Global Optimality of Single-Timescale Actor-Critic under Continuous State-Action Space: A Study on Linear Quadratic Regulator

Xuyang Chen[1] , **Jingliang Duan**[2] , **Lin Zhao**[1]
[1]National University of Singapore
[2]University of Science and Technology Beijing
chenxuyang@u.nus.edu, duanjl@ustb.edu.cn, elezhli@nus.edu.sg

Abstract

Actor-critic methods have achieved state-of-the-art performance in various challenging tasks. However, theoretical understandings of their performance remain elusive and challenging. Existing studies mostly focus on practically uncommon variants such as double-loop or two-timescale stepsize actor-critic algorithms for simplicity. These results certify local convergence on finite state- or action-space only. We push the boundary to investigate the classic single-sample single-timescale actor-critic on continuous (infinite) state-action space, where we employ the canonical linear quadratic regulator (LQR) problem as a case study. We show that the popular single-timescale actor-critic can attain an epsilon-optimal solution with an order of epsilon to -2 sample complexity for solving LQR on the demanding continuous state-action space. Our work provides new insights into the performance of single-timescale actor-critic, which further bridges the gap between theory and practice.

1 Introduction

Actor-critic (AC) methods achieved substantial success in solving many difficult reinforcement learning (RL) problems [LeCun *et al.*, 2015; Mnih *et al.*, 2016; Silver *et al.*, 2017]. In addition to a policy update, AC methods employ a parallel critic update to bootstrap the Q-value for policy gradient estimation, which often enjoys reduced variance and fast convergence in training.

Despite the empirical success, theoretical analysis of AC in the most practical form remains challenging. Existing works mostly focus on either the double-loop or the two-timescale variants. In double-loop AC, the actor is updated in the outer loop only after the critic takes sufficiently many steps to have an accurate estimation of the Q-value in the inner loop [Yang *et al.*, 2019; Kumar *et al.*, 2019; Wang *et al.*, 2019]. Hence, the convergence of the critic is decoupled from that of the actor. The analysis is separated into a policy evaluation sub-problem in the inner loop and a perturbed gradient descent in the outer loop. In two-timescale AC, the actor and the critic are updated simultaneously in each iteration using stepsizes of different timescales. The actor stepsize (denoted by α_t in the sequel) is typically smaller than that of the critic (denoted by β_t in the sequel), with their ratio going to zero as the iteration number goes to infinity (i.e., $\lim_{t\to\infty} \alpha_t/\beta_t = 0$). The two-timescale allows the critic to approximate the correct Q-value asymptotically. This special stepsize design essentially decouples the analysis of the actor and the critic.

The aforementioned AC variants are considered mainly for the ease of analysis, which, however, are uncommon in practical implementations. In practice, the single-timescale AC, where the actor and the critic are updated simultaneously using constantly proportional stepsizes (i.e., with $\alpha_t/\beta_t = c > 0$), is more favorable due to its simplicity of implementation and empirical sample efficiency [Schulman *et al.*, 2015; Mnih *et al.*, 2016]. For online learning, the actor and the critic update only once with a single sample in each iteration using proportional stepsizes. This single-sample single-timescale AC is the most classic AC algorithm extensively discussed in the literature and introduced in [Sutton and Barto, 2018]. However, its analysis is significantly more difficult than other variants, primarily due to the more inaccurate value estimation of the critic update and the stronger coupling between critic and actor. More recent works [Chen *et al.*, 2021; Olshevsky and Gharesifard, 2023; Chen and Zhao, 2022] investigated its local convergence and on the finite state- or action-space only. Given that most practical applications in real world are of continuous state-action space, it is demanding to ask the following challenging question:

Can the classic single-sample single-timescale AC find a global optimal policy on continuous state-action space?

To this end, we take a first step to consider the Linear Quadratic Regulation (LQR), a fundamental continuous state-action space control problem that is commonly employed to study the performance and the limits of RL algorithms [Fazel *et al.*, 2018; Yang *et al.*, 2019; Tu and Recht, 2018; Duan *et al.*, 2023]. We analyze the same classic single-sample single-timescale AC algorithm as those studied in the references listed in Table 1. As compared in Table 1, our result is the first to show the global optimality on continuous (infinite) state-action space, while achieving the sample complexity as the previous studies.

Specifically, we consider the time-average cost, which is a more common case for LQR formulation and more difficult to analyze than the discounted cost. The single-sample

Reference	Setting		Optimality	Sample Complexity
	State Space	action space		
[Chen *et al.*, 2021]	infinite	finite	local	$\mathcal{O}(\epsilon^{-2})$
[Olshevsky and Gharesifard, 2023]	finite	finite	local	$\mathcal{O}(\epsilon^{-2})$
[Chen and Zhao, 2022]	infinite	finite	local	$\mathcal{O}(\epsilon^{-2})$
This Paper	infinite	infinite	global	$\mathcal{O}(\epsilon^{-2})$

Table 1: Comparison with other single-sample single-timescale actor-critic algorithms

single-timescale AC algorithm for solving LQR consists of three parallel updates in each iteration: the cost estimator, the critic, and the actor. Unlike the aforementioned double-loop or two-timescale, there is no specialized design in single-sample single-timescale AC that facilitates a decoupled analysis of its three interconnected updates. In fact, it is both conservative and difficult to bound the three iterations separately. Moreover, the existing perturbed gradient analysis can no longer be applied to establish the convergence of the actor either.

To tackle these challenges in analysis, we instead directly bound the overall interconnected iteration system altogether, without resorting to conservative decoupled analysis. In particular, despite the inaccurate estimation in all three updates, we prove the estimation errors diminish to zero if the (constant) ratio of the stepsizes between the actor and the critic is below a threshold. The identified threshold provides new insights into the practical choices of the stepsizes for single-timescale AC.

Compared with other single-sample single-timescale AC (see Table 1), the state-action space we study is infinite. We emphasize that moving from finite to infinite state-action space is highly nontrivial and requires significant analysis. Existing works [Chen *et al.*, 2021; Chen and Zhao, 2022] derived key intermediate results such as many Lipschitz constants relying on the finite size of the state-action space ($|\mathcal{S}|, |\mathcal{A}|$). These results however become immaterial in the infinite state-action space scenario. Some other analysis [Olshevsky and Gharesifard, 2023] concatenates all state-action pairs to create a finite-dimensional feature matrix. However, this will not be possible when the state-action space is infinite. Consequently, existing analyses are not applicable in our context.

We also distinguish our work from other model-free RL algorithms for solving LQR in Table 2, in addition to AC methods. The zeroth-order methods and the policy iteration method are included for completeness. In particular, we note that [Zhou and Lu, 2023] analyzed the single-timescale AC under a multi-sample setting, where the critics are updated by the least square temporal difference (LSTD) estimator. The idea is still to obtain an accurate policy gradient estimation at each iteration by using sufficient samples (in LSTD), and then follow the common perturbed gradient analysis to prove the convergence of the actor, which decouples the convergence analysis of the actor and the critic. Moreover, the analysis requires a strong assumption on the uniform boundedness of the critic parameters. In comparison, our analysis does not require this assumption and considers the more classic and challenging single-sample setting which is also considered by the previous works as listed in Table 1.

Overall, our contributions are summarized as follows:

• Our work furthers the theoretical understanding of AC on continuous state-action space, which represents the most practical usages. We for the first time show that the single-sample single-timescale AC can provably find the ϵ-accurate global optimum with a sample complexity of $\mathcal{O}(\epsilon^{-2})$ for tasks with unbounded continuous state-action space. The previous works consider the more restricted finite state-action space settings with only local convergence guarantee [Chen *et al.*, 2021; Olshevsky and Gharesifard, 2023; Chen and Zhao, 2022].

• We also contribute to the work of RL on continuous control tasks. It is novel that even with the actor updated by a roughly estimated gradient, the single-sample single-timescale AC algorithm can still find the global optimal policy for LQR, under general assumptions. Compared with all other model-free RL algorithms for solving LQR (see Table 2), our work adopts the simplest single-sample single-timescale structure, which may serve as the first step towards understanding the limits of AC methods on continuous control tasks. In addition, compared with the state-of-the-art double-loop AC for solving LQR [Yang *et al.*, 2019], we improve the sample complexity from $\mathcal{O}(\epsilon^{-5})$ to $\mathcal{O}(\epsilon^{-2})$. We also show the algorithm is much more sample-efficient empirically compared to a few classic works in Experiments, which unveils the practical wisdom of AC algorithm.

1.1 Related Work

In this section, we review the existing works that are most relevant to ours.

Actor-Critic methods. The AC algorithm was proposed by [Konda and Tsitsiklis, 1999]. [Kakade, 2001] extended it to the natural AC algorithm. The asymptotic convergence of AC algorithms has been well established in [Kakade, 2001; Bhatnagar *et al.*, 2009; Castro and Meir, 2010; Zhang *et al.*, 2020]. Many recent works focused on the finite-time convergence of AC methods. Under the double-loop setting, [Yang *et al.*, 2019] established the global convergence of AC methods for solving LQR. [Wang *et al.*, 2019] studied the global convergence of AC methods with both the actor and the critic being parameterized by neural networks. [Kumar *et al.*, 2019] studied the finite-time local convergence of a few AC variants with linear function approximation. Under the two-timescale AC setting, [Wu *et al.*, 2020; Xu *et al.*, 2020] established the finite-time convergence to a stationary point at a sample complexity of $\mathcal{O}(\epsilon^{-2.5})$. Under the single-timescale setting, all the related works [Chen *et al.*, 2021; Olshevsky and Gharesifard, 2023; Chen and Zhao, 2022]

Reference	Algorithm	Structure	
[Fazel *et al.*, 2018]	zeroth-order	double-loop	
[Malik *et al.*, 2019]	zeroth-order		
[Yang *et al.*, 2019]	actor-critic		
[Krauth *et al.*, 2019]	policy iteration	multi-sample	
[Zhou and Lu, 2023]	actor-critic	single-timescale	multi-sample
This paper	actor-critic	single-timescale	single-sample

Table 2: Comparison with other model-free RL algorithms for solving LQR.

have been reviewed in the Introduction.

RL algorithms for LQR. RL algorithms in the context of LQR have seen increased interest in the recent years. These works can be mainly divided into two categories: model-based methods [Dean *et al.*, 2018; Mania *et al.*, 2019; Cohen *et al.*, 2019; Dean *et al.*, 2020] and model-free methods. Our main interest lies in the model-free methods. Notably, [Fazel *et al.*, 2018] established the first global convergence result for LQR under the policy gradient method using zeroth-order optimization. [Krauth *et al.*, 2019] studied the convergence and sample complexity of the LSTD policy iteration method under the LQR setting. On the subject of adopting AC to solve LQR, [Yang *et al.*, 2019] provided the first finite-time analysis with convergence guarantee and sample complexity under the double-loop setting. [Zhou and Lu, 2023] considered the multi-sample (LSTD) and single-timescale setting. For the more practical yet challenging single-sample single-timescale AC, there is no such theoretical guarantee so far, which is the focus of this paper.

Notation. We use non-bold letters to denote scalars and use lower and upper case bold letters to denote vectors and matrices respectively. We also use $\|\boldsymbol{\omega}\|$ to denote the ℓ_2-norm of a vector $\boldsymbol{\omega}$, $\|\boldsymbol{A}\|$ to denote the spectral norm of a matrix $\boldsymbol{A}$, and $\|\boldsymbol{A}\|_F$ to denote the Frobenius norm of a matrix $\boldsymbol{A}$. We use $\mathrm{Tr}(\cdot)$ to denote the trace of a matrix. For any symmetric matrix $\boldsymbol{M} \in \mathbb{R}^{n\times n}$, let $\mathrm{svec}(\boldsymbol{M}) \in \mathbb{R}^{n(n+1)/2}$ denote the vectorization of the upper triangular part of $\boldsymbol{M}$ such that $\|\boldsymbol{M}\|_F^2 = \langle \mathrm{svec}(\boldsymbol{M}), \mathrm{svec}(\boldsymbol{M})\rangle$. Besides, let $\mathrm{smat}(\cdot)$ denote the inverse of $\mathrm{svec}(\cdot)$ so that $\mathrm{smat}(\mathrm{svec}(\boldsymbol{M})) = \boldsymbol{M}$. Finally, we denote by $\boldsymbol{A}\otimes_s\boldsymbol{B}$ the symmetric Kronecker product [Schacke, 2004] of two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$.

2 Preliminaries

In this section, we introduce the AC algorithm and provide the theoretical background of LQR.

2.1 Actor-Critic Algorithms

We consider the reinforcement learning for the standard Markov Decision Process (MDP) defined by $(\mathcal{X},\mathcal{U},\mathcal{P},c)$, where $\mathcal{X}$ is the state space, $\mathcal{U}$ is the action space, $\mathcal{P}(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t,\boldsymbol{u}_t)$ denotes the transition kernel that the agent transits to state $\boldsymbol{x}_{t+1}$ after taking action $\boldsymbol{u}_t$ at current state $\boldsymbol{x}_t$, and $c(\boldsymbol{x}_t,\boldsymbol{u}_t)$ is the running cost. A policy $\pi_{\boldsymbol{\theta}}(\boldsymbol{u}|\boldsymbol{x})$ parameterized by $\boldsymbol{\theta}$ is defined as a mapping from a given state to a probability distribution over actions.

In this paper, we aim to find a policy $\pi_{\boldsymbol{\theta}}$ that minimizes the infinite-horizon time-average cost, which is given by

$$J(\boldsymbol{\theta}) := \lim_{T\to\infty} \mathbb{E}_{\boldsymbol{\theta}} \frac{\sum_{t=0}^{T} c(\boldsymbol{x}_t,\boldsymbol{u}_t)}{T} = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\rho_{\boldsymbol{\theta}},\boldsymbol{u}\sim\pi_{\boldsymbol{\theta}}}[c(\boldsymbol{x},\boldsymbol{u})], \tag{1}$$

where $\rho_{\boldsymbol{\theta}}$ denotes the stationary state distribution generated by policy $\pi_{\boldsymbol{\theta}}$. In the time-average cost setting, the state-action value (Q-value) of policy $\pi_{\boldsymbol{\theta}}$ is defined as

$$Q_{\boldsymbol{\theta}}(\boldsymbol{x},\boldsymbol{u}) = \mathbb{E}_{\boldsymbol{\theta}}[\sum_{t=0}^{\infty}(c(\boldsymbol{x}_t,\boldsymbol{u}_t) - J(\boldsymbol{\theta}))|\boldsymbol{x}_0 = \boldsymbol{x}, \boldsymbol{u}_0 = \boldsymbol{u}],$$

which describes the accumulated differences between running costs and average cost for selecting $\boldsymbol{u}$ in state $\boldsymbol{x}$ and thereafter following policy $\pi_{\boldsymbol{\theta}}$ [Sutton and Barto, 2018]. Based on this definition, we can use the policy gradient theorem [Sutton *et al.*, 1999] to express the gradient of $J(\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$ as

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}\sim\rho_{\boldsymbol{\theta}},\boldsymbol{u}\sim\pi_{\boldsymbol{\theta}}}[\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(\boldsymbol{u}|\boldsymbol{x}) Q_{\boldsymbol{\theta}}(\boldsymbol{x},\boldsymbol{u})]. \tag{2}$$

One can also choose to update the policy using the natural policy gradient [Kakade, 2001], which is given by

$$\nabla^N_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = F(\boldsymbol{\theta})^{\dagger} \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}). \tag{3}$$

where

$$F(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}\sim\rho_{\boldsymbol{\theta}},\boldsymbol{u}\sim\pi_{\boldsymbol{\theta}}}[\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(\boldsymbol{u}|\boldsymbol{x}) \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(\boldsymbol{u}|\boldsymbol{x})^{\top}]$$

is the Fisher information matrix and $F(\boldsymbol{\theta})^{\dagger}$ denotes its Moore Penrose pseudoinverse.

Optimizing $J(\boldsymbol{\theta})$ in (1) with (2) requires evaluating the Q-value of the current policy $\pi_{\boldsymbol{\theta}}$, which is usually unknown. AC estimates both the Q-value and the policy. The critic update approximates Q-value towards the actual value of the current policy $\pi_{\boldsymbol{\theta}}$ using temporal difference (TD) learning [Sutton and Barto, 2018]. The actor improves the policy to reduce the time-average cost $J(\boldsymbol{\theta})$ via policy gradient descent. Note that the AC with a natural policy gradient is also known as natural AC, which is a variant of AC.

2.2 Actor-Critic for Linear Quadratic Regulator

In this paper, we aim to demystify the convergence property of AC by focusing on the infinite-horizon time-average linear quadratic regulator (LQR) problem:

$$\begin{aligned} \underset{\{\boldsymbol{u}_t\}}{\text{minimize}} \quad & J(\{\boldsymbol{u}_t\}) := \lim_{T\to\infty} \frac{1}{T}\mathbb{E}[\sum_{t=1}^{T} \boldsymbol{x}_t^{\top}\boldsymbol{Q}\boldsymbol{x}_t + \boldsymbol{u}_t^{\top}\boldsymbol{R}\boldsymbol{u}_t] \\ \text{subject to} \quad & \boldsymbol{x}_{t+1} = \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{B}\boldsymbol{u}_t + \boldsymbol{\epsilon}_t, \end{aligned} \tag{4}$$

where $\boldsymbol{x}_t \in \mathbb{R}^d$ is the state and $\boldsymbol{u}_t \in \mathbb{R}^k$ is the control action at time t; $\boldsymbol{A} \in \mathbb{R}^{d\times d}$ and $\boldsymbol{B} \in \mathbb{R}^{d\times k}$ are system matrices, and the $(\boldsymbol{A}, \boldsymbol{B})$-pair is stabilizable; $\boldsymbol{Q} \in \mathbb{S}^{d\times d}$ and $\boldsymbol{R} \in \mathbb{S}^{k\times k}$ are symmetric positive definite performance matrices, and hence, the $(\boldsymbol{A}, \boldsymbol{Q}^{1/2})$-pair is immediately observable; $\boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \boldsymbol{D_0})$ are i.i.d Gaussian random variables with positive definite covariance $\boldsymbol{D_0} \succ 0$. From the optimal control theory [Anderson and Moore, 2007], the optimal policy of (4) is a linear feedback of the state

$$\boldsymbol{u_t} = -\boldsymbol{K}^*\boldsymbol{x}_t, \tag{5}$$

where $\boldsymbol{K}^* \in \mathbb{R}^{k\times d}$ is the optimal policy which can be uniquely found by solving an Algebraic Riccati Equation (ARE) [Anderson and Moore, 2007] depending on $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{Q}$, $\boldsymbol{R}$. This means that finding $\boldsymbol{K}^*$ using ARE relies on the complete model knowledge.

In the sequel, we pursue finding the optimal policy in a *model-free* way by using the AC method, without knowing or estimating $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{Q}, \boldsymbol{R}$. The structure of the optimal policy in (5) allows us to reformulate (4) as a static optimization problem over all feasible policy matrix $\boldsymbol{K} \in \mathbb{R}^{k\times d}$. To encourage exploration, we parameterize the policy as

$$\{\pi_{\boldsymbol{K}}(\cdot|\boldsymbol{x}) = \mathcal{N}(-\boldsymbol{K}\boldsymbol{x}, \sigma^2\boldsymbol{I}_k), \boldsymbol{K} \in \mathbb{R}^{k\times d}\}, \tag{6}$$

where $\mathcal{N}(\cdot,\cdot)$ denotes the Gaussian distribution and $\sigma > 0$ is the standard deviation of the exploration noise. In other words, given a state $\boldsymbol{x}_t$, the agent will take an action $\boldsymbol{u}_t$ according to $\boldsymbol{u}_t = -\boldsymbol{K}\boldsymbol{x}_t + \sigma\boldsymbol{\zeta}_t$, where $\boldsymbol{\zeta}_t \sim \mathcal{N}(0, \boldsymbol{I}_k)$. As a consequence, the optimization problem defined in (4) under policy (6) can be reformulated as

$$\underset{\boldsymbol{K}}{\text{minimize}}\ J(\boldsymbol{K}) := \lim_{T\to\infty}\frac{1}{T}\mathbb{E}[\sum_{t=1}^{T}\boldsymbol{x}_t^\top\boldsymbol{Q}\boldsymbol{x}_t + \boldsymbol{u}_t^\top\boldsymbol{R}\boldsymbol{u}_t] \tag{7}$$

subject to

$$\begin{aligned}\boldsymbol{u}_t &= -\boldsymbol{K}\boldsymbol{x}_t + \sigma\boldsymbol{\zeta}_t,\\ \boldsymbol{x}_{t+1} &= \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{B}\boldsymbol{u}_t + \boldsymbol{\epsilon}_t.\end{aligned} \tag{8}$$

Therefore, the closed-loop form of system (8) is given by

$$\boldsymbol{x}_{t+1} = (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})\boldsymbol{x}_t + \boldsymbol{\xi}_t, \tag{9}$$

where $\boldsymbol{\xi}_t = \boldsymbol{\epsilon}_t + \sigma\boldsymbol{B}\boldsymbol{\zeta}_t \sim \mathcal{N}(0, \boldsymbol{D_\sigma})$ with $\boldsymbol{D_\sigma} = \boldsymbol{D_0} + \sigma^2\boldsymbol{B}\boldsymbol{B}^\top$. Note that optimizing over the set of stochastic policies (6) will lead to the same optimal $\boldsymbol{K}^*$. From (9), a policy $\boldsymbol{K}$ is stabilizing if and only if $\rho(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}) < 1$, where $\rho(\cdot)$ denotes the spectral radius. It is well known that if $\boldsymbol{K}$ is stabilizing, the Markov chain in (9) yields a stationary state distribution $\rho_{\boldsymbol{K}} \sim \mathcal{N}(0, \boldsymbol{D_K})$, where $\boldsymbol{D_K}$ satisfies the following Lyapunov equation (by taking the variance of (9))

$$\boldsymbol{D_K} = \boldsymbol{D}_\sigma + (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})\boldsymbol{D_K}(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})^\top. \tag{10}$$

Similarly, we define $\boldsymbol{P_K}$ as the unique positive definite solution to (Bellman equation under $\boldsymbol{K}$)

$$\boldsymbol{P_K} = \boldsymbol{Q} + \boldsymbol{K}^\top\boldsymbol{R}\boldsymbol{K} + (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})^\top\boldsymbol{P_K}(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}). \tag{11}$$

Based on $\boldsymbol{D_K}$ and $\boldsymbol{P_K}$, the following lemma characterizes $J(\boldsymbol{K})$ and its gradient $\nabla_{\boldsymbol{K}}J(\boldsymbol{K})$.

Lemma 1 ([Yang *et al.*, 2019]). *For any stabilizing policy $\boldsymbol{K}$, the time-average cost $J(\boldsymbol{K})$ and its gradient $\nabla_{\boldsymbol{K}}J(\boldsymbol{K})$ take the following forms*

$$J(\boldsymbol{K}) = \mathrm{Tr}(\boldsymbol{P_K}\boldsymbol{D}_\sigma) + \sigma^2\mathrm{Tr}(\boldsymbol{R}), \tag{12a}$$

$$\nabla_{\boldsymbol{K}}J(\boldsymbol{K}) = 2\boldsymbol{E_K}\boldsymbol{D_K}, \tag{12b}$$

where $\boldsymbol{E_K} := (\boldsymbol{R} + \boldsymbol{B}^\top\boldsymbol{P_K}\boldsymbol{B})\boldsymbol{K} - \boldsymbol{B}^\top\boldsymbol{P_K}\boldsymbol{A}$.

Then, the natural gradient of $J(\boldsymbol{K})$ can be calculated as [Fazel *et al.*, 2018; Yang *et al.*, 2019]

$$\nabla^N_{\boldsymbol{K}}J(\boldsymbol{K}) = \nabla_{\boldsymbol{K}}J(\boldsymbol{K})\boldsymbol{D}^{-1}_{\boldsymbol{K}} = \boldsymbol{E_K}, \tag{13}$$

which eliminates the burden of estimating $\boldsymbol{D_K}$. Note that we omit the constant coefficient since it can be absorbed by the stepsize.

Calculating the natural gradient $\nabla^N_{\boldsymbol{K}}J(\boldsymbol{K})$ requires estimating $\boldsymbol{P_K}$, which depends on $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{Q}, \boldsymbol{R}$. To estimate the gradient without the knowledge of the model, we instead directly utilize the Q-value.

Lemma 2 ([Bradtke *et al.*, 1994; Yang *et al.*, 2019]). *For any stabilizing policy $\boldsymbol{K}$, the Q-value $Q_{\boldsymbol{K}}(\boldsymbol{x}, \boldsymbol{u})$ takes the following form*

$$\begin{aligned}Q_{\boldsymbol{K}}(\boldsymbol{x}, \boldsymbol{u}) = (\boldsymbol{x}^\top, \boldsymbol{u}^\top)\boldsymbol{\Omega_K}\begin{pmatrix}\boldsymbol{x}\\ \boldsymbol{u}\end{pmatrix} - \mathrm{Tr}(\boldsymbol{P_K}\boldsymbol{D_K})\\ - \sigma^2\mathrm{Tr}(\boldsymbol{R} + \boldsymbol{P_K}\boldsymbol{B}\boldsymbol{B}^\top),\end{aligned} \tag{14}$$

where

$$\boldsymbol{\Omega_K} := \begin{bmatrix}\boldsymbol{\Omega}^{11}_{\boldsymbol{K}} & \boldsymbol{\Omega}^{12}_{\boldsymbol{K}}\\ \boldsymbol{\Omega}^{21}_{\boldsymbol{K}} & \boldsymbol{\Omega}^{22}_{\boldsymbol{K}}\end{bmatrix} := \begin{bmatrix}\boldsymbol{Q} + \boldsymbol{A}^\top\boldsymbol{P_K}\boldsymbol{A} & \boldsymbol{A}^\top\boldsymbol{P_K}\boldsymbol{B}\\ \boldsymbol{B}^\top\boldsymbol{P_K}\boldsymbol{A} & \boldsymbol{R} + \boldsymbol{B}^\top\boldsymbol{P_K}\boldsymbol{B}\end{bmatrix}. \tag{15}$$

Clearly, if we can estimate $\boldsymbol{\Omega_K}$, then $\boldsymbol{E_K}$ in (13) can be readily estimated by using $\boldsymbol{\Omega}^{21}_{\boldsymbol{K}}$ and $\boldsymbol{\Omega}^{22}_{\boldsymbol{K}}$, which represent the bottom left corner block and bottom right corner block of matrix $\boldsymbol{\Omega_K}$, respectively.

3 Single-sample Single-timescale Actor-Critic

In this section, we describe the single-sample single-timescale AC algorithm for solving LQR. In view of the structure of the Q-value given in (14) and the fact that [Schacke, 2004]

$$(\boldsymbol{x}^\top, \boldsymbol{u}^\top)\,\boldsymbol{\Omega_K}\begin{pmatrix}\boldsymbol{x}\\ \boldsymbol{u}\end{pmatrix} = \phi(\boldsymbol{x}, \boldsymbol{u})^\top\mathrm{svec}(\boldsymbol{\Omega_K}), \tag{16}$$

where

$$\phi(\boldsymbol{x}, \boldsymbol{u}) := \mathrm{svec}[\begin{pmatrix}\boldsymbol{x}\\ \boldsymbol{u}\end{pmatrix}\begin{pmatrix}\boldsymbol{x}\\ \boldsymbol{u}\end{pmatrix}^\top] \tag{17}$$

and $\mathrm{svec}(\cdot)$ denotes the vectorization of the upper triangular part of a symmetric matrix as defined in [Schacke, 2004]. We can then parameterize the Q-estimator (critic) by

$$\hat{Q}_{\boldsymbol{K}}(\boldsymbol{x}, \boldsymbol{u}; \boldsymbol{\omega}, b) = \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{u})^\top\boldsymbol{\omega} + b,$$

where $\phi(\boldsymbol{x}, \boldsymbol{u})$ defined in (17) is the feature function and $\boldsymbol{\omega}$ is the critic. Using the TD(0) learning, the critic update is followed by

$$\begin{aligned}\boldsymbol{\omega}_{t+1} = \boldsymbol{\omega}_t + \beta_t[(c_t - J(\boldsymbol{K}) + \boldsymbol{\phi}(\boldsymbol{x}_{t+1}, \boldsymbol{u}_{t+1})^\top\boldsymbol{\omega}_t\\ + b - \boldsymbol{\phi}(\boldsymbol{x}_t, \boldsymbol{u}_t)^\top\boldsymbol{\omega}_t - b)]\boldsymbol{\phi}(\boldsymbol{x}_t, \boldsymbol{u}_t),\end{aligned} \tag{18}$$

where β_t is the stepsize of the critic and $\boldsymbol{K}$ denotes the policy under which the state-action pairs are sampled. Note that the constant b is not required for updating the linear coefficient $\boldsymbol{\omega}$.

Taking the expectation of $\boldsymbol{\omega}_{t+1}$ in (18) with respect to the stationary distribution, conditioned on $\boldsymbol{\omega}_t$, the expected subsequent critic can be written as

$$\mathbb{E}[\boldsymbol{\omega}_{t+1}|\boldsymbol{\omega}_t] = \boldsymbol{\omega}_t + \beta_t(\boldsymbol{b}_{\boldsymbol{K}} - \boldsymbol{A}_{\boldsymbol{K}}\boldsymbol{\omega}_t), \tag{19}$$

where

$$\begin{aligned} \boldsymbol{A}_{\boldsymbol{K}} &= \mathbb{E}_{(\boldsymbol{x},\boldsymbol{u})}[\boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{u})(\boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{u}) - \boldsymbol{\phi}(\boldsymbol{x}',\boldsymbol{u}'))^\top], \\ \boldsymbol{b}_{\boldsymbol{K}} &= \mathbb{E}_{(\boldsymbol{x},\boldsymbol{u})}[(c(\boldsymbol{x},\boldsymbol{u}) - J(\boldsymbol{K}))\boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{u})]. \end{aligned} \tag{20}$$

Note that for ease of exposition, we denote $(\boldsymbol{x}', \boldsymbol{u}')$ as the next state-action pair after $(\boldsymbol{x}, \boldsymbol{u})$ and abbreviate $\mathbb{E}_{\boldsymbol{x}\sim\rho_{\boldsymbol{K}},\boldsymbol{u}\sim\pi_{\boldsymbol{K}}(\cdot|\boldsymbol{x})}$ as $\mathbb{E}_{(\boldsymbol{x},\boldsymbol{u})}$.

Assumption 1. *We consider the policy class $\mathbb{K}$ such that $\forall \boldsymbol{K} \in \mathbb{K}$, $\boldsymbol{K}$ is norm bounded and the spectral radius satisfies $\rho(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}) \leq \lambda$ for some constant $\lambda \in (0, 1)$.*

The above assumes the uniform boundedness of the policy (actor) parameter $\boldsymbol{K}$, which is common in the literature of actor-critic algorithms [Karmakar and Bhatnagar, 2018; Barakat *et al.*, 2022; Zhou and Lu, 2023]. One potential approach to address the boundedness assumption involves formulating a projection map capable of diminishing the magnitude of $\|\boldsymbol{K}\|$ when it exceeds the specified boundary [Konda and Tsitsiklis, 1999; Bhatnagar *et al.*, 2009], which is deferred to future research endeavors.

As previously discussed, a policy $\boldsymbol{K}$ is considered stabilizing if and only if $\rho(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}) < 1$. Therefore, Assumption 1 also implies the stability of policy $\boldsymbol{K}$, which is equivalent to assuming the existence of $\boldsymbol{A}_{\boldsymbol{K}}$ due to the expectation being taken over the stationary distribution. Such assumption is standard in the literature [Wu *et al.*, 2020; Chen *et al.*, 2021; Olshevsky and Gharesifard, 2023]. Without loss of generality, we slightly strengthen the requirement to $\rho(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}) \leq \lambda$ for some constant $\lambda \in (0, 1)$. This is made to avoid tedious computation of the probability of bounded learning trajectories. It is worth noting that one could alternatively assume $\rho(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}) < 1$ and deduce that the same results presented in the sequel with additional high probability characterization.

We then provide the coercive property of cost function $J(\boldsymbol{K})$, illustrating that $J(\boldsymbol{K})$ tends towards infinity as $\|\boldsymbol{K}\|$ approaches infinity or when $\rho(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})$ approaches 1.

Lemma 3 (Coercive Property)**.** *The cost function $J(\boldsymbol{K})$ defined in* (7) *is coercive, that is, for any sequence $\{\boldsymbol{K}_i\}_{i=1}^{\infty}$ of stabilizing policies, we have*

$$J(\boldsymbol{K}_i) \to +\infty, \quad \textit{if } \|\boldsymbol{K}_i\| \to +\infty \ \textit{ or } \ \rho(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}_i) \to 1.$$

Lemma 3 demonstrates the safety of boundary cutting ($\|\boldsymbol{K}_i\| \to +\infty$, $\rho(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}_i) \to 1$), ensuring that the optimal $\boldsymbol{K}^*$ that minimizes $J(\boldsymbol{K})$ resides within the class $\mathbb{K}$, thereby justifying Assumption 1. Additionally, we present some numerical examples in Section 5 to support this assumption.

As the existence of $\boldsymbol{A}_{\boldsymbol{K}}$ and $\boldsymbol{b}_{\boldsymbol{K}}$ are ensured by Assumption 1, given a policy $\pi_{\boldsymbol{K}}$, it is not hard to show that if the update in (19) has converged to some limiting point $\boldsymbol{\omega}^*_{\boldsymbol{K}}$, i.e., $\lim_{t\to\infty} \boldsymbol{\omega}_t = \boldsymbol{\omega}^*_{\boldsymbol{K}}$, $\boldsymbol{\omega}^*_{\boldsymbol{K}}$ must be the solution of $\boldsymbol{A}_{\boldsymbol{K}}\boldsymbol{\omega} = \boldsymbol{b}_{\boldsymbol{K}}$.

Lemma 4. *Suppose $K \in \mathbb{K}$. Then the matrix $\boldsymbol{A}_{\boldsymbol{K}}$ defined in* (20) *is invertible and $\boldsymbol{A}_{\boldsymbol{K}}\boldsymbol{\omega} = \boldsymbol{b}_{\boldsymbol{K}}$ has a unique solution $\boldsymbol{\omega}^*_{\boldsymbol{K}}$ that satisfies*

$$\boldsymbol{\omega}^*_{\boldsymbol{K}} = \operatorname{svec}(\boldsymbol{\Omega}_{\boldsymbol{K}}). \tag{21}$$

where $\boldsymbol{\Omega}_{\boldsymbol{K}}$ is defined in (15)*.*

Since $\operatorname{smat}(\cdot)$ represents the inverse of $\operatorname{svec}(\cdot)$, it follows that $\boldsymbol{\Omega}_{\boldsymbol{K}}$ can be expressed as $\operatorname{smat}(\boldsymbol{\omega}^*_{\boldsymbol{K}})$, thereby completing the estimation of $\boldsymbol{\Omega}_{\boldsymbol{K}}$.

Combining (13), (15), and (21), we can express the natural gradient of $J(\boldsymbol{K})$ using $\boldsymbol{\omega}^*_{\boldsymbol{K}}$:

$$\nabla^N_{\boldsymbol{K}} J(\boldsymbol{K}) = \boldsymbol{\Omega}^{22}_{\boldsymbol{K}}\boldsymbol{K} - \boldsymbol{\Omega}^{21}_{\boldsymbol{K}} = \operatorname{smat}(\boldsymbol{\omega}^*_{\boldsymbol{K}})^{22}\boldsymbol{K} - \operatorname{smat}(\boldsymbol{\omega}^*_{\boldsymbol{K}})^{21},$$

where $\operatorname{smat}(\boldsymbol{\omega}^*_{\boldsymbol{K}})^{21}$ and $\operatorname{smat}(\boldsymbol{\omega}^*_{\boldsymbol{K}})^{22}$ represent the bottom left corner block and bottom right corner block of matrix $\operatorname{smat}(\boldsymbol{\omega}^*_{\boldsymbol{K}})$, respectively.

This allows us to estimate the natural policy gradient using the critic parameters $\boldsymbol{\omega}_t$, and then update the actor in a model-free manner

$$\boldsymbol{K}_{t+1} = \boldsymbol{K}_t - \alpha_t \widehat{\nabla^N_{\boldsymbol{K}_t} J(\boldsymbol{K}_t)}, \tag{22}$$

where α_t is the actor stepsize and $\widehat{\nabla^N_{\boldsymbol{K}_t} J(\boldsymbol{K}_t)}$ is the natural gradient estimation depending on $\boldsymbol{\omega}_t$:

$$\widehat{\nabla^N_{\boldsymbol{K}_t} J(\boldsymbol{K}_t)} = \operatorname{smat}(\boldsymbol{\omega}_t)^{22}\boldsymbol{K}_t - \operatorname{smat}(\boldsymbol{\omega}_t)^{21}. \tag{23}$$

Furthermore, we introduce a cost estimator η_t to estimate the time-average cost $J(\boldsymbol{K}_t)$. Combining the critic update (18) and the actor update (22)-(23), the single-sample single-timescale AC for solving LQR is listed below.

Algorithm 1 Single-Sample Single-timescale Actor-Critic for Linear Quadratic Regulator

1: **Input** initialize actor parameter $\boldsymbol{K}_0 \in \mathbb{K}$, critic parameter $\boldsymbol{\omega}_0$, time-average cost η_0, stepsizes α_t for actor, β_t for critic, and γ_t for cost estimator.
2: **for** $t = 0, 1, 2, \cdots, T-1$ **do**
3: Sample $\boldsymbol{x}_t$ from the stationary distribution $\rho_{\boldsymbol{K}_t}$.
4: Take action $\boldsymbol{u}_t \sim \pi_{\boldsymbol{K}_t}(\cdot|\boldsymbol{x}_t)$ and receive cost $c_t = c(\boldsymbol{x}_t, \boldsymbol{u}_t)$ and the next state $\boldsymbol{x}'_t$.
5: Obtain $\boldsymbol{u}'_t \sim \pi_{\boldsymbol{K}_t}(\cdot|\boldsymbol{x}'_t)$.
6: $\delta_t = c_t - \eta_t + \boldsymbol{\phi}(\boldsymbol{x}'_t, \boldsymbol{u}'_t)^\top\boldsymbol{\omega}_t - \boldsymbol{\phi}(\boldsymbol{x}_t, \boldsymbol{u}_t)^\top\boldsymbol{\omega}_t$
7: $\eta_{t+1} = \mathit{proj}_{\mathcal{B}_{\bar{\eta}}}(\eta_t + \gamma_t(c_t - \eta_t))$
8: $\boldsymbol{\omega}_{t+1} = \mathit{proj}_{\mathcal{B}_{\bar{\omega}}}(\boldsymbol{\omega}_t + \beta_t\delta_t\boldsymbol{\phi}(\boldsymbol{x}_t, \boldsymbol{u}_t))$
9: $\boldsymbol{K}_{t+1} = \boldsymbol{K}_t - \alpha_t(\operatorname{smat}(\boldsymbol{\omega}_t)^{22}\boldsymbol{K}_t - \operatorname{smat}(\boldsymbol{\omega}_t)^{21})$
10: **end for**

Note that *single-sample* refers to the fact that only one sample is used to update the critic per actor step. Line 3 of Algorithm 1 samples from the stationary distribution induced by the policy $\pi_{\boldsymbol{K}_t}$, which is a mild requirement in the analysis of uniformly ergodic Markov chain, such as in the LQR problem [Yang *et al.*, 2019]. It is only made to simplify the theoretical analysis. Indeed, as shown in [Tu and Recht, 2018], when $\boldsymbol{K} \in \mathbb{K}$, (9) is geometrically β-mixing and thus its distribution converges to the stationary distribution exponentially. In practice, one can run the Markov chain in (9) a sufficient number of steps and sample one state from the last step to approximate the stationary distribution. In addition, *single-timescale* refers to the fact that the stepsizes for the critic and the actor updates are constantly proportional.

Since the update of the critic parameter in (18) requires the time-average cost $J(\boldsymbol{K}_t)$, Line 7 provides an estimation of it. Besides, on top of (18), we additionally introduce a projection in Line 8 and Line 9 to keep the critic norm-bounded. The projection follows the standard definition, i.e., $proj_{\mathcal{B}_y}(\boldsymbol{x})$ means project $\boldsymbol{x}$ to the set $\mathcal{B}_y := \{\boldsymbol{x} | \|\boldsymbol{x}\| \leq y\}$. This is common in the literature [Wu *et al.*, 2020; Yang *et al.*, 2019; Chen and Zhao, 2022]. In our analysis, the projection is relaxed using its nonexpansive property.

4 Main Theory

In this section, we establish the global optimality and analyze the finite-time performance of Algorithm 1. All the proofs can be found in the Supplementary Material.

Theorem 1. *Suppose that Assumptions 1 hold and choose* $\alpha_t = \frac{c}{\sqrt{T}}, \beta_t = \gamma_t = \frac{1}{\sqrt{T}}$, *where c is a small positive constant. It holds that*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}(\eta_t - J(\boldsymbol{K}_t))^2 = \mathcal{O}(\frac{1}{\sqrt{T}}),$$
$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\boldsymbol{\omega}_t - \boldsymbol{\omega}^*_{\boldsymbol{K}_t}\|^2 = \mathcal{O}(\frac{1}{\sqrt{T}}),$$
$$\min_{0\leq t<T}\mathbb{E}[J(\boldsymbol{K}_t) - J(\boldsymbol{K}^*)] = \mathcal{O}(\frac{1}{\sqrt{T}}).$$

The theorem shows that the cost estimator, the critic, and the actor all converge at a sub-linear rate of $\mathcal{O}(T^{-\frac{1}{2}})$. The $\mathcal{O}$ notation hides the polynomials of the dependence parameters. Note that we have explicitly characterized all the necessary problem parameters in the proofs before the last step of the analysis of the interconnected system. One can easily keep all the problem parameters in the interconnected system analysis and get the order for all parameters. To focus on the key factors and for ease of comprehension, we only show the convergence rate in terms of the iteration number.

Correspondingly, to obtain an ϵ-optimal policy, the required sample complexity is $\mathcal{O}(\epsilon^{-2})$. This order is consistent with the existing results on single-sample single-timescale AC [Chen *et al.*, 2021; Olshevsky and Gharesifard, 2023; Chen and Zhao, 2022]. Nevertheless, our result is the first finite-time analysis of the single-sample single-timescale AC with a global optimality guarantee and considers the challenging continuous state-action space.

4.1 Proof Sketch

The main challenge in the finite-time analysis lies in that the estimation errors of the time-average cost, the critic, and the natural policy gradient are strongly coupled. To overcome this issue, we view the propagation of these errors as an interconnected system and analyze them comprehensively. To see the merit of our analysis framework, we sketch the main proof steps of Theorem 1 in the following. The supporting lemmas and theorems mentioned below can be found in the Supplementary Material.

We define three measures A_T, B_T, C_T which denote average values of the cost estimation error, the critic error, and the square norm of natural policy gradient, respectively:

$$A_T := \frac{\sum_{t=0}^{T-1}\mathbb{E}y_t^2}{T}, B_T := \frac{\sum_{t=0}^{T-1}\mathbb{E}\|\boldsymbol{z}_t\|^2}{T}, C_T := \frac{\sum_{t=0}^{T-1}\mathbb{E}\|\boldsymbol{E}_{\boldsymbol{K}_t}\|^2}{T},$$

where $y_t := \eta_t - J(\boldsymbol{K}_t)$ is the cost estimation error and $\boldsymbol{z}_t := \boldsymbol{\omega}_t - \boldsymbol{\omega}^*_t$ with $\boldsymbol{\omega}^*_t := \boldsymbol{\omega}^*_{\boldsymbol{K}_t}$ is the critic error. Note that $\boldsymbol{E}_{\boldsymbol{K}_t} = \nabla^N_{\boldsymbol{K}_t} J(\boldsymbol{K}_t)$ is the natural policy gradient according to (13).

We first derive implicit (coupled) upper bounds for the cost estimation error y_t, the critic error $\boldsymbol{z}_t$, and the natural gradient $\boldsymbol{E}_{\boldsymbol{K}_t}$, respectively. After that, we solve an interconnected system of inequalities in terms of $A_T,\ B_T,\ C_T$ to establish the finite-time convergence.

Step 1: Cost estimation error analysis. From the cost estimator update rule (Line 7 of Algorithm 1), we decompose the cost estimation error into (neglecting the projection for the time being):

$$\begin{aligned} y_{t+1}^2 = &(1-2\gamma_t)y_t^2 + 2\gamma_t y_t(c_t - J(\boldsymbol{K}_t)) \\ &+ 2y_t(J(\boldsymbol{K}_t) - J(\boldsymbol{K}_{t+1})) \\ &+ [J(\boldsymbol{K}_t) - J(\boldsymbol{K}_{t+1}) + \gamma_t(c_t - \eta_t)]^2. \end{aligned} \tag{24}$$

The second term on the right hand side of (24) is a noise term introduced by random sampling of state-action pairs, which reduces to 0 after taking the expectations. The third term is the variation of the moving targets $J(\boldsymbol{K}_t)$ tracked by cost estimator. It is bounded by $y_t, \boldsymbol{z}_t, \boldsymbol{E}_{\boldsymbol{K}_t}$ utilizing the Lipschitz continuity of $J(\boldsymbol{K}_t)$ (Lemma 9), the actor update rule (23), and the Cauchy-Schwartz inequality. The last term reflects the variance in cost estimation, which is bounded by $\mathcal{O}(\gamma_t)$.

Step 2: Critic error analysis. By the critic update rule (Line 8 of Algorithm 1), we decompose the squared error by (neglecting the projection for the time being)

$$\begin{aligned} \|\boldsymbol{z}_{t+1}\|^2 = &\|\boldsymbol{z}_t\|^2 + 2\beta_t\langle \boldsymbol{z}_t, \bar{\boldsymbol{h}}(\boldsymbol{\omega}_t, \boldsymbol{K}_t)\rangle + 2\beta_t\boldsymbol{\Lambda}(O_t, \boldsymbol{\omega}_t, \boldsymbol{K}_t) \\ &+ 2\beta_t\langle \boldsymbol{z}_t, \Delta\boldsymbol{h}(\boldsymbol{O}_t, \eta_t, \boldsymbol{K}_t)\rangle + 2\langle \boldsymbol{z}_t, \boldsymbol{\omega}^*_t - \boldsymbol{\omega}^*_{t+1}\rangle \\ &+ \|\beta_t(\boldsymbol{h}(\boldsymbol{O}_t, \boldsymbol{\omega}_t, \boldsymbol{K}_t) + \Delta\boldsymbol{h}(\boldsymbol{O}_t, \eta_t, \boldsymbol{K}_t)) \\ &+ (\boldsymbol{\omega}^*_t - \boldsymbol{\omega}^*_{t+1})\|^2, \end{aligned} \tag{25}$$

where the definitions of $\boldsymbol{h}, \bar{\boldsymbol{h}}, \Delta\boldsymbol{h}, \boldsymbol{\Lambda}$, and $\boldsymbol{O}_t$ can be found in (28) in the Supplementary Material. The second term on the right hand side of (25) is bounded by $-\mu\|\boldsymbol{z}_t\|^2$, where μ is a lower bound of $\sigma_{\min}(A_{\boldsymbol{K}_t})$ proved in Lemma 10. The third term is a random noise introduced by sampling, which reduces to 0 after taking expectation. The fourth term is caused by inaccurate cost and critic estimations, which can be bounded by the norm of y_t and $\boldsymbol{z}_t$. The fifth term tracks the difference between the drifting critic targets. We control it by the Lipschitz continuity of the critic target established in Lemma 11. The last term reflects the variances of various estimations, which is bounded by $\mathcal{O}(\beta_t)$.

Step 3: Natural gradient norm analysis. From the actor update rule (Line 9 of Algorithm 1) and the almost smoothness property of LQR (Lemma 12), we derive

$$\begin{aligned}&2\mathrm{Tr}(\boldsymbol{D}_{\boldsymbol{K}_{t+1}}\boldsymbol{E}_{\boldsymbol{K}_t}^\top\boldsymbol{E}_{\boldsymbol{K}_t}) = \frac{1}{\alpha_t}[J(\boldsymbol{K}_t) - J(\boldsymbol{K}_{t+1})]\\ &- 2\mathrm{Tr}(\boldsymbol{D}_{\boldsymbol{K}_{t+1}}(\hat{\boldsymbol{E}}_{\boldsymbol{K}_t} - \boldsymbol{E}_{\boldsymbol{K}_t})^\top\boldsymbol{E}_{\boldsymbol{K}_t})\\ &+ \alpha_t\mathrm{Tr}(\boldsymbol{D}_{\boldsymbol{K}_{t+1}}\hat{\boldsymbol{E}}_{\boldsymbol{K}_t}^\top(\boldsymbol{R} + \boldsymbol{B}^\top\boldsymbol{P}_{\boldsymbol{K}_t}\boldsymbol{B})\hat{\boldsymbol{E}}_{\boldsymbol{K}_t}),\end{aligned} \tag{26}$$

where $\hat{\boldsymbol{E}}_{\boldsymbol{K}_t}$ denotes the estimation of the natural gradient $\boldsymbol{E}_{\boldsymbol{K}_t}$. The first term on the left hand side of (26) can be considered as the scaled square norm of the natural gradient. The first term on the right hand side compares the actor's performances between consecutive updates, which is bounded via Abel summation by parts. The second term evaluates the inaccurate natural gradient estimation, which is then bounded by the critic error $\boldsymbol{z}_t$ and the natural gradient $\boldsymbol{E}_{\boldsymbol{K}_t}$. The last term can be considered as the variance of the perturbed natural gradient update, which is bounded by $\mathcal{O}(\alpha_t)$.

Step 4: Interconnected iteration system analysis. Taking expectation and summing (24), (25), (26) from 0 to $T-1$, we obtain the following interconnected iteration system:

$$\begin{aligned}A_T \leq&\mathcal{O}(\frac{1}{\sqrt{T}}) + h_2 B_T + h_2 C_T,\\ B_T \leq&\mathcal{O}(\frac{1}{\sqrt{T}}) + h_4\sqrt{A_T B_T} + h_5 C_T,\\ C_T \leq&\mathcal{O}(\frac{1}{\sqrt{T}}) + h_7\sqrt{B_T C_T},\end{aligned} \tag{27}$$

where $h_2, h_4, h_5,$ and h_7 are positive constants defined in (47). By solving the above inequalities, we further prove that if $h_2 h_4^2 + h_2 h_4^2 h_7^2 + 2h_5 h_7^2 < 1$, then A_T, B_T, C_T converge at a rate of $\mathcal{O}(T^{-\frac{1}{2}})$. This condition can be easily satisfied by choosing the stepsize ratio c to be smaller than a threshold defined in (51).

Step 5: Global convergence analysis. To prove the global optimality, we utilize the gradient domination condition of LQR (Lemma 13):

$$J(\boldsymbol{K}) - J(\boldsymbol{K}^*) \leq \frac{1}{\sigma_{\min}(\boldsymbol{R})}\|\boldsymbol{D}_{\boldsymbol{K}^*}\|\mathrm{Tr}(\boldsymbol{E}_{\boldsymbol{K}}^\top\boldsymbol{E}_{\boldsymbol{K}}).$$

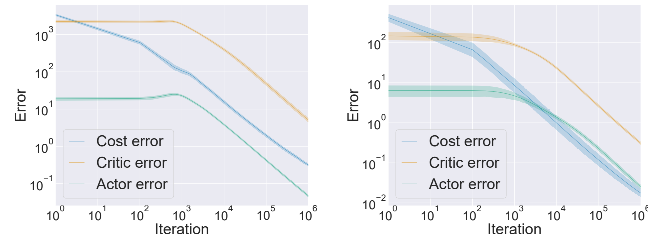
This property shows that the actor performance error can be bounded by the norm of the natural gradient ($\mathrm{Tr}(\boldsymbol{E}_{\boldsymbol{K}}^\top\boldsymbol{E}_{\boldsymbol{K}})$). Since we have proved the average natural gradient norm C_T converges to zero, summation over both sides of the above inequality yields

$$\min_{0\leq t<T}\mathbb{E}[J(\boldsymbol{K}_t) - J(\boldsymbol{K}^*)] =\mathcal{O}(\frac{1}{\sqrt{T}}),$$

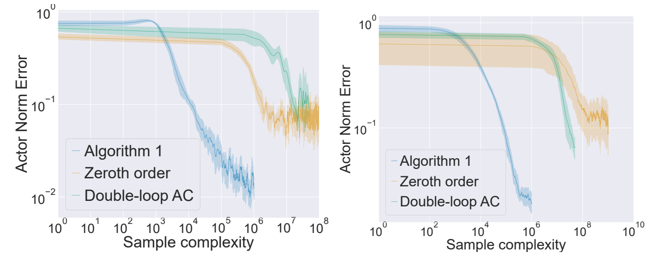
which is the convergence of the actor performance error. We thus complete the proof of Theorem 1.

5 Experiments

While our main contribution lies in the theoretical analysis, we also present several examples to validate the efficiency of Algorithm 1. We provide two examples to illustrate our theoretical results. The first example (first column in Figure 1) is a two-dimensional system and the second example (second column in Figure 1) is a four-dimensional system. The detailed parameters are shown in Supplementary Material.



(a) Learning results of Algorithm 1



(b) Comparison of Algorithm 1 with two other algorithms

Figure 1: **(a)** Learning results of Algorithm 1. In the figure, the cost error refers to $\frac{1}{T}\sum_{t=0}^{T-1}(\eta_t - J(\boldsymbol{K}_t))^2$, Critic error refers to $\frac{1}{T}\sum_{t=0}^{T-1}\|\boldsymbol{\omega}_t - \boldsymbol{\omega}_{\boldsymbol{K}_t}^*\|^2$, and the Actor error refers to $\frac{1}{T}\sum_{t=0}^{T-1}[J(\boldsymbol{K}_t) - J(\boldsymbol{K}^*)]$, corresponding to the conclusion in Theorem 1 empirically.
(b) Comparison of Algorithm 1 with two other algorithms. The actor norm error refers to $\|\boldsymbol{K} - \boldsymbol{K}^*\|_F$. In this figure, the solid lines correspond to the mean and the shaded regions correspond to 95% confidence interval over 10 independent runs.

The performance of Algorithm 1 is shown in Figure 1, where the left column corresponds to the two-dimensional system and the right column to the four-dimensional system. The solid lines plot the mean values and the shaded regions denote the 95% confidence interval over 10 independent runs. Consistent with our theorem, Figure 1(a) shows that the cost estimation error, the critic error, and the actor performance error all diminish at a rate of at least $\mathcal{O}(T^{-\frac{1}{2}})$. The convergence also suggests that the intermediate closed-loop linear systems during iteration are uniformly stable.

We compare Algorithm 1 with the zeroth-order method [Fazel *et al.*, 2018] and the double-loop AC algorithm [Yang *et al.*, 2019] (listed in Algorithm 2 and Algorithm 3 respectively, in Supplementary Material). We plotted the relative errors of the actor parameters for all three methods in Figure 1(b). As it can be seen that Algorithm 1 demonstrates superior sample efficiency compared to the other two algorithms.

6 Conclusion and Discussion

In this paper, we establish the finite-time analysis for the single-sample single-timescale AC method under the LQR setting. We for the first time show that this method can find a global optimal policy under the general continuous state-action space, which contributes to understanding the limits of the AC on continuous control tasks.

Acknowledgements

This work was supported by the Singapore Ministry of Education Tier 1 Academic Research Fund (22-5460-A0001) and Tier 2 AcRF (T2EP20123-0037). (Corresponding author: Lin Zhao.) The work of J. Duan was supported in part by the NSF China under Grant 52202487 and in part by the Young Elite Scientists Sponsorship Program by CAST under Grant 2023QNRC001.

References

[Anderson and Moore, 2007] Brian DO Anderson and John B Moore. *Optimal control: linear quadratic methods*. Courier Corporation, 2007.

[Barakat *et al.*, 2022] Anas Barakat, Pascal Bianchi, and Julien Lehmann. Analysis of a target-based actor-critic algorithm with linear function approximation. In *International Conference on Artificial Intelligence and Statistics*, pages 991–1040. PMLR, 2022.

[Bhatnagar *et al.*, 2009] Shalabh Bhatnagar, Richard S Sutton, Mohammad Ghavamzadeh, and Mark Lee. Natural actor–critic algorithms. *Automatica*, 45(11):2471–2482, 2009.

[Bradtke *et al.*, 1994] Steven J Bradtke, B Erik Ydstie, and Andrew G Barto. Adaptive linear quadratic control using policy iteration. In *Proceedings of 1994 American Control Conference-ACC'94*, volume 3, pages 3475–3479. IEEE, 1994.

[Castro and Meir, 2010] Dotan Di Castro and Ron Meir. A convergent online single time scale actor critic algorithm. *The Journal of Machine Learning Research*, 11:367–410, 2010.

[Chen and Zhao, 2022] Xuyang Chen and Lin Zhao. Finite-time analysis of single-timescale actor-critic. *arXiv preprint arXiv:2210.09921*, 2022.

[Chen *et al.*, 2021] Tianyi Chen, Yuejiao Sun, and Wotao Yin. Closing the gap: Tighter analysis of alternating stochastic gradient methods for bilevel problems. *Advances in Neural Information Processing Systems*, 34:25294–25307, 2021.

[Cohen *et al.*, 2019] Alon Cohen, Tomer Koren, and Yishay Mansour. Learning linear-quadratic regulators efficiently with only $\sqrt{T}$ regret. In *International Conference on Machine Learning*, pages 1300–1309. PMLR, 2019.

[Dean *et al.*, 2018] Sarah Dean, Horia Mania, Nikolai Matni, Benjamin Recht, and Stephen Tu. Regret bounds for robust adaptive control of the linear quadratic regulator. *Advances in Neural Information Processing Systems*, 31, 2018.

[Dean *et al.*, 2020] Sarah Dean, Horia Mania, Nikolai Matni, Benjamin Recht, and Stephen Tu. On the sample complexity of the linear quadratic regulator. *Foundations of Computational Mathematics*, 20(4):633–679, 2020.

[Duan *et al.*, 2023] Jingliang Duan, Jie Li, Xuyang Chen, Kai Zhao, Shengbo Eben Li, and Lin Zhao. Optimization landscape of policy gradient methods for discrete-time static output feedback. *IEEE Transactions on Cybernetics*, 2023.

[Fazel *et al.*, 2018] Maryam Fazel, Rong Ge, Sham Kakade, and Mehran Mesbahi. Global convergence of policy gradient methods for the linear quadratic regulator. In *International Conference on Machine Learning*, pages 1467–1476. PMLR, 2018.

[Joarder and Omar, 2011] Anwar H Joarder and M Hafidz Omar. On statistical characteristics of the product of two correlated chi-square variables. *Journal of Applied Statistical Science*, 19(4):89–101, 2011.

[Kakade, 2001] Sham M Kakade. A natural policy gradient. *Advances in neural information processing systems*, 14, 2001.

[Karmakar and Bhatnagar, 2018] Prasenjit Karmakar and Shalabh Bhatnagar. Two time-scale stochastic approximation with controlled markov noise and off-policy temporal-difference learning. *Mathematics of Operations Research*, 43(1):130–151, 2018.

[Konda and Tsitsiklis, 1999] Vijay Konda and John Tsitsiklis. Actor-critic algorithms. *Advances in neural information processing systems*, 12, 1999.

[Krauth *et al.*, 2019] Karl Krauth, Stephen Tu, and Benjamin Recht. Finite-time analysis of approximate policy iteration for the linear quadratic regulator. *Advances in Neural Information Processing Systems*, 32, 2019.

[Kumar *et al.*, 2019] Harshat Kumar, Alec Koppel, and Alejandro Ribeiro. On the sample complexity of actor-critic method for reinforcement learning with function approximation. *arXiv preprint arXiv:1910.08412*, 2019.

[LeCun *et al.*, 2015] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[Magnus, 1978] Jan R Magnus. The moments of products of quadratic forms in normal variables. *Statistica Neerlandica*, 32(4):201–210, 1978.

[Malik *et al.*, 2019] Dhruv Malik, Ashwin Pananjady, Kush Bhatia, Koulik Khamaru, Peter Bartlett, and Martin Wainwright. Derivative-free methods for policy optimization: Guarantees for linear quadratic systems. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2916–2925. PMLR, 2019.

[Mania *et al.*, 2019] Horia Mania, Stephen Tu, and Benjamin Recht. Certainty equivalence is efficient for linear quadratic control. *Advances in Neural Information Processing Systems*, 32, 2019.

[Mnih *et al.*, 2016] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937. PMLR, 2016.

[Nagar, 1959] Anirudh L Nagar. The bias and moment matrix of the general k-class estimators of the parameters in simultaneous equations. *Econometrica: Journal of the Econometric Society*, pages 575–595, 1959.

[Olshevsky and Gharesifard, 2023] Alex Olshevsky and Bahman Gharesifard. A small gain analysis of single timescale actor critic. *SIAM Journal on Control and Optimization*, 61(2):980–1007, 2023.

[Rencher and Schaalje, 2008] Alvin C Rencher and G Bruce Schaalje. *Linear models in statistics*. John Wiley & Sons, 2008.

[Schacke, 2004] Kathrin Schacke. On the kronecker product. *Master's thesis, University of Waterloo*, 2004.

[Schulman *et al.*, 2015] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR, 2015.

[Silver *et al.*, 2017] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.

[Sutton and Barto, 2018] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[Sutton *et al.*, 1999] Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12, 1999.

[Tu and Recht, 2018] Stephen Tu and Benjamin Recht. Least-squares temporal difference learning for the linear quadratic regulator. In *International Conference on Machine Learning*, pages 5005–5014. PMLR, 2018.

[Wang *et al.*, 2019] Lingxiao Wang, Qi Cai, Zhuoran Yang, and Zhaoran Wang. Neural policy gradient methods: Global optimality and rates of convergence. *arXiv preprint arXiv:1909.01150*, 2019.

[Wu *et al.*, 2020] Yue Frank Wu, Weitong Zhang, Pan Xu, and Quanquan Gu. A finite-time analysis of two time-scale actor-critic methods. *Advances in Neural Information Processing Systems*, 33:17617–17628, 2020.

[Xu *et al.*, 2020] Tengyu Xu, Zhe Wang, and Yingbin Liang. Non-asymptotic convergence analysis of two timescale (natural) actor-critic algorithms. *arXiv preprint arXiv:2005.03557*, 2020.

[Yang *et al.*, 2019] Zhuoran Yang, Yongxin Chen, Mingyi Hong, and Zhaoran Wang. Provably global convergence of actor-critic: A case for linear quadratic regulator with ergodic cost. *Advances in neural information processing systems*, 32, 2019.

[Zhang *et al.*, 2020] Shangtong Zhang, Bo Liu, Hengshuai Yao, and Shimon Whiteson. Provably convergent two-timescale off-policy actor-critic with function approximation. In *International Conference on Machine Learning*, pages 11204–11213. PMLR, 2020.

[Zhou and Lu, 2023] Mo Zhou and Jianfeng Lu. Single timescale actor-critic method to solve the linear quadratic regulator with convergence guarantees. *Journal of Machine Learning Research*, 24(222):1–34, 2023.