Convergence Guarantees of Model-free Policy Gradient Methods for LQR with Stochastic Data

Bowen Song
Institute for Systems Theory and Automatic Control
University of Stuttgart, Germany
bowen.song@ist.uni-stuttgart.de

Andrea Iannelli
Institute for Systems Theory and Automatic Control
University of Stuttgart, Germany
andrea.iannelli@ist.uni-stuttgart.de

Abstract

Policy gradient (PG) methods are the backbone of many reinforcement learning algorithms due to their good performance in policy optimization problems. As a gradient-based approach, PG methods typically rely on knowledge of the system dynamics. However, this information is not always available, and in such cases, trajectory data can be utilized to approximate first-order information. When the data are noisy, gradient estimates become inaccurate and a formal investigation that encompasses uncertainty estimation and the analysis of its propagation through the algorithm is currently missing. To address this, our work focuses on the Linear Quadratic Regulator (LQR) problem for systems subject to additive stochastic noise. After briefly summarizing the state of the art for cases with a known model, we focus on scenarios where the system dynamics are unknown, and approximate gradient information is obtained using zeroth-order optimization techniques. We analyze the theoretical properties by computing the error in the estimated gradient and examining how this error affects the convergence of PG algorithms. Additionally, we provide global convergence guarantees for various versions of PG methods, including those employing adaptive step sizes and variance reduction techniques, which help increase the convergence rate and reduce sample complexity. One contribution of this work is the study of the robustness of model-free PG methods, aiming to identify their limitations in the presence of noise and propose improvements to enhance their applicability. Numerical simulations show that these theoretical analyses provide valuable guidance in tuning the algorithm parameters, thereby making these methods more reliable in practically relevant scenarios.

1 Introduction

Reinforcement learning [3, 4, 35] has demonstrated a significant influence across a wide range of applications. A key concept within reinforcement learning is policy optimization, where the policy is parameterized and directly optimized over based on a predefined performance metric [16]. Several successful policy optimization methods have been developed, including policy gradient [36], actor-critic [19], proximal policy optimization [29]. This work focuses on policy gradient (PG) methods, which are based on the simple idea of minimizing a cost function over the parameterized policy by improving performance through a gradient descent-type update. Studying the convergence

18th European Workshop on Reinforcement Learning (EWRL 2025).

properties of PG methods, particularly their global convergence to the optimal policy, is an active area of research [6, 10, 11, 41, 42, 45, 46, 15]. For instance, PG methods have been applied in [41] to solve $\mathcal{H}_2$ cost function subject to $\mathcal{H}_\infty$ constraints, and they have been applied in [42] to the Markov decision processes.

The application of reinforcement learning to the linear quadratic regulator (LQR) problem has been extensively explored due to its value as an analytically tractable benchmark, making it ideal for studying reinforcement learning in environments with continuous state and action spaces [2, 8, 10, 20, 24, 28, 38, 14]. Policy gradient methods have gained attention in recent studies focusing on the LQR problem. Interest in applying these methods to the LQR setting increased significantly after the work in [10] demonstrated global convergence properties of policy gradient methods applied to the deterministic LQR problem. In [46], the performance limitations of PG applied to LQR problem are studied from a control-theoretic perspective. The work [45] explores a primal-dual policy gradient method to solve the constrained LQR problem.

Solving the LQR problem requires the availability of a model of the system, as it is the case in [41, 42, 45, 46], where the analysis assumes knowledge of the exact model. However, in real-world applications, a complete system description is often unavailable, making it necessary to combine policy optimization schemes with data-driven techniques. To address this challenge, various approaches have been developed. For instance, methods like those in [1, 5, 31] combine system identification with model-based LQR design and use regret as a performance metric of the learning process. It is also possible to use the same two-step approach to solve the LQR problem with gradient-based schemes. The work [33] integrates recursive least squares with policy iteration optimization, while [30] combines recursive least squares with the policy gradient methods. Both works assume noise-free data, whereas [34, 43] extend these frameworks to bounded-noise settings. These approaches are termed indirect data-driven methods, because first data is used to estimate a model, which is then integrated into a model-based certainty equivalence design. An alternative category is direct data-driven control, which directly uses data to design the controller, bypassing the intermediate model estimation process. For instance, a model-free policy iteration algorithm is proposed in [39]. Several model-free policy gradient methods have also been introduced and theoretically analyzed. In [10], model-free PG methods are proposed and formulated using zeroth-order optimization to estimate gradients for the LQR cost from finite but noise-free system trajectories. Similarly, in [37], PG methods are applied to a continuous-time output feedback system using zeroth-order gradient estimation alongside variance reduction techniques. The work [44] focuses on stabilizing the unknown system using policy gradient methods. In the aforementioned direct data-driven works [10, 37, 44], only the case where the system's trajectories are noise-free is studied. However, it is essential to analyze what happens when gradients and other objects involved in the policy update are estimated using noisy data. In [11], Model-free PG methods are applied to the linear system with multiplicative noise, addressing uncertainty in the system matrix. The works [37, 44, 11] all build on [10], leveraging the inherent robustness of PG methods: informally, when the gradient estimation error is sufficiently small, one can still guarantee a decrease in the cost function. Another line of research tackles the model-free LQR problem differently. The work [25] derives convergence properties by carefully selecting step sizes when facing uncertainty of the gradient. Follow-up works [27, 22] refine the analysis by incorporating alternative gradient estimation techniques, yielding improved sample efficiency and robustness guarantees. In those works, the noise is stochastic but bounded. These results assume stochastic but bounded noise. There are also studies considering stochastic unbounded noise. For example, [13] applies PG methods to finite-horizon LQR, while [17, 40] propose model-free first-order methods, which estimate individual components of the gradient and differ from the aforementioned zeroth-order approaches. To date, there is no comprehensive analysis that jointly addresses the uncertainty introduced by the estimation process using the zeroth-order method and its propagation through the algorithm, and provides probabilistic convergence guarantees and sample complexity bounds for the infinite-horizon LQR problem.

In this work, we analyze the sample complexity and robustness to noise of PG methods for the model-free LQR with additive stochastic noise. Due to the presence of noise with unbounded support, we consider an average infinite-horizon cost. We first characterize the properties of this cost function and analyze model-based versions of a few PG algorithms. For the gradient descent (GD) and natural policy gradient (NPG) methods, we propose an adaptive stepsize scheme that has a provable faster convergence rate compared to the fixed step size methods discussed in [10, 11, 37]. Then, for both the model-free gradient descent and natural policy gradient methods, we employ zeroth-order optimization to approximate the gradient and covariance from noisy data. To handle the challenge of unbounded gradient estimates due to unbounded noise, we introduce an event-based analysis,

extending the approach of [13]. We provide convergence guarantees for these model-free PG methods with a sample complexity that takes into account the noisy source of estimation error. Furthermore, we introduce and analyze a variance reduction technique within the zeroth-order optimization framework, and prove that this achieves a guaranteed improvement in the sample complexity. Even though variance reduction techniques for model-free LQR were already presented in [37], the contribution here represents a significant extension as we work in the noisy scenario and we are able to provide a provable improvement in the number of required samples. Finally, we provide a comparative analysis of model-free PG methods with and without noisy data, to demonstrate the importance of the analysis considering noise, offer guidance of parameter tuning in applying policy gradient methods to applications, and also illustrate the improvement of introducing adaptive step sizes and variance reduction. Our analysis shows that model-free PG with stochastic data enjoys qualitatively similar convergence guarantees to those in the noise-free scenario, but it points out limitations in terms of the number of samples necessaries, stepsize ranges, and convergence rate that are caused by noise.

The paper is organized as follows. Section 2 introduces the problem setting and provides some preliminaries. Section 3 studies convergence properties of a few representative PG methods for the average cost case and proposes their extension to adaptive stepsizes. Section 4 investigates convergence and sample complexity of model-free versions of gradient descent and natural policy gradient methods by placing particular emphasis on the effect of noisy data on the algorithms, together with adaptive step sizes and variance reduction techniques. Section 5 serves as a concluding summary of the work. Numerical results are provided to illustrate the main findings of this work by examining the behavior of the analyzed policy gradient methods under varying noise levels and algorithmic configurations. Due to page limitations, the detailed results are presented in the appendix D.

Notations

We denote by $A \succeq 0$ and $A \succ 0$ a positive semidefinite and positive definite matrix A, respectively. The symbols $\lambda_1(A)$ denote the smallest eigenvalue of the matrix A. For matrices, $\|\cdot\|_F$, $\|\cdot\|$ denote the Frobenius norm and induced 2-norm, respectively. For vectors, $\|\cdot\|$ denotes the Euclidean norm. I represents the identity matrix. $\mathcal{N}(0, \Sigma)$ denotes a Gaussian distribution with 0 mean and covariance $\Sigma \succ 0$.

2 Preliminaries

In this work, we consider the following averaged infinite horizon optimal control problem, where the plant is subject to additive stochastic noise:

$$\min_{u_t} \lim_{T\to+\infty} \frac{1}{T} \mathop{\mathbb{E}}_{x_0,w_t} \sum_{t=0}^{T-1} \left(x_t^\top Q x_t + u_t^\top R u_t \right), \tag{1a}$$

$$\text{s. t. } x_{t+1} = Ax_t + Bu_t + w_t, x_0 \sim \mathcal{D}, \tag{1b}$$

where $x_t \in \mathbb{R}^{n_x}$ is the system state and $u_t \in \mathbb{R}^{n_u}$ is the system input; $A \in \mathbb{R}^{n_x \times n_x}$, $B \in \mathbb{R}^{n_x \times n_u}$, (A, B) is unknown but stabilizable; $w_t \sim \mathcal{N}(0, \Sigma_w)$; $Q, R \succ 0$ are the weighting matrix. We define the state covariance at the initial time as $\Sigma_0 := \mathop{\mathbb{E}}_{x_0} \left[x_0 x_0^\top \right]$ and then $\mathcal{D} := \mathcal{N}(0, \Sigma_0)$.

The input u_t is parameterized as a linear state feedback control with gain K, i.e. $u_t = Kx_t$. The optimal control cost only depends on K and is denoted by C:

$$C(K) := \lim_{T\to+\infty} \frac{1}{T} \mathop{\mathbb{E}}_{x_0,w_t} \sum_{t=0}^{T-1} x_t^\top \underbrace{\left(Q + K^\top R K \right)}_{=:Q_K} x_t. \tag{2}$$

We introduce basic properties of problem (1) that are particularly relevant when policy gradient methods are used to solve it. While these results are mostly well-known, they form the foundation for the main results of this work in Sections 3 and 4. The state response x_t for $t \geq 1$ associated with any gain K is given by:

$$x_t = (A_K)^t x_0 + \sum_{k=0}^{t-1} (A_K)^{t-k} w_{k-1}. \tag{3}$$

We define the set $\mathcal{S}$ as the set of matrices $K \in \mathbb{R}^{n_x \times n_u}$ that stabilizes the system (A, B), meaning that the matrix $A_K := A + BK$ is Schur stable:

$$\mathcal{S} := \left\{ K \in \mathbb{R}^{n_x \times n_u} | \|A_K\| < 1 \right\}. \tag{4}$$

Now, we introduce some important definitions associated with $K \in \mathcal{S}$. The covariance matrix at time t is defined as $\Sigma_t := \mathop{\mathbb{E}}_{x_0, w_t}[x_t x_t^\top]$. From this definition and the state response given in (3), we obtain:

$$\Sigma_{t+1} = A_K \Sigma_t A_K^\top + \Sigma_w, \quad t \in \mathbb{Z}_+. \tag{5}$$

The average covariance matrix associated with $K \in \mathcal{S}$ is defined as:

$$\Sigma_K := \lim_{T \to +\infty} \frac{1}{T} \sum_{t=0}^{T-1} \Sigma_t. \tag{6}$$

The cost function $C(K)$ associated with $K \in \mathcal{S}$ can be equivalently expressed as:

$$C(K) = \mathrm{Tr}\left(P_K \Sigma_w\right) = \mathrm{Tr}\left(Q_K \Sigma_K\right), \tag{7}$$

where P_K is defined as the solution of the following Lyapunov equation:

$$P_K = Q_K + A_K^\top P_K A_K. \tag{8}$$

From the expression of the cost function (7) and well-known results [21], the optimal K^*, which minimizes the cost function C, is given by:

$$K^* = -(R + B^\top P_{K^*} B)^{-1} B^\top P_{K^*} A, \tag{9a}$$

$$P_{K^*} = Q + A^\top P_{K^*} A - A^\top P_{K^*} B (R + B^\top P_{K^*} B)^{-1} B^\top P_{K^*} A. \tag{9b}$$

The average covariance matrix associated with the optimal K^* is denoted as Σ_{K^*}. Using (7), the gradient of $C(K)$ with $K \in \mathcal{S}$ can be expressed as follows [46]:

$$\nabla C(K) = 2 E_K \Sigma_K, \tag{10}$$

where $E_K := \left(R + B^\top P_K B\right) K + B^\top P_K A$.

It is known that the cost function C defined in (2) is generally non-convex. In the context of non-convex optimization, the convergence to the optimal solution of gradient descent cannot usually be guaranteed. However, the cost function $C(K)$ satisfies a special condition known as *gradient domination*.

Lemma 1 (Gradient Domination) *The function C on the set $\mathcal{S}$ is gradient dominated. That is, for any $K \in \mathcal{S}$, the following inequality holds:*

$$C(K) - C(K^*) \leq \mu \|\nabla C(K)\|_F^2, \tag{11}$$

with $\mu := \frac{1}{4} \|\Sigma_{K^*}\| \|\Sigma_w^{-2}\| \|R^{-1}\|$.

From (11) and the fact that $\Sigma_w \succ 0$, it follows that the cost function C has a unique minimizer and no other stationary points. The proof of Lemma 1 is provided in [32, Appendix A.1], and is a straightforward extension of [10, Lemma 3] to the averaged infinite horizon setting considered here.

Lemma 2 (Almost Smoothness on $\mathcal{S}$) *For any $K, K' \in \mathbb{S}$, the following inequality holds:*

$$\left| C(K') - C(K) - 2\mathrm{Tr}\left((K' - K)^\top E_K \Sigma_{K'} \right) \right| \leq \|\Sigma_{K'}\| \left\| R + B^\top P_K B \right\| \|K' - K\|_F^2, \tag{12}$$

The proof of Lemma 2 is provided in [32, Appendix A.2], which is an extension of [10, Lemma 6]. To understand why (12) is referred to as almost smoothness, assume additionally that: we can approximate Σ_K with $\Sigma_{K'}$, i.e. $\Sigma_K \approx \Sigma_{K'}$, and that we can upper bound the term $\|\Sigma_K\| \left\| R + B^\top P_K B \right\|$ by L. Then, recalling (12), we see that (10) is equivalent to:

$$\left| C(K') - C(K) - \mathrm{Tr}\left((K' - K)^\top \nabla C(K) \right) \right| \leq L \|K' - K\|_F^2, \quad \forall K,\ K' \in \mathcal{S}.$$

which is the classic descent lemma for a smooth matrix function C. This approximation allows us to interpret the cost function as nearly smooth when the aforementioned assumptions are satisfied. Since we aim to use (12) to develop a gradient descent algorithm, it is crucial to quantify the relationship between Σ_K and $\Sigma_{K'}$ and find the upper bound L, which is discussed in Appendix A.

3 Convergence of Model-based Policy Gradient

In this section, we show the global convergence properties of various model-based policy gradient methods, that is, algorithms that operate with perfect knowledge of the system matrices (A, B). We first present the theorem guaranteeing the convergence of the model-based gradient descent method, based on the properties introduced in (12).

Theorem 1 (Gradient Descent with Adaptive Step Size) *Suppose the initial $K_0 \in \mathcal{S}$, and consider the gradient descent iteration*

$$K_{i+1} = K_i - \eta_i \nabla C(K_i), \quad \forall i \in \mathbb{Z}_+, \tag{13}$$

where the step size satisfies $\eta_i \leq h_{\mathrm{GD}}(C(K_i))$, and $h_{\mathrm{GD}}(C(K_i))$ is defined in (36) *in the proof. Then, the following relationship holds:*

$$C(K_{i+1}) - C(K^*) \leq \left(1 - \frac{2\eta_i \lambda_1(R)\lambda_1^2(\Sigma_w)}{\|\Sigma_{K^*}\|}\right)(C(K_i) - C(K^*)). \tag{14}$$

For $\eta_i = h_{\mathrm{GD}}(C(K_i)), i \in \mathbb{Z}_+$ and given any accuracy gap $\epsilon > 0$, if the number of iterations N satisfies:

$$N \geq \frac{\|\Sigma_{K^*}\|}{2\eta_0 \lambda_1(R)\lambda_1^2(\Sigma_w)} \log \frac{C(K_0) - C(K^*)}{\epsilon},$$

then $C(K_N) - C(K^*) \leq \epsilon$.

The proof of Theorem 1 can be found in Appendix B.1. From Theorem 1, we see that the gradient descent method can achieve any desired accuracy ϵ within a finite number of iterations N and converges linearly. According to Theorem 1, the step size is adaptive according to the cost $C(K_i)$ at the current iterate K_i. A similar analysis can be conducted for the other two policy gradient methods: the natural policy gradient method: $K_{i+1} = K_i - \eta \nabla C(K_i)\Sigma_{K_i}^{-1}$ and the Gauss-Newton method: $K_{i+1} = K_i - \eta(R + B^\top P_{K_i} B)^{-1}\nabla C(K_i)\Sigma_{K_i}^{-1}$. Theoretical results supporting this analysis are provided in Theorem 5 and Theorem 6 in Appendix B.

Lack of robustness of model-based PG methods

The guarantees on PG methods reviewed in the previous sections rely on the availability of an exact system model. When this is not available, gradients and covariances are estimated from data trajectories, and for plants such as (1b), they will be subject to stochastic errors. We exemplify with a simple toy example the performance of a policy GD method when noisy gradients are used. We consider a linear time-invariant (LTI) system described by the following dynamics [8, 39]:

$$x_{t+1} = \underbrace{\begin{bmatrix} 1.01 & 0.01 & 0 \\ 0.01 & 1.01 & 0.01 \\ 0 & 0.01 & 1.01 \end{bmatrix}}_{A} x_t + \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{B} u_t + w_t. \tag{15}$$

The weighting matrices Q and R are set to

$$Q = 0.001 I_3, \quad R = I_3. \tag{16}$$

The initial condition $\hat{K}_0$ is chosen as the optimal gain for the LQR problem with $(A, B, 50Q, R)$. To model gradient estimation errors due to noisy data, we define the gradient estimate as follows:

$$\hat{\nabla} C(K) := \nabla C(K) + \Delta, \tag{17}$$

where $\Delta \in \mathbb{R}^{3\times 3}$ is a matrix with entries Δ_{ij} sampled according to the distribution $\mathcal{N}(0, \sigma^2)$, where the values of $\sigma \geq 0$ will be discussed later. The gradient descent algorithm updates the gain $\hat{K}_i$ as follows:

$$\hat{K}_{i+1} = \hat{K}_i - \eta \hat{\nabla} C(\hat{K}_i), \quad \forall i \in \mathbb{Z}_+. \tag{18}$$

The simulation results are illustrated in Figure 1, where the y-axis represents the average obtained from a Monte Carlo simulation over 10,000 data samples. When the gradient is exactly known (i.e. $\sigma = 0$, black dotted line), we can use theoretical bounds on the step size to guarantee convergence

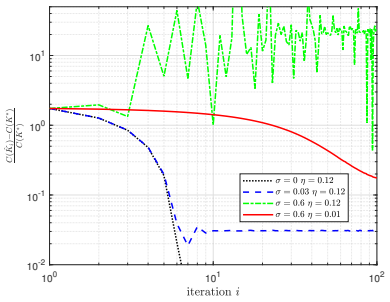

Figure 1: Performance of Gradient Descent with Inexact Noisy Gradient for Different Values of Step Size and Noise

of the gradient descent algorithm. In this case, the converging behavior of $\eta = 0.12$ to the optimal policy is shown as an example. When the gradient estimation is affected by a small amount of noise ($\sigma = 0.03$, blue dashed line), the same step size yields convergence but to a suboptimal value of gain K. As the noise level increases ($\sigma = 0.6$, green dash-dotted line), the algorithm starts showing non-converging behavior. In this case, reducing the step size addresses the issue, as shown by the red solid line where the step size is empirically decreased to 0.03. It is worth observing that this reduction in step size results in a slower convergence rate. These results exemplify that when PG algorithms are run using finite noisy data trajectories to compute gradients (or equivalently covariances), the convergence guarantees obtained in the model-based scenario no longer hold. Specifically, the algorithm might converge to suboptimal solutions or even diverge. It is also shown that, in the latter case, by properly modifying the algorithms' parameters (such as the step size), converging behaviors can be achieved.

To characterize the fundamental limitations of such algorithms and improve their performance it is then paramount to: quantify the uncertainty on gradients and covariances due to the use of finite noisy data; and analyze how this uncertainty propagates in the policy optimization algorithm. To the best of the authors' knowledge, there is no holistic analysis that captures all these aspects and provides guarantees for model-free PG algorithms in the presence of noisy trajectory data. The main technical contribution of this work is to bridge this gap by providing: probabilistic guarantees of convergence and suboptimality bounds; closed-form expressions for the algorithm's parameters (e.g. step size, variance reduction) that guarantee converging behavior.

4 Convergence of Model-free Policy Gradient

In this section, we analyze the model-free policy gradient methods. Unlike model-based approaches, the system dynamics (A, B) are unknown in the model-free setting. Instead of relying on system identification, we use zeroth-order optimization, where gradients and covariance estimates are estimated directly from trajectory data. In Section 4.1, we begin by analyzing estimation algorithm for the gradient utilized in the proposed method. We derive novel finite-sample error bounds that explicitly account for the effects of noisy data. Building on this quantitative bound, Section 4.2 introduces a variance reduction technique to provably reduce the required number of rollouts. Leveraging these results, we establish probabilistic convergence guarantees for gradient descent (Section 4.3), emphasizing the more general scenario with time-adaptive step sizes. A similar analysis can also be carried out for the natural policy gradient method, where the additional term Σ_k must be estimated from data. Error quantification and convergence guarantees can likewise be established. For further details, we refer the reader to Appendix C.4. The distinction between the noisy and noise-free analyses is also discussed in Appendix C.5, which highlights the impact of accounting for noisy data.

4.1 Model-free Gradient and Covariance Estimation

In the model-free setting, zeroth-order optimization techniques [7, 23] can be utilized to approximate the gradient and average covariance matrix using only function evaluations. The procedures for

gradient and covariance estimation are outlined in Algorithm 1. We present two theorems to analyze the estimation errors of both $\hat{\nabla}C(K)$ and $\hat{\Sigma}_K$, obtained from Algorithm 1.

Algorithm 1 Gradient and covariance estimation

Require: Gain matrix $K \in \mathcal{S}$, number of rollouts n, rollout length l, exploration radius r, an upper bound of the initial state L_0

for $k = 1, ..., n$ **do**

1. Generate a sample gain matrix $\hat{K}_k = K + U_k$, where U_k is drawn uniformly at random over matrices of compatible dimensions with Frobenius norm r;

2. Generate a sample initial state $x_0^{(k)}$ with $\|x_0^{(k)}\| \leq L_0$;

3. Excite the closed-loop system with $u_t^{(k)} = \hat{K}_k x_t^{(k)}$ for l-steps starting from $x_0^{(k)}$, yielding the state sequence $\left\{x_t^{(k)}\right\}_{t=0}^{l-1}$ originating from (1b);

4. Collect the empirical cost estimate $\hat{C}_k := \frac{1}{l}\sum_{t=0}^{l-1} x_t^{(k)\top}(Q + \hat{K}_k^\top R \hat{K}_k) x_t^{(k)}$ and the empirical covariance matrix $\hat{\Sigma}_i = \frac{1}{l}\sum_{t=0}^{l-1} x_t^{(k)} x_t^{(k)\top}$;

end for

return Gradient estimate $\hat{\nabla}C(K) := \frac{1}{n}\sum_{k=1}^{n} \frac{n_x n_u}{r^2}\hat{C}_k U_k$ and covariance estimate $\hat{\Sigma}_K := \frac{1}{n}\sum_{k=1}^{n}\hat{\Sigma}_k$.

Theorem 2 (Error Bound of $\hat{\nabla}C(K)$) *Given an arbitrary tolerance $\epsilon > 0$, which can be expressed as $\epsilon = \epsilon_d + \epsilon_l + \epsilon_n + \epsilon_r$, and an arbitrary probability $\delta \in (0, 1)$, which can be expressed as $\delta = 1 - (1 - \delta_d)(1 - \delta_n)(1 - \delta_x)$ with $\delta_x, \delta_n, \delta_d \in (0, 1)$, for a given $K \in \mathcal{S}$, the estimated gradient $\hat{\nabla}C(K)$ from Algorithm 1 enjoys the following bound:*

$$\mathbb{P}\left\{\|\hat{\nabla}C(K) - \nabla C(K)\| \leq \epsilon\right\} \geq 1 - \delta, \tag{19}$$

if the parameters r, l, n in Algorithm 1 satisfy:

$$r \leq r_{\max}(C(K), \epsilon_r), \quad l \geq l_{\min}(C(K), \epsilon_l), \tag{20a}$$

$$n \geq \max\big(N_1(C(K), \epsilon_r, \epsilon_l, \epsilon_n, \delta_n, \delta_d), N_2(C(K), \epsilon_r, \epsilon_l, \epsilon_n, \epsilon_d, \delta_x, \delta_n, \delta_d)\big). \tag{20b}$$

where the detailed expressions of functions $r_{\max}$, $l_{\min}$, N_1 and N_2 are given in (47)*,* (48)*,* (49a)*, and* (49b) *in Appendix C.6, respectively.*

The proof of Theorem 2 is given in [32, Appendix C.5]. Similarly, the error bound for the estimation of $\hat{\Sigma}_K$ is established in Theorem 7 in Appendix C.1.

4.2 Model-free Gradient Estimation with Variance Reduction

In the previous section, we employed zeroth-order optimization to estimate the gradient of the cost at the current value of the policy. However, this approach often suffers from high variance, resulting in a slow learning process [12]. Using a baseline is a common variance reduction technique in policy gradient methods [9, 26]. In this section, we propose employing the finite-horizon cost function as a baseline and show its performance improvement. For a state-dependent baseline function $b(x)$, the estimated gradient with variance reduction $\hat{\nabla}C_v(K)$ is expressed as

$$\hat{\nabla}C_v(K) := \frac{1}{n}\sum_{k=1}^{n} \frac{n_x n_u}{r^2}(\hat{C}_k - b(x_0^{(k)}))U_k, \tag{21}$$

where $U_k, x_0^{(k)}, \hat{C}_k, r$ are the same as those defined in Algorithm 1. We select the baseline function $b_s(x_0)$ as

$$b_s(x_0) = \frac{1}{l}\sum_{t=0}^{l-1} \mathbb{E}_{w_t}\left[x_t^\top (Q + K^\top R K) x_t\right], \tag{22}$$

which can be estimated in the model-free setting using trajectory data. The detailed procedure for computing this baseline, as well as the rationale behind its choice, is presented in [32, Section 4.2].

Algorithm 2 Baseline function estimation $b_s(x_0)$ for variance reduction of $\hat{\nabla}C(K)$

Require: Gain matrix $K \in \mathcal{S}$, number of rollouts to estimate the baseline function n_v, rollout length l, initial state x_0

for $k = 1, ..., n_v$ **do**

1. Excite the closed-loop system with $u_t = Kx_t$ for l-steps starting from x_0, yielding the state sequence $\left\{x_t^{(k)}\right\}_{t=0}^{l-1}$;

2. Collect the empirical finite-horizon cost estimate $\hat{C}_k^V := \frac{1}{l}\sum_{t=0}^{l-1} x_t^{(k)\top}(Q + K_i^\top RK)x_t^{(k)}$;

end for

return Baseline function estimate $\hat{b}_s(x_0) := \frac{1}{n_v}\sum_{k=1}^{n_v} \hat{C}_k^V$.

Using the estimated baseline function $\hat{b}_s$, the gradient estimation algorithm can be summarized in Algorithm 3 given in Appendix C.2. Compared with Algorithm 1, Algorithm 3 introduces an additional step involving the baseline function to reduce the variance of the estimated gradient. The following theorem shows that this modification of the algorithm brings a provable improvement in the sample complexity required to estimate the gradient with a given accuracy:

Theorem 3 (Performance Improvement) *Given the same tolerance $\epsilon = \epsilon_d + \epsilon_l + \epsilon_n + \epsilon_r$ and probability $\delta = 1 - (1-\delta_d)(1-\delta_n)(1-\delta_x)$ introduced in Theorem 2, for a given $K \in \mathcal{S}$, then the estimated gradient $\hat{\nabla}C(K)$ from Algorithm 3 enjoys the following bound:*

$$\mathbb{P}\left\{\|\hat{\nabla}C(K) - \nabla C(K)\| \leq \epsilon\right\} \geq 1 - \delta, \tag{23}$$

if n_b, l and r in Algorithm 3 satisfy:

$$r \leq r_{\max}(C(K), \epsilon_r), l \geq l_{\min}(C(K), \epsilon_l), \tag{24a}$$

$$n_b \geq \max(N_1, N_3(\hat{b}_s(x_0^{(k)})) \tag{24b}$$

where the detailed expression of N_3 is given in (55) *in Appendix C.6 and $r_{\max}$, $l_{\min}$ and N_1 were introduced in Theorem 2. Moreover, let the desired probability $\delta_v \in (0,1)$ be expressed as $\delta_v = 1 - (1-\tilde{\delta}_v)(1-\tilde{\delta}_x)$. If the number of rollouts n_v to estimate the baseline function (as defined in Algorithm 2) satisfies*

$$n_v \geq \tilde{n}_{\min}(C(K), l, \tilde{\delta}_v, \tilde{\delta}_x), \tag{25}$$

then:

$$\mathbb{P}\{N_2 \geq N_3\} \geq 1 - \delta_v, \tag{26}$$

where N_2 was first introduced in (20b) *in Theorem 2 and the detailed expression of $\tilde{n}_{\min}$ is provided in* (56) *in Appendix C.6.*

The proof of Theorem 3 is given in [32, Appendix C.6]. Based on Theorem 3, we can compare the performance of Algorithm 1 without variance reduction and Algorithm 3 with variance reduction. The exploration radius $r_{\max}$ and rollout length $l_{\min}$ (24a) remain the same for both algorithms. The number of rollouts (24b) is determined by the maximum of two arguments. N_1 remains unchanged across the two algorithms, as it depends on the smoothing function and the original function. From the second part of Theorem 3 (Equation (26)), it then follows that the required number of rollouts is at least non-increasing when the baseline function estimates are sufficiently accurate. δ_x in Theorem (2) represents the probability that the data samples remain bounded. Increasing δ_x leads to a larger bound for the data samples. This upper bound is utilized in the matrix concentration inequality to determine the required number of rollouts. From the expressions of N_1, N_2 (see (49a) and (49b)), we observe that, for large Σ_w and thus large noise, the required number of rollouts is determined by N_2 (in Algorithm 1) and N_3 (in Algorithm 3). In other words, the proposed variance reduction gives a provable reduction of the number of rollouts (and thus of samples) for scenarios with large noise, which is intuitive and expected.

4.3 Gradient Descent with Adaptive Step Size

Building on the gradient uncertainty quantified in the previous section, we now proceed to study the convergence of the model-free gradient descent algorithm.

Theorem 4 (Model-free Gradient Descent with Adaptive Step Size) *Suppose the initial $\hat{K}_0 \in \mathcal{S}$, and consider the gradient descent with adaptive step size:*

$$\hat{K}_{i+1} = \hat{K}_i - \eta_i \hat{\nabla} C(\hat{K}_i), \quad \forall i \in \mathbb{Z}_+, \tag{27}$$

where $\hat{\nabla} C(\hat{K}_i)$ is the gradient estimate from Algorithm 1 and $0 < \eta_i \leq h_{\mathrm{GD}}(C(\hat{K}_i))$, where the function h_{GD} was introduced in Theorem 1. Given any accuracy $\epsilon > 0$, and $\sigma \in (0,1)$, define $\eta_{\mathrm{GD}} := \inf_i \eta_i$ and the number of iterations n_{GD}:

$$n_{\mathrm{GD}} = \frac{\|\Sigma_{K^*}\|}{2(1-\sigma)\eta_{\mathrm{GD}}\lambda_1(R)\lambda_1^2(\Sigma_w)} \log \frac{C(\hat{K}_0) - C(K^*)}{\epsilon}.$$

Given any probability $\delta \in (0,1)$ satisfying $\delta n_{\mathrm{GD}} \in (0,1)$, if the estimation error of the gradient $\hat{\nabla} C(\hat{K}_i)$ satisfies:

$$\mathbb{P}\left\{ \|\hat{\nabla} C(\hat{K}_i) - \nabla C(\hat{K}_i)\| \leq \frac{\sigma\epsilon\lambda_1(R)\lambda_1^2(\Sigma_w)}{h_C(C(\hat{K}_i))\|\Sigma_{K^*}\|} \right\} \geq 1 - \delta,$$

where h_C was introduced in Lemma 4. This can be ensured by choosing (l, r, n) according to Theorem 2. Then for any $C(\hat{K}_i) \geq C(K^*) + \epsilon$, the following inequality holds:

$$\mathbb{P}\left\{ C(\hat{K}_{i+1}) - C(K^*) \leq \gamma_i (C(\hat{K}_i) - C(K^*)) \right\} \geq 1 - \delta,$$

where $\gamma_i := 1 - (1-\sigma)\frac{2\eta_i\lambda_1(R)\lambda_1^2(\Sigma_w)}{\|\Sigma_{K^*}\|}$ and $\gamma_i < 1, \forall i \in \mathbb{Z}_+$.
As a result, the gradient descent method enjoys the following performance bound:

$$\mathbb{P}\left\{ \min_{i \in [0, n_{\mathrm{GD}}]} C(\hat{K}_i) - C(K^*) \leq \epsilon \right\} \geq 1 - \delta n_{\mathrm{GD}}.$$

To achieve the desired accuracy of the estimates stated in Theorem 4, the corresponding values of l_i, r_i, n_i, as defined in Algorithm 4 given in Appendix C.3, should be selected based on Theorem 2. Theorem 4 ensures that the cost will converge to the optimal solution with a predefined accuracy gap ϵ, meaning $C(K^*) + \epsilon$ will be reached with a certain probability. However, due to the estimation error in the gradient, further improvements in cost stop when the realized cost $C(\hat{K}_i) \leq C(K^*) + \epsilon$. To achieve higher accuracy, it is necessary to minimize the gradient estimation error. Nevertheless, the algorithm can only converge to the optimal when the gradient estimation is exact, which is not possible with a finite number of rollouts. The interpretation of the σ is given in Remark 1. Regarding the step size η_i stated in Theorem 4, from the expression of h_{GD} given in (36), we observe that as the noise level Σ_w increases, the step size η_i decreases, reflecting the observation analyzed in Section 3.

5 Conclusions

In this work, we applied the policy gradient method to the Linear Quadratic Regulator problem with stochastic noise, analyzing both model-based and model-free approaches. Convergence guarantees were established for the model-based gradient descent and natural policy gradient methods with adaptive step sizes, demonstrating their ability to converge to the global optimum. For the model-free gradient descent and natural policy gradient methods, we showed that they can converge to the optimal solution within any desired accuracy. However, it is crucial that the algorithm's parameters are tuned based on factors such as the noise magnitude affecting the data trajectories. To improve convergence rates and reduce sample complexity, in addition to adaptive step sizes, we introduced a variance reduction technique. We also provided a qualitative discussion on the impact of noise on model-free policy gradient methods, offering insights into their robustness and providing useful guidelines for their tuning under practical scenarios. Several open problems remain to be addressed. As previously mentioned, implementing the model-free Gauss-Newton method presents an interesting opportunity for future research. While this work focused primarily on direct data-driven policy gradient approaches, it would also be valuable to explore and compare indirect data-driven policy gradient methods, which integrate online parameter estimation with model-based gradient descent. A thorough comparison of direct and indirect PG methods in the presence of stochastic noise is an important topic for future study.

References

[1] Y. Abbasi-Yadkori and C. Szepesvári. Regret bounds for the adaptive control of linear quadratic systems. In *Proceedings of the 24th Annual Conference on Learning Theory*, pages 1–26. PMLR, 2011.

[2] M. Abeille and A. Lazaric. Thompson Sampling for Linear-Quadratic Control Problems. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pages 1246–1254. PMLR, 2017.

[3] A. M. Annaswamy. Adaptive control and intersections with reinforcement learning. *Annual Review of Control, Robotics, and Autonomous Systems*, 6(1):65–93, 2023.

[4] D. Bertsekas. *Reinforcement Learning and Optimal Control*. Athena Scientific optimization and computation series. Athena Scientific, 2019.

[5] A. Cassel, A. Cohen, and T. Koren. Logarithmic regret for learning linear quadratic regulators efficiently. In *Proceedings of the 37th International Conference on Machine Learning*, pages 1328–1337. PMLR, 2020.

[6] S. Cen and Y. Chi. Global convergence of policy gradient methods in reinforcement learning, games and control. arXiv preprint arXiv:2310.05230, 2023.

[7] A. R. Conn, K. Scheinberg, and L. N. Vicente. *Introduction to Derivative-Free Optimization*. Society for Industrial and Applied Mathematics, 2009.

[8] S. Dean, H. Mania, N. Matni, B. Recht, and S. Tu. On the sample complexity of the linear quadratic regulator. *Foundations of Computational Mathematics*, 20, 10 2017.

[9] T. Degris, P. M. Pilarski, and R. S. Sutton. Model-free reinforcement learning with continuous action in practice. In *2012 American Control Conference (ACC)*, pages 2177–2182, 2012.

[10] M. Fazel, R. Ge, S. Kakade, and M. Mesbahi. Global convergence of policy gradient methods for the linear quadratic regulator. In *Proceedings of the 35th International Conference on Machine Learning*, pages 1467–1476. PMLR, 2018.

[11] B. Gravell, P. M. Esfahani, and T. Summers. Learning optimal controllers for linear systems with multiplicative noise via policy gradient. *IEEE Transactions on Automatic Control*, 66(11):5283–5298, 2021.

[12] I. Grondman, L. Busoniu, G. A. D. Lopes, and R. Babuska. A survey of actor-critic reinforcement learning: Standard and natural policy gradients. *IEEE Transactions on Systems, Man, and Cybernetics*, 42(6):1291–1307, 2012.

[13] B. Hambly, R. Xu, and H. Yang. Policy gradient methods for the noisy linear quadratic regulator over a finite horizon. *SIAM Journal on Control and Optimization*, 59(5):3359–3391, 2021.

[14] B. Hambly, R. Xu, and H. Yang. Policy gradient methods find the nash equilibrium in n-player general-sum linear-quadratic games. *Journal of Machine Learning Research*, 24(139):1–56, 2023.

[15] Y. Han, M. Razaviyayn, and R. Xu. Policy gradient converges to the globally optimal policy for nearly linear-quadratic regulators. *SIAM Journal on Control and Optimization*, 63(4):2936–2963, 2025.

[16] B. Hu, K. Zhang, N. Li, M. Mesbahi, M. Fazel, and T. Başar. Toward a theoretical foundation of policy optimization for learning control policies. *Annual Review of Control, Robotics, and Autonomous Systems*, 6(Volume 6, 2023):123–158, 2023.

[17] C. Ju, G. Kotsalis, and G. Lan. A model-free first-order method for linear quadratic regulator with $\tilde{o}(1/\varepsilon)$ sampling complexity. *SIAM Journal on Control and Optimization*, 63(3):2098–2123, 2025.

[18] S. M. Kakade. A natural policy gradient. In *Advances in Neural Information Processing Systems*. MIT Press, 2001.

[19] V. Konda and J. Tsitsiklis. Actor-critic algorithms. In *Advances in Neural Information Processing Systems*, pages 1008–13. MIT Press, 1999.

[20] D. Lee and J. Hu. Primal-dual q-learning framework for LQR design. *IEEE Transactions on Automatic Control*, 64(9):3756–3763, 2019.

[21] F. Lewis, D. Vrabie, and V. Syrmos. *Optimal Control*. EngineeringPro collection. Wiley, 2012.

[22] W. Li, P. Kounatidis, Z.-P. Jiang, and A. A. Malikopoulos. On the robustness of derivative-free methods for linear quadratic regulator. arXiv preprint arXiv:2506.12596, 2025.

[23] S. Liu, P.-Y. Chen, B. Kailkhura, G. Zhang, A. O. Hero III, and P. K. Varshney. A primer on zeroth-order optimization in signal processing and machine learning: Principals, recent advances, and applications. *IEEE Signal Processing Magazine*, 37(5):43–54, 2020.

[24] V. G. Lopez and M. A. Müller. An efficient off-policy reinforcement learning algorithm for the continuous-time LQR problem. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, pages 13–19, 2023.

[25] 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 *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89, pages 2916–2925. PMLR, 2019.

[26] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 1928–1937. PMLR, 2016.

[27] A. N. Moghaddam, A. Olshevsky, and B. Gharesifard. Sample complexity of the linear quadratic regulator: A reinforcement learning lens. arXiv preprint arXiv:2404.10851, 2025.

[28] B. Recht. A tour of reinforcement learning: The view from continuous control. *Annual Review of Control, Robotics, and Autonomous Systems*, 2(Volume 2, 2019):253–279, 2019.

[29] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347, 2017.

[30] L. Sforni, G. Carnevale, I. Notarnicola, and G. Notarstefano. On-policy data-driven linear quadratic regulator via combined policy iteration and recursive least squares. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, pages 5047–5052, 2023.

[31] M. Simchowitz and D. Foster. Naive exploration is optimal for online LQR. In *Proceedings of the 37th International Conference on Machine Learning*, pages 8937–8948. PMLR, 2020.

[32] B. Song and A. Iannelli. Convergence guarantees of model-free policy gradient methods for LQR with stochastic data. arXiv preprint arXiv:2405.13592, 2024.

[33] B. Song and A. Iannelli. The role of identification in data-driven policy iteration: A system theoretic study. *International Journal of Robust and Nonlinear Control*, n/a(n/a), 2024.

[34] B. Song and A. Iannelli. Robustness of online identification-based policy iteration to noisy data. *at - Automatisierungstechnik*, 73(6):398–412, 2025.

[35] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.

[36] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Proceedings of the 12th International Conference on Neural Information Processing Systems*, page 1057–1063. MIT Press, 1999.

[37] S. Takakura and K. Sato. Structured output feedback control for linear quadratic regulator using policy gradient method. *IEEE Transactions on Automatic Control*, 69(1):363–370, 2024.

[38] S. Tu and B. Recht. Least-squares temporal difference learning for the linear quadratic regulator. In *Proceedings of the 35th International Conference on Machine Learning*, pages 5005–5014. PMLR, 2018.

[39] F. A. Yaghmaie, F. Gustafsson, and L. Ljung. Linear quadratic control using model-free reinforcement learning. *IEEE Transactions on Automatic Control*, 68(2):737–752, Feb 2023.

[40] Z. Yang, Y. Chen, M. Hong, and Z. Wang. Provably global convergence of actor-critic: A case for linear quadratic regulator with ergodic cost. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[41] 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*, 59(6):4081–4109, 2021.

[42] K. Zhang, A. Koppel, H. Zhu, and T. Başar. Global convergence of policy gradient methods to (almost) locally optimal policies. *SIAM Journal on Control and Optimization*, 58(6):3586–3612, 2020.

[43] F. Zhao, A. Chiuso, and F. Dörfler. Policy gradient adaptive control for the LQR: Indirect and direct approaches. arXiv preprint arXiv:2505.03706, 2025.

[44] F. Zhao, X. Fu, and K. You. Convergence and sample complexity of policy gradient methods for stabilizing linear systems. *IEEE Transactions on Automatic Control*, 70(3):1455–1466, 2025.

[45] F. Zhao, K. You, and T. Başar. Global convergence of policy gradient primal–dual methods for risk-constrained LQRs. *IEEE Transactions on Automatic Control*, 68(5):2934–2949, 2023.

[46] I. Ziemann, A. Tsiamis, H. Sandberg, and N. Matni. How are policy gradient methods affected by the limits of control? In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pages 5992–5999, 2022.

A Perturbation of $\Sigma_K, C, \nabla C$

In this section, we demonstrate that the average covariance matrix Σ_K, the cost function $C(K)$, and its gradient $\nabla C(K)$ are locally Lipschitz continuous with respect to the policy K. This property is crucial for the analysis of model-free policy optimization.

Lemma 3 (Σ_K Perturbation) *Suppose $K', K \in \mathcal{S}$ are such that:*

$$\|K - K'\| \leq \frac{\lambda_1(\Sigma_w)\lambda_1(Q)}{8C(K)\|B\|} =: h(C(K)),$$

it holds that:

$$\|\Sigma_K - \Sigma_{K'}\| \leq \underbrace{8\left(\frac{C(K)}{\lambda_1(Q)}\right)^2 \frac{\|B\|}{\lambda_1(\Sigma_w)}}_{=:h_\Sigma(C(K))}\|K - K'\|.$$

Lemma 4 (C Perturbation) *Suppose $K', K \in \mathcal{S}$ are such that:*

$$\|K - K'\| \leq \min\{h\left(C(K)\right), \|K\|\},$$

it holds that:

$$\|C(K') - C(K)\| \leq h_C(C(K))\|K - K'\|,$$

where $h_C(C(K))$ is defined as:

$$h_C(C(K)) := 6\left(\frac{C(K)}{\lambda_1(\Sigma_w)\lambda_1(Q)}\right)^2 \left(2b_K^2(C(K))\|R\|\|B\| + b_K(C(K))\|R\|\right)\mathrm{Tr}(\Sigma_w), \tag{28}$$

with

$$b_K(C(K)) := \frac{1}{\lambda_1(R)}\left(\sqrt{\frac{(C(K) - C(K^*))\left(\|R\| + \|B\|^2\frac{C(K)}{\lambda_1(\Sigma_w)}\right)}{\lambda_1(\Sigma_w)}} + \|B\|\|A\|\frac{C(K)}{\lambda_1(\Sigma_w)}\right). \tag{29}$$

Lemma 5 (∇C Perturbation) *Suppose K' is such that:*

$$\|K - K'\| \leq \min\{h(C(K)), \|K\|\},$$

then there exists a polynomial $h_\nabla(C(K))$ such that:

$$\|\nabla C(K) - \nabla C(K')\| \leq h_\nabla(C(K))\|K - K'\|,$$

where $h_\nabla(C(K))$ is defined as:

$$h_\nabla(C(K)) := \alpha_1(C(K)) + \alpha_3(C(K)). \tag{30}$$

with

$$\alpha_1(C(K)) := 2\sqrt{\frac{(C(K) - C(K^*))}{\sigma_1(\Sigma_w)}\left(\|R\| + \frac{\|B\|^2C(K)}{\lambda_1(\Sigma_w)}\right)}h_\Sigma(C(K)); \tag{31}$$

$$\alpha_3(C(K)) := \left(\|R\| + \frac{\|B\|^2C(K)}{\lambda_1(\Sigma_0)} + \alpha_2(C(K))\left(\|B\|\|A\| + b_K(C(K))\|B\|^2\right)\right); \tag{32}$$

$$\alpha_2(C(K)) := 6\left(\frac{C(K)}{\lambda_1(\Sigma_w)\lambda_1(Q)}\right)^2 \left(2b_K^2(C(K))\|R\|\|B\| + b_K(C(K))\|R\|\right). \tag{33}$$

The proofs of Lemma 3, Lemma 4 and Lemma 5 are provided in [32, Appendix A.3, Appendix A.4, Appendix A.5], respectively. These proofs follow the approach used in [10, Lemma 16, Lemma 24, Lemma 25] by adapting the expressions of Σ_K, C and ∇C. In summary, Lemma 3, Lemma 4 and Lemma 5 establish that Σ_K, C and ∇C are locally Lipschitz continuous.

B Model-based Policy Gradient Methods

B.1 Proof of Theorem 1

Proof 1 *Using Lemma 2 and following the step by [10, Lemma 21], if*

$$\eta_i \leq \frac{1}{32}\min\left\{\left(\frac{\lambda_1(Q)\lambda_1(\Sigma_w)}{C(K_i)}\right)^2 \frac{1}{\|B\|\|\nabla C(K_i)\|(1+\|A+BK_i\|)},\ \frac{\lambda_1(Q)}{2C(K_i)\|R+B^\top P_{K_i}B\|}\right\}, \tag{34}$$

then,

$$C(K_{i+1}) - C(K^*) \leq \left(1 - \frac{2\eta_i\lambda_1(R)\lambda_1^2(\Sigma_w)}{\|\Sigma_K^*\|}\right)(C(K_i) - C(K^*)). \tag{35}$$

Based on the upper bounds on $\|\nabla C(K)\|$, we can derive the following from (34)*:*

$$\eta_i \leq \underbrace{\frac{1}{32}\min\left\{\left(\frac{\lambda_1(Q)\lambda_1(\Sigma_w)}{C(K_i)}\right)^2 \frac{1}{2\|B\|b_\nabla(C(K_i))}, \frac{\lambda_1(Q)}{2C(K_i)\left(\|R\| + \frac{\|B\|^2C(K_i)}{\lambda_1(\Sigma_w)}\right)}\right\}}_{=:h_{\mathrm{GD}}(C(K))}, \tag{36}$$

where b_∇ is defined as

$$b_\nabla(C(K)) := \sqrt{4\left(\frac{C(K)}{\lambda_1(Q)}\right)^2 \frac{(C(K)-C(K^*))}{\sigma_1(\Sigma_w)}\left(\|R\| + \frac{\|B\|^2C(K)}{\lambda_1(\Sigma_w)}\right)}. \tag{37}$$

From (35)*, we know that $C(K_{i+1}) \leq C(K_i)$. Then we have $h_{\mathrm{GD}}(C(K_{i+1})) \geq h_{\mathrm{GD}}(C(K_i))$. The remainder of the proof proceeds by setting $\eta_i = \eta_0, \forall i \in \mathbb{Z}_+$, which represents the lower bound of the convergence rate, and then following the steps outlined in [10, Lemma 22].*

B.2 Natural Policy Gradient

The natural policy gradient method adjusts the standard policy gradient by considering the geometry of the parameter space through the Fisher information matrix [18]. We now present the theorem that establishes the convergence guarantee of the natural policy gradient method

Theorem 5 (Natural Policy Gradient with Adaptive Step Size) *Suppose the initial $K_0 \in \mathcal{S}$, and consider the natural policy gradient iteration*

$$K_{i+1} = K_i - \eta_i \nabla C(K_i)\Sigma_{K_i}^{-1}, \quad \forall i \in \mathbb{Z}_+, \tag{38}$$

where the step size satisfies

$$\eta_i \leq \frac{1}{2\|R\| + \frac{2\|B\|^2C(K_i)}{\lambda_1(\Sigma_w)}}. \tag{39}$$

Then the following relationship holds:

$$C(K_{i+1}) - C(K^*) \leq \left(1 - \frac{2\eta_i\lambda_1(R)\lambda_1(\Sigma_w)}{\|\Sigma_{K^*}\|}\right)(C(K_i) - C(K^*)). \tag{40}$$

For $\eta_i = \frac{1}{2\|R\| + \frac{2\|B\|^2C(K_i)}{\lambda_1(\Sigma_w)}}$, $\forall i \in \mathbb{Z}_+$, and given any accuracy gap $\epsilon > 0$, if the number of iterations N satisfies:

$$N \geq \frac{\|\Sigma_{K^*}\|}{2\lambda_1(\Sigma_w)}\left(\frac{\|R\|}{\lambda_1(R)} + \frac{\|B\|^2C(K_0)}{\lambda_1(R)\lambda_1(\Sigma_w)}\right)\log\frac{C(K_0)-C(K^*)}{\epsilon},$$

then $C(K_N) - C(K^*) \leq \epsilon$.

The proof of Theorem 5 relies on Lemma 2 and follows the procedure outlined in [10, Lemma 15]. Theorem 5 establishes the global convergence properties of the natural policy gradient method, when an appropriate step size η_i is selected. The introduction of adaptive step sizes improves the convergence rate, as the step sizes increase adaptively with the decreasing of the cost.

B.3 Gauss-Newton Method

We now present the theorem that provides the convergence guarantee for the Gauss-Newton method.

Theorem 6 (Gauss-Newton Method) *Suppose the initial $K_0 \in \mathcal{S}$, and consider the Gauss-Newton iteration*

$$K_{i+1} = K_i - \eta(R + B^\top P_{K_i} B)^{-1} \nabla C(K_i) \Sigma_{K_i}^{-1}, \quad \forall i \in \mathbb{Z}_+, \tag{41}$$

with $\eta \leq \frac{1}{2}$. Then, the following relationship holds:

$$C(K_{i+1}) - C(K^*) \leq \left(1 - \frac{2\eta\lambda_1(\Sigma_w)}{\|\Sigma_K^*\|}\right)(C(K_i) - C(K^*)). \tag{42}$$

For the maximum fixed step size $\eta = \frac{1}{2}$ and any accuracy gap $\epsilon > 0$, if the number of iterations N satisfies:

$$N \geq \frac{\|\Sigma_K^*\|}{\lambda_1(\Sigma_w)} \log \frac{C(K_0) - C(K^*)}{\epsilon},$$

then $C(K_N) - C(K^*) \leq \epsilon$.

The proof of the Theorem 6 builds on the property shown in Lemma 2 and follows similar steps to the steps in [10, Lemma 14]. It is important to note that the choice of step size here is independent of the specific system parameters. To achieve the fastest convergence rate, one can always select the step size $\eta = \frac{1}{2}$, which is equivalent to the policy iteration algorithm.

C Model-free Policy Gradient Methods

C.1 Error Bound of covariance matrix

Theorem 7 (Error bound of $\hat{\Sigma}_K$) *Given an arbitrary tolerance $\epsilon' > 0$, which can be expressed as $\epsilon' = \epsilon'_l + \epsilon'_n + \epsilon'_r$, and an arbitrary probability $\delta' \in (0,1)$, which can be expressed as $\delta' = 1 - (1-\delta'_n)(1-\delta'_x)$ with $\delta'_n,\ \delta'_x \in (0,1)$, for a given $K \in \mathcal{S}$, the estimated average covariance $\hat{\Sigma}_K$ from Algorithm 1 enjoys the following bound:*

$$\mathbb{P}\left\{\|\hat{\Sigma}_K - \Sigma_K\| \leq \epsilon'\right\} \geq 1 - \delta', \tag{43}$$

if the parameters r, l, n in Algorithm 1 satisfy:

$$r \leq r'_{\max}(C(K), \epsilon'_r), l \geq l'_{\min}(C(K), \epsilon'_l), n \geq n'_{\min}(C(K), \epsilon'_r, \epsilon'_l, \epsilon'_n, \delta'_x, \delta'_n), \tag{44}$$

where the detailed expressions of functions $r'_{\max}$, $l'_{\min}$ and $n'_{\min}$ are given in (59) (60) *and* (61) *in the Proof C.6, respectively.*

The proof of Theorem 7 is given in [32, Appendix C.6].

C.2 Gradient estimation with variance reduction algorithm

Algorithm 3 Gradient estimation with variance reduction

Require: Gain matrix $K \in \mathcal{S}$, number of rollouts n_b, rollout length l, exploration radius r, an upper bound of the initial state L_0

for $k = 1, ..., n_b$ **do**

1. Generate a sample initial state $x_0^{(k)}$ with $\|x_0^{(k)}\| \leq L_0$;

2. Estimate the baseline function $\hat{b}_s(x_0^{(k)})$ using Algorithm 2

3. Generate a sample gain matrix $\hat{K}_k = K + U_k$, where U_k is drawn uniformly at random over matrices of compatible dimensions with Frobenius norm r;

4. Excite the closed-loop system with $u_t^{(k)} = \hat{K}_k x_t^{(k)}$ for l-steps starting from $x_0^{(k)}$, yielding the state sequence $\left\{x_t^{(k)}\right\}_{t=0}^{l-1}$;

5. Collect the empirical finite-horizon cost estimate $\hat{C}_k := \frac{1}{l}\sum_{t=0}^{l-1} x_t^{(k)\top}(Q + \hat{K}_k^\top R \hat{K}_k) x_t^{(k)}$

end for

return Gradient estimate $\hat{\nabla} C(K) := \frac{1}{n_b}\sum_{k=1}^{n_b} \frac{n_x n_u}{r^2}(\hat{C}_k - \hat{b}_s(x_0^{(k)}))U_k$

C.3 Model-free gradient descent with adaptive step size

Algorithm 4 Model-free gradient descent with adaptive step size

Require: An initial stabilizing gain matrix $\hat{K}_0$, desired accuracy ϵ and probability δ.
for $i = 1, ..., \infty$ **do**
1. Compute the required rollouts n_i, exploration radius r_i and rollout length l_i based on Theorem 2 or 3 to achieve the accuracy stated in Theorem 4.
2. Use Algorithm 1 or 3 to estimate the gradient $\hat{\nabla} C(\hat{K}_i)$.
3. Update the gradient as $\hat{K}_{i+1} = \hat{K}_i - \eta_i \hat{\nabla} C(\hat{K}_i)$, with $\eta_i \leq h_{\mathrm{GD}}(C(\hat{K}_i))$.

end for

In Algorithm 4, the required rollouts n_i, exploration radius r_i and rollout length l_i are determined adaptively to the cost $C(\hat{K}_i)$, i.e., they are computed online inside the for-loop. This design is aligned with Theorem 4, which shows that the cost function $C(\hat{K}_i)$ decreases probabilistically with each iteration. As the cost $C(\hat{K}_i)$ decreases, this leads to a reduction in the required rollouts n_i and rollout length l_i, while the exploration radius r_i increases. Alternatively, the algorithm can be modified to compute these parameters offline, where for all iterations i, n_i, l_i, r_i are calculated once based on the initial cost $C(\hat{K}_0)$.

C.4 Natural Policy Gradient with Adaptive Step Size

Analogous to the model-free gradient descent method, we now turn to analyzing the convergence of the natural policy gradient method.

Theorem 8 (Model-free Natural Policy Gradient with Adaptive Step Size) *Suppose the initial $\hat{K}_0 \in \mathcal{S}$, and consider natural policy gradient with adaptive step size:*

$$\hat{K}_{i+1} = \hat{K}_i - \eta_i \hat{\nabla} C(\hat{K}_i) \hat{\Sigma}_{\hat{K}_i}^{-1}, \quad \forall i \in \mathbb{Z}_+, \tag{45}$$

where $\hat{\nabla} C(\hat{K}_i)$ and $\hat{\Sigma}_{\hat{K}_i}$ are the gradient and covariance estimates from Algorithm 1 and $0 < \eta_i \leq \frac{1}{2\|R\| + \frac{2\|B\| C(\hat{K}_i)}{\lambda_1(\Sigma_w)}}$.
Given any accuracy $\epsilon > 0$ and $\sigma \in (0,1)$, defining $\eta_{\mathrm{NPG}} := \inf_i \eta_i$ and the number of iterations n_{NPG}:

$$n_{\mathrm{NPG}} \geq \frac{\|\Sigma_{K^*}\|}{2(1-\sigma)\eta_{\mathrm{NPG}} \lambda_1(\Sigma_w)} \log \frac{C(K_0) - C(K^*)}{\epsilon}.$$

Given any probability $\delta \in (0,1)$ satisfying $\delta n_{\mathrm{NPG}} \in (0,1)$, if the estimation error of the gradient $\hat{\nabla} C(\hat{K}_i)$ and the covariance $\hat{\Sigma}_{\hat{K}_i}$ satisfy:

$$\mathbb{P}\left\{ \|\hat{\nabla} C(\hat{K}_i) - \nabla C(\hat{K}_i)\| \leq \frac{\sigma \epsilon \lambda_1(R) \lambda_1^2(\Sigma_w)}{4 h_\nabla(C(\hat{K}_i)) \|\Sigma_K^*\|} \right\} \geq \sqrt{1-\delta}, \tag{46a}$$

$$\mathbb{P}\left\{ \|\hat{\Sigma}_{\hat{K}_i} - \Sigma_{\hat{K}_i}\| \leq \frac{\sigma \epsilon \lambda_1(R) \lambda_1^3(\Sigma_w)}{4 h_\nabla(C(\hat{K}_i)) \|\Sigma_K^*\| \sqrt{b_\nabla(C(\hat{K}_i))}} \right\} \geq \sqrt{1-\delta}, \tag{46b}$$

where h_∇ was introduced in Lemma 5 and b_∇ was defined in (37)*. This can be ensured by choosing (l_i, r_i, n_i) according to Theorem 2 and Theorem 7. then for any $C(\hat{K}_i) \geq C(K^*) + \epsilon$, the following inequality holds:*

$$\mathbb{P}\left\{ C(\hat{K}_{i+1}) - C(K^*) \leq \kappa_i (C(\hat{K}_i) - C(K^*)) \middle| C(\hat{K}_i) \right\} \geq 1 - \delta,$$

where $\kappa_i := \left(1 - (1-\sigma) \frac{2\eta_i \lambda_1(R) \lambda_1(\Sigma_w)}{\|\Sigma_K^*\|}\right)$ and $\kappa_i < 1, \forall i \in \mathbb{Z}_+$.
As a result, the natural policy gradient method enjoys the following performance bound:

$$\mathbb{P}\left\{ \min_{i \in [0, n_{\mathrm{NPG}}]} C(\hat{K}_i) - C(K^*) \leq \epsilon \right\} \geq 1 - \delta n_{\mathrm{NPG}},$$

The proof of Theorem 2 is provided in [32, Appendix C.9]. From Theorem 2, we can select the values of $l_i^\nabla, r_i^\nabla, n_i^\nabla$ to satisfy the requirement for the gradient estimation error, as indicated in (46a). Similarly, we can select $l_i^\Sigma, r_i^\Sigma, n_i^\Sigma$ to satisfy the requirement for the covariance estimation error, as described in (46b). To ensure

both requirements are met, we can then set: $n_i \geq \max\{n_i^\nabla, n_i^\Sigma\}, l_i \geq \max\{l_i^\nabla, l_i^\Sigma\}, r_i \leq \min\{r_i^\nabla, r_i^\Sigma\}$. These choices guarantee that the estimates of both the gradient and the covariance matrix satisfy their respective accuracy requirements. This combined selection of parameters is implemented in Algorithm 5 to ensure convergence with the desired accuracy and probability.

Algorithm 5 Model-free natural policy gradient with adaptive step size

Require: An initial stabilizing gain matrix $\hat{K}_0$, desired accuracy ϵ and probability δ.

for $i = 0, ..., \infty$ **do**

1. Compute the required rollouts n_i, exploration radius r_i and rollout length l_i based on Theorem 2 to achieve the accuracy stated in Theorem 8.
2. Use Algorithm 1 to estimate the gradient $\hat{\nabla}C(\hat{K}_i)$ and covariance $\hat{\Sigma}_{\hat{K}_i}$.
3. Update the gradient as $\hat{K}_{i+1} = \hat{K}_i - \eta_i \hat{\nabla}C(\hat{K}_i)\hat{\Sigma}_{\hat{K}_i}^{-1}$ with $\eta_i \leq \frac{1}{2\|R\| + \frac{2\|B\|C(\hat{K}_i)}{\lambda_1(\Sigma_w)}}$.

end for

Remark 1 (Qualitative effect of gradient and covariance errors on convergence)

C.5 Comparison of Model-free PG for Noise and Noise-free Case

In Algorithm 1, three key quantities, exploration radius r, rollout length l, and the required rollouts n, play a central role in determining the estimation errors on gradient and covariance estimation. The presence of stochastic noise acting on the system (1b) during data collection influences these quantities in the following ways:

- **Exploration radius:** The selection of the exploration radius is fully determined by the desired accuracy and the specific model parameters. The choice of r directly influences the estimation error by affecting the bias term U_i, which represents the difference between quantities $\|\nabla C(K)\|$ and $\|\nabla C(K + U_k)\|$, as well as $\|\Sigma_K\|$ and $\|\Sigma_{K+U_k}\|$. These differences are crucial in the error analysis, as detailed in Appendix [32, Appendix C.3] and [32, Appendix C.6], respectively. The estimation error of the gradient is discussed in detail in (47) for the noise case and in [10, Lemma 27] for the noise-free case. Similarly, the estimation error of the covariance is analyzed in (59) for the noise case and in [10, Theorem 30] for the noise-free case. Due to the differences in cost functions, the noise-free case considers the infinite-horizon cost, while the noisy case considers the average infinite-horizon cost. Consequently, for the noisy case, r is determined by Σ_w whereas for the noise-free case, it is determined by Σ_0.
- **Rollout length:** The rollout length l is different for the noise-free case and the noise case. In both scenarios, the rollout length plays a crucial role in estimating the gradient and covariance, whose true values are defined over an infinite horizon. However, finite approximations must be used. In the noise-free case, as described in [10, Lemma 23], the rollout length is determined by the desired accuracy, model parameters, and initial covariance Σ_0. In the presence of noise, as discussed in [32, Lemma C.1], the rollout length is influenced not only by Σ_0, but also by the noise covariance Σ_w. Therefore, the rollout length in the noisy case must account for both the system's initial conditions and noise disturbances. These relationships are quantitatively explored in through (48) and (60).
- **Required rollouts:** The most significant difference between the noisy and noise-free cases comes from using the matrix concentration inequality for the trajectories consisting of noisy data. In the noise-free case, the boundedness of the data is guaranteed by an upper bound on the initial state, i.e., $\|x_0^{(k)}\| \leq L_0$ (see [10, Lemmas 27, 29]). In contrast, in the noisy case, particularly with unbounded Gaussian noise, the state remains bounded only with a certain probability ([32, Appendix C.4]). This boundedness in probability is then used to apply matrix concentration inequalities, providing an upper bound on the covariance of the samples([32, Appendix C.5]).

C.6 Detailed Expressions

$$r_{\max} = \min\{h(C(K)), \|K\|, \frac{\epsilon_r}{h_\nabla(C(K))}\}, \tag{47}$$

where h_∇ is defined in (30).

$$l_{\min} = \frac{2(C(K) + rh_C C(K))^2}{\epsilon_l \lambda_1(\Sigma_w)} \left(\frac{\|\Sigma_0\|\lambda_1(\Sigma_w) + C(K) + rh_C C(K)}{\lambda_1(Q)\lambda_1^2(\Sigma_w)} + \frac{1}{\lambda_1(Q)} \right). \tag{48}$$

$$N_1 := \frac{2\min\{n_x, n_u\}}{\epsilon_n^2}\left(\alpha_4^2(C(K),\epsilon_r) + \frac{\alpha_5(C(K),\epsilon_r)\epsilon_n}{3\sqrt{\min(n_x,n_u)}}\right)\log\left[\frac{n_x+n_u}{\delta_n}\right], \tag{49a}$$

$$N_2 := \frac{2\min\{n_x, n_u\}}{\epsilon_d^2}\left(\alpha_7^2(C(K),\delta_x) + \frac{\alpha_8(C(K),\delta_x)\epsilon_d}{3\sqrt{\min(n_x,n_u)}}\right)\log\left[\frac{n_x+n_u}{\delta_d}\right]; \tag{49b}$$

where $\alpha_4, \alpha_5, \alpha_7$ and α_8 are defined in (50), (51), (53) and (54) respectively.

$$\alpha_4(C(K),\epsilon_1) := \frac{n_x n_u\left(C(K) + rh_C(C(K))\right)}{r} + \epsilon_1 + b_\nabla(C(K)). \tag{50}$$

$$\alpha_5(C(K),\epsilon_1) := \max\{n_x,n_u\}^2\left(\frac{n_x n_u\left(C(K) + rh_C(C(K))\right)}{r}\right)^2 + \left(\epsilon_1 + b_\nabla(C(K))\right)^2, \tag{51}$$

with h_C defined in (28) and b_∇ defined in (37).

$$\alpha_6(C(K),\delta_x) := \frac{n_x n_u}{r}\frac{C(K) + rh_C(C(K))}{\lambda_1(\Sigma_w)}\left(L_0 + (l-1)\frac{\mathrm{Tr}(\Sigma_{\mathrm{w}})}{1-(1-\delta_x)^{1/l}}\right)^2. \tag{52}$$

$$\alpha_7(C(K),\delta_x) := \epsilon_l + \epsilon_{nr} + b_\nabla(C(K)) + \alpha_6(C(K),\delta_x). \tag{53}$$

$$\alpha_8(C(K),\delta_x) := \max\{n_x,n_u\}^2\alpha_6^2(C(K),\epsilon_x) + \left(\epsilon_l + \epsilon_{nr} + b_\nabla(C(K))\right)^2. \tag{54}$$

$$N_3 := \frac{2\min\{n_x, n_u\}}{\epsilon_d^2}\left(\alpha_7^2(C(K),\delta_x) + \frac{\alpha_{11}(C(K),\delta_x,\hat{b}_s(x_0))\epsilon_d}{3\sqrt{\min(n_x,n_u)}}\right)\log\left[\frac{n_x+n_u}{\delta_d}\right], \tag{55}$$

where

$$\alpha_{11}(C(K),\delta_x,\hat{b}_s(x_0)) := \max\{n_x,n_u\}^2\alpha_{12}^2(\hat{b}_s(x_0)) + \left(\epsilon_l + \epsilon_{nr} + b_\nabla(C(K))\right)^2,$$

and

$$\alpha_{12}(\hat{b}_s(x_0)) := \frac{n_x n_u}{r}\{\max\{\hat{C}_k - b_s^*(x_0), b_s^*(x_0)\} + \|b_s(x_0) - \hat{b}_s(x_0)\|\}.$$

$$\tilde{n}_{\min} = \frac{2}{(\epsilon_v')^2}\left((\bar{C}_E^V + \bar{C}^V)^2 + \frac{((\bar{C}_E^V)^2 + (\bar{C}^V(\tilde{\delta}_x))^2)\epsilon_v}{3}\right)\log\left[\frac{2}{(\tilde{\delta}_v)}\right]. \tag{56}$$

where

$$\bar{C}^V(\tilde{\delta}_x) := \frac{C(K)}{\lambda_1(\Sigma_w)}\left(L_0 + (l-1)\frac{\mathrm{Tr}(\Sigma_{\mathrm{w}})}{1-(1-\delta_x)^{1/l}}\right)^2, \tag{57}$$

and

$$\bar{C}_E^V := \frac{C(K)}{\lambda_1(\Sigma_w)}\left(L_0 + (l-1)\|\Sigma_w\|\right)^2. \tag{58}$$

$$r'_{\max} := \min\left\{h(C(K)), \|K\|, \frac{\epsilon_r'}{b_\nabla(C(K))}\right\}. \tag{59}$$

$$l'_{\min} := \frac{2C(K)}{\epsilon_l'\lambda_1(\Sigma_w)}\left(\frac{C(K)\|\Sigma_0\|}{\lambda_1(Q)\lambda_1(\Sigma_w)} + \frac{C^2(K)}{\lambda_1(Q)\lambda_1^2(\Sigma_w)} + \frac{C(K)}{\lambda_1(Q)}\right). \tag{60}$$

$$n'_{\min} := \frac{2n_x}{(\epsilon_n')^2}\left(\alpha_{10}(C(K),\delta_x') + \frac{\alpha_9(C(K),\delta_x')\epsilon_n'}{3\sqrt{n_x}}\right)\log\left[\frac{2n_x}{\delta_n'}\right], \tag{61}$$

where $\alpha_9(C(K),\delta_x')$ and $\alpha_{10}(C(K),\delta_x')$ are defined as

$$\alpha_9(C(K),\delta_x') := \frac{C(K) + rh_C(C(K))}{\lambda_1(Q)} + (\bar{L}'(\delta_x'))^2, \tag{62}$$

$$\alpha_{10}(C(K),\delta_x') := n_x^2\left[(\bar{L}'(\delta_x'))^2 + \left(\frac{C(K) + rh_C(C(K))}{\lambda_1(Q)}\right)^2\right], \tag{63}$$

with $\bar{L}'(\delta_x') := L_0 + (l-1)\frac{\mathrm{Tr}(\Sigma_{\mathrm{w}})}{1-(1-\delta_x)^{1/l}}$.

D Simulation Results

In this subsection, we use numerical simulations [1] to demonstrate the benefits of incorporating the variance reduction technique and adaptive step sizes in model-free policy gradient algorithms and the overall effect of noise on them. The matrices (A, B, Q, R) and $\hat{K}_0$ are the same as described in Section 3. The rollout length, exploration radius, and number of rollouts are set to $l = 100,\ r = 0.05,\ n = 1000$, respectively. The simulation results are obtained from a Monte Carlo simulation over 10 data samples.

[1] The Matlab codes used to generate these results (in Section 3 and Section D) are accessible from the repository: `https://github.com/col-tasas/2025-PGforLQRwithStochastic`

D.1 Noise Level and Stepsize

The policy is updated using the gradient descent method defined in (27). Figure 2 illustrates the relative suboptimality gap of the cost associated with the iterated policy $\hat{K}_i$ as a function of the iteration index i, for different noise levels and step sizes.

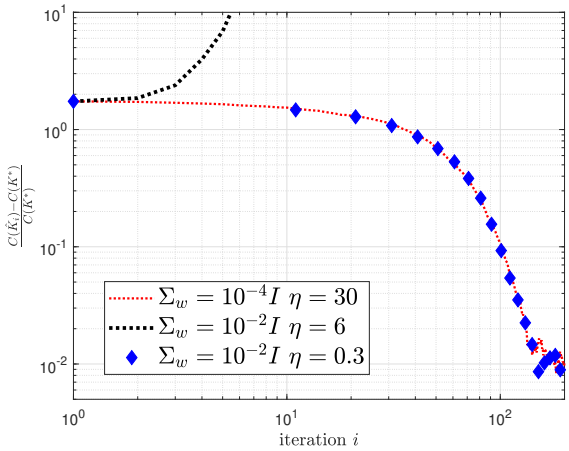

Figure 2: Model-free GD: Effect of Noise Level on Step Size

When the noise level is low, $\Sigma_w = 10^{-4}I$, the step size is empirically chosen as 30, as shown by the red dashed line. In this case, we observe a persistent suboptimality gap, which originates from the estimation error of the gradient inherent to the zeroth-order method. When the noise level increases to $\Sigma_w = 10^{-2}I$, maintaining the same step size leads to divergence of the cost function (as shown in black dotted line). As stated in Theorem 4, when the noise level is higher, it is necessary to decrease the step size to ensure convergence. By reducing the step size to $\eta = 0.3$, as illustrated by the blue points in the figure, the cost converges again with suboptimality gap.

D.2 GD with/out variance reduction

The update of the policy is according to the gradient descent method, whose update law was defined in (27). The step size η is chosen based on the noise level: for $\Sigma_w = 10^{-4}I$ is set to 30, while for $\Sigma_w = 10^{-2}I$, it is reduced to 0.3, qualitatively in accordance with the bounds established in Theorem 4. For the variance reduction technique, the number of rollouts to estimate the baseline function, n_b, is set as 20.

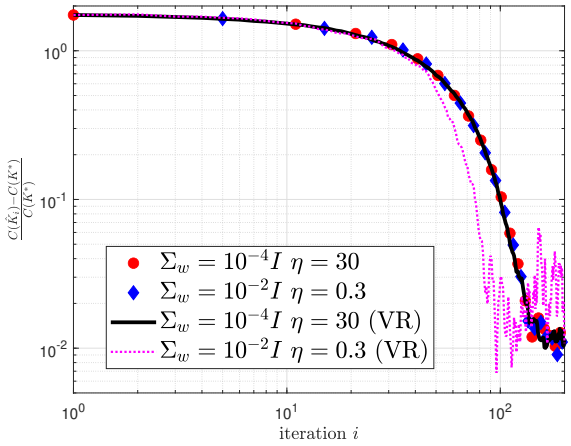

Figure 3: Model-free GD with/out Variance Reduction

Figure 3 illustrates the relative suboptimality gap of the cost associated with the iterated policy $\hat{K}_i$ as a function of the iteration index i. Several observations can be made regarding the impact of variance reduction. We compare two groups of curves corresponding to different noise levels: the red points and black solid lines for the noise level $\Sigma_w = 10^{-4}$, and the blue points and magenta lines for $\Sigma_w = 10^{-2}$. The dotted point curves represent results without variance reduction. It can be observed that the performance improvement achieved through variance reduction is more pronounced in high-noise scenarios, which is consistent with the analysis in Theorem 3.

D.3 Model-free NPG with Adaptive Step Sizes

The natural policy gradient algorithm updates the gain $\hat{K}_i$ according to the update law defined in (45). A fixed step size is chosen as $\eta = \frac{a}{b+c\mathrm{Tr}(\mathrm{P}_{\mathrm{K}_0})}$ which follows the structure of the step size derived in Theorem 8 and is applied uniformly across all noise levels $\Sigma_w = 10^{-4}I, , 10^{-2}I, , 10^{0}I$. Due to the conservative nature of the theoretical bound, the parameters a, b, and c are selected empirically as 0.09, 1, and 2, respectively. In addition, an adaptive step size is considered, defined as $\eta_i = \frac{a}{b+c\mathrm{Tr}(\mathrm{P}_{\mathrm{K}_i})}$, where $\mathrm{Tr}(P_{K_i})$ reflects the current cost associated with the policy at iteration i.

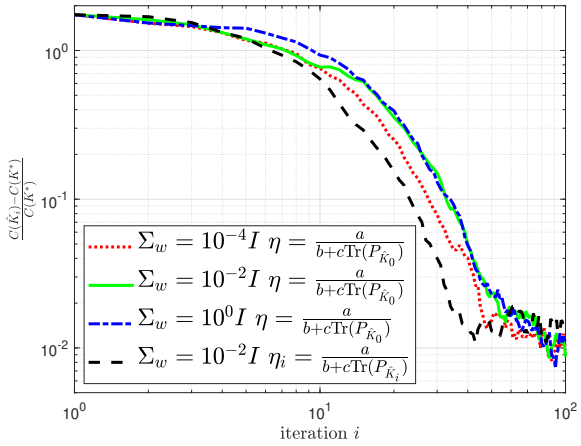

Figure 4: Model-free NPG with/out Adaptive Step Size

Comparing the black dashed line and the green solid line in Figure 4, we observe that using adaptive step sizes yields a faster convergence rate than fixed step sizes, further highlighting the benefits of adaptive strategies. As shown in Figure 4 (red dotted, green solid, and blue dot–dashed lines), the model-free NPG algorithm achieves nearly the same accuracy even as the noise Σ_w increases, provided that l, r, and n remain unchanged. At higher noise levels, the estimation errors of both $\hat{\nabla}C(\hat{K}_i)$ and $\hat{\Sigma}_{\hat{K}_i}$ grow. These errors are explicitly accounted for in our analysis (Theorems 2 and 7). Due to the natural policy gradient update rule (see (45)), the estimated gradient is multiplied by the inverse of the estimated covariance, which causes the errors in these two quantities to partially offset each other. If the analysis were performed directly on the product $\hat{\nabla}C(\hat{K}_i)\hat{\Sigma}_{\hat{K}_i}^{-1}$ rather than treating them separately (as in [32, Appendix C.9, Proof of Theorem 4.5]), substantially stronger theoretical guarantees could be established. This observation points to an important direction for future research.