2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Learning Zero-Sum Linear Quadratic Games with Improved Sample Complexity

Jiduan Wu Anas Barakat Ilyas Fatkhullin Niao He

Abstract— **Zero-sum Linear Quadratic (LQ) games are fundamental in optimal control and can be used (i) as a dynamic game formulation for risk-sensitive or robust control, or (ii) as a benchmark setting for multi-agent reinforcement learning with two competing agents in continuous state-control spaces. In contrast to the well-studied single-agent linear quadratic regulator problem, zero-sum LQ games entail solving a challenging nonconvex-nonconcave min-max problem with an objective function that lacks coercivity. Recently, Zhang et al. [1] discovered an implicit regularization property of natural policy gradient methods which is crucial for safety-critical control systems since it preserves the robustness of the controller during learning. Moreover, in the model-free setting where the knowledge of model parameters is not available, Zhang et al. proposed the first polynomial sample complexity algorithm to reach an ϵ-neighborhood of the Nash equilibrium while maintaining the desirable implicit regularization property. In this work, we propose a simpler nested Zeroth-Order (ZO) algorithm improving sample complexity by several orders of magnitude. Our main result guarantees a $\widetilde{\mathcal{O}}(\epsilon^{-3})$ sample complexity under the same assumptions using a single-point ZO estimator. Furthermore, when the estimator is replaced by a two-point estimator, our method enjoys a better $\widetilde{\mathcal{O}}(\epsilon^{-2})$ sample complexity. Our key improvements rely on a more sample-efficient nested algorithm design and finer control of the ZO natural gradient estimation error.**

I. Introduction

While policy optimization has a long history in control for unknown and parameterized system models (see for e.g., [2]), recent successes in reinforcement learning and continuous control tasks have renewed the interest in direct policy search thanks to its flexibility and scalability to high-dimensional problems. Despite these desirable features, theoretical guarantees for policy gradient methods have remained elusive until very recently because of the nonconvexity of the induced optimization landscape. In particular, in contrast to control-theoretic approaches which are often model-based and estimate the system dynamics first before designing optimal controllers, the computational and sample complexities of model-free policy gradient methods were only recently analyzed. We refer the interested reader to a nice recent survey about learning control policies [3]. For instance, while the classic Linear Quadratic Regulator (LQR) problem induces a nonconvex optimization problem over the set of stable control gain matrices, the gradient domination property [4] and the coercivity of the cost function respectively allow to derive global convergence to optimal policies for policy gradient methods and ensure stable feedback policies at each iteration [5]. As exact gradients are often unavailable when system dynamics are unknown, derivative-free optimization techniques using cost values have been employed to design model-free policy gradient methods to solve LQR problems [5]. Alternative approaches to solve LQR include system identification [6], [7], iterative solution of Algebraic Riccati Equation [8], [9] and convex semi-definite program formulations [10]. However, such methods are not easily adaptable to the simulation-based model-free setting.

Besides the desired stability constraint, other requirements such as robustness and risk sensitivity constraints also play an important role in the design of controllers for safety-critical control systems. Indeed, system perturbations, modeling imprecision, and adversarial uncertainty are ubiquitous in control systems and may lead to severe degradation in performance [11], [12]. Robustness constraints can be incorporated into control design via different approaches including using statistical models for disturbances such as for linear quadratic Gaussian design, adopting a game theory perspective via designing ‘minimax’ controllers and incorporating an $\mathcal{H}_\infty$ norm bound of input-output operators as in $\mathcal{H}_\infty$ control [13]. Classical linear models for robust control include the LQ disturbances attenuation problem and the linear exponential quadratic Gaussian problem which are well-known to be equivalent to zero-sum LQ games [13], [14], [15]. Besides its relevance for robust control problem formulation, zero-sum LQ games also constitute a benchmark problem for multi-agent continuous control problems involving two competing agents. However, solving this problem faces (at least) two distinct challenges requiring to deal with (a) a constrained nonconvex-nonconcave problem and (b) lack of coercivity, unlike for the classic LQR problem for which descent over the objective ensures feasibility and stability of the iterates during learning.

While the formulation of zero-sum LQ games dates back at least to the seventies [15]*, the sample complexity analysis of model-free policy gradient algorithms solving this problem was only recently explored in the literature [1]. More precisely, Zhang et al. [1] showed that an ϵ-Nash equilibrium of finite horizon zero-sum LQ games can be learned via nested model-free Natural Policy Gradient (NPG) algorithms with polynomial sample complexity in the accuracy ϵ. Interestingly, the aforementioned algorithms enjoy an Implicit Regularization (IR) property which maintains

This work was supported by ETH AI Center doctoral fellowship, ETH Foundations of Data Science (ETH-FDS), ETH Research Grant funded via ETH Zurich Foundation, NCCR Automation, and Swiss National Science Foundation (SNSF) Project Funding No. 200021-207343. The authors are with the Department of Computer Science, ETH Zurich, Switzerland. Corresponding email: jiduwu@student.ethz.ch.

*This formulation is under the continuous-time setting.

2023 62nd IEEE Conference on Decision and Control (CDC) | 979-8-3503-0124-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/CDC49753.2023.10384025

the robustness of the controllers during learning [1], [14]. In particular, the iterates of the algorithms are guaranteed to stay in some feasible set where the worst-case cost is finite without using any explicit regularization or projection operation. In the present work, we show that significantly less samples are required to guarantee both the IR property and the convergence to an ϵ-Nash equilibrium of the zero-sum LQ games problem while only having access to ZO information. Our contributions can be summarized as follows:

Contributions. Our main result states that our derivative-free nested policy gradient algorithm requires $\widetilde{\mathcal{O}}(\epsilon^{-3})$ samples to reach an ϵ-neighborhood of the Nash equilibrium (NE) of the zero-sum LQ games problem, improving over the best-known-so-far $\widetilde{\mathcal{O}}(\epsilon^{-9})$[†] total sample complexity established in [1]. We also show that our algorithm enjoys the IR property upon choosing adequate values for ZO estimation parameters such as the batch sizes and the perturbation radius which are less restrictive compared to prior work [1]. Our improvement follows from (a) a simpler algorithm design reducing the number of calls to the inner-loop maximizing procedure, (b) a better sample complexity to solve the inner maximization problem and (c) an improved sample complexity for solving the resulting minimization problem in our outer-loop procedure using a careful decomposition of the estimation error caused by policy gradient estimation. We further improve the sample complexity to $\widetilde{\mathcal{O}}(\epsilon^{-2})$ using a two-point ZO estimator under a stronger sampling assumption.

Paper organization. The rest of this paper is structured as follows. In section II, we discuss related work. In section III, we introduce the stochastic zero-sum LQ games problem together with useful background. We present our model-free nested natural policy gradient algorithm to solve the problem in section IV and section V presents our main results along with a proof sketch to highlight the key steps leading to sample complexity improvement. In Section VI, we further validate our results by numerical simulations. We conclude this paper with possible future directions. The proofs of our results and the detailed version of some results are deferred to an extended version [16].

II. Related Work

Policy optimization for LQ problems. Compared to zero-sum LQ games, policy optimization for single-agent LQ problems is a well-understood topic. Theoretical guarantees for model-based and model-free algorithms searching for the optimal policy were established in [5] for the discrete-time infinite-horizon setting. Several subsequent works improved over the polynomial sample complexity in [5] using single and two-point ZO estimation [17], [18]. Additionally, the LQ model has been studied under different settings including finite-horizon [19] and continuous-time [20], [21], [22]. First-order methods have also been recently investigated for solving LQR [23], [24]. Bu et al. [25] provided convergence analysis for possibly indefinite infinite-horizon LQR problems. Guo et al. [26] designed Goldstein subdifferential algorithms to solve the nonsmooth $\mathcal{H}_\infty$ control problem and left sample complexity analysis in the model-free setting as an important future direction. Other related problems include Markovian jump systems [27], output control design [20] , decentralized control [28] and nonlinear dynamics [29]. Interested readers are referred to the thorough review paper [3] on policy optimization methods for learning control policies.

Zero-sum LQ games and beyond. Recent research efforts have been devoted to studying the more challenging zero-sum LQ games problem [14], [30], [31], [1]. Zhang et al. [30] proposed projected nested gradient-based algorithms in which the projection step is difficult to implement in practice. Later, Bu et al. [31] removed the projection step, but their analysis requires access to the exact solution of the inner maximization problem and cannot be easily extended to the model-free case. Meanwhile, Zhang et al. [14] introduced a nested natural gradient-based algorithm that demonstrates the IR property for the infinite-horizon $\mathcal{H}_2/\mathcal{H}_\infty$ control problem in the model-based case. In the model-free setting, Al-Tamimi et al. [32] proposed a Q-learning-based method to solve zero-sum LQ games without providing a sample complexity analysis. In the context of mean-field games, counterparts of LQR and zero-sum LQ games were developed in [33], [34], where the formulation of mean-field zero-sum LQ games reduces to two zero-sum LQ games problems. Recently, a N-player general-sum game formulation of LQR was studied in [35], [36], [37]. However, such a problem in the 2-player case is different from our zero-sum formulation.

III. Preliminaries

Notations. For any matrix $M \in \mathbb{R}^{n\times n}$, we denote by $\|M\|$ and $\|M\|_F$ its operator and Frobenius norms respectively. The spectral radius of a matrix M is denoted by $\rho(M)$ and a matrix is said to be (Schur) stable if $\rho(M) < 1$, i.e., all the absolute values of the eigenvalues of the matrix M are (strictly) smaller than 1. The smallest eigenvalue of a symmetric matrix M is denoted by $\lambda_{\min}(M)$. For N diagonal matrices X_i for $i \in \{0, \cdots N-1\}$ for some integer $N \geq 1$, the block-diagonal matrix with diagonal entries $X_0, \cdots, X_{N-1}$ is denoted by $\mathrm{diag}(X_{0-(N-1)})$. The uniform distribution over a set S is denoted as Unif(S).

Stochastic Zero-Sum Linear Quadratic Dynamic Games. We consider the zero-sum LQ games problem (following the exposition in [1]) where the system state evolves as follows:

$$x_{h+1} = A_h x_h + B_h u_h + D_h w_h + \xi_h,\ h \in \{0, \cdots, N-1\}, \tag{1}$$

where N is a finite nonzero horizon, $x_0 \in \mathbb{R}^m$ is an initial random state and where for any stage $h \in \{0, \cdots, N-1\}$, $x_h \in \mathbb{R}^m$ is the system state, $u_h \in \mathbb{R}^d$ and $w_h \in \mathbb{R}^n$ are the control inputs of the min and max players respectively[‡] and ξ_h is a random variable describing noisy perturbations to

[†] Notice that the total sample complexity was not provided in [1] but can be easily derived from their results, see Remark 4 for more details.

[‡] These controls depend on the history of state-control pairs at each time step h for now, stationary control policies will be sufficient as will be mentioned later on.

the system while A_h, B_h, D_h are (possibly) time-dependent system matrices with appropriate dimensions.

Assumption 1: The initial state x_0 and the noise ξ_h for $h \in \{0, \cdots, N-1\}$ are independent random variables following a distribution with zero-mean and positive-definite covariance. Moreover, there exists a positive scalar ϑ such that for all $h \in \{0, \cdots, N-1\}$, $\|x_0\| \leq \vartheta$ and $\|\xi_h\| \leq \vartheta$ almost surely.[§]

Our objective is to solve the following zero-sum game:

$$\inf_{(u_h)} \sup_{(w_h)} \mathbb{E}_{\boldsymbol{\xi}} \left[\sum_{h=0}^{N-1} (x_h^\top Q_h x_h + u_h^\top R_h^u u_h - w_h^\top R_h^w w_h) + c_N \right] \tag{2}$$

where $c_N := x_N^\top Q_N x_N$ and $\boldsymbol{\xi} := \left[x_0^\top, \xi_0^\top, \cdots, \xi_{N-1}^\top\right]^\top$ and the system states follow the linear time-varying system dynamics described in (1) and for every $h \in \{0, \cdots, N-1\}, Q_h \succeq 0, R_h^u, R_h^w \succ 0$ are symmetric matrices defining the quadratic objective. In view of our robust control motivation, the two players can be seen as a min controller and a max disturbance. Under standard assumptions which we do not mention here for brevity[¶], the saddle-point control policies solving (2) are unique and have the linear state-feedback form. Thus, we can restrict our search to gain matrices $K_h \in \mathbb{R}^{d \times m}$ and $L_h \in \mathbb{R}^{n \times m}$ such that the controls are given by $u_h = -K_h x_h, w_h = -L_h x_h$ for $h \in \{0, \cdots, N-1\}$. Therefore, we will mainly focus on solving the following min-max policy optimization problem resulting from (2):

$$\min_{(K_h)} \max_{(L_h)} \mathbb{E}_{\boldsymbol{\xi}} \left[\sum_{h=0}^{N-1} x_h^\top M_h x_h + c_N \right], \tag{3}$$

where $M_h := Q_h + K_h^\top R_h^u K_h - L_h^\top R_h^w L_h$ and the system state follows the dynamics $x_{h+1} = (A_h - B_h K_h - D_h L_h) x_h + \xi_h$ for $h \in \{0, \cdots, N-1\}$.

a) Compact reformulation: To simplify the exposition and our analysis, we rewrite problem (3) under a more compact form following the reformulation proposed in [1]. Consider the following notations:

$$\begin{aligned}
&\boldsymbol{x} := [x_0^\top, \cdots, x_N^\top]^\top, \boldsymbol{u} := [u_0^\top, \cdots, u_{N-1}^\top]^\top, \\
&\boldsymbol{w} := [w_0^\top, \cdots, w_{N-1}^\top]^\top, \boldsymbol{\xi} = [x_0^\top, \xi_0^\top, \cdots, \xi_{N-1}^\top]^\top, \\
&\boldsymbol{A} := \begin{bmatrix} \mathbf{0}_{m \times mN} & \mathbf{0}_{m \times m} \\ \operatorname{diag}(A_{0-(N-1)}) & \mathbf{0}_{mN \times m} \end{bmatrix}, \boldsymbol{Q} := \operatorname{diag}(Q_{0-N}), \\
&\boldsymbol{D} := \begin{bmatrix} \mathbf{0}_{m \times nN} \\ \operatorname{diag}(D_{0-(N-1)}) \end{bmatrix}, \boldsymbol{B} := \begin{bmatrix} \mathbf{0}_{m \times dN} \\ \operatorname{diag}(B_{0-(N-1)}) \end{bmatrix}, \\
&\boldsymbol{R}^u := \operatorname{diag}(R^u_{0-(N-1)}), \boldsymbol{R}^w := \operatorname{diag}(R^w_{0-(N-1)}), \\
&\boldsymbol{K} := \begin{bmatrix} \operatorname{diag}(K_{0-(N-1)}) & \mathbf{0}_{dN \times m} \end{bmatrix}, \qquad (4) \\
&\boldsymbol{L} := \begin{bmatrix} \operatorname{diag}(L_{0-(N-1)}) & \mathbf{0}_{nN \times m} \end{bmatrix}. \qquad (5)
\end{aligned}$$

We denote by $\mathcal{S}_1 \subset \mathbb{R}^{dN \times m(N+1)}$ and $\mathcal{S}_2 \subset \mathbb{R}^{nN \times m(N+1)}$ the matrix subspaces induced by the sparsity patterns described in (4), (5) for the gain matrices $\boldsymbol{K}$ and $\boldsymbol{L}$ respectively. The subspaces $\mathcal{S}_1$, $\mathcal{S}_2$ where we search for the NE solution $(\boldsymbol{K}^*, \boldsymbol{L}^*)$, are of dimensions $d_{\boldsymbol{K}} := dmN$ and $d_{\boldsymbol{L}} := nmN$ respectively. Then, problem (3) can be rewritten as:

$$\min_{\boldsymbol{K} \in \mathcal{S}_1} \max_{\boldsymbol{L} \in \mathcal{S}_2} \mathcal{G}(\boldsymbol{K}, \boldsymbol{L}) := \mathbb{E}_{\boldsymbol{\xi}}[\boldsymbol{x}^\top (\boldsymbol{Q} + \boldsymbol{K}^\top \boldsymbol{R}^u \boldsymbol{K} - \boldsymbol{L}^\top \boldsymbol{R}^w \boldsymbol{L}) \boldsymbol{x}], \tag{6}$$

where the transition dynamics are described by $\boldsymbol{x} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{D}\boldsymbol{w} + \boldsymbol{\xi} = (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K} - \boldsymbol{D}\boldsymbol{L})\boldsymbol{x} + \boldsymbol{\xi}$. Notice that our search for gain matrices $\boldsymbol{K}, \boldsymbol{L}$ is restricted to the matrices of the form described in (4), (5) as this set of sparse matrices is sufficient to find the NE we are looking for. For any gain matrices $\boldsymbol{K}$ and $\boldsymbol{L}$, we can rewrite the objective function value $\mathcal{G}(\boldsymbol{K}, \boldsymbol{L})$ as follows:

$$\begin{aligned}
\mathcal{G}(\boldsymbol{K}, \boldsymbol{L}) &= \mathbb{E}_{\boldsymbol{\xi}}[\mathcal{G}_{\boldsymbol{\xi}}(\boldsymbol{K}, \boldsymbol{L})] = \operatorname{Tr}(\boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}} \boldsymbol{\Sigma}_0) \\
&= \operatorname{Tr}\big((\boldsymbol{Q} + \boldsymbol{K}^\top \boldsymbol{R}^u \boldsymbol{K} - \boldsymbol{L}^\top \boldsymbol{R}^w \boldsymbol{L}) \boldsymbol{\Sigma}_{\boldsymbol{K},\boldsymbol{L}}\big),
\end{aligned}$$

where $\mathcal{G}_{\boldsymbol{\xi}}(\boldsymbol{K}, \boldsymbol{L}) := \boldsymbol{\xi}^\top \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}} \boldsymbol{\xi}$, $\boldsymbol{\Sigma}_0 := \mathbb{E}_{\boldsymbol{\xi}}[\boldsymbol{\xi}\boldsymbol{\xi}^\top] \succ 0$ (see Assumption 1) and the matrices $\boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}}$, $\boldsymbol{\Sigma}_{\boldsymbol{K},\boldsymbol{L}} := \mathbb{E}_{\boldsymbol{\xi}}[\operatorname{diag}(x_0 x_0^\top, \cdots, x_N x_N^\top]$ are the unique solutions to the recursive Lyapunov equations

$$\boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}} = \boldsymbol{A}_{\boldsymbol{K},\boldsymbol{L}}^\top \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}} \boldsymbol{A}_{\boldsymbol{K},\boldsymbol{L}} + \boldsymbol{Q} + \boldsymbol{K}^\top \boldsymbol{R}^u \boldsymbol{K} - \boldsymbol{L}^\top \boldsymbol{R}^w \boldsymbol{L}, \tag{7}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{K},\boldsymbol{L}} = \boldsymbol{A}_{\boldsymbol{K},\boldsymbol{L}} \boldsymbol{\Sigma}_{\boldsymbol{K},\boldsymbol{L}} \boldsymbol{A}_{\boldsymbol{K},\boldsymbol{L}}^\top + \boldsymbol{\Sigma}_0, \tag{8}$$

where $\boldsymbol{A}_{\boldsymbol{K},\boldsymbol{L}} := \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K} - \boldsymbol{D}\boldsymbol{L}$. The objective $\mathcal{G}(\boldsymbol{K}, \boldsymbol{L})$ is nonconvex-nonconcave in general (see Lemma 3.1 in [1]). From the above compact formulation, we observe that the finite-horizon case can be seen as a special case of infinite-horizon zero-sum LQ games with special constraints on sparsity patterns of matrices defined in (4), (5). Using this perspective, the time-varying case where model parameters such as A_h, B_h vary over $h \in \{0, \cdots, N-1\}$ is included in the compact formulation as shown in [1].

b) Policy Gradients.: The gradients of $\mathcal{G}$ w.r.t. $\boldsymbol{K}, \boldsymbol{L}$ (see [1]) are given by the following expressions:

$$\begin{aligned}
&\nabla_{\boldsymbol{K}} \mathcal{G}(\boldsymbol{K}, \boldsymbol{L}) = 2\boldsymbol{F}_{\boldsymbol{K},\boldsymbol{L}} \boldsymbol{\Sigma}_{\boldsymbol{K},\boldsymbol{L}}, && (9) \\
&\boldsymbol{F}_{\boldsymbol{K},\boldsymbol{L}} := (\boldsymbol{R}^u + \boldsymbol{B}^\top \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}} \boldsymbol{B})\boldsymbol{K} - \boldsymbol{B}^\top \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}}(\boldsymbol{A} - \boldsymbol{D}\boldsymbol{L}), \\
&\nabla_{\boldsymbol{L}} \mathcal{G}(\boldsymbol{K}, \boldsymbol{L}) = 2\boldsymbol{E}_{\boldsymbol{K},\boldsymbol{L}} \boldsymbol{\Sigma}_{\boldsymbol{K},\boldsymbol{L}}, && (10) \\
&\boldsymbol{E}_{\boldsymbol{K},\boldsymbol{L}} := (-\boldsymbol{R}^w + \boldsymbol{D}^\top \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}} \boldsymbol{D})\boldsymbol{L} - \boldsymbol{D}^\top \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}}(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}).
\end{aligned}$$

If $\boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}} \succeq 0$ and $\boldsymbol{R}^w - \boldsymbol{D}^\top \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}} \boldsymbol{D} \succ 0$ for a stationary point $(\boldsymbol{K}, \boldsymbol{L})$ of $\mathcal{G}$, then this stationary point is the unique NE of the game (see Lemma 3.2 in [1]).

Remark 1: In our finite-horizon scenario, $\rho(\boldsymbol{A}_{\boldsymbol{K},\boldsymbol{L}}) = 0$ since $\boldsymbol{A}_{\boldsymbol{K},\boldsymbol{L}}^{N+1} = 0$. This means that the pair $(\boldsymbol{K}, \boldsymbol{L})$ defined in (4)-(5) is always stable. This property leads to the existence and uniqueness of the solution of the Lyapunov equation.

IV. Nested Derivative-Free Natural Policy Gradient (NPG) Algorithm

In this section, we present our model-free and derivative-free nested NPG algorithm inspired from the recent work [1]. We start with the deterministic exact version of the algorithm assuming access to exact natural policy gradients.

[§] The almost sure boundedness can be relaxed to consider sub-Gaussian distributions as noticed in prior work [17], [38].

[¶] See Assumption 2.4 in [1] for instance and the explanations in Remark 2.5 therein for further details, see also [13].

A. Exact Nested NPG Algorithm

To prepare the stage for the model-free setting, we briefly introduce the nested NPG algorithm in the deterministic setting, i.e., when we have access to the policy gradients w.r.t. both control variables $\boldsymbol{K}, \boldsymbol{L}$ as reported in (9). This algorithm was considered for example in [1] and we follow a similar exposition in this subsection. We first solve the inner maximization problem in (6) for any fixed control gain matrix $\boldsymbol{K}$ to obtain a solution $\boldsymbol{L}(\boldsymbol{K})$ before solving the outer-loop minimization problem with the resulting objective $\mathcal{G}(\boldsymbol{K}, \boldsymbol{L}(\boldsymbol{K}))$. The following proposition that we report here from Lemma 3.3 in [1] guarantees that there exists a unique solution $\boldsymbol{L}(\boldsymbol{K})$ to the inner maximization problem whenever the control gain matrix $\boldsymbol{K}$ lies in a set which is known to contain the optimal control gain matrix solving the min-max problem.

Lemma 1: (Inner-loop well-definedness condition [1]) Consider the Riccati equation

$$\boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})} = \boldsymbol{Q} + \boldsymbol{K}^\top \boldsymbol{R}^u \boldsymbol{K} + \boldsymbol{A}_{\boldsymbol{K}}^\top \widetilde{\boldsymbol{P}}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})} \boldsymbol{A}_{\boldsymbol{K}}, \qquad (11)$$

where $\boldsymbol{A}_{\boldsymbol{K}} := \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}$ and $\widetilde{\boldsymbol{P}}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})} := \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})} + \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})} \boldsymbol{D}(\boldsymbol{R}^w - \boldsymbol{D}^\top \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})} \boldsymbol{D})^{-1} \boldsymbol{D}^\top \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})}$ and define the set

$$\begin{aligned}\mathcal{K} := \big\{ \boldsymbol{K} \in \mathcal{S}_1 \ & |(11) \text{ admits a solution } \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})} \succeq 0, \\ & \text{and } \boldsymbol{R}^w - \boldsymbol{D}^\top \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})} \boldsymbol{D} \succ 0 \big\}. \qquad (12)\end{aligned}$$

Then, for any $\boldsymbol{K} \in \mathcal{K}$, there exists a unique solution $\boldsymbol{L}(\boldsymbol{K})$ to the inner maximization problem in 6 given by

$$\boldsymbol{L}(\boldsymbol{K}) = (-\boldsymbol{R}^w + \boldsymbol{D}^\top \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})} \boldsymbol{D})^{-1} \boldsymbol{D}^\top \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})} (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}).$$

Moreover, for any $\boldsymbol{K} \in \mathcal{K}$ and any $\boldsymbol{L} \in \mathcal{S}_2$, $\boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}} \preceq \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})}$.

We are now ready to introduce the nested NPG algorithm which can be written as follows using positive stepsizes τ_1, τ_2 and indices $k \geq 0, t \geq 0$ for the inner and outer loops respectively:

$$\text{Inner loop: } \boldsymbol{L}_{k+1} = \boldsymbol{L}_k + \tau_1 \boldsymbol{E}_{\boldsymbol{K}_t,\boldsymbol{L}_k},\ k = 0, 1, \dots \qquad (13)$$

$$\text{Outer loop: } \boldsymbol{K}_{t+1} = \boldsymbol{K}_t - \tau_2 \boldsymbol{F}_{\boldsymbol{K}_t,\boldsymbol{L}(\boldsymbol{K}_t)},\ t = 0, 1, \dots \qquad (14)$$

With the choice of natural policy gradients, careful choice of inner-loop problem accuracy ϵ_1, and the deployment of the nested structure, our algorithm achieves an important IR effect: The iterates remain in the feasible set defining admissible stable controls without any explicit regularization of the problem, as shown in [1]. Maintaining the feasibility of the iterates during learning is important since it translates to preserving the robustness of the controllers in the face of adversarial perturbations. More formally, it was shown in Theorem 3.7 in [1] that (a) the sequence $(\boldsymbol{P}_{\boldsymbol{K}_t,\boldsymbol{L}(\boldsymbol{K}_t)})_t$ is well-defined, satisfies the conditions in (12) for every $t \geq 0$ and is (most importantly) non-increasing and bounded below in the sense of positive definiteness; and as a consequence (b) for every $t \geq 0, \boldsymbol{K}_t \in \mathcal{K}$ when $\boldsymbol{K}_0 \in \mathcal{K}$.

B. Derivative-Free Nested NPG Algorithm

In this section, we describe our algorithm to solve problem (6) in the model-free setting where we do not have access to exact gradients. In this setting for which system parameters are unknown, namely $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{D}, \boldsymbol{Q}, \boldsymbol{R}^u, \boldsymbol{R}^w$, we can simulate system trajectories, $(x_h)_{h=0,\cdots,N}$, using a pair of control gain matrices $(\boldsymbol{K}, \boldsymbol{L})$ and we have access to ZO information consisting of the (stochastic) cost $\mathcal{G}_{\boldsymbol{\xi}}(\boldsymbol{K}, \boldsymbol{L})$ incurred by this pair of controllers. In Algorithms 1 and 2, we denote by (1P) and (2P) the single-point and two-point ZO estimation procedures respectively.

Inner loop ZO-NPG algorithm (see Algorithm 1). In the light of the update rule (13) in the deterministic exact setting, for any fixed matrix $\boldsymbol{K}$ and any time index k, we replace the gradient $\nabla_{\boldsymbol{L}} \mathcal{G}(\boldsymbol{K}, \boldsymbol{L}_k)$ and the covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{K},\boldsymbol{L}_k}$ by ZO estimates denoted as $\widetilde{\nabla}_{\boldsymbol{L}} \mathcal{G}(\boldsymbol{K}, \boldsymbol{L}_k)$ and $\widetilde{\boldsymbol{\Sigma}}_{\boldsymbol{K},\boldsymbol{L}_k}$ respectively. A brief pseudocode of the inner-loop algorithm is described in Algorithm 1. Its detailed sampling and computation procedures can be found in Algorithm 1 of [1] and hence omitted here.

Algorithm 1 Model-free Inner-loop ZO-NPG Algorithm

```
Input: Given $\boldsymbol{K} \in \mathcal{K}$ and $\boldsymbol{L}_0$, number of iterations $T_{\text{in}}$,
    stepsize $\tau_1$, sample size $M_1$, perturbation radius $r_1$.
Output: $\boldsymbol{L}_{T_{\text{in}}}$
for $k = 0, 1, \cdots, T_{\text{in}}$ do
    Sample trajectories and estimate gradients and covariance matrices using (1P)/(2P) ZO estimation.
    Update $\boldsymbol{L}_{k+1} = \boldsymbol{L}_k + \tau_1 \widetilde{\boldsymbol{E}}_{\boldsymbol{K},\boldsymbol{L}_k}$ where
        $\widetilde{\boldsymbol{E}}_{\boldsymbol{K},\boldsymbol{L}_k} := \frac{1}{2} \widetilde{\nabla}_{\boldsymbol{L}} \mathcal{G}(\boldsymbol{K}, \boldsymbol{L}_k) \widetilde{\boldsymbol{\Sigma}}_{\boldsymbol{K},\boldsymbol{L}_k}^{-1}$
end for
```

Outer loop ZO-NPG (see Algorithm 2). Similarly to the inner loop procedure, we now replace the unknown quantities $\nabla_{\boldsymbol{K}} \mathcal{G}(\boldsymbol{K}_t, \boldsymbol{L}(\boldsymbol{K}_t))$ and $\boldsymbol{\Sigma}_{\boldsymbol{K}_t,\boldsymbol{L}(\boldsymbol{K}_t)}$ in (14) by ZO estimates. As for the exact solution $\boldsymbol{L}(\boldsymbol{K}_t)$ to the inner maximization problem, we use the output of the inner loop ZO-NPG algorithm instead. Notice that the zeroth-order single-point estimate $\widetilde{\nabla}_{\boldsymbol{K}} \mathcal{G}(\boldsymbol{K}, \boldsymbol{L})$ as defined in Algorithm 2 is an unbiased estimate of the gradient of the smoothed objective $\mathcal{G}_r$ in the sense that: $\mathbb{E}[\widetilde{\nabla}_{\boldsymbol{K}} \mathcal{G}(\boldsymbol{K}, \boldsymbol{L})] = \nabla_{\boldsymbol{K}} \mathcal{G}_r(\boldsymbol{K}, \boldsymbol{L})$, $\mathcal{G}_r(\boldsymbol{K}, \boldsymbol{L}) := \mathbb{E}[\mathcal{G}(\boldsymbol{K} + r_2 \boldsymbol{V}, \boldsymbol{L})]$, where $\boldsymbol{V}$ is uniformly sampled on a unit ball in $\mathcal{S}_2$.

Comparison to the derivative free NPG algorithm in [1]. We would like to point out here an important difference between our proposed algorithm and the zeroth order NPG algorithm in [1] which inspired this work. This difference lies in the outer loops of the algorithms: namely comparing Algorithm 2 and Algorithm 2 in [1]. In their work, at each time step t of the outer loop, Algorithm 1 (which provides an approximate solution of the maximization problem) is called for each perturbation $\boldsymbol{K}_t^m$ (for $m = 0, \cdots, M_2 - 1$) of the control gain matrix $\boldsymbol{K}_t$ (see step 6: in their Algorithm 2) in order to control the gradient estimation error. In contrast to their work, observe that we only call Algorithm 1 once at each outer loop iteration t in Algorithm 2 and use the approximate maximizer $\boldsymbol{L}_t$ to compute our zeroth order estimates for updating the control gain matrix sequence $(\boldsymbol{K}_t)$. This observation is crucial for our sample complexity improvement as will be discussed in the next section.

Remark 2: The single-point estimation [39] might suffer high variance for a small smoothing radius r. We can reduce the variance and hence the sample complexity by using two-point estimation.

Algorithm 2 Outer-loop Nested Natural Policy Gradient

```
Input: $\boldsymbol{K}_0 \in \mathcal{K}$, number of iterations $T$, sample size $M_2$,
    perturbation radius $r$, stepsize $\tau_2$, horizon $N$, dimension
    $d_{\boldsymbol{K}} = dmN$.
Output: $\boldsymbol{K}_{\text{out}} = \boldsymbol{K}_i$ where $i \sim \text{Unif}(\{0, \cdots, T-1\})$.
for $t = 0, 1, \cdots, T$ do
    Call Algorithm 1 to obtain $\boldsymbol{L}_t$.
    for $m = 0, 1, \cdots, M_2 - 1$ do
        Sample policies
        • (1P): Sample $\boldsymbol{K}_t^m = \boldsymbol{K}_t + r\boldsymbol{V}_m$ where $\boldsymbol{V}_m$ is
          uniformly drawn from $\mathcal{S}_1$ with $\|\boldsymbol{V}_m\|_F = 1$.
        • (2P): Sample $\boldsymbol{K}_t^{1,m} = \boldsymbol{K}_t + r\boldsymbol{V}_m$, $\boldsymbol{K}_t^{2,m} = \boldsymbol{K}_t -$
          $r\boldsymbol{V}_m$ where $\boldsymbol{V}_m$ is uniformly drawn from $\mathcal{S}_1$ with
          $\|\boldsymbol{V}_m\|_F = 1$.
        Simulate trajectories
        • (1P): Simulate a first trajectory using control
          $(\boldsymbol{K}_t^m, \boldsymbol{L}_t)$ for horizon $N$ under one realization
          of noises $\boldsymbol{\xi}_m$ and collect the cost $\mathcal{G}_{\boldsymbol{\xi}_m}(\boldsymbol{K}_t^m, \boldsymbol{L}_t)$.
        • (2P): Simulate two trajectories using controls
          $(\boldsymbol{K}_t^{1,m}, \boldsymbol{L}_t)$ and $(\boldsymbol{K}_t^{2,m}, \boldsymbol{L}_t)$ for horizon $N$ under
          the same realization of noises $\boldsymbol{\xi}_m$ and collect
          $\mathcal{G}_{\boldsymbol{\xi}_m}(\boldsymbol{K}_t^{1,m}, \boldsymbol{L}_t)$, $\mathcal{G}_{\boldsymbol{\xi}_m}(\boldsymbol{K}_t^{2,m}, \boldsymbol{L}_t)$ .
        Simulate another independent trajectory using con-
        trol $(\boldsymbol{K}_t, \boldsymbol{L}_t)$ for horizon $N$ starting from $x_{0,m}$ and
        compute
            $\widetilde{\boldsymbol{\Sigma}}^m_{\boldsymbol{K}_t,\boldsymbol{L}_t} = \text{diag}\big(x_{0,m}x_{0,m}^\top, \cdots, x_{N,m}x_{N,m}^\top\big)$
    end for
    Update $\boldsymbol{K}_{t+1} = \boldsymbol{K}_t - \tau_2 \widetilde{\nabla}_{\boldsymbol{K}}\mathcal{G}(\boldsymbol{K}_t, \boldsymbol{L}_t)\widetilde{\boldsymbol{\Sigma}}^{-1}_{\boldsymbol{K}_t,\boldsymbol{L}_t}$ where
    $\widetilde{\nabla}_{\boldsymbol{K}}\mathcal{G}(\boldsymbol{K}_t, \boldsymbol{L}_t)$ equals
    (1P): $\frac{1}{M_2}\sum_{m=0}^{M_2-1} \frac{d_{\boldsymbol{K}}}{r}\mathcal{G}_{\boldsymbol{\xi}_m}(\boldsymbol{K}_t^m, \boldsymbol{L}_t)\boldsymbol{V}_m$,
    (2P):
    $\frac{1}{M_2}\sum_{m=0}^{M_2-1} \frac{d_{\boldsymbol{K}}}{2r}\big(\mathcal{G}_{\boldsymbol{\xi}_m}(\boldsymbol{K}_t^{1,m}, \boldsymbol{L}_t) - \mathcal{G}_{\boldsymbol{\xi}_m}(\boldsymbol{K}_t^{2,m}, \boldsymbol{L}_t)\big)\boldsymbol{V}_m$,
    and $\widetilde{\boldsymbol{\Sigma}}_{\boldsymbol{K}_t,\boldsymbol{L}_t} = \frac{1}{M_2}\sum_{m=0}^{M_2-1}\widetilde{\boldsymbol{\Sigma}}^m_{\boldsymbol{K}_t,\boldsymbol{L}_t}$.
end for
```

V. Sample Complexity Analysis

In this section, we analyze the sample complexity of the algorithm introduced in Section IV, i.e., the number of samples of system trajectories required to reach an ϵ-neighborhood of the NE.

When using estimated natural gradients, the monotonicity of the sequence $(\boldsymbol{P}_{\boldsymbol{K}_t,\boldsymbol{L}(\boldsymbol{K}_t)})_{t\geq 0}$ is violated and the iterates $(\boldsymbol{K}_t)$ are no longer guaranteed to lie in the set $\mathcal{K}$ as we previously described in Section IV-A for the deterministic counterpart of the algorithm. In the following, we consider a subset $\hat{\mathcal{K}}$ of $\mathcal{K}$ for which we prove that IR holds (with high probability) similarly to the result we reported in Lemma 1 for good enough ZO estimates as we shall precisely state later in this section. Consider an initial point $\boldsymbol{K}_0 \in \mathcal{K}$ and define the set

$$\begin{aligned}\hat{\mathcal{K}} := \Big\{ &\boldsymbol{K} \in \mathcal{S}_1 \mid (11) \text{ admits a solution } \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})} \succeq 0 \\ &\text{and } \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})} \preceq \boldsymbol{P}_{\boldsymbol{K}_0,\boldsymbol{L}(\boldsymbol{K}_0)} + \frac{\lambda_{\min}(\boldsymbol{H}_{\boldsymbol{K}_0,\boldsymbol{L}(\boldsymbol{K}_0)})}{2\|\boldsymbol{D}\|^2}\cdot \boldsymbol{I} \Big\},\end{aligned} \tag{15}$$

where $\boldsymbol{H}_{\boldsymbol{K},\boldsymbol{L}} := \boldsymbol{R}^w - \boldsymbol{D}^\top \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}}\boldsymbol{D}$. Notice that $\hat{\mathcal{K}} \subset \mathcal{K}$ since

$$\begin{aligned}&\boldsymbol{R}^w - \boldsymbol{D}^\top \boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})}\boldsymbol{D} \\ &\succeq \boldsymbol{R}^w - \boldsymbol{D}^\top\Big(\boldsymbol{P}_{\boldsymbol{K}_0,\boldsymbol{L}(\boldsymbol{K}_0)} + \frac{\lambda_{\min}(\boldsymbol{H}_{\boldsymbol{K}_0,\boldsymbol{L}(\boldsymbol{K}_0)})}{2\|\boldsymbol{D}\|^2}\cdot \boldsymbol{I}\Big)\boldsymbol{D} \\ &\succeq \frac{\lambda_{\min}(\boldsymbol{H}_{\boldsymbol{K}_0,\boldsymbol{L}(\boldsymbol{K}_0)})}{2}\cdot \boldsymbol{I} \succ 0.\end{aligned} \tag{16}$$

As can be observed from (15), we need to control the error induced by the inner loop solver which provides an approximation of $\boldsymbol{L}(\boldsymbol{K})$ in order to show the recurrence of the iterates $\boldsymbol{K}_t$ in the set $\hat{\mathcal{K}}$ with high probability. This inner maximization problem which takes the form of an LQR problem has been previously addressed in the literature in several works using for example a gradient ascent or a natural gradient ascent algorithm in both model-based and model-free settings [5], [1], [17]. We report in the next result an informal version of Theorem 4.1 in [1] for the inner maximization problem in view of deriving the total sample complexity of our nested algorithm.

Lemma 2: (Inner-loop sample complexity [1]) Let $\delta_1, \epsilon_1 \in (0,1)$ and let $\boldsymbol{K} \in \mathcal{K}$. Using $\widetilde{\mathcal{O}}(\epsilon_1^{-2}\log \delta_1^{-1})$ samples, Algorithm 1 outputs with probability at least $1-\delta_1$ a control gain matrix $\boldsymbol{L}$ satisfying: $\mathcal{G}(\boldsymbol{K}, \boldsymbol{L}(\boldsymbol{K})) - \mathcal{G}(\boldsymbol{K}, \boldsymbol{L}) \leq \epsilon_1$, $\|\boldsymbol{L}(\boldsymbol{K}) - \boldsymbol{L}\|_F \leq \sqrt{\lambda_{\min}^{-1}(\boldsymbol{H}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})}) \cdot \epsilon_1}$.

Remark 3: This $\widetilde{\mathcal{O}}(\epsilon_1^{-2})$ sample complexity reported in Lemma 2 can be further improved to $\widetilde{O}(\epsilon_1^{-1})$ using ZO two-point estimation [40].

It follows from Lemma 2 that any control gain matrix $\boldsymbol{L}$ produced by Algorithm 1 lies in the following bounded set:

$$\hat{\mathcal{L}} := \Big\{ \boldsymbol{L} \in \mathcal{S}_2 \mid \|\boldsymbol{L}(\boldsymbol{K}) - \boldsymbol{L}\|_F \leq H, \boldsymbol{K} \in \hat{\mathcal{K}} \Big\}, \tag{17}$$
$$H := \sup_{\boldsymbol{K}\in\hat{\mathcal{K}}} \lambda_{\min}^{-1}(\boldsymbol{H}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})}) \leq 2\lambda_{\min}^{-1}(\boldsymbol{H}_{\boldsymbol{K}_0,\boldsymbol{L}(\boldsymbol{K}_0)}).$$

Using the sets $\hat{\mathcal{K}}$ and $\hat{\mathcal{L}}$ respectively defined in (15) and (17), we are now ready to state the IR of our model-free nested natural gradient algorithm w.r.t. both control gain matrices $\boldsymbol{K}$ and $\boldsymbol{L}$. More specifically, we will prove that the pair of iterates $(\boldsymbol{K}_t, \boldsymbol{L}_t)$ generated by Algorithms 1 and 2 will be maintained in the bounded set $\hat{\mathcal{K}} \times \hat{\mathcal{L}}$ with high probability for every t if we properly choose the batch sample size M_2, the smoothing radius r and the inner-loop accuracy ϵ_1. Before stating the IR result, we state some nice Lipschitzness properties over the set $\hat{\mathcal{K}} \times \hat{\mathcal{L}}$ that will contribute to our analysis.

Proposition 1: Let $\boldsymbol{K}_0 \in \mathcal{K}$ and consider the corresponding set $\hat{\mathcal{K}}$. For any $(\boldsymbol{K}, \boldsymbol{L}) \in \hat{\mathcal{K}} \times \hat{\mathcal{L}}$, $\boldsymbol{K}' \in \mathcal{K}, \boldsymbol{L}'$, there exist positive constants D_1, D_2 such that if $\|\boldsymbol{K}' - \boldsymbol{K}\| \leq D_1$, $\|\boldsymbol{L}' - \boldsymbol{L}\| \leq D_2$, then there exist positive constants l_1, l_2 such that $\|\boldsymbol{F}_{\boldsymbol{K}',\boldsymbol{L}} - \boldsymbol{F}_{\boldsymbol{K},\boldsymbol{L}}\| \leq l_1 \|\boldsymbol{K}' - \boldsymbol{K}\|$, and $\|\boldsymbol{F}_{\boldsymbol{K},\boldsymbol{L}'} - \boldsymbol{F}_{\boldsymbol{K},\boldsymbol{L}}\| \leq l_2 \|\boldsymbol{L}' - \boldsymbol{L}\|$. Similar results also hold when replacing $\boldsymbol{F}_{\boldsymbol{K},\boldsymbol{L}}$ by $\boldsymbol{E}_{\boldsymbol{K},\boldsymbol{L}}, \boldsymbol{\Sigma}_{\boldsymbol{K},\boldsymbol{L}}$, and $\boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}}$.

The smoothness and continuity over the set $\hat{\mathcal{K}} \times \hat{\mathcal{L}}$ naturally motivate us to borrow the ideas from stochastic optimization. In particular, it is tempting to follow the analysis of stochastic nested algorithms for global Lipschitz smooth functions, see for instance [41]. Unfortunately, such analysis is not directly applicable since the properties stated in Proposition 1, only hold locally within the set $\hat{\mathcal{K}} \times \hat{\mathcal{L}}$, therefore one needs to ensure that the iterates of Algorithm 2 remain in this set. This can be achieved by controlling the value matrix $\boldsymbol{P}_{\boldsymbol{K},\boldsymbol{L}(\boldsymbol{K})}$ along the iterations. When the exact (natural) gradients are available, [1] utilize this idea to show that the sequence $(\boldsymbol{P}_{\boldsymbol{K}_t,\boldsymbol{L}(\boldsymbol{K}_t)})$ is monotone along the trajectory in the positive semi-definite sense and refer to this property as *implicit regularization*. However, in the case when the estimated gradients (from ZO estimation) are used, the situation is more challenging. Such sequence is no longer monotone and the deviation from monotonicity must be controlled.

In the following, we state one of our key technical results, which ensures that the iterates will remain in the set $\hat{\mathcal{K}} \times \hat{\mathcal{L}}$ with high probability. The key technical improvement over the similar result in Theorem 4.2 of [1] is that we require a much smaller number of samples for achieving this. This improvement is crucial for achieving our better total sample complexity stated in Theorems 1 and 2.

Proposition 2: (Implicit regularization using single-point estimation) Let Assumption 1 hold. Let $\boldsymbol{K}_0 \in \mathcal{K}$ and consider the corresponding $\hat{\mathcal{K}}$ set defined in (15). For any $\delta_1 \in (0,1), \epsilon_1 > 0$ and for any $\boldsymbol{K} \in \mathcal{K}$, Algorithm 1 with single-point estimation outputs $\boldsymbol{L}$ such that $\mathcal{G}(\boldsymbol{K}, \boldsymbol{L}(\boldsymbol{K})) - \mathcal{G}(\boldsymbol{K}, \boldsymbol{L}) \leq \epsilon_1$ with probability at least $1 - \delta_1$ using $M_1 = \widetilde{\mathcal{O}}(\epsilon_1^{-2})$ samples. Moreover for any $\delta_2 \in (0,1)$ and any integer $T \geq 1$, if the estimation parameters in Algorithm 2 satisfy $M_2 = \tilde{\mathcal{O}}(T^2)$, $\tau_2 = \mathcal{O}(1)$, $r_2 = \mathcal{O}(T^{-1/2})$, $\epsilon_1 = \mathcal{O}(T^{-1})$, $\delta_1 = \mathcal{O}(\delta_2/T)$, then, it holds with probability at least $1 - \delta_2$ that $\boldsymbol{K}_t \in \hat{\mathcal{K}}$ for all $t = 1, \cdots, T$.

Proof: The key step in the proof is a descent-like inequality for the value matrix sequence $(\boldsymbol{P}_{\boldsymbol{K}_t,\boldsymbol{L}(\boldsymbol{K}_t)})$ (in the positive semi-definite sense) which holds with high probability :

$$\begin{aligned}
&\boldsymbol{P}_{\boldsymbol{K}_{t+1},\boldsymbol{L}(\boldsymbol{K}_{t+1})} - \boldsymbol{P}_{\boldsymbol{K}_t,\boldsymbol{L}(\boldsymbol{K}_t)} \\
&\quad \preceq \tau_2 (c_1 \cdot r_2^2 + c_2 \cdot \epsilon_1 + c_3 \cdot \|V(\widetilde{\boldsymbol{F}}_{\boldsymbol{K}_t,\boldsymbol{L}_t})\|) \cdot I \\
&\quad\quad - \frac{\tau_2}{4} \boldsymbol{F}_{\boldsymbol{K}_t,\boldsymbol{L}(\boldsymbol{K}_t)}^\top \boldsymbol{F}_{\boldsymbol{K}_t,\boldsymbol{L}(\boldsymbol{K}_t)} \qquad (18)
\end{aligned}$$

where c_1, c_2, c_3 are positive constants and $V(\widetilde{\boldsymbol{F}}_{\boldsymbol{K}_t,\boldsymbol{L}_t}) := (\widetilde{\boldsymbol{F}}_{\boldsymbol{K}_t,\boldsymbol{L}_t} - \mathbb{E}[\widetilde{\boldsymbol{F}}_{\boldsymbol{K}_t,\boldsymbol{L}_t}])^\top (\widetilde{\boldsymbol{F}}_{\boldsymbol{K}_t,\boldsymbol{L}_t} - \mathbb{E}[\widetilde{\boldsymbol{F}}_{\boldsymbol{K}_t,\boldsymbol{L}_t}])$. From (18), we can observe that the deviation can be upperbounded by three sources of estimation errors: a $\mathcal{O}(r_2^2)$ bias term induced by the ZO estimate, the inner-loop error ϵ_1, and a variance-like term induced by the ZO estimation procedure. Hence, the deviation can be controlled by choosing $\epsilon_1 = \mathcal{O}(1/T)$, $r_2 = \mathcal{O}(T^{-1/2})$ and a large enough M_2 such that $V(\widetilde{\boldsymbol{F}}_{\boldsymbol{K}_t,\boldsymbol{L}_t}) = \mathcal{O}(1/T)$. This control allows to show that $\boldsymbol{K}_{t+1}$ can be kept in $\hat{\mathcal{K}}$ for $t = 0, \cdots, T-1$. Inequality (18) follows from the Lipschitzness properties in Proposition 1 and borrows ideas from the analysis of stochastic double-loop algorithms for functions with similar curvature properties such as Lipschitz smoothness and continuity (see supplementary material of [41], for example). ■

Theorem 1: Under the setting of Proposition 2, for every integer $T \geq 1$, it holds with probability at least $1 - \delta_2$ that

$$\frac{1}{T} \sum_{t=0}^{T-1} \|\boldsymbol{F}_{\boldsymbol{K}_t,\boldsymbol{L}(\boldsymbol{K}_t)}\|_F^2 = \mathcal{O}\left(\frac{1}{T}\right).$$

In other words, Algorithm 2 reaches with high probability an ϵ-stationary point (i.e., $\|\boldsymbol{F}_{\boldsymbol{K}_{\text{out}},\boldsymbol{L}(\boldsymbol{K}_{\text{out}})}\|_F^2 \leq \epsilon$) and hence an ϵ-neighborhood of the NE[‖] with a total sample complexity given by $T(T_{\text{in}} M_1 + M_2) = \widetilde{\mathcal{O}}(\epsilon^{-3})$.

Proof: The convergence rate result follows from multiplying (18) by $\boldsymbol{\Sigma_0}$, taking the trace, and computing the telescoping sum. ■

Remark 4: Our $\widetilde{\mathcal{O}}(\epsilon^{-3})$ total sample complexity result improves over the $\widetilde{\mathcal{O}}(\epsilon^{-9})$ sample complexity shown in [1]. The improvement of our algorithms comes from three elements: (a) we have a looser requirement for the inner-loop problem accuracy $\epsilon_1 = \mathcal{O}(T^{-1})$ while in [1] $\epsilon_1 = \mathcal{O}(T^{-2})$; (b) we achieve a better sample complexity for the outer-loop problem using a more careful decomposition of the estimation error caused by the estimated natural gradients: we only require $r_2 = \mathcal{O}(T^{-1/2})$ while [1] chose $r_2 = \mathcal{O}(T^{-1})$ and (c) we reduce the number of inner-loop algorithm calls with a more natural version of the model-free nested algorithm (see the comparison at the end of Section IV). Hence the outer-loop sample complexity is improved from $\mathcal{O}(\epsilon^{-5})$ to $TM_2 = \widetilde{\mathcal{O}}(\epsilon^{-3})$. Combining all of these three elements, we improve the total sample complexity provided in [1] which is given by: $T(T_{in} M_1 M_2 + T_{in} M_1) = \mathcal{O}(\epsilon^{-9})$[**].

[‖] Here the correspondence between stationary point and NE can be found in Lemma 3.2 of [1].

[**] Notice that the total sample complexity for inner and outer loops together was not explicitely stated in [1], but can be inferred from their intermediate results.

In the following theorem, we utilize the two-point zeroth order estimation method which enjoys smaller variance and hence leads to improved sample complexity.

Theorem 2: (Sample complexity using two-point estimation) Let Assumption 1 hold. Let $\boldsymbol{K}_0 \in \mathcal{K}$ and consider the corresponding set $\hat{\mathcal{K}}$ defined in (15). For any $\delta_1 \in (0,1), \epsilon_1 > 0$ and for any $\boldsymbol{K} \in \mathcal{K}$, Algorithm 1 with two-point estimation outputs $\boldsymbol{L}$ such that $\mathcal{G}(\boldsymbol{K}, \boldsymbol{L}(\boldsymbol{K})) - \mathcal{G}(\boldsymbol{K}, \boldsymbol{L}) \leq \epsilon_1$ with probability at least $1 - \delta_1$ using $M_1 = \widetilde{O}(\epsilon_1^{-1})$ samples. Moreover for any $\delta_2 \in (0,1)$ and any integer $T \geq 1$, if the estimation parameters in Algorithm 2 satisfy $M_2 = \tilde{\mathcal{O}}(T)$, $\tau_2 = \mathcal{O}(1)$, $r_2 = \mathcal{O}(T^{-1/2})$, $\epsilon_1 = \mathcal{O}(T^{-1})$, $\delta_1 = \mathcal{O}(\delta_2/T)$. Then, it holds with probability at least $1 - \delta_2$ that $\boldsymbol{K}_t \in \hat{\mathcal{K}}$ for all $t = 1, \cdots, T$ and $\frac{1}{T}\sum_{t=0}^{T-1} \|\boldsymbol{F}_{\boldsymbol{K}_t, \boldsymbol{L}(\boldsymbol{K}_t)}\|_F^2 = \mathcal{O}\left(\frac{1}{T}\right)$. In other words, Algorithm 2 returns an ϵ-stationary point (i.e., $\|\boldsymbol{F}_{\boldsymbol{K}_{\text{out}}, \boldsymbol{L}(\boldsymbol{K}_{\text{out}})}\|_F^2 \leq \epsilon$) after $\mathcal{O}(\epsilon^{-1})$ iterations. The total sample complexity is given by $T\left(T_{\text{in}} M_1 + M_2\right) = \widetilde{\mathcal{O}}(\epsilon^{-2})$.

Remark 5: (Two-point estimation) To obtain Theorem 2, we assume to have access to cost values at two different controllers $\boldsymbol{K}_t^1$ and $\boldsymbol{K}_t^2$ under the same realization of noise $\boldsymbol{\xi}_m$. This assumption can be limiting since it implies that $\boldsymbol{\xi}_m$ is generated in advance. Recently developed techniques of first-order estimation for single agent LQR (instead of ZO) [23] might help to avoid this assumption in the future.

VI. Simulations

In this section, we present simulation results[††] to further validate our contribution. We mainly present simulation results to show that (i) Algorithm 2 in [1] (benchmark algorithm) and Algorithm 2 converge when solving the same zero-sum LQ game using the same set of algorithm parameters; (ii) Algorithm 2 is more sample-efficient compared to the benchmark algorithm.

a) Simulation setup: All the experiments are executed with Python 3.8.5 on a high-performance computing cluster where the reserved memory for executing experiments is 2000 MB. For the sake of comparison, we adopt the same set of model parameters as [1]. Here we repeat the setting for completeness. We use single-point zeroth-order estimation in the following simulations. The horizon length H is set to 5 and $A_t = A$, $B_t = B$, $D_t = D$, $Q_t = Q$, $R_t^u = R^u$, and $R_t^w = R_w$, where

$$A = \begin{bmatrix} 1 & 0 & -5 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & -10 & 0 \\ 0 & 3 & 1 \\ -1 & 0 & 2 \end{bmatrix},$$

$$D = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.2 & 0 \\ 0 & 0 & 0.2 \end{bmatrix}, \quad Q = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix},$$

$$R^u = \begin{bmatrix} 4 & -1 & 0 \\ -1 & 4 & -2 \\ 0 & -2 & 3 \end{bmatrix}, \quad R^w = 5 \cdot I.$$

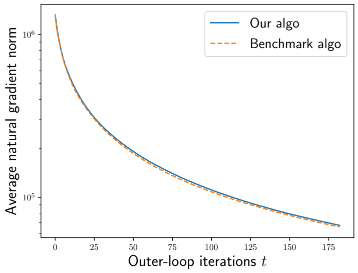

Fig. 1: Comparison between Algorithm 2 and the benchmark algorithm when using a fixed number of inner-loop iterations, exact inner-loop natural gradient, and estimated outer-loop natural gradients with $M_2 = 5 \times 10^5$, $\tau_1 = 0.1$, $\tau_2 = 2 \times 10^{-3}$, and $r_2 = 0.02$. In the figure, we show the convergence in terms of $T^{-1}\sum_{t=0}^{T-1} \|\boldsymbol{F}_{\boldsymbol{K}_t, \boldsymbol{L}(\boldsymbol{K}_t)}\|_F^2$.

Using the Nash equilibrium solution $(\boldsymbol{K}^*, \boldsymbol{L}^*)$ of the above game, we have $\mathcal{G}(\boldsymbol{K}^*, \boldsymbol{L}^*) \approx 3.2330$ and $\lambda_{\min}(\boldsymbol{H}_{\boldsymbol{K}^*, \boldsymbol{L}^*}) \approx 4.2860$. For the purpose of comparison, we choose the same set of parameters for both the benchmark algorithm and Algorithm 2 in this paper. We choose $\boldsymbol{\Sigma}_0 = 0.05 \cdot I$, and default values of other parameters are as follows $r_2 = 0.08$, $M_2 = 5 \times 10^5$, $\epsilon_1 = 10^{-4}$, $\tau_1 = 0.1$, $\tau_2 = 4.67 \times 10^{-4}$,

$$\boldsymbol{K}_0 = \begin{bmatrix} \operatorname{diag}(K, K, K, K, K) & \boldsymbol{0}_{15 \times 3} \end{bmatrix},$$
$$K := \begin{bmatrix} -0.08 & 0.35 & 0.62 \\ -0.21 & 0.19 & 0.32 \\ -0.06 & 0.10 & 0.41 \end{bmatrix}, \quad \boldsymbol{L}_0 = \boldsymbol{0}_{15 \times 18}.$$

b) Sample complexity improvement: In the implementation of Algorithm 2, we adopt a constant number T_{in} with default value 10 of inner-loop iterations instead of assuming access to ϵ_1 to determine when to terminate the inner-loop iterations[‡‡]. In Figure 1, our algorithm shows a comparable convergence rate compared to the benchmark algorithm when using exact inner-loop natural gradients. These results indicate that Algorithm 2 is more sample-efficient than the benchmark algorithm. Indeed, in the benchmark algorithm, an inner-loop problem needs to be solved using samples at each sample step $m = 0, \cdots, M_2 - 1$, which will demand many more samples to solve inner-loop problems when the exact inner-loop solutions are not accessible. Hence the comparable rates imply the advantage of our algorithm compared with the benchmark algorithm.

VII. Conclusion

In this work, we showed a $\tilde{\mathcal{O}}(\epsilon^{-3})$ sample complexity for a derivative-free nested natural policy gradient algorithm for solving the stochastic zero-sum linear quadratic dynamic game problem, improving over prior work. We further improved this sample complexity to $\tilde{\mathcal{O}}(\epsilon^{-2})$ using zeroth order two-point estimation. Possible future research directions include (a) extending our analysis to continuous-time and infinite-horizon settings beyond our finite-horizon setting using techniques such as sensitivity analysis for stable

[††] The codes can be found at https://github.com/wujiduan/Zero-sum-LQ-Games.git.

[‡‡] In the implementation of the benchmark algorithm, we assume exact access to the solution of the inner-loop problem given each perturbed $\boldsymbol{K}_t^m$, i.e., $\boldsymbol{L}(\boldsymbol{K}_t^m)$ $m = 0, \cdots, M_2 - 1$ for the efficiency of the simulations.

continuous-time Lyapunov equations [42], (b) improving the dependence on problem dimensions and considering more general noise distributions since the boundedness of noises is not required by the stability constraint under the finite-horizon setting, and (c) establishing lower bounds for solving this problem. Designing theoretically grounded single-loop algorithms for zero-sum LQ games and considering more involved dynamics such as certain nonlinear dynamics [29], [43] offer avenues of future research that merit further investigation.

References

[1] K. Zhang, X. Zhang, B. Hu, and T. Basar, "Derivative-free policy optimization for linear risk-sensitive and robust control design: Implicit regularization and sample complexity," *Advances in Neural Information Processing Systems*, vol. 34, pp. 2949–2964, 2021.

[2] P. Makila and H. Toivonen, "Computational methods for parametric LQ problems–a survey," *IEEE Transactions on Automatic Control*, vol. 32, no. 8, pp. 658–671, 1987.

[3] B. Hu, K. Zhang, N. Li, M. Mesbahi, M. Fazel, and T. Başar, "Towards a theoretical foundation of policy optimization for learning control policies," *Annual Review of Control, Robotics, and Autonomous Systems*, 2022.

[4] B. Polyak, "Gradient methods for the minimisation of functionals," *USSR Computational Mathematics and Mathematical Physics*, p. 864–878, 1963.

[5] M. Fazel, R. Ge, S. Kakade, and M. Mesbahi, "Global convergence of policy gradient methods for the linear quadratic regulator," in *International conference on machine learning*, pp. 1467–1476, PMLR, 2018.

[6] L. Ljung, *System Identification*. Birkhäuser Boston, 1998.

[7] C.-N. Fiechter, "PAC adaptive control of linear systems," in *Proc. 14th International Conference on Machine Learning*, pp. 116–124, Morgan Kaufmann, 1997.

[8] G. Hewer, "An iterative technique for the computation of the steady state gains for the discrete optimal regulator," *IEEE Transactions on Automatic Control*, vol. 16, no. 4, pp. 382–384, 1971.

[9] P. Lancaster and L. Rodman, *Algebraic Riccati equations*. Clarendon press, 1995.

[10] V. Balakrishnan and L. Vandenberghe, "Semidefinite programming duality and linear time-invariant systems," *IEEE Transactions on Automatic Control*, vol. 48, no. 1, pp. 30–41, 2003.

[11] S. P. Bhattacharyya and L. H. Keel, "Robust control: the parametric approach," in *Advances in control education 1994*, pp. 49–52, Elsevier, 1995.

[12] M. C. Campi and M. R. James, "Nonlinear discrete-time risk-sensitive optimal control," *International Journal of Robust and Nonlinear Control*, vol. 6, no. 1, pp. 1–19, 1996.

[13] T. Başar and P. Bernhard, "$\mathcal{H}_\infty$-optimal control and related minimax design problems," *Springer Book Archive-Mathematics*, 1995.

[14] K. Zhang, B. Hu, and T. Başar, "Policy optimization for $\mathcal{H}_2$ linear control with $\mathcal{H}_\infty$ robustness guarantee: Implicit regularization and global convergence," *SIAM Journal on Control and Optimization*, vol. 59, no. 6, pp. 4081–4109, 2021.

[15] E. Mageirou and Y. Ho, "Decentralized stabilization via game theoretic methods," *Automatica*, vol. 13, no. 4, pp. 393–399, 1977.

[16] J. Wu, A. Barakat, I. Fatkhullin, and N. He, "Learning zero-sum linear quadratic games with improved sample complexity," 2023.

[17] D. Malik, A. Pananjady, K. Bhatia, K. Khamaru, P. Bartlett, and M. Wainwright, "Derivative-free methods for policy optimization: Guarantees for linear quadratic systems," in *The 22nd international conference on artificial intelligence and statistics*, pp. 2916–2925, PMLR, 2019.

[18] H. Mohammadi, M. Soltanolkotabi, and M. R. Jovanović, "On the linear convergence of random search for discrete-time LQR," *IEEE Control Systems Letters*, vol. 5, no. 3, pp. 989–994, 2020.

[19] B. Hambly, R. Xu, and H. Yang, "Policy gradient methods for the noisy linear quadratic regulator over a finite horizon," *arXiv preprint arXiv:2011.10300*, Jun 2021.

[20] I. Fatkhullin and B. Polyak, "Optimizing static linear feedback: Gradient method," *SIAM Journal on Control and Optimization*, vol. 59, no. 5, pp. 3887–3911, 2021.

[21] H. Mohammadi, A. Zare, M. Soltanolkotabi, and M. R. Jovanović, "Convergence and sample complexity of gradient methods for the model-free linear–quadratic regulator problem," *IEEE Transactions on Automatic Control*, vol. 67, no. 5, pp. 2435–2450, 2021.

[22] M. Giegrich, C. Reisinger, and Y. Zhang, "Convergence of policy gradient methods for finite-horizon stochastic linear-quadratic control problems," *arXiv preprint arXiv:2211.00617*, 2022.

[23] C. Ju, G. Kotsalis, and G. Lan, "A model-free first-order method for linear quadratic regulator with $\tilde{O}(1/\varepsilon)$ sampling complexity," *arXiv preprint arXiv:2212.00084*, Feb 2023.

[24] Z. Yang, Y. Chen, M. Hong, and Z. Wang, "On the global convergence of actor-critic: A case for linear quadratic regulator with ergodic cost," *arXiv preprint arXiv:1907.06246*, Jul 2019.

[25] J. Bu and M. Mesbahi, "Global convergence of policy gradient algorithms for indefinite least squares stationary optimal control," *IEEE Control Systems Letters*, vol. 4, no. 3, pp. 638–643, 2020.

[26] X. Guo and B. Hu, "Global convergence of direct policy search for state-feedback $\mathcal{H}_\infty$ robust control: A revisit of nonsmooth synthesis with goldstein subdifferential," in *Thirty-Sixth Conference on Neural Information Processing Systems*, 2022.

[27] Y. Sun and M. Fazel, "Learning optimal controllers by policy gradient: Global optimality via convex parameterization," in *2021 60th IEEE Conference on Decision and Control (CDC)*, p. 4576–4581, IEEE, Dec 2021.

[28] H. Feng and J. Lavaei, "On the exponential number of connected components for the feasible set of optimal decentralized control problems," in *2019 American Control Conference (ACC)*, pp. 1430–1437, IEEE, 2019.

[29] Y. Han, M. Razaviyayn, and R. Xu, "Policy gradient finds global optimum of nearly linear-quadratic control systems," in *OPT 2022: Optimization for Machine Learning (NeurIPS 2022 Workshop)*, 2022.

[30] K. Zhang, Z. Yang, and T. Basar, "Policy optimization provably converges to nash equilibria in zero-sum linear quadratic games," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[31] J. Bu, L. J. Ratliff, and M. Mesbahi, "Global convergence of policy gradient for sequential zero-sum linear quadratic dynamic games," *arXiv preprint arXiv:1911.04672*, 2019.

[32] A. Al-Tamimi, F. L. Lewis, and M. Abu-Khalaf, "Model-free Q-learning designs for discrete-time zero-sum games with application to H-infinity control," in *2007 European Control Conference (ECC)*, p. 1668–1675, Jul 2007.

[33] R. Carmona, K. Hamidouche, M. Laurière, and Z. Tan, "Linear-quadratic zero-sum mean-field type games: Optimality conditions and policy optimization," *arXiv preprint arXiv:2009.00578*, 2020.

[34] R. Carmona, M. Laurière, and Z. Tan, "Linear-quadratic mean-field reinforcement learning: Convergence of policy gradient methods," *arXiv preprint arXiv:1910.04295*, 2019.

[35] E. Mazumdar, L. J. Ratliff, M. I. Jordan, and S. S. Sastry, "Policy-gradient algorithms have no guarantees of convergence in linear quadratic games," *arXiv preprint arXiv:1907.03712*, 2019.

[36] B. Hambly, R. Xu, and H. Yang, "Policy gradient methods find the nash equilibrium in n-player general-sum linear-quadratic games," *arXiv preprint arXiv:2107.13090*, Aug 2022.

[37] H. Yang, *Policy gradient methods for linear quadratic problems*. PhD thesis, University of Oxford, 2022.

[38] L. Furieri and M. Kamgarpour, "First order methods for globally optimal distributed controllers beyond quadratic invariance," in *American Control Conference (ACC)*, p. 4588–4593, Jul 2020.

[39] A. D. Flaxman, A. T. Kalai, and H. B. McMahan, "Online convex optimization in the bandit setting: gradient descent without a gradient," *arXiv preprint cs/0408007*, 2004.

[40] A. Agarwal and O. Dekel, "Optimal algorithms for online convex optimization with multi-point bandit feedback.," in *Colt*, pp. 28–40, Citeseer, 2010.

[41] T. Lin, C. Jin, and M. Jordan, "On gradient descent ascent for nonconvex-concave minimax problems," in *International Conference on Machine Learning*, pp. 6083–6093, PMLR, 2020.

[42] G. Hewer and C. Kenney, "The sensitivity of the stable Lyapunov equation," *SIAM journal on control and optimization*, vol. 26, no. 2, pp. 321–344, 1988.

[43] Y. Han, M. Razaviyayn, and R. Xu, "Policy gradient converges to the globally optimal policy for nearly linear-quadratic regulators," *arXiv preprint arXiv:2303.08431*, 2023.