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

Data-enabled Policy Optimization for the Linear Quadratic Regulator

Feiran Zhao, Florian Dörfler, Keyou You

Abstract— **Policy optimization (PO), an essential approach of reinforcement learning for a broad range of system classes, requires significantly more system data than indirect (identification-followed-by-control) methods or behavioral-based direct methods even in the simplest linear quadratic regulator (LQR) problem. In this paper, we take an initial step towards bridging this gap by proposing the data-enabled policy optimization (DeePO) method, which requires only a finite number of sufficiently exciting data to iteratively solve the LQR problem via PO. Based on a data-driven closed-loop parameterization, we are able to directly compute the policy gradient from a batch of persistently exciting data. Next, we show that the nonconvex PO problem satisfies a projected gradient dominance property by relating it to an equivalent convex program, leading to the global convergence of DeePO. Moreover, we apply regularization methods to enhance the certainty-equivalence and robustness of the resulting controller and show an implicit regularization property. Finally, we perform simulations to validate our results.**

I. Introduction

As a cornerstone of modern control theory, the linear quadratic regulator (LQR) problem has been the benchmark for data-driven control methods that seek to design a controller from raw system data. The manifold approaches to data-driven control can be broadly categorized as *indirect* (when identifying a dynamical model followed by model-based control design) versus *direct* (when bypassing the identification step). The use of direct data-driven control is usually motivated when the dynamical model is difficult to establish, or is too complex for model-based control design. As an end-to-end approach, the direct methods are conceptually simple and easy to implement in practice.

A representative instance of direct data-driven control is policy optimization (PO), an essential approach for applications of reinforcement learning (RL) [1]–[3]. As an iterative method, PO directly searches over the policy space to optimize a performance metric of interest. Based on zeroth-order optimization techniques, it uses multiple system trajectories to estimate the policy gradient. There has been a resurgent interest in studying theoretical properties of PO on the LQR problem such as convergence and sample complexity; see e.g., [4]–[7] and the comprehensive survey [8]. Even though global convergence has been shown for the nonconvex PO problem by a *gradient dominance* property [4], there exists a considerable gap in the sample complexity between PO and indirect methods, which have proved themselves to be more sample-efficient [9], [10] for solving the LQR problem. This gap is due to the exploration or trial-and-error nature of RL, or more specifically, that the cost used for gradient estimate can only be evaluated *after* a whole trajectory is observed. Thus, the existing PO methods require numerous system trajectories to find an optimal policy, even in the simplest LQR setting.

Recent years have witnessed an emerging line of direct methods inspired by the *Fundamental Lemma* [11], which states that the behavior of a linear time-invariant (LTI) system can be characterized by the range space of raw data matrices. This result implies a non-parametric representation of LTI systems, giving rise to a notable implicit design called data-enabled predictive control (DeePC) [12], which has seen many successful implementations in different practical scenarios [13]. The fundamental lemma has also been utilized to solve various explicit control design and analysis problems [14]–[16]. In particular, it has been shown in [14] that using subspace relations, the closed-loop LTI system can be parameterized by input-state data, leading to a data-based convex reformulation of the LQR problem. Compared with PO, this approach is significantly more sample-efficient as it only requires a batch of persistently exciting (PE) data. Indeed, the PE condition is equivalent to identifiability for LTI systems and should be a minimal assumption for most control design problems [15], [17], e.g., the LQR problem. There have been many recent works leveraging regularization methods to promote certainty-equivalence and robustness of the LQR [18]–[20], and to bridge behavioral-based direct and indirect methods [21]. All these methods use only a small batch of PE data compared to data-hungry zeroth-order PO methods [4]–[6]. This leads to a natural question: does there exist a data-efficient PO method for solving the LQR problem?

In this paper, we provide an affirmative answer to the above question. By leveraging the data-driven closed-loop parameterization [14], we propose an iterative method called **d**ata-**e**nabl**e**d **p**olicy **o**ptimization (DeePO) to solve the LQR problem. Instead of estimating the policy gradient from the cost of observed trajectories, we show that after a change of optimization variables, the gradient can be directly characterized from a batch of PE data. Even though the resulting optimization problem is nonconvex, it can be parameterized as a data-based convex program. By exploiting this relation and using a recent PO result [22], we further show that the LQR cost is *projected gradient* dominated, while it is only *gradient* dominated in [4], [5]. By establishing that the cost

Research of F. Zhao and K. You was supported by National Key R&D Program of China (2022ZD0116700) and National Natural Science Foundation of China (62033006, 62325305).

F. Zhao and K. You are with the Department of Automation and BNRist, Tsinghua University, Beijing 100084, China. (e-mail: zhaofr18@mails.tsinghua.edu.cn, youky@tsinghua.edu.cn.) F. Dörfler is with the Department of Information Technology and Electrical Engineering, ETH Zurich, 8092 Zurich, Switzerland. (e-mail: dorfler@control.ee.ethz.ch)

is also locally smooth, we show that the projected gradient method converges to the global optimum. We also investigate how regularization [18]–[20] affects the convergence of DeePO. In particular, we show that the certainty-equivalence regularizer leads to an *implicit regularization* property, meaning that the DeePO algorithm without regularization behaves as if it is regularized. This property has been advocated as an important feature of gradient-based methods for solving many nonconvex problems [23], [24]. Finally, we perform a numerical case study to validate our theoretical results. We are hopeful that the discovered DeePO method with significantly relaxed data requirements offers a possible path towards direct adaptive LQR control.

The rest of this paper is organized as follows. In Section II, we revisit the LQR problem and recapitulate the data-driven LQR formulation. In Section III, we propose the DeePO method to iteratively solve the LQR problem and show its global convergence. Section IV studies the effects of two regularizers on the convergence of DeePO. Section V uses a numerical example to validate our main results. Conclusion and future work in Section VI complete this paper.

Notation. We use I_n to denote the n-by-n identity matrix. We use $\underline{\sigma}(\cdot)$ to denote the minimal singular value of a matrix. We use $\|\cdot\|$ to denote the 2-norm of a vector or a matrix, and $\|\cdot\|_F$ the Frobenius norm. We use $\rho(\cdot)$ to denote the spectral radius of a square matrix. We use $\text{poly}(\cdot)$ to denote a polynomial function. We use $\dagger$ to denote the right inverse of a full row rank matrix.

II. Problem Formulation

In this section, we first revisit the model-based LQR problem. By recapitulating its direct data-driven formulation from [14], we then propose our PO reformulation.

A. The Model-based LQR problem

Consider a discrete-time LTI system

$$x(t+1) = Ax(t) + Bu(t), \tag{1}$$

where $x(t) \in \mathbb{R}^n$ and $u(t) \in \mathbb{R}^m$ are the state and control input, respectively. We assume that (A, B) are controllable.

The LQR problem is phrased as finding a state-feedback gain $K \in \mathbb{R}^{m\times n}$ to minimize the quadratic cost

$$J(K) := \mathbb{E}_{x(0)\sim\mathcal{D}}\left[\sum_{t=0}^{\infty}(x(t)^\top Qx(t) + u(t)^\top Ru(t))\right], \tag{2}$$

where $Q \succ 0, R \succ 0$ are penalty matrices, and $\{x(t), u(t)\}$ is the trajectory following (1) and $u(t) = Kx(t)$ starting from the initial state $x(0)$. The distribution $\mathcal{D}$ of $x(0)$ satisfies $\mathbb{E}[x(0)] = 0$ and $\mathbb{E}[x(0)x(0)^\top] = I_n$. It is well-known that the unique optimal gain to (2) is

$$K^* = -(R + B^\top P^* B)^{-1} B^\top P^* A,$$

where P^* is the unique positive semi-definite solution to the algebraic Riccati equation [25]

$$P^* = A^\top P^* A + Q - A^\top P^* B(R + B^\top P^* B)^{-1} B^\top P^* A.$$

We aim to solve the LQR problem in a direct data-driven approach when (A, B) are unknown, but we assume the access to a T-length dataset of states and control inputs.

B. Direct data-driven formulation

Define the offline data matrices

$$\begin{aligned} X_- &= \begin{bmatrix} x(0) & x(1) & \dots & x(T-1)\end{bmatrix} \in \mathbb{R}^{n\times T},\\ U_- &= \begin{bmatrix} u(0) & u(1) & \dots & u(T-1)\end{bmatrix} \in \mathbb{R}^{m\times T},\\ X_+ &= \begin{bmatrix} x(1) & x(2) & \dots & x(T)\end{bmatrix} \in \mathbb{R}^{n\times T}, \end{aligned}$$

which satisfy the system dynamics (1)

$$X_+ = AX_- + BU_-. \tag{3}$$

Throughout the paper, we assume that the following block matrix of input and state data

$$D_- = \begin{bmatrix} U_- \\ X_- \end{bmatrix} \in \mathbb{R}^{(m+n)\times T}$$

has full row rank

$$\text{rank}(D_-) = m + n, \tag{4}$$

i.e., the information in the data is sufficiently rich. This condition is necessary for identifying (A, B) from data and for solving the data-driven LQR problem [15]. As shown in [14], it can be ensured provided that the input data U_- is PE of order $n+1$. Note that the columns of (X_-, U_-, X_+) are not necessarily consecutive data samples. In fact, they could be from independent or multiple averaged experiments as long as they satisfy (3) and (4) [14].

Under the rank condition (4), there exists a matrix $G \in \mathbb{R}^{T\times n}$ that satisfies

$$\begin{bmatrix} K \\ I_n \end{bmatrix} = D_- G \tag{5}$$

for any given K. That is, K can be parameterized by $K = U_- G$ where G satisfies a linear constraint $X_- G = I_n$. Then, the closed-loop matrix can be expressed in a data-driven fashion as [14]

$$A + BK = [B \;\; A]\begin{bmatrix} K \\ I_n \end{bmatrix} = (AX_- + BU_-)G = X_+ G,$$

leading to the following closed-loop system

$$x(t+1) = X_+ Gx(t). \tag{6}$$

Furthermore, the LQR problem becomes

$$\begin{aligned} &\underset{G}{\text{minimize}} \;\; J(G),\\ &\text{subject to } G \in \mathcal{S}_G := \{G | X_- G = I_n, \rho(X_+ G) < 1\}. \end{aligned} \tag{7}$$

Here, $J(G)$ is the LQR cost following (6) and $u(t) = U_- Gx(t)$, and $\mathcal{S}_G$ is the feasible set. In contrast to the model-based LQR, the problem (7) is characterized by raw data matrices. Though (7) can be reformulated as a semi-definite program (SDP) using techniques from [14], [18], it is computationally challenging to solve for a large data size.

In this paper, we take an iterative PO perspective to solve (7) viewing G as the optimization matrix. We aim to

design a gradient-based method to find an optimal G while maintaining feasibility, and recover the control from (5) as $K = U_-G$. Since (7) is a challenging constrained nonconvex problem, we leverage a novel convex parameterization to establish the global convergence.

III. Data-enabled policy optimization

In this section, we first present our novel PO method for solving (7). Then, we propose a convex parameterization of (7) to derive the projected gradient dominance property of $J(G)$. By establishing the local smoothness of $J(G)$, we are able to show the global convergence of our method.

A. Data-enabled policy optimization to solve (7)

For $G \in \mathcal{S}_G$, the cost $J(G)$ is finite and has the following closed-form expressions [14]

$$J(G) = \text{Tr}\{P_G\} = \text{Tr}\{(Q + G^\top U_-^\top R U_- G)\Sigma_G\}, \quad (8)$$

where P_G satisfies the Lyapunov equation

$$P_G = Q + G^\top U_-^\top R U_- G + G^\top X_+^\top P_G X_+ G, \quad (9)$$

and $\Sigma_G := \mathbb{E}_{x(0)\sim\mathcal{D}}[\sum_{t=0}^{\infty} x(t)x(t)^\top]$ is the state covariance matrix of the closed-loop system (6) satisfying

$$\Sigma_G = I_n + X_+ G \Sigma_G G^\top X_+^\top.$$

We have the following gradient expression for $J(G)$.

Lemma 1: For $G \in \mathcal{S}_G$, the gradient of $J(G)$ is $\nabla J(G) = 2E_G\Sigma_G$ with $E_G := (U_-^\top R U_- + X_+^\top P_G X_+)G$.

The expression of $\nabla J(G)$ is data-driven since both E_G and Σ_G can be computed using raw data matrices under the rank condition (4).

The feasible set $\mathcal{S}_G$ contains a linear constraint $X_-G = I_n$, which motivates the use of projected gradient methods to ensure feasibility. Define the nullspace of X_- as

$$\mathcal{N}(X_-) := \{G \in \mathbb{R}^{T\times n} | X_- G = 0\},$$

and the projection operator $\Pi_{X_-} := I_T - X_-^\dagger X_-$ onto $\mathcal{N}(X_-)$. The projected gradient update is then given by

$$G^+ = G - \eta \Pi_{X_-} \nabla J(G), \quad (10)$$

where $\eta \geq 0$ is the stepsize. We refer to this method as data-enabled policy optimization (DeePO) since the update (10) can be efficiently computed by raw data matrices, and the control can be recovered from (5) as $K = U_-G$. As an iterative search method, the initial policy G^0 requires to satisfy $G^0 \in \mathcal{S}_G$.

Due to non-convexity of both the objective $J(G)$ and the constraint $\mathcal{S}_G$, it is challenging to provide global convergence guarantees for DeePO. Moreover, an optimal solution to (7) is not unique. In fact, it has been shown in [20, Lemma 2.1] that the solution set is

$$\{G | G = G^* + \Delta, \Delta \in \mathcal{N}(D_-)\} \text{ with } G^* = D_-^\dagger \begin{bmatrix} K^* \\ I_n \end{bmatrix}, \quad (11)$$

which contains a considerable nullspace. Nevertheless, based on a recent work [22] that proves optimality via convex parameterization, we are able to show a projected gradient dominance property of $J(G)$.

B. Optimality via a convex parameterization

We first relate (7) to a convex parameterization via a change of variables $G = L\Sigma^{-1}$ as

$$\begin{aligned} &\underset{L,\Sigma}{\text{minimize}} \;\; f(L,\Sigma) := \text{Tr}\{Q\Sigma\} + \text{Tr}\{L\Sigma^{-1}L^\top U_-^\top R U_-\}, \\ &\text{subject to} \;\; \Sigma = X_- L, \;\; \begin{bmatrix} \Sigma - I_n & X_+ L \\ L^\top X_+^\top & \Sigma \end{bmatrix} \succeq 0. \end{aligned} \quad (12)$$

Let $\mathcal{S}$ be its feasible set. The equivalence between the two problems (7) and (12) are established below.

Lemma 2: For any $(L,\Sigma) \in \mathcal{S}$, Σ is invertible and $L\Sigma^{-1} \in \mathcal{S}_G$. Moreover, for $G \in \mathcal{S}_G$ it holds that

$$J(G) = \min_{L,\Sigma} \{f(L,\Sigma), \text{s.t.} (L,\Sigma) \in \mathcal{S}, L\Sigma^{-1} = G\}. \quad (13)$$

In the following lemma, we show the convexity of the parameterization (12).

Lemma 3: The feasible set $\mathcal{S}$ of (12) is convex in (L,Σ), and $f(L,\Sigma)$ is differentiable over an open domain that contains $\mathcal{S}$. Moreover, $f(L,\Sigma)$ is convex over $\mathcal{S}$.

We now formally define the gradient dominance property.

Definition 1: A differentiable function $g(x) : \mathbb{R}^n \to \mathbb{R}$ with a finite global minimum g^* is gradient dominated of degree p over a set $\mathcal{X} \subseteq \text{dom}(g)$ if

$$g(x) - g^* \leq \lambda_{\mathcal{X}} \|\nabla g(x)\|^p, \quad \forall x \in \mathcal{X}, \text{ for some } \lambda_{\mathcal{X}} > 0.$$

The gradient dominance property means that all the stationary points are optimal. Moreover, the convergence rate of gradient-based methods usually depends on the values of the degree p. Particularly, for smooth objective function $p = 1$ leads to a sublinear rate and $p = 2$ leads to a linear rate.

Equipped with Lemmas 2 and 3, we apply [22, Theorem 1] to show the gradient dominance property of $J(G)$ over any sublevel set $S_G(a) := \{G | J(G) \leq a\}$ with $a > 0$.

*Lemma 4 (**Projected gradient dominance of degree 1**):* For $G \in \mathcal{S}_G(a)$, there exists $\mu(a) > 0$ such that

$$J(G) - J^* \leq \mu(a) \|\Pi_{X_-} \nabla J(G)\|,$$

where J^* is the optimal LQR cost to (7).

In contrast to the existing literature [4] on PO for the LQR, the cost $J(G)$ here is *projected gradient* dominated, meaning that G is optimal if the projected gradient $\Pi_{X_-}\nabla J(G)$ is equal to zero. By using Lemma 4, we next show global convergence of the projected gradient descent in (10).

C. Global convergence of DeePO

We first prove the smoothness of $J(G)$. Since $J(G)$ tends extremely to infinity as G approaches the boundary $\partial\mathcal{S}_G$, we can only show that $J(G)$ is *locally* smooth over any sublevel set. Define the Hessian acting on the direction $Z \in \mathbb{R}^{T\times n}$ as $\nabla^2 J(G)[Z,Z] := \frac{d^2}{dt^2} J(G+tZ)\Big|_{t=0}$, and the directional derivative of P_G as $P_G'[Z] := \frac{d}{dt} P_{G+tZ}\Big|_{t=0}$. Then, we have the following closed-form expression for the Hessian.

Lemma 5: For $G \in \mathcal{S}_G$ and a feasible direction $Z \in \mathbb{R}^{T\times n}$, the Hessian of $J(G)$ is characterized by

$$\begin{aligned} \nabla^2 J(G)[Z,Z] = &\, 2\text{Tr}\{Z^\top (U_-^\top R U_- + X_+^\top P_G X_+) Z \Sigma_G\} \\ &+ 4\text{Tr}\{Z^\top X_+^\top P_G'[Z] X_+ G \Sigma_G\}, \end{aligned}$$

where $P'_G[Z] = \sum_{i=0}^{\infty}(G^\top X_+^\top)^i(Z^\top E_G + E_G^\top Z)(X_+G)^i$.

Define $\|\nabla^2 J(G)\| := \sup_{\|Z\|_F=1}\left|\nabla^2 J(G)[Z,Z]\right|$. We show an upper bound for $\|\nabla^2 J(G)\|$ over a sublevel set.

*Lemma 6 (**Local smoothness**):* For $G \in \mathcal{S}_G(a)$, it holds $\|\nabla^2 J(G)\| \le \text{poly}(a, \|U_-\|, \|X_+\|_F, \|R\|, \underline{\sigma}(Q)) := l(a)$, where $l(a)$ is the smoothness constant of $J(G)$ over $\mathcal{S}_G(a)$. That is, for any $G, G' \in \mathcal{S}_G(a)$ satisfying $G + \delta(G' - G) \in \mathcal{S}_G(a), \forall \delta \in [0,1]$, the following inequality holds

$$J(G') \le J(G) + \langle \nabla J(G), G' - G\rangle + l(a)\|G' - G\|^2/2.$$

Under the gradient dominance property of degree 1 in Lemma 4 and the local smoothness in Lemma 6, we now show the global sublinear convergence of DeePO. The key is to select an appropriate stepsize such that the policy sequence is feasible and stays in the sublevel set associated with the initial policy $G^0 \in \mathcal{S}_G$. For simplicity, let μ_0 and l_0 denote the projected gradient dominance and smoothness constants of $J(G)$ over $\mathcal{S}_G(J(G^0))$, respectively. We present our convergence result in the following theorem.

*Theorem 1 (**Global convergence**):* For $G^0 \in \mathcal{S}_G$ and a stepsize $\eta \in (0, 1/l_0]$, the update (10) leads to $G^k \in \mathcal{S}_G(J(G^0)), \forall k \in \mathbb{N}$. Moreover, for any $\epsilon > 0$ and

$$k \ge 2\mu_0^2/(\epsilon(2\eta - l_0\eta^2)),$$

the update (10) enjoys the following performance bound

$$J(G^k) - J^* \le \epsilon.$$

We compare with the traditional PO for the LQR [4]–[6]. Their approach relies on a zeroth-order estimate of the policy gradient, which inevitably requires numerous system trajectories to approximate the cost. In sharp contrast, DeePO directly computes the gradient from a batch of raw data matrices based on a data-based representation of the closed-loop system. This remarkable feature enables DeePO to work with only a small batch of PE data. Moreover, the state-of-the-art sample complexity (in terms of number of sampled trajectories, the length of which can be very long) of PO in [4]–[6] is $\mathcal{O}(\log(1/\epsilon))$, while our sample complexity (in terms of number of state-input pairs) is independent of ϵ. Even though both two approaches achieve global convergence (albeit with vastly different amounts of data), DeePO is more flexible as it is compatible with regularization methods used to enhance the robustness to noisy data, which will be shown in the next section. To the best of our knowledge, there are no robustifying regularization methods that have been applied to the PO method for the LQR problem.

IV. DeePO for the regularized LQR

For the direct data-driven LQR formulation [18]–[20], regularization plays an important role in promoting certainty-equivalence and robust stability when the data is corrupted with noise. This section investigates how regularization affects the convergence of DeePO.

A. Certainty-equivalence regularizer

Consider the regularized LQR problem

$$\begin{aligned} \underset{G}{\text{minimize}} \;\; & J_\lambda(G) := J(G) + \lambda\|\Pi_{D_-} G\Sigma_G^{1/2}\|^2, \\ \text{subject to} \;\; & G \in \mathcal{S}_G, \end{aligned} \tag{14}$$

where $\lambda \ge 0$ is a user-defined constant and $\Pi_{D_-} := I - D_-^\dagger D_-$ is the projection matrix onto the nullspace of D_-. For the noiseless data (X_-, U_-, X_+) here, the orthogonality regularizer in (14) does not change the optimal cost but only singles out a solution G^* satisfying $\Pi_{D_-} G^* = 0$ from the solution set in (11). When the data is corrupted with noises, it promotes certainty-equivalence, i.e., when λ tends to infinity the solution of (14) coincides with that of indirect data-driven control with an underlying maximum likelihood system identification attenuating the effect of noise; we refer interested readers to [20, Section III] for more discussions.

Note that we have added the weighting $\Sigma_G^{1/2}$ to the regularizer (c.f. [20, (15)]) to make it compatible with the convex parameterization (12). As a result, (14) can be formulated with $L\Sigma^{-1} = G$ as the following convex problem

$$\begin{aligned} \underset{L,\Sigma}{\text{minimize}} \;\; & f_\lambda(L,\Sigma) := \text{Tr}\{Q\Sigma\} \\ & \quad + \text{Tr}\{L\Sigma^{-1}L^\top(\lambda\Pi_{D_-}^\top\Pi_{D_-} + U_-^\top R U_-)\}, \\ \text{subject to} \;\; & \Sigma = X_-L, \;\; \begin{bmatrix}\Sigma - I_n & X_+L \\ L^\top X_+^\top & \Sigma\end{bmatrix} \succeq 0. \end{aligned} \tag{15}$$

Comparing (15) with (12), we see that $f_\lambda(L,\Sigma)$ upon amounts to $f(L,\Sigma)$ adding a convex regularizer, and hence $f_\lambda(L,\Sigma)$ is convex. Indeed, by standard matrix analysis [26], its Hessian acting on the direction $(\tilde{L}, \tilde{\Sigma})$ satisfies

$$\begin{aligned} &\nabla^2 f_\lambda(L,\Sigma)[(\tilde{L},\tilde{\Sigma}),(\tilde{L},\tilde{\Sigma})] = \nabla^2 f(L,\Sigma)[(\tilde{L},\tilde{\Sigma}),(\tilde{L},\tilde{\Sigma})] \\ &+ 2\lambda\|(\Pi_{D_-}\tilde{L} - \Pi_{D_-}L\Sigma^{-1}\tilde{\Sigma})\Sigma^{-1/2}\|_F^2 \\ &\ge \nabla^2 f(L,\Sigma)[(\tilde{L},\tilde{\Sigma}),(\tilde{L},\tilde{\Sigma})]. \end{aligned}$$

Moreover, following analogous arguments as in Section III, $J_\lambda(G)$ can also be shown to be locally smooth. Based on previous analysis, the projected gradient update

$$G^+ = G - \eta\Pi_{X_-}\nabla J_\lambda(G) \tag{16}$$

converges to the optimal solution of (14) under a proper stepsize selection.

B. Robustness-promoting regularizer

Regularization can also be used to enhance robust stability. Consider the following regularized LQR problem

$$\begin{aligned} \underset{G}{\text{minimize}} \;\; & J_\gamma(G) := J(G) + \gamma\text{Tr}\{G\Sigma_G G\}, \\ \text{subject to} \;\; & G \in \mathcal{S}_G, \end{aligned} \tag{17}$$

where $\gamma \ge 0$ is a user-defined constant. To see why it promotes the robust stability for noisy data, we note that the state covariance matrix is given by $\Sigma_G = I_n + X_+G\Sigma_G G^\top X_+^\top$. Thus, a small $\text{Tr}\{G\Sigma_G G^\top\}$ can reduce the effect of noises in X_+. Different from the certainty-equivalence regularization, the regularizer in (17) bias the LQR solution even when the data is noiseless, reflecting a trade-off between performance and robustness.

The problem (17) can be formulated with $L\Sigma^{-1} = G$ as

$$\begin{aligned} \underset{L,\Sigma}{\text{minimize}} \;\; & f_\gamma(L,\Sigma) := \text{Tr}\{Q\Sigma\} \\ & \quad + \text{Tr}\{L\Sigma^{-1}L^\top(\gamma I_T + U_-^\top R U_-)\}, \\ \text{subject to} \;\; & \Sigma = X_-L, \;\; \begin{bmatrix}\Sigma - I_n & X_{t+1}L \\ L^\top X_{t+1}^\top & \Sigma\end{bmatrix} \succeq 0. \end{aligned} \tag{18}$$

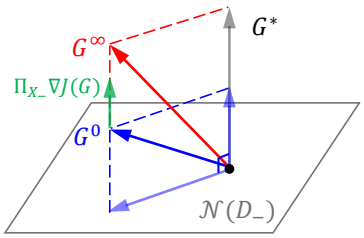

Fig. 1. Supspace relations among $\mathcal{N}(D_-)$, $\Pi_{X_-}\nabla J(G)$, and G^*.

Clearly, $f_\lambda(L,\Sigma)$ is also convex since

$$\nabla^2 f_\gamma(L,\Sigma)[(\tilde{L},\tilde{\Sigma}),(\tilde{L},\tilde{\Sigma})] = \nabla^2 f(L,\Sigma)[(\tilde{L},\tilde{\Sigma}),(\tilde{L},\tilde{\Sigma})] + 2\gamma\|(\tilde{L}-L\Sigma^{-1}\tilde{\Sigma})\Sigma^{-\frac{1}{2}}\|_F^2 \geq \nabla^2 f(L,\Sigma)[(\tilde{L},\tilde{\Sigma}),(\tilde{L},\tilde{\Sigma})].$$

By analogous reasoning and combining the smoothness of the regularizer, the projected gradient update

$$G^+ = G - \eta\Pi_{X_-}\nabla J_\gamma(G) \tag{19}$$

converges to the optimal solution of (17) under a proper stepsize selection.

C. Implicit regularization

Apart from the convergence, we observe an interesting *implicit regularization* property of the certainty-equivalence regularized LQR problem (14) formally defined below.

*Definition 2 (**Implicit regularization**):* For the regularized LQR problem (14), suppose that a convergent algorithm generates a sequence of $\{G^k\}$. If $G^\infty := \lim_{k\to\infty} G^k$ satisfies $\Pi_{D_-}G^\infty = 0$, then the algorithm is called *regularized*; If it is regularized with $\lambda = 0$, then it is called *implicitly* regularized.

The concept of implicit regularization has been adopted in many recent works on nonconvex optimization, including matrix factorization [23] and PO for robust LQR problems [24]. As its name suggests, it means that the algorithm without regularization behaves as if it is regularized. Note that implicit regularization is a property of a certain algorithm for solving a certain nonconvex problem. In the following theorem, we specify the conditions for the update (16) to be implicitly regularized for problem (14).

*Theorem 2 (**Implicit regularization**):* Consider (14) with $\lambda = 0$ and suppose that G^0 satisfies $\Pi_{D_-}G^0 = 0$. Then, the update (16) leads to $\Pi_{D_-}G^k = 0, k \in \{0,1,\dots\}$.

By Theorem 2, a sufficient condition for implicit regularization is

$$G^0 = D_-^\dagger \begin{bmatrix} K^0 \\ I_n \end{bmatrix},$$

provided with a stabilizing policy K^0. Theorem 2 also helps understand the optimization landscape of DeePO. Fig. 1 illustrates the relations among the nullspace $\mathcal{N}(D_-)$, the projected gradient, and an optimal solution G^*. Since $\Pi_{X_-}\nabla J(G)$ is orthogonal to $\mathcal{N}(D_-)$, the resulted policy of DeePO can be read as $G^\infty = \Pi_{D_-}G^0 + G^*$.

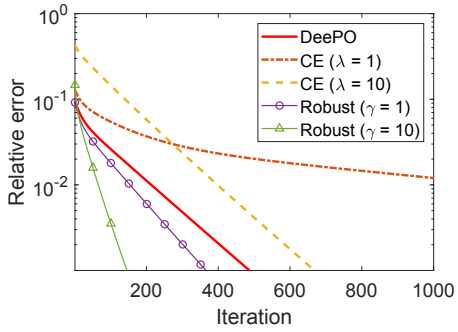

Fig. 2. Convergence of the DeePO methods.

V. Simulations

In this section, we perform simulations to validate the convergence of DeePO and the effects of regularization.

A. Numerical example

We randomly generate a dynamical model (A,B) with $n = 4, m = 2$ from a standard normal distribution and normalize A such that $\rho(A) = 0.8$, i.e., the open-loop system is stable. The resulting model parameters (A,B) are

$$A = \begin{bmatrix} -0.137 & 0.146 & -0.297 & 0.283 \\ 0.487 & 0.095 & 0.417 & 0.301 \\ -0.018 & 0.049 & 0.175 & 0.435 \\ 0.143 & 0.317 & -0.293 & -0.107 \end{bmatrix},$$

$$B = \begin{bmatrix} 1.639 & 0.930 \\ 0.264 & 1.793 \\ -1.464 & -1.183 \\ -0.776 & -0.111 \end{bmatrix}.$$

It is straightforward to check that (A,B) is controllable. Let $Q = I_4$ and $R = I_2$. We use Gaussian distribution to generate a batch of sufficiently exciting data (U_-,X_-) with $T = 10$ that satisfies (4), and compute X_+ by (3). In the sequel, we only use (U_-,X_-,X_+) to perform the DeePO methods and validate the convergence.

B. Convergence of the DeePO methods

We consider three algorithms, i.e, DeePO in (10), DeePO with the certainty-equivalence regularizer in (16) and with the robustness regularizer in (19). For all the three algorithms, we set the stepsize to $\eta = 2\times 10^{-3}$ for a fair comparison. For DeePO and DeePO with robustness regularizer, we set the initial policy as

$$G^0 = D_-^\dagger \begin{bmatrix} K^0 \\ I_4 \end{bmatrix} \in \mathcal{S}_G$$

with $K^0 = 0$ since the system is open-loop stable. For DeePO with certainty-equivalence regularizer, we set

$$G^0 = D_-^\dagger \begin{bmatrix} 0 \\ I_4 \end{bmatrix} + \Pi_{D_-}M \in \mathcal{S}_G,$$

where the elements of $M \in \mathbb{R}^{T\times n}$ are randomly sampled from a Gaussian distribution $\mathcal{N}(0,0.01)$ (otherwise due to the implicit regularization, there will be no difference in

the convergence curve compared with DeePO). To see how regularization parameters affect the performance, we select $\lambda = 1, 10$ for the certainty-equivalence regularizer and $\gamma = 1, 10$ for the robustness regularizer.

We illustrate the performance of the three algorithms in Fig. 2, where their relative errors are defined as $(J(G^k) - J^*)/J^*$, $(J_\lambda(G^k) - J_\lambda^*)/J_\lambda^*$, and $(J_\gamma(G^k) - J_\gamma^*)/J_\gamma^*$, respectively. While Theorem 1 only shows a more conservative sublinear convergence rate, all the three algorithms converge linearly in the simulation. The DeePO algorithm with certainty-equivalence regularizer (denoted by CE in Fig. 2) has the slowest convergence. The case for $\lambda = 10$ converges faster than the case $\lambda = 1$ due to the faster decay of the regularizer $\lambda\|\Pi_{D_-} G\Sigma_G^{1/2}\|^2$, and it achieves the same rate as the unregularized DeePO algorithm. Under the robustness regularizer, the DeePO algorithm has the fastest convergence, and $\gamma = 10$ leads to a larger convergence rate. Nevertheless, the resulted policy is different from those of the other two algorithms as discussed in Section IV-B. Finally, we note that all the algorithms only use 10 pairs of state-input data to achieve an arbitrary relative error. In sharp contrast, the zeroth-order optimization method in [6] uses 10^5 trajectories (of manually tuned length to approximate the cost well) to achieve 0.01 relative error for an LTI system with $m = n = 3$.

VI. Conclusion

In this paper, we have proposed the DeePO method that only requires a finite number of PE data to solve the LQR problem. By relating the nonconvex optimization problem to a convex program, we have shown the global convergence of DeePO. Furthermore, we have shown that the regularization method can be applied to enhance certainty-equivalence and robust stability without affecting its convergence.

In future, it would be valuable to discover a strongly convex reparameterization of (7), which may improve the sublinear convergence rate to linear. It would also be interesting to study DeePO in a more general setting, e.g., the LQR with noisy inputs. Since DeePO is an efficient iterative method, it is expected to be able to applied to online control, where the control performance is constantly improved by collecting more real-time data. We are also hopeful that it can be used to solve the adaptive LQR for time-varying systems.

References

[1] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[2] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," in *International Conference on Learning Representations*, 2016.

[3] B. Recht, "A tour of reinforcement learning: The view from continuous control," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 2, pp. 253–279, 2019.

[4] 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*, 2018, pp. 1467–1476.

[5] 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, 2022.

[6] 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 *22nd International Conference on Artificial Intelligence and Statistics*, 2019, pp. 2916–2925.

[7] F. Zhao, K. You, and T. Başar, "Global convergence of policy gradient primal-dual methods for risk-constrained LQRs," *IEEE Transactions on Automatic Control*, 2023, to appear, available at https://ieeexplore.ieee.org/document/10005813.

[8] 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*, 2023, to appear, available at https://arxiv.org/abs/2210.04810.

[9] S. Tu and B. Recht, "The gap between model-based and model-free methods on the linear quadratic regulator: An asymptotic viewpoint," in *Conference on Learning Theory*, 2019, pp. 3036–3083.

[10] M. Simchowitz and D. Foster, "Naive exploration is optimal for online LQR," in *International Conference on Machine Learning*. PMLR, 2020, pp. 8937–8948.

[11] J. C. Willems, P. Rapisarda, I. Markovsky, and B. L. De Moor, "A note on persistency of excitation," *Systems & Control Letters*, vol. 54, no. 4, pp. 325–329, 2005.

[12] J. Coulson, J. Lygeros, and F. Dörfler, "Data-enabled predictive control: In the shallows of the DeePC," in *18th European Control Conference (ECC)*, 2019, pp. 307–312.

[13] I. Markovsky and F. Dörfler, "Behavioral systems theory in data-driven analysis, signal processing, and control," *Annual Reviews in Control*, vol. 52, pp. 42–64, 2021.

[14] C. De Persis and P. Tesi, "Formulas for data-driven control: Stabilization, optimality, and robustness," *IEEE Transactions on Automatic Control*, vol. 65, no. 3, pp. 909–924, 2019.

[15] H. J. Van Waarde, J. Eising, H. L. Trentelman, and M. K. Camlibel, "Data informativity: a new perspective on data-driven analysis and control," *IEEE Transactions on Automatic Control*, vol. 65, no. 11, pp. 4753–4768, 2020.

[16] J. Berberich, J. Köhler, M. A. Müller, and F. Allgöwer, "Data-driven model predictive control with stability and robustness guarantees," *IEEE Transactions on Automatic Control*, vol. 66, no. 4, pp. 1702–1717, 2020.

[17] S. Kang and K. You, "Minimum input design for direct data-driven property identification of unknown linear systems," *arXiv preprint arXiv:2208.13454*, 2022.

[18] C. De Persis and P. Tesi, "Low-complexity learning of linear quadratic regulators from noisy data," *Automatica*, vol. 128, p. 109548, 2021.

[19] F. Dörfler, P. Tesi, and C. De Persis, "On the certainty-equivalence approach to direct data-driven LQR design," *IEEE Transactions on Automatic Control*, 2023, to appear, available at https://ieeexplore.ieee.org/document/10061542.

[20] ——, "On the role of regularization in direct data-driven LQR control," in *61st IEEE Conference on Decision and Control (CDC)*, 2022, pp. 1091–1098.

[21] F. Dörfler, J. Coulson, and I. Markovsky, "Bridging direct and indirect data-driven control formulations via regularizations and relaxations," *IEEE Transactions on Automatic Control*, vol. 68, no. 2, pp. 883–897, 2023.

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

[23] S. Arora, N. Cohen, W. Hu, and Y. Luo, "Implicit regularization in deep matrix factorization," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[24] 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.

[25] D. Bertsekas, *Dynamic programming and optimal control*. Athena Scientific, Massachusetts, 2012, vol. 1.

[26] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge University Press, Cambridge, England, 2012.