooperatively evaluate the

[1] School of Industrial and Systems Engineering [2] School of Electrical and Computer Engineering, Georgia Instiute of Technology, GA, 30332, USA. Correspondence to: Thinh T. Doan <thinhdoan@gatech.edu>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019. Copyright 2019 by the author(s).

global accumulative reward based only on their local interactions. For solving this problem, our focus is to consider a distributed variant of $\mathsf{TD}(0)$ algorithm with linear function approximation, where our goal is to provide a finite-time analysis of such distributed $\mathsf{TD}(0)$ in the context of MARL. To the best of our knowledge, such finite-time analysis for distributed $\mathsf{TD}(0)$ is not available in the existing literature.

1.1. Existing Literature

Despite its simple implementation, theoretical analysis for the performance of TD is quite complicated. In general, TD method is not the true stochastic gradient descent (SGD) for solving any static optimization problems, making it challenging to characterize the consistency and quantify the progress of this method. A dominant approach to study the asymptotic convergence of TD learning is to use tools from stochastic approximation (SA), specifically, the ordinary differential equation (ODE) method. In particular, Tsitsiklis and Van Roy considered a policy evaluation problem on a discounted MDP for both finite and infinite state spaces with linear function approximation (Tsitsiklis & Roy, 1997). By viewing TD as a stochastic approximation for solving a suitable Bellman equation, they characterized the almost sure convergence of this method based on the ODE approach. That is, under the right conditions, the SA update asymptotically follows the trajectory of a stable ODE. The convergence of SA is then equivalent to the convergence of this ODE solution, which can be shown by using Lyapunov theorem in control theory. Following this work, Borkar and Meyn provided a general and unified framework for the convergence of SA with broad applications in RL (Borkar & Meyn, 2000). More general results in this area can be found in the monograph by Borkar (Borkar, 2008).

While the asymptotic convergence of TD algorithms is well-known, very little is known about their finite-time analysis (or the rate of convergence). Indeed, it is not obvious how to derive such convergence rate by using ODE approach. A concentration bound was given in (Thoppe & Borkar; Borkar, 2008) for the SA algorithm under a strict stability assumption of the iterates. Recently, a finite-time analysis of $\mathsf{TD}(0)$ algorithm with linear function approximation was simultaneously studied in (Dalal et al., 2018; Bhandari et al., 2018) for a single agent problem. These works carefully characterize the progress of $\mathsf{TD}(0)$ update and derive its convergence rate by utilizing the standard techniques of SGD and the results in (Tsitsiklis & Roy, 1997).

Within the context of MARL, an asymptotic convergence of the distributed gossiping $\mathsf{TD}(0)$ with linear function approximation was probably first studied in (Mathkar & Borkar, 2017), where the authors utilize the standard techniques of ODE approach. Such results were also studied implicitly in the context of distributed actor-critic methods in (Zhang et al., 2018). On the other hand, unlike recent works about finite-time analysis in a single agent setup (Dalal et al., 2018; Bhandari et al., 2018), the rate of convergence of distributed $\mathsf{TD}(0)$ is missing in the existing literature of MARL, which is the focus of this paper.

Finally, we mention some related RL methods for solving policy evaluation problems in both single agent RL and MARL, such as, the gradient temporal difference methods studied in (Sutton et al., 2009b;a; Liu et al., 2015; Macua et al., 2015; Stanković & Stanković, 2016; Wai et al., 2018), least squares temporal difference (LSTD) (Bradtke & Barto, 1996; Tu & Recht, 2018), and least squares policy evaluation (LSPE) (Nedić & Bertsekas, 2003; Yu & Bertsekas, 2009). Although they share some similarity with TD learning, these methods belong to a different class of algorithms, which involve more iteration complexity in their updates.

1.2. Main Contributions

In this paper, we study a distributed variant of the $\mathsf{TD}(0)$ algorithm for solving a policy evaluation problem in MARL. Our distributed algorithm is composed of the popular consensus step and local $\mathsf{TD}(0)$ updates at the agents. Our main contribution is to provide a finite-time analysis for the convergence of distributed $\mathsf{TD}(0)$ over time-varying networks. We obtain an explicit upper bound on the rate of convergence of this algorithm as a function of the network topology and the discount factor. Our results mirror what we would expect from using distributed SGD for solving convex optimization problems. For example, when the stepsizes are chosen independently with the problem's parameters, the function value estimated at each agent's time-weighted estimates converges to a neighborhood around the optimal value at a rate $\mathcal{O}(1\,/\,k)$ under constant stepsizes and asymptotically converges to the optimal value at a rate $\mathcal{O}(1\,/\,\sqrt{k+1})$ under time-varying stepsizes. Moreover, our rates also show the dependence on the network topology and the discount factor associated with the accumulative reward. These convergence rates mirrors the ones from using distributed stochastic gradient descent for solving convex optimization problems. On the other hand, we observe the same results in the case of strongly convex optimization problems for both constant and time-varying stepsizes, when the stepsizes are chosen based on some knowledge of the problem's parameter. We note that such an explicit formula for the rate of distributed $\mathsf{TD}(0)$ algorithm is not available in the literature.

2. Centralized Temporal-Difference Learning

We briefly review here the problem of policy evaluation for a given stationary policy μ over a Markov Decision Process (MDP). This will facilitate our development of multi-agent reinforcement learning in the next section. We consider a discounted reward MDP defined by 5-tuple $(\mathcal{S},\mathcal{U},\mathcal{P},\mathcal{R},\gamma)$, where $\mathcal{S}$ is a finite set of states, $\mathcal{S}=\{1,\ldots,n\}$. In addition, $\mathcal{U}$ is the set of control actions, $\mathcal{P}$ is the set of transition probability matrices associated with the Markov chain, $\mathcal{R}$ is the reward function, and $\gamma\in(0,1)$ is the discount factor.

At each time $k\geq 0$, the agent observes the current state $s(k)=i$ and applies an action $\mu(s(k))$, where $\mu:\mathcal{S}\rightarrow\mathcal{U}$. The system then moves to the next state $s'(k)=j$ with some probability $p_{ij}(\mu(i))$ decided by the action $\mu(i)$. Moreover, the agent receives the instantaneous reward $r(k)$. That is, for each transition from i to j an immediate reward r is observed according to $\mathcal{R}(i,j)$. In the sequel, since the policy is stationary, we drop μ in our notation for convenience. The discounted accumulative reward $J^*:\mathcal{S}\rightarrow\mathbb{R}$ associated with this Markov chain is defined for all $i\in\mathcal{S}$ as

$$J^*(i)\triangleq\mathbb{E}\left[\sum_{k=0}^{\infty}\gamma^k\mathcal{R}(s(k),s'(k))\;\Big|\;s(0)=i\right],\qquad(1)$$

which is the solution of the following Bellman equation (Sutton & Barto, 1998; Bertsekas & Tsitsiklis, 1999)

$$J^*(i)=\sum_{j=1}^{n}p_{ij}\left[\mathcal{R}(i,j)+\gamma J^*(j)\right],\quad\forall i\in\mathcal{S}.\qquad(2)$$

We are interested in the case when the number of states is very large, and so computing J^* exactly may be intractable. To mitigate this, we use low-dimensional approximation $\tilde{J}$ of J^*, restricting $\tilde{J}$ to be in a linear subspace. While more advanced nonlinear approximations using, for example, neural nets as in the recent works (Mnih et al., 2015; Silver et al., 2016) may lead to more powerful approximations, the simplicity of the linear model allows us to analyze it in detail. The linear function approximation $\tilde{J}$ is parameterized by a weight vector $\theta\in\mathbb{R}^K$, with

$$\tilde{J}(i,\theta)=\sum_{\ell=1}^{K}\theta_\ell\phi_\ell(i),\qquad(3)$$

for a given set of K feature vectors $\phi_\ell:\mathcal{S}\rightarrow\mathbb{R}$, $\ell\in\{1,\ldots,K\}$, where $K\ll n$. Let $\phi(i)$ be a vector defined as

$$\phi(i)=(\phi_1(i),\ldots,\phi_K(i))^T\in\mathbb{R}^K.$$

And let $\Phi\in\mathbb{R}^{n\times K}$ be a matrix, whose $i-$th row is the row vector $\phi(i)^T$ and whose $\ell-$th column is the vector $\phi_\ell=(\phi_\ell(1),\ldots,\phi_\ell(n))^T\in\mathbb{R}^n$, that is

$$\Phi=\begin{bmatrix} | & & | \\ \phi_1 & \cdots & \phi_K \\ | & & | \end{bmatrix}=\begin{bmatrix} — & \phi(1)^T & — \\ \cdots & \cdots & \cdots \\ — & \phi(n)^T & — \end{bmatrix}.$$

Thus, $\tilde{J}(\theta)=\Phi\theta$, giving the gradient of $\tilde{J}$ w.r.t θ as

$$\nabla\tilde{J}=\Phi^T\quad\text{and}\quad\nabla\tilde{J}(i,\theta)=\phi(i),\quad\forall i\in\mathcal{S}.$$

The goal now is to find a $\tilde{J}$ that is the best approximation of J^* based on the generated data by applying the stationary policy μ on the MDP. That is, we seek an optimal weight θ^* such that the distance between $\tilde{J}$ and J^* is minimized. For solving this problem, we are interested in the TD(0) algorithm, which is equivalent to a stochastic approximation for solving the Bellman equation (2) (Bertsekas & Tsitsiklis, 1999). In particular, we assume that at each time k there is an oracle giving one data tuple $(s(k),s'(k),r(k))$, probably through simulation. The method of TD(0) then updates θ as

$$\theta(k+1)=\theta(k)+\alpha(k)d(k)\nabla\tilde{J}(s(k),\theta(k)),\qquad(4)$$

where $d(k)\in\mathbb{R}$ is the temporal difference at time k

$$d(k)=r(k)+\gamma\tilde{J}(s'(k),\theta(k))-\tilde{J}(s(k),\theta(k)).$$

Here, $d(k)$ represents the difference between the outcome $r(k)+\gamma\tilde{J}(s'(k),\theta(k))$ of the current stage and the current estimate $\tilde{J}(s(k),\theta(k))$. Thus, $d(k)$ provides us an indicator whether to increase or decrease our current variable $\theta(k)$.

In (Tsitsiklis & Roy, 1997), to study the convergence of TD(0), the authors viewed J^* as a fixed point of the Bellman operator $\mathcal{T}:\mathbb{R}^n\rightarrow\mathbb{R}^n$ defined as

$$(\mathcal{T}J)(i)=\sum_{j=1}^{n}p_{ij}\left\{\mathcal{R}(i,j)+\gamma J(j)\right\},\quad\forall i\in\mathcal{S}.$$

and showed that $\{\theta(k)\}$ generated by TD(0) converges to θ^* almost surely, where θ^* is the unique solution of the projected Bellman equation $\Pi\,\mathcal{T}(\Phi\theta^*)=\Phi\theta^*$, and satisfies

$$\|\Phi\theta^*-J^*\|_D\leq\frac{1}{1-\gamma}\|\Pi\,J^*-J^*\|_D.\qquad(5)$$

Here, $\Pi\,J$ denotes the projection of a vector J to the linear subspace spanned by the feature vectors ϕ_ℓ

$$\Pi\,J=\underset{y\in\text{span}\{\phi_\ell\}}{\arg\min}\|y-J\|_D,$$

where $\|J\|_D^2=J^TDJ$ is the weighted norm of J associated with the $n\times n$ diagonal matrix D, whose diagonal entry are $(\pi(1),\ldots,\pi(n))$, the stationary distribution associated with $\mathcal{P}$. Moreover, we denote by $\|\cdot\|$ the Euclidean and Frobenius norms for a vector and a matrix, respectively.

As shown in (Tsitsiklis & Roy, 1997) θ^* satisfies $\mathbf{A}\theta^*=b$, where $\mathbf{A}$ is positive definite, i.e., $x^T\mathbf{A}x>0\;\forall x$, and

$$\mathbf{A}=\mathbb{E}_\pi\left[\phi(s)(\phi(s)-\gamma\phi(s'))^T\right],\quad b=\mathbb{E}_\pi[r\phi(s)].\qquad(6)$$

It is worth noting that although TD(0) can be viewed as a stochastic approximation method for solving (2), it is not a SGD method since the temporal direction $d(k)\nabla\tilde{J}(s(k),\theta(k))$ is not a true stochastic gradient of any static objective function. This makes analyzing the finite-time convergence of TD(0) more challenging since standard techniques of SGD are not applicable. Our focus, therefore, is to provide such a finite-time analysis for the distributed variant of TD(0) algorithm in the context of MARL.

3. Multi-Agent Reinforcement Learning

We consider a multi-agent reinforcement learning system of N agents modeled by a Markov decision process. We assume that the agents can communicate with each other through a given sequence of time-varying undirected graphs $\mathcal{G}(k) = (\mathcal{V}, \mathcal{E}(k))$, where $\mathcal{V} = \{1, \ldots, N\}$ and $\mathcal{E}(k) = \mathcal{V} \times \mathcal{V}$ are the vertex and edge sets at time k, respectively. This framework can be mathematically characterized by a 6-tuple $(\mathcal{S}, \{\mathcal{U}_v\}, \mathcal{P}, \{\mathcal{R}_v\}, \gamma, \mathcal{G}(k))$ for $v \in \mathcal{V}$ at each time step k. Here, $\mathcal{S} = \{1, \ldots, n\}$ is the global finite state space observed by the agents, $\mathcal{U}_v$ is the set of control available to each agent v, and $\mathcal{R}_v$ is each agent's reward function.

At any time k, each agent v observes the current states $s(k)$ and applies an action $\mu_v(k) \in \mathcal{U}_v$, where μ_v is a stationary policy of agent v. Based on the joint actions of the agents, the system moves to the new state $s'(k)$ and agent v receives an instantaneous local reward $r_v(k)$, defined by $\mathcal{R}_v(s(k), s'(k))$ for each transition of states. The goal of the agents is to cooperatively find the total accumulative reward J^* over the network defined as

$$J^*(i) \triangleq \mathbb{E}\left[\sum_{k=0}^{\infty} \frac{\gamma^k}{N} \sum_{v \in \mathcal{V}} \mathcal{R}_v(s(k), s'(k)) \;\Big|\; s(0) = i\right], \quad (7)$$

which also satisfies the following Bellman equation

$$J^*(i) = \sum_{j=1}^{n} p_{ij} \left\{ \frac{1}{N} \sum_{v \in \mathcal{V}} \mathcal{R}_v(i,j) + \gamma J^*(j) \right\}, \qquad i \in \mathcal{S}.$$

Similar to the centralized problem, we are interested in finding a linear approximation $\tilde{J}$ of J^* as given in Eq. (3). In addition, since each agent knows only its own reward function, the agents have to cooperate to find $\tilde{J}$. In the following, for solving such problem we provide a distributed variant of the TD(0) algorithm presented in Section 2, where the agents only share their estimates of the optimal θ^* to its neighbors but not their local rewards. Similar to the centralized problem (a.k.a Eq. (6)), θ^* satisfies

$$\mathbf{A}\theta^* = \frac{1}{N} \sum_{v \in \mathcal{V}} b_v, \quad (8)$$

where the positive defnite matrix $\mathbf{A}$ and b_v are defined as

$$\mathbf{A} = \mathbb{E}_\pi \left[\phi(s)(\phi(s) - \gamma\phi(s'))^T\right], \quad b_v = \mathbb{E}_\pi[r_v \phi(s)]. \quad (9)$$

3.1. Distributed Consensus-Based TD(0) Learning

In this section, we study a distributed consensus-based variant of the centralized TD(0) method, formally stated in Algorithm 1. In particular, agent v maintains their own estimate $\theta_v \in \mathbb{R}^K$ of the optimal θ^*. At any iteration $k \geq 0$, each agent v only receives the estimates θ_u from its neighbors $u \in \mathcal{N}_v(k)$, where $\mathcal{N}_v(k) := \{u \in \mathcal{V} \mid (v,u) \in \mathcal{E}(k)\}$ is the set of node v's neighbors at time k. Agent v then observes one data tuple $(s(k), s'(k), r_v(k))$ returned by the oracle, following an update to its estimate $\theta_v(k+1)$ by using Eq. (10). Here, $W_{vu}(k)$ is the weight which agent v assigns for $\theta_u(k)$. Finally, $[\cdot]_\mathcal{X}$ denotes the projection to a set $\mathcal{X} \subset \mathbb{R}^K$, whose condition is given shortly.

Algorithm 1 Distributed TD(0) Algorithm

```
1. Initialize: Each agent $v$ arbitrarily initializes $\theta_v(0) \in \mathcal{X}$
   and the sequence of stepsizes $\{\alpha(k)\}_{k \in \mathbb{N}}$.
   Set $\hat{\theta}_v(0) = \theta_v(0)$ and $S_v(0) = 0$.
2. Iteration: For $k = 0, 1, \ldots,$ agent $v \in \mathcal{V}$ implements
   a. Exchange $\theta_v(k)$ with agent $u \in \mathcal{N}_v(k)$
   b. Observe a tuple $(s(k), s'(k), r_v(k))$
   c. Execute local updates
         $y_v(k) = \sum_{u \in \mathcal{N}_v(k)} W_{vu}(k)\theta_u(k)$
         $d_v(k) = r_v(k) + \theta_v(k)^T \left(\gamma\phi(s'(k)) - \phi(s(k))\right)$      (10)
         $\theta_v(k+1) = \left[y_v(k) + \alpha(k) d_v(k) \phi(s(k))\right]_\mathcal{X}$
   d. Update the output
         $S_v(k+1) = S_v(k) + \alpha(k)$
         $\hat{\theta}_v(k+1) = \frac{S(k)\hat{\theta}_v(k) + \alpha(k)\theta_v(k)}{S(k+1)}$
```

The update of Eq. (10) has a simple interpretation: agent v first computes y_v by forming a weighted average of its own value θ_v and the values θ_u received from its neighbor $u \in \mathcal{N}_v$, with the goal of seeking consensus on their estimates. Agent v then moves along its own temporal direction $d_v(k)\nabla\tilde{J}(s(k), \theta_v(k))$ to update its estimate, pushing the consensus point toward θ^*. In Eq. (10) each agent v only shares θ_v with its neighbors but not its immediate reward r_v. In a sense, the agents implement in parallel N local TD(0) methods and then combine their estimate through consensus steps to find the global approximate reward $\tilde{J}$.

3.2. Convergence Rates of Distributed TD(0)

We state here the main results of this paper, the convergence rates of the distributed TD(0) algorithm. In particular, we provide an explicit formula for the upper bound on the rates of TD(0) for both constant and diminishing stepsizes. Our bounds mirror the results that we would expect from the ones using distributed SGD for solving convex optimization problems. For ease of exposition, we delay the analysis of these results to Sections 4.2 and 4.3.

Our main results are established based on the assumption that the data tuple $\{(s(k), s'(k), r_v(k)\}$ are sampled i.i.d from stationary distribution for all k and v. However, within a tuple, $s'(k)$ and $r_v(k)$ are dependent on $s(k)$. We note that the i.i.d condition is often assumed in the literature when dealing with the rates of RL algorithms, see for example; (Dalal et al., 2018; Bhandari et al., 2018). Such a condition is not easy to remove since the dependence between samples can make the analysis become extremely complicated in general. One possible way to collect i.i.d samples is to generate independently a number of trajectories and using first-visit methods, see (Bertsekas & Tsitsiklis, 1999). On the other hand, sampling from stationary distribution can be done by taking last samples of a long trajectory.

Moreover, we make the following fairly standard assumptions in the existing literature of consensus and reinforcement learning (Tsitsiklis & Roy, 1997; Dalal et al., 2018; Bhandari et al., 2018; Nedić et al., 2018). To the rest of this paper, we will assume that these assumptions always hold.

Assumption 1. *There exists an integer $\mathcal{B}$ such that the following graph is connected for all positive integers ℓ*

$$(\mathcal{V}\,,\,\mathcal{E}(\ell\mathcal{B})\cup\mathcal{E}(\ell\mathcal{B}+1)\ldots\cup\mathcal{E}((\ell+1)\mathcal{B}-1)).$$

Assumption 2. *There exists a positive constant β such that $\mathbf{W}(k)=[W_{vu}(k)]\in\mathbb{R}^{N\times N}$ is doubly stochastic and $W_{vv}(k)\geq\beta\ \forall v\in\mathcal{V}$. Moreover, $W_{vu}(k)\in[\beta,1)$ if $(v,u)\in\mathcal{N}_v(k)$ otherwise $W_{vu}(k)=0$ for all $v,u\in\mathcal{V}$.*

Assumption 3. *The Markov chain associated with $\mathcal{P}$ is irreducible.*

Assumption 4. *All the local rewards are uniformly bounded, i.e., there exist constants C_v, for all $v\in\mathcal{V}$ such that $|\,\mathcal{R}_v(s,s')\,|\leq C_v$, for all $s,s'\in\mathcal{S}$.*

Assumption 5. *The feature vectors $\{\phi_\ell\}$, for all $\ell\in\{1,\ldots,K\}$, are linearly independent, i.e., the matrix Φ has full column rank. In addition, we assume that all feature vectors $\phi(s)$ are uniformly bounded, i.e., $\|\phi(s)\|\leq 1$.*

Assumption 6. *The convex compact set $\mathcal{X}\subset\mathbb{R}^K$ contains the fixed point θ^* of the projected Bellman equation.*

Assumption 1 ensures the long-term connectivity and information propagation between the agents, while Assumption 2 imposes the underlying topology of $\mathcal{G}(k)$ where each agent only communicates with its neighbors. Assumptions 1 and 2 yield the following condition (Nedić et al., 2018)

$$\|\mathbf{W}(k)\ldots\mathbf{W}(k+\mathcal{B}-1)\mathbf{Q}\Theta\|\leq\eta\|\mathbf{Q}\Theta\|.\ \forall\Theta,\tag{11}$$

In Eq. (11), $\Theta\in\mathbb{R}^{N\times K}$ and $\mathbf{Q}\in\mathbb{R}^{N\times N}$ are defined as

$$\Theta\triangleq\begin{bmatrix}- & \theta_1^T & -\\ \cdots & \cdots & \cdots\\ - & \theta_N^T & -\end{bmatrix},\qquad \mathbf{Q}=\mathbf{I}-\frac{1}{N}\mathbf{1}\mathbf{1}^T,\tag{12}$$

where $\mathbf{I}$ and $\mathbf{1}$ are the identity matrix and the vector in $\mathbb{R}^N$ with all entries equal to 1, respectively. Moreover, denote by $\sigma_2(\mathbf{W}(k))$ the second largest singular value of $\mathbf{W}(k)$ and $\eta\in(0,1)$ a parameter representing the spectral properties of the sequence of graphs $\{\mathcal{G}(k)\}$ defined as

$$\eta=\min\ \left\{1-1/(2N^3),\ \sup_{k\geq 0}\sigma_2(\mathbf{W}(k))\right\}.\tag{13}$$

For convinience, we define $\delta:=\eta^{\frac{1}{\mathcal{B}}}$. Assumption 3 guarantees that there exists a unique stationary distribution π with positive entries, while under Assumption 4 the accumulative reward J^* is well defined. Under Assumption 5, the projection operator Π is well defined. If there are some dependent ϕ_ℓ, we can simply disregard those dependent feature vectors. Moreover, the uniform boundedness of ϕ_ℓ can be guaranteed through feature normalization.

Finally, Assumption 6 is used to guarantee the stability of agents' updates, which is often assumed in the literature of MARL and stochastic approximation, see for example; (Zhang et al., 2018; Borkar, 2008). We note that this projection step is only used for the purpose of our convergence analysis. In practice, we may not need this step to implement Algorithm 1 since the consensus step likely keeps the agents' estimates close to each other while the TD direction drives these estimates to an optimal solution.

Denote by $\sigma_{\min}$ and $\sigma_{\max}$ the smallest and largest singular value of $\mathbf{A}$, respectively. Let $R_0=\max_{\theta\in\mathcal{X}}\|\theta-\theta^*\|$. We now present our first results, the convergence rates of the approximate value function estimated at each agent's output to the optimal value. That is, we provide the speed of convergence of $\tilde{J}(\hat{\theta}_v(k))$ to $\Phi\theta^*$, for each $v\in\mathcal{V}$. These results are established based on proper conditions on stepsizes $\alpha(k)$ chosen independently of the problem's parameters.

Theorem 1. *Let $\theta_v(k)$, for all $v\in\mathcal{V}$, be generated by Algorithm 1. In addition, given the constant $L>0$ in Lemma 1, let β_0 and β_1 be two positive constants defined as*

$$\begin{aligned}\beta_0&=\mathbb{E}\left[\|\bar{\theta}(0)-\theta^*\|^2\right]\\&\quad+\frac{\alpha(0)\mathbb{E}\left[\|\Theta(0)\|\right](L+2N\sigma_{\max}R_0)}{N\eta(1-\delta)}\\ \beta_1&=\frac{4L(L+N\sigma_{\max}R_0)}{N\eta(1-\delta)}.\end{aligned}\tag{14}$$

1. If $\alpha(k)=\alpha$ for some positive constant α then $\forall v\in\mathcal{V}$

$$\|\tilde{J}(\hat{\theta}_v(k))-\tilde{J}(\theta^*)\|_D^2\leq\frac{\beta_0}{\alpha(1-\gamma)}\frac{1}{k+1}+\frac{\beta_1\alpha}{(1-\gamma)}.\tag{15}$$

2. If $\{\alpha(k)\}=1\,/\sqrt{k+1}$ for all $k\geq 0$ then $\forall v\in\mathcal{V}$

$$\|\tilde{J}(\hat{\theta}_v(k))-\tilde{J}(\theta^*)\|_D^2\leq\frac{\beta_0+\beta_1(1+\ln(k+1))}{2(1-\gamma)\sqrt{k+1}}.\tag{16}$$

As shown in Eq. (15), our rate mirrors what we would expect in using distributed SGD for solving a convex optimization problem with a constant stepsize, i.e., the convergence of the function value to a neighborhood around the optimal value occurs at $\mathcal{O}(1\,/\,k+1)$. In addition, the rate of the distributed $\mathsf{TD}(0)$ also depends inversely on $1-\gamma$ and $1-\delta$. Here, the term $1\,/\,(1-\gamma)$ is expected, as can been seen from Eq. (5). Moreover, $1-\delta$ is the spectral gap of $\mathbf{W}$ and its inverse represents the connectivity of the underlying communication graph between agents. For different graphs, we have different values of δ, see for example (Nedić et al., 2018). Similar observation holds for the case of time-varying stepsizes $\alpha(k) = 1\,/\,\sqrt{k+1}$, where we would expect an asymptotic rate at $\mathcal{O}(1\,/\,\sqrt{k+1})$, with the same dependence on the inverse of $1-\gamma$ and $1-\delta$.

Second, we derive the convergence rate of $\hat{\theta}_v(k)$, for all $v\in\mathcal{V}$, to the optimal solution θ^*, where $\sigma_{\min}$ of $\mathbf{A}$ is assumed to be known. In particular, the stepsizes $\alpha(k)$ are chosen based on this $\sigma_{\min}$. We again observe the same rates as we would expect from using distributed SGD for solving strongly convex optimization problems. In addition, these rates depend on the condition number $\sigma_{\max}/\sigma_{\min}$ of $\mathbf{A}$, as often observed in distributed SGD.

Theorem 2. *Let $\theta_v(k)$, for all $v\in\mathcal{V}$, be generated by Algorithm 1. In addition, given the constant $L>0$ in Lemma 1, let β_2 and β_3 be two positive constants defined as*

$$\begin{aligned}\beta_2 &= \frac{4(L+N\sigma_{\max}R_0)\mathbb{E}\left[\|\Theta(0)\|\right]}{N\eta}\\ \beta_3 &= \frac{16L(L+N\sigma_{\max}R_0)}{N\eta(1-\delta)}.\end{aligned}\tag{17}$$

1 *If* $\alpha(k)=\alpha\in(0,1/\sigma_{\min})$ *then* $\forall v\in\mathcal{V}$

$$\begin{aligned}\mathbb{E}[\|\theta_v(k)-\theta^*\|^2] &\leq 2\mathbb{E}\Big(\|\bar{\theta}(0)-\theta^*\|^2+2\|\Theta(0)\|^2\Big)\rho^k\\ &\quad+\frac{\beta_2}{1-\rho}\alpha+\frac{\beta_3}{(1-\rho)(1-\delta)}\alpha^2,\end{aligned}\tag{18}$$

where $\rho=\max\{1-\sigma_{\min}\alpha,\ \delta\}\in(0,1)$.

2 *If* $\alpha(k)=\alpha_0\,/\,(k+1)$ *where* $\alpha_0>1\,/\,\sigma_{\min}$ *then* $\forall v\in\mathcal{V}$

$$\begin{aligned}&\mathbb{E}\left[\|\hat{\theta}_v(k)-\theta^*\|^2\right]\\ &\leq\left(\frac{\beta_2}{2\sigma_{\min}(1-\delta)}+\frac{\alpha_0\beta_3}{4\sigma_{\min}}\right)\frac{\ln(k+1)}{k+1}.\end{aligned}\tag{19}$$

4. Finite-Time Analysis of Distributed $\mathsf{TD}(0)$

In this section, our goal is to provide the proofs of the main results in this paper, that is, the proofs of Theorems 1 and 2. We start by introducing more notation and stating some preliminary results corresponding to the updates of consensus and TD steps.

4.1. Notation and Preliminary Results

Using $\mathbf{A}, b_v$ are given in Eq. (9) we denote by

$$\begin{aligned}h_v(k) &= b_v-\mathbf{A}\theta_v(k)\\ M_v(k) &= d_v(k)\phi(s(k))-[b_v-\mathbf{A}\theta_v(k)],\end{aligned}\tag{20}$$

Then, we rewrite Eq. (10) as

$$\begin{aligned}y_v(k) &= \sum_{u\in\mathcal{N}_v(k)}W_{vu}(k)\theta_u(k)\\ \tilde{\theta}_v(k) &= y_v(k)+\alpha(k)(h_v(k)+M_v(k))\\ e_v(k) &= \tilde{\theta}_v(k)-[\tilde{\theta}_v(k)]_{\mathcal{X}}\\ \theta_v(k+1) &= \Big[\tilde{\theta}_v(k)\Big]_{\mathcal{X}}=\tilde{\theta}_v(k)-e_v(k),\end{aligned}\tag{21}$$

Thus, using $\mathbf{W}(k)$ in Assumption 2 and Θ in Eq. (12), the matrix form of Eq. (21) is

$$\begin{aligned}\mathbf{Y}(k) &= \mathbf{W}(k)\Theta(k)\\ \tilde{\Theta}(k) &= \mathbf{W}(k)\Theta(k)+\alpha(k)(\mathbf{H}(k)+\mathbf{M}(k))\\ \mathbf{E}(k) &= \tilde{\Theta}(k)-[\tilde{\Theta}(k)]_{\mathcal{X}}\\ \Theta(k+1) &= \tilde{\Theta}(k)-\mathbf{E}(k),\end{aligned}\tag{22}$$

where $\mathbf{H}(k)$, $\mathbf{M}(k)$, and $\mathbf{E}(k)$ are the matrices, whose $v-$th rows are $h_v(k)^T$, $M_v(k)^T$, and $e_v(k)^T$, respectively. Moreover, $[\tilde{\Theta}(k)]_{\mathcal{X}}$ is the row-wise projection of $\tilde{\Theta}(k)$. Given the vectors θ_v we denote by $\bar{\theta}$ their average, i.e., $\bar{\theta}\triangleq 1\,/N\,\sum_{v\in\mathcal{V}}\theta_v$. Thus, Assumption 2 and Eq. (22) gives

$$\bar{\theta}(k+1)=\bar{\theta}(k)+\alpha(k)(\bar{h}(k)+\bar{m}(k))-\bar{e}(k),\tag{23}$$

Finally, we provide here some preliminary results, which are useful to derive our main results in the next section. For convenience, we put their proofs in the supplementary document. We first provides an upper bound for the consensus error defined at time k as $\Theta(k)-\mathbf{1}\bar{\theta}(k)^T=\mathbf{Q}\Theta(k)$ in the following lemma, where $\mathbf{Q}$ is given in Eq. (12).

Lemma 1. *Let $\theta_v(k)$, for all $v\in\mathcal{V}$, be generated by Algorithm 1. Let $\{\alpha(k)\}$ be a nonnegative nonincreasing sequence of stepsizes. Then there exists a constant $L>0$ such that*

1. *The consensus error $\mathbf{Q}\Theta(k)$ satisfies*

$$\|\mathbf{Q}\Theta(k)\|\leq\delta^k\frac{\|\Theta(0)\|}{\eta}+\frac{2L}{\eta}\sum_{t=0}^{k-1}\delta^{k-1-t}\alpha(t).\tag{24}$$

2. *In addition, we obtain*

$$\sum_{t=0}^{k}\alpha(t)\|\mathbf{Q}\Theta(t)\|\leq\frac{\alpha(0)\|\Theta(0)\|}{\eta(1-\delta)}+\frac{2L}{\eta(1-\delta)}\sum_{t=0}^{k}\alpha^2(t).\tag{25}$$

Second, we provide an upper bound in expectation for the optimal distance $\|\bar{\theta}(k)-\theta^*\|$.

Lemma 2. *Let $\theta_v(k)$, for all $v\in\mathcal{V}$, be generated by Algorithm 1. In addition, let $\{\alpha(k)\}$ be a nonnegative nonincreasing sequence of stepsizes. Then we have*

$$\begin{aligned}&\mathbb{E}\left[\|\bar{\theta}(k+1)-\theta^*\|^2\right]\\ &\leq\mathbb{E}[\|\bar{\theta}(k)-\theta^*\|^2]+2\alpha(k)\mathbb{E}[(\bar{\theta}(k)-\theta^*)^T\bar{h}(k)]\\ &\qquad+\frac{4L^2\alpha^2(k)}{N}+\frac{2L}{N}\alpha(k)\mathbb{E}[\|\mathbf{Q}\Theta(k)\|].\end{aligned}\tag{26}$$

4.2. Proof of Theorem 1

By Eq. (8) we have $\bar{b} = \mathbf{A}\theta^*$. Thus, Eq. (20) gives

$$\begin{aligned} \bar{h}(k) &= \bar{b} - \mathbf{A}\bar{\theta}(k) = \mathbf{A}(\theta^* - \bar{\theta}(k)) \\ &= \mathbf{A}(\theta^* - \theta_u(k)) + \mathbf{A}(\theta_u(k) - \bar{\theta}(k)). \end{aligned} \tag{27}$$

Recall that $\tilde{J}(\bar{\theta}(k)) = \Phi\bar{\theta}(k)$. In addition, since the data are sampled i.i.d from the stationary distribution, we have $\Phi^T D \Phi = \mathbb{E}\left[\phi(s)\phi(s)^T\right]$. Thus, given a $\theta \in \mathbb{R}^K$ consider

$$\begin{aligned} &\|\tilde{J}(\theta) - \tilde{J}(\theta^*)\|_D^2 = (\theta - \theta^*)^T\Phi^T D\Phi(\theta - \theta^*) \\ &= \mathbb{E}\left[(\theta - \theta^*)^T\phi(s)\phi(s)^T(\theta - \theta^*)\right] \\ &= \mathbb{E}\left[\|\phi(s)^T(\theta - \theta^*)\|^2\right]. \end{aligned} \tag{28}$$

Fix an index $u \in \mathcal{V}$. Using Eq. (27) we consider

$$\begin{aligned} &\mathbb{E}\left[(\bar{\theta}(k) - \theta^*)^T\bar{h}(k)\right] \\ &= \mathbb{E}\left[(\theta_u(k) - \theta^*)^T\bar{h}(k)\right] + \mathbb{E}\left[(\bar{\theta}(k) - \theta_u(k))^T\bar{h}(k)\right] \\ &\overset{(27)}{=} \mathbb{E}\left[(\theta_u(k) - \theta^*)^T\mathbf{A}(\theta^* - \theta_u(k))\right] \\ &\qquad + 2\mathbb{E}\left[(\theta_u(k) - \theta^*)^T\mathbf{A}(\theta_u(k) - \bar{\theta}(k))\right] \\ &\qquad + \mathbb{E}\left[(\bar{\theta}(k) - \theta_u(k))^T\mathbf{A}(\theta_u(k) - \bar{\theta}(k))\right] \\ &\leq \mathbb{E}\left[(\theta_u(k) - \theta^*)^T\mathbf{A}(\theta^* - \theta_u(k))\right] \\ &\qquad + 2\mathbb{E}\left[(\theta_u(k) - \theta^*)^T\mathbf{A}(\theta_u(k) - \bar{\theta}(k))\right] \\ &\leq \mathbb{E}\left[(\theta_u(k) - \theta^*)^T\mathbf{A}(\theta^* - \theta_u(k))\right] \\ &\qquad + 2\sigma_{\max}R_0\mathbb{E}\left[\|\theta_u(k) - \bar{\theta}(k)\|\right], \end{aligned} \tag{29}$$

where the first inequality is due to $\mathbf{A}$ is positive definite. Using the definition of $\mathbf{A}$ in Eq. (9) and Eqs. (27) and (28) we analize the first term on the right-hand side of Eq. (29)

$$\begin{aligned} &\mathbb{E}\left[(\theta_u(k) - \theta^*)^T\mathbf{A}(\theta^* - \theta_u(k))\right] \\ &= \mathbb{E}\left[(\theta_u(k) - \theta^*)^T\phi(s)\Big(\phi(s) - \gamma\phi(s')\Big)^T(\theta^* - \theta_u(k))\right] \\ &= -\mathbb{E}\left[(\theta_u(k) - \theta^*)^T\phi(s)\phi(s)^T(\theta_u(k) - \theta^*)\right] \\ &\quad + \gamma\mathbb{E}\left[(\theta_u(k) - \theta^*)^T\phi(s)\phi(s')^T(\theta_u(k) - \theta^*)\right] \\ &\leq -\mathbb{E}\left[\|\phi(s)^T(\theta_u(k) - \theta^*)\|^2\right] \\ &+ \gamma\sqrt{\mathbb{E}\left[\|\phi(s)^T(\theta_u(k) - \theta^*)\|^2\right]}\sqrt{\mathbb{E}\left[\|\phi(s')^T(\theta_u(k) - \theta^*)\|^2\right]} \\ &= -(1-\gamma)\mathbb{E}\left[\|\phi(s)^T(\theta_u(k) - \theta^*)\|^2\right] \\ &\overset{(28)}{=} -(1-\gamma)\|\tilde{J}(\theta_u(k)) - \tilde{J}(\theta^*)\|_D^2, \end{aligned}$$

where the inequality is due to the Cauchy-Schwarz inequality. Substituting the preceding relation into Eq. (29) and using $\|\theta_u(k) - \bar{\theta}(k)\| \leq \|\mathbf{Q}\Theta(k)\|$ we obtain

$$\begin{aligned} \mathbb{E}\left[(\bar{\theta}(k) - \theta^*)^T\bar{h}(k)\right] &\leq -(1-\gamma)\|\tilde{J}(\theta_u(k)) - \tilde{J}(\theta^*)\|_D^2 \\ &\qquad + 2\sigma_{\max}R_0\mathbb{E}\left[\|\mathbf{Q}\Theta(k)\|\right]. \end{aligned} \tag{30}$$

Using Eq. (30) into Eq. (26) gives

$$\begin{aligned} &\mathbb{E}\left[\|\bar{\theta}(k+1) - \theta^*\|^2\right] \\ &\leq \mathbb{E}\left[\|\bar{\theta}(k) - \theta^*\|^2\right] + \frac{4L^2}{N}\alpha^2(k) \\ &\quad + 2\left(\frac{L}{N} + 2\sigma_{\max}R_0\right)\mathbb{E}\left[\alpha(k)\,\|\mathbf{Q}\Theta(k)\|\right] \\ &\quad - 2(1-\gamma)\alpha(k)\|\tilde{J}(\theta_u(k)) - \tilde{J}(\theta^*)\|_D^2. \end{aligned}$$

Rearranging and summing up both sides of the preceding relation over k from 0 to K for some constant $K > 0$ gives

$$\begin{aligned} &2(1-\gamma)\sum_{k=0}^{K}\alpha(k)\|\tilde{J}(\theta_u(k)) - \tilde{J}(\theta^*)\|_D^2 \\ &\leq \mathbb{E}\left[\|\bar{\theta}(0) - \theta^*\|^2\right] + \frac{4L^2}{N}\sum_{k=0}^{K}\alpha^2(k) \\ &\qquad + \frac{2L + 4N\sigma_{\max}R_0}{N}\sum_{k=0}^{K}\mathbb{E}\left[\alpha(k)\|\mathbf{Q}\Theta(k)\|\right] \\ &\leq \mathbb{E}\left[\|\bar{\theta}(0) - \theta^*\|^2\right] + \frac{2\alpha(0)\mathbb{E}\left[\|\Theta(0)\|\right](L + 2N\sigma_{\max}R_0)}{N\eta(1-\delta)} \\ &\qquad + \frac{8L(L + N\sigma_{\max}R_0)}{N\eta(1-\delta)}\sum_{k=0}^{K}\alpha^2(k), \end{aligned} \tag{31}$$

where the last inequality is due to Eq. (25). We now consider two choices of $\alpha(k)$ with β_0 and β_1 as defined in Eq. (14).

1. Let $\alpha(k) = \alpha > 0$. Dividing Eq. (31) by $2\alpha(1-\gamma)(K+1)$ and using the Jensen's inequality yields Eq. (15).

2. Let $\alpha(k) = 1\,/\,\sqrt{k+1}$. Using the integral test yields

$$\sum_{t=0}^{K}\alpha(k) \geq 2\sqrt{K+1}, \quad \sum_{t=0}^{K}\alpha^2(k) \leq (1 + \ln(K+1)).$$

Thus, dividing Eq. (31) by $2(1-\gamma)\sum_{k=0}^{K}\alpha(k)$ and using the Jensen's inequality give Eq. (16).

4.3. Proof of Theorem 2

Fix a $u \in \mathcal{V}$. Note that $2(x-y)^T\mathbf{A}(y-z) = \|x-z\|_{\mathbf{A}}^2 - \|x-y\|_{\mathbf{A}}^2 - \|z-y\|_{\mathbf{A}}^2$ $\forall x, y, z$. Thus, Eq. (27) gives

$$\begin{aligned} &2\mathbb{E}\left[(\bar{\theta}(k) - \theta^*)^T\bar{h}(k)\right] \\ &= 2\mathbb{E}\left[(\theta_u(k) - \theta^*)^T\mathbf{A}(\theta^* - \bar{\theta}(k))\right] \\ &\qquad + 2\mathbb{E}\left[(\bar{\theta}(k) - \theta_u(k))^T\mathbf{A}(\theta^* - \bar{\theta}(k))\right] \\ &= -\mathbb{E}\left[\|\bar{\theta}(k) - \theta^*\|_{\mathbf{A}}^2\right] - \mathbb{E}\left[\|\theta_u(k) - \theta^*\|_{\mathbf{A}}^2\right] \\ &\qquad + \mathbb{E}\left[\|\bar{\theta}(k) - \theta_u(k)\|_{\mathbf{A}}^2\right] \\ &\qquad + 2\mathbb{E}\left[(\bar{\theta}(k) - \theta_u(k))^T\mathbf{A}(\theta^* - \bar{\theta}(k))\right], \end{aligned}$$

which gives

$$\begin{aligned}
&2\mathbb{E}\left[(\bar{\theta}(k)-\theta^*)^T\bar{h}(k)\right]\\
&= -\mathbb{E}\left[\|\bar{\theta}(k)-\theta^*\|_{\mathbf{A}}^2\right] - \mathbb{E}\left[\|\theta_u(k)-\theta^*\|_{\mathbf{A}}^2\right]\\
&\quad + \mathbb{E}\left[(\bar{\theta}(k)-\theta_u(k))^T\mathbf{A}(\theta^*-\theta_u(k))\right]\\
&\quad + \mathbb{E}\left[(\bar{\theta}(k)-\theta_u(k))^T\mathbf{A}(\theta^*-\bar{\theta}(k))\right]\\
&\le -\sigma_{\min}\mathbb{E}\left[\|\bar{\theta}(k)-\theta^*\|^2\right] - \sigma_{\min}\mathbb{E}\left[\|\theta_u(k)-\theta^*\|^2\right]\\
&\quad + 2R_0\sigma_{\max}\mathbb{E}\left[\|\bar{\theta}(k)-\theta_u(k)\|\right].
\end{aligned}\tag{32}$$

Using Eqs. (32) into Eq. (26) gives

$$\begin{aligned}
&\mathbb{E}\left[\|\bar{\theta}(k+1)-\theta^*\|^2\right]\\
&\le \mathbb{E}\left[\|\bar{\theta}(k)-\theta^*\|^2\right] + 2\alpha(k)\mathbb{E}\left[(\bar{\theta}(k)-\theta^*)^T\bar{h}(k)\right]\\
&\quad + \frac{4L^2}{N}\alpha^2(k) + \frac{2L}{N}\mathbb{E}\left[\alpha(k)\,\|\mathbf{Q}\Theta(k)\|\right]\\
&\le (1-\sigma_{\min}\alpha(k))\,\mathbb{E}\left[\|\bar{\theta}(k)-\theta^*\|^2\right] + \frac{4L^2}{N}\alpha^2(k)\\
&\quad + \frac{2(L+N\sigma_{\max}R_0)}{N}\mathbb{E}\left[\alpha(k)\,\|\mathbf{Q}\Theta(k)\|\right]\\
&\quad - \sigma_{\min}\alpha(k)\mathbb{E}\left[\|\theta_u(k)-\theta^*\|^2\right].
\end{aligned}\tag{33}$$

We now consider two choices of stepsizes $\alpha(k)$ with β_2,β_3 given in Eq. (17) as follows.

1. Let $\alpha(k)=\alpha\in(0,1\,/\,\sigma_{\min})$ and recall that $\rho=\max\{1-\sigma_{\min}\alpha,\ \delta\}\in(0,1)$. In addition, Eq. (24) yields

$$\|\mathbf{Q}\Theta(k)\| \le \frac{\|\Theta(0)\|}{\eta} + \frac{2L\alpha}{\eta(1-\delta)}\cdot$$

Thus, recursively updating Eq. (33), dropping the negative term, and using the preceding relation yield

$$\begin{aligned}
&\mathbb{E}\left[\|\bar{\theta}(k+1)-\theta^*\|^2\right]\\
&\le \rho^{k+1}\mathbb{E}\left[\|\bar{\theta}(0)-\theta^*\|^2\right] + \frac{4L^2}{N}\alpha^2\sum_{t=0}^{k}\rho^{k-t}\\
&\quad + \frac{2(L+N\sigma_{\max}R_0)}{N\eta}\mathbb{E}\left[\|\Theta(0)\|\right]\alpha\sum_{t=0}^{k}\rho^{k-t}\\
&\quad + \frac{L+N\sigma_{\max}R_0}{N\eta}\frac{4L\alpha^2}{1-\delta}\sum_{t=0}^{k}\rho^{k-t}\\
&\le \rho^{k+1}\mathbb{E}\left[\|\bar{\theta}(0)-\theta^*\|^2\right]\\
&\quad + \frac{2(L+N\sigma_{\max}R_0)}{N\eta}\frac{2\mathbb{E}\left[\|\Theta(0)\|\right]\alpha}{1-\rho}\\
&\quad + \frac{4L(2L+N\sigma_{\max}R_0)}{N\eta(1-\delta)}\frac{\alpha^2}{1-\rho}\cdot
\end{aligned}\tag{34}$$

On the other hand, Eq. (24) yields

$$\mathbb{E}\left[\|\mathbf{Q}\Theta(k)\|^2\right] \le 2\mathbb{E}\left[\|\Theta(0)\|^2\right]\delta^{2k} + \frac{8L^2\alpha^2}{(1-\delta)^2}\cdot\tag{35}$$

Thus, we obtain Eq. (18) by using Eqs. (34) and (35), and $\mathbb{E}[\|\theta_u(k)-\theta^*\|^2]\le 2\mathbb{E}[\|\bar{\theta}(k)-\theta^*\|^2]+2\mathbb{E}[\|\theta_u(k)-\bar{\theta}(k)\|^2]$.

2. Let $\alpha(k)=\alpha_0\,/\,(k+1)$ where $\alpha_0>1\,/\,\sigma_{\min}$, implying $1-\sigma_{\min}\alpha(k)\le k/(k+1)$. Thus, Eq. (33) gives

$$\begin{aligned}
&\mathbb{E}\left[\|\bar{\theta}(k+1)-\theta^*\|^2\right]\\
&\le \frac{k}{k+1}\mathbb{E}\left[\|\bar{\theta}(k)-\theta^*\|^2\right] + \frac{4L^2\alpha_0^2}{N}\frac{1}{(k+1)^2}\\
&\quad + \frac{2\alpha_0(L+N\sigma_{\max}R_0)}{N}\frac{\mathbb{E}\left[\|\mathbf{Q}\Theta(k)\|\right]}{k+1}\\
&\quad - \alpha(0)\sigma_{\min}\frac{\mathbb{E}\left[\|\theta_u(k)-\theta^*\|^2\right]}{k+1}\\
&\le \frac{4L^2\alpha_0^2}{N}\sum_{t=0}^{k}\frac{1}{(t+1)}\frac{1}{k+1}\\
&\quad + \frac{2\alpha_0(L+N\sigma_{\max}R_0)}{N}\frac{\sum_{t=0}^{k}\mathbb{E}\left[\|\mathbf{Q}\Theta(t)\|\right]}{k+1}\\
&\quad - \alpha_0\sigma_{\min}\frac{\sum_{t=0}^{k}\mathbb{E}\left[\|\theta_u(k)-\theta^*\|^2\right]}{k+1}.
\end{aligned}\tag{36}$$

The integral test gives $\sum_{t=0}^{k}1\,/\,(k+1)\le 1+\ln(k+1)$. In addition, Eq. (24) yields

$$\begin{aligned}
&\sum_{t=0}^{k}\mathbb{E}\left[\|\mathbf{Q}\Theta(t)\|\right]\\
&\le \frac{\mathbb{E}\left[\|\mathbf{Q}\Theta(0)\|\right]}{\eta}\sum_{t=0}^{k}\delta^t + \frac{2L}{\eta}\sum_{t=0}^{k}\sum_{\ell=0}^{t}\delta^{t-1-\ell}\alpha(\ell)\\
&\le \frac{\mathbb{E}\left[\|\mathbf{Q}\Theta(0)\|\right]}{\eta(1-\delta)} + \frac{2L}{\eta}\sum_{\ell=0}^{k}\alpha(\ell)\sum_{t=\ell+1}^{k}\delta^t\\
&\le \frac{\mathbb{E}\left[\|\mathbf{Q}\Theta(0)\|\right]}{\eta(1-\delta)} + \frac{2L\alpha_0(1+\ln(k+1))}{\eta(1-\delta)}.
\end{aligned}$$

Thus, using the preceding relation into Eq. (36), rearranging the terms, and using the Jensen's inequality gives Eq. (19).

5. Conclusion and Discussion

In this paper, we consider a distributed consensus-based variant of the popular TD(0) algorithm for estimating the value function of a given stationary policy. Our main contribution is to provide a finite-time analysis for the performance of distributed TD(0), which has not been addressed in the existing literature of MARL. In particular, our results mirror what we would expect from using distributed SGD for solving static convex optimization problems. A few interesting questions left from this work are the finite-time analysis for the general distributed TD(λ) and when the policy is not stationary, e.g., distributed actor-critic methods. We believe that this paper establishes fundamental results that enable one to tackle these problems, which we leave for our future research.

Acknowledgements

This work is partially supported by ARL DCIST CRA W911NF-17-2-0181 and the SRC JUMP program.

References

Abbeel, P., Coates, A., Quigley, M., and Ng, A. An application of reinforcement learning to aerobatic helicopter flight. In *Advances in Neural Information Processing Systems 19*, pp. 1–8. 2007.

Bennis, M., Perlaza, S. M., Blasco, P., Han, Z., and Poor, H. V. Self-organization in small cell networks: A reinforcement learning approach. *IEEE Transactions on Wireless Communications*, 12(7):3202–3212, 2013.

Bertsekas, D. and Tsitsiklis, J. *Neuro-Dynamic Programming*. Athena Scientific, Belmont, MA, 2nd edition, 1999.

Bhandari, J., Russo, D., and Singal, R. A finite time analysis of temporal difference learning with linear function approximation. In *COLT*, 2018.

Borkar, V. *Stochastic Approximation: A Dynamical Systems Viewpoint*. Cambridge University Press, 2008.

Borkar, V. and Meyn, S. The o.d.e. method for convergence of stochastic approximation and reinforcement learning. *SIAM Journal on Control and Optimization*, 38(2):447–469, 2000.

Bradtke, S. and Barto, A. Linear least-squares algorithms for temporal difference learning. *Machine Learning*, 22 (1):33–57, Mar 1996.

Chen, C., Seff, A., Kornhauser, A., and Xiao, J. Deepdriving: Learning affordance for direct perception in autonomous driving. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, pp. 2722–2730, Washington, DC, USA, 2015.

Cortes, J., Martinez, S., Karatas, T., and Bullo, F. Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation*, 20(2):243–255, 2004.

Dalal, G., Szörényi, B., Thoppe, G., and Mannor, S. Finite sample analyses for td(0) with function approximation. In *AAAI*, 2018.

Dayan, P. The convergence of TD(λ) for general λ, 1992.

Gu, S., Holly, E., Lillicrap, T., and Levine, S. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3389–3396, 2017.

Gurvits, L., Lin, L. J., and Hanson, S. J. Incremental learning of evaluation functions for absorbing Markov chains: New methods and theorems, 1994.

Kar, S., Moura, J. M. F., and Poor, H. V. Qd-learning: A collaborative distributed strategy for multi-agent reinforcement learning through consensus + innovations. *IEEE Trans. Signal Processing*, 61:1848–1862, 2013.

Liu, B., Liu, J., Ghavamzadeh, M., Mahadevan, S., and Petrik, M. Finite-sample analysis of proximal gradient td algorithms. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, pp. 504–513, 2015.

Macua, S. V., Chen, J., Zazo, S., and Sayed, A. H. Distributed policy evaluation under multiple behavior strategies. *IEEE Transactions on Automatic Control*, 60(5): 1260–1274, 2015.

Mathkar, A. and Borkar, V. S. Distributed reinforcement learning via gossip. *IEEE Transactions on Automatic Control*, 62(3):1465–1470, 2017.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A., Veness, J., G. Bellemare, M., Graves, A., Riedmiller, M., K. Fidjeland, A., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., and Hassabis, D. Human-level control through deep reinforcement learning. *Nature*, 518:529–33, 02 2015.

Nedić, A. and Bertsekas, D. P. Least squares policy evaluation algorithms with linear function approximation. *Discrete Event Dynamic Systems*, 13(1):79–110, Jan 2003.

Nedić, A., Olshevsky, A., and Rabbat, M. G. Network topology and communication-computation tradeoffs in decentralized optimization. *Proceedings of the IEEE*, 106 (5):953–976, 2018.

Ogren, P., Fiorelli, E., and Leonard, N. E. Cooperative control of mobile sensor networks:adaptive gradient climbing in a distributed environment. *IEEE Transactions on Automatic Control*, 49(8):1292–1302, 2004.

Pineda, F. J. Mean-field theory for batched TD(λ). *Neural Computation*, 9:1403–1419, 1997.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T. P., Leach, M., Kavukcuoglu, K., Graepel, T., and Hassabis, D. Mastering the game of go with deep neural networks and tree search. *Nature*, 529:484–489, 2016.

Stanković, M. S. and Stanković, S. S. Multi-agent temporal-difference learning with linear function approximation: Weak convergence under time-varying network topologies. In *2016 American Control Conference (ACC)*, pp. 167–172, 2016.

Sutton, R. S. Learning to predict by the methods of temporal differences. *Machine Learning*, 3(1):9–44, Aug 1988.

Sutton, R. S. and Barto, A. G. *Reinforcement Learning: An Introduction*. MIT Press, 1st edition, 1998.

Sutton, R. S., Maei, H. R., Precup, D., Bhatnagar, S., Silver, D., Szepesvári, C., and Wiewiora, E. Fast gradient-descent methods for temporal-difference learning with linear function approximation. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 993–1000, 2009a.

Sutton, R. S., Maei, H. R., and Szepesvári, C. A convergent o(n) temporal-difference algorithm for off-policy learning with linear function approximation. In *Advances in Neural Information Processing Systems 21*, pp. 1609–1616. 2009b.

Szepesvari, C. *Algorithms for Reinforcement Learning*. Morgan and Claypool Publishers, 2010.

Tesauro, G. Temporal difference learning and td-gammon. *Commun. ACM*, 38(3):58–68, 1995.

Thoppe, G. and Borkar, V. S. A Concentration Bound for Stochastic Approximation via Alekseev's Formula. Available at: https://arxiv.org/abs/1506.08657.

Tsitsiklis, J. N. and Roy, B. V. An analysis of temporal-difference learning with function approximation. *IEEE Transactions on Automatic Control*, 42(5):674–690, 1997.

Tsitsiklis, J. N. and Roy, B. V. Average cost temporal-difference learning. *Automatica*, 35:1799–1808, 1999.

Tu, S. and Recht, B. Least-squares temporal difference learning for the linear quadratic regulator. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, pp. 5012–5021, 2018.

Wai, H.-T., Yang, Z., Wang, Z., and Hong, M. Multi-agent reinforcement learning via double averaging primal-dual optimization. In *Annual Conference on Neural Information Processing Systems*, pp. 9672–9683, 2018.

Yu, H. and Bertsekas, D. P. Convergence results for some temporal difference methods based on least squares. *IEEE Transactions on Automatic Control*, 54(7):1515–1531, 2009.

Zhang, K., Yang, Z., Liu, H., Zhang, T., and Basar, T. Fully decentralized multi-agent reinforcement learning with networked agents. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5872–5881, 2018.