Model-Free Design of Stochastic LQR Controller from Reinforcement Learning and Primal-Dual Optimization Perspective

Man Li, Jiahu Qin, Wei Xing Zheng, Yaonan Wang, Yu Kang

Abstract

To further understand the underlying mechanism of various reinforcement learning (RL) algorithms and also to better use the optimization theory to make further progress in RL, many researchers begin to revisit the linear-quadratic regulator (LQR) problem, whose setting is simple and yet captures the characteristics of RL. Inspired by this, this work is concerned with the model-free design of stochastic LQR controller for linear systems subject to Gaussian noises, from the perspective of both RL and primal-dual optimization. From the RL perspective, we first develop a new model-free off-policy policy iteration (MF-OPI) algorithm, in which the sampled data is repeatedly used for updating the policy to alleviate the data-hungry problem to some extent. We then provide a rigorous analysis for algorithm convergence by showing that the involved iterations are equivalent to the iterations in the classical policy iteration (PI) algorithm. From the perspective of optimization, we first reformulate the stochastic LQR problem at hand as a constrained non-convex optimization problem, which is shown to have strong duality. Then, to solve this non-convex optimization problem, we propose a model-based primal-dual (MB-PD) algorithm based on the properties of the resulting Karush-Kuhn-Tucker (KKT) conditions. We also give a model-free implementation for the MB-PD algorithm by solving a transformed dual feasibility condition. More importantly, we show that the dual and primal update steps in the MB-PD algorithm can be interpreted as the policy evaluation and policy improvement steps in the PI algorithm, respectively. Finally, we provide one simulation example to show the performance of the proposed algorithms.

Key words: Stochastic linear quadratic regulator; model-free design; reinforcement learning; optimization; duality.

1 Introduction

Reinforcement learning (RL), a branch of machine learning, aims to learn policies to optimize a long-term decision-making process in a dynamic environment [1, 2]. In general, there are two main characteristics, which are also readily considered as the difficulties of the RL setting [1]. First, the value of a state is the total amount of reward/cost an agent expects to accumulate over the future, starting from that state. That is, early decisions affect the outcomes both near and far in the future. Second, the learner’s knowledge about the environment is incomplete, e.g., the underlying mechanism governing the dynamics of the state-transition trajectories is unknown. To overcome these difficulties, a lot of effective RL algorithms have been developed [3–5], and these algorithms have been successfully used in numerous fields, including smart grids [6], multi-robot systems [7], intelligent transportation [8], and networked control [9]. From these works, we know that in spite of the fact that RL cannot be regarded as a standard optimization problem due to the lack of an explicit objective function [10], it still has strong ties to the optimization formulation, since its goal is to optimize the long-term cumulative performance [11]. In view of this, existing fruitful optimization algorithms and theories might enable researchers to make further progress in RL, so that the integration of elementary RL methods with optimization frameworks is worthwhile to study. Some efforts towards this direction have been made, see, for example, [12, 13].

It is worth mentioning that following this line, numerous researchers begin to revisit the linear-quadratic regulator (LQR) problem [14–19], the task of which is to determine an optimal feedback controller minimizing a quadratic cost function for linear systems [20]. One important reason for the revisit of the LQR problem is that it is simple and yet captures the two characteristics of RL problems mentioned in the previous paragraph. Such properties of the LQR problem provide convenience for understanding the behavior and the underlying mechanism of various RL algorithms. Another reason is that the traditional methods for addressing the LQR problem require to directly solve an algebraic Riccati equation (ARE), which depends upon exact information of the system matrices; however, the exact system matrices are usually unknown in real applications. Although model identification provides a way to obtain the estimated system model, the propagation and accumulation of modeling errors may cause undesirable system behaviors. Therefore, it is meaningful to study the model-free design of the LQR controller with the consideration of the integration of RL algorithms with optimization theories. Some relevant works can be found in [15–19].

Specifically, in [15] and [16], the authors reformulate the LQR problem as a non-convex optimization problem, and merge ideas from optimal control theory, mathematical optimization, and sampled-based RL to solve it. To seek the solutions to the resulting non-convex optimization
problem, model-free stochastic policy gradient algorithms along with zeroth-order optimization techniques are developed with provable global convergence [15] and explicit convergence rate [16]. The work [17] considers distributed RL for decentralized LQR problem, and proposes a zeroth-order distributed policy optimization algorithm to learn the local optimal controllers in a distributed manner, by leveraging the ideas of policy gradient, zero-order optimization, as well as consensus algorithms. Note that in the above mentioned works, the key to realizing model-free control is to use the zeroth-order optimization approach to estimate the policy gradient. This approach often suffers from large variance, as it involves the randomness of an entire trajectory. To address this difficulty, some different algorithms have been proposed [18, 19]. The work [18] develops a model-free approximate policy iteration algorithm based on Q-function, in which the way to guarantee a good regret is also discussed. In [19], an on-policy actor-critic algorithm, which is an online alternating update algorithm for bilevel optimization, is developed with a non-asymptotic convergence analysis, and then it is extended to the off-policy case to alleviate the data-hungry problem.

Despite the fact that extensive research has been carried out on the LQR problem when considering the integration of RL frameworks with optimization techniques, these works, such as [15–19], mainly focus on the theoretical understanding of the performance of common RL algorithms. They do not provide any explanations for the proposed RL algorithms from the perspective of optimization or establish any relationship between RL and optimization-based methods. Different from the above mentioned works, the authors in [21] formulate the LQR problem as a new optimization problem, and further explain its duality results from a Q-learning viewpoint. The duality results proposed in [21] are based on the results given in [22], where the duality for the stochastic LQR problem pertaining to linear systems with stochastic noise is investigated in terms of semidefinite programming (SDP). These works motivate some questions. Can we formulate the stochastic LQR problem as an optimization problem whose optimal points can be interpreted from the perspective of various RL algorithms? Can we explain the RL algorithms used to solve stochastic LQR problem from the perspective of optimization?

To answer these questions, we investigate the model-free design of stochastic LQR controller from the perspective of both RL and optimization. We first solve the stochastic LQR problem in a model-free way from the perspective of RL. Then, we reconsider this problem by some reformulation and transformations, and solve it from the perspective of primal-dual optimization. Besides that, we also provide a theorem to show that the algorithms developed from the above two perspectives are equivalent in the sense that there is a corresponding relation in the involved iterations. The contributions of this work are summarized as follows:

1. To solve the stochastic LQR problem, we develop a model-free off-policy policy iteration (MF-OPPI) algorithm and a model-free primal-dual (MB-PD) algorithm from the perspective of RL and primal-dual optimization, respectively. Both algorithms can be implemented in a model-free way, so that one can circumvent model identification, which may cause undesirable behaviors due to the propagation and accumulation of modeling errors. More importantly, we show that the dual and primal update steps in the MB-PD algorithm can be interpreted as the policy evaluation and policy improvement steps in the classical policy iteration (PI) algorithm, respectively. This result provides a novel primal-dual optimization perspective to understand the classical PI algorithm.

2. From the perspective of RL, we develop a new MF-OPPI algorithm to address the stochastic LQR problem, in which the data generated by exerting the behavior policy on the real system is repeatedly used to evaluate the target policy so as to alleviate the data-hungry problem to some extent. Compared with the off-policy actor-critic algorithm proposed in [19], the newly developed MF-OPPI algorithm is simple and easy to implement, since it does not require to use neural networks for function approximation.

3. From the optimization perspective, we first reformulate the stochastic LQR problem as a constrained non-convex optimization problem, based on the works [21] and [22]. However, due to the consideration of the influence of stochastic noise and the relation with RL algorithms, the proposed optimization formulation is different from the ones given in [21] and [22]. Such differences bring about some challenges in the equivalent formulation of primal-dual problems as well as the analysis for the strong duality. To address the resulting constrained non-convex optimization problem, we develop a new MB-PD algorithm with a model-free implementation, by using the properties of the resulting Karush-Kuhn-Tucker (KKT) conditions. This algorithm circumvents the requirement of probing noise exerted on the system for persistently exciting, which is necessary for the newly proposed MF-OPPI algorithm and other RL-based algorithms proposed in [23–25], thereby avoiding undesirable oscillation.

The rest of this paper is organized as follows. We first formulate the problem of interest and provide some preliminaries in Section 2. Subsequently, we investigate this problem from the perspective of RL and primal-dual optimization in Section 3 and Section 4, respectively. Then, in Section 5, we show that the algorithms developed from the above two perspectives are equivalent in the sense that there is a corresponding relation in the involved iterations. Next, we provide a simulation example in Section 6. Some concluding remarks are finally given in Section 7.

Notations: We use $I_n$ and $0_n$ to denote an identity matrix and a null matrix in $\mathbb{R}^{n \times n}$, respectively. Denote by $0_n$, a zero vector of $n$ dimensions. A set of symmetric matrices of $n \times n$ dimensions is denoted by $\mathbb{S}_n$. $\mathbb{S}_n^+$ ($\mathbb{S}_n^-$) is a set of symmetric positive (negative) definite matrices. For a matrix $M$, we use $M_{ij}$ to denote the element in the $i$-th row and the $j$-th column. $\text{Tr}(M)$ is the trace of a matrix $M$, and $\rho(M)$ denotes its spectral radius. A symmetric positive (semi)-definite matrix $M$ is denoted by $M \succeq 0$ ($M \succeq 0$).

For a symmetric matrix $M \in \mathbb{R}^{n \times n}$, $\text{vech}(M) \in \mathbb{R}^{\frac{n(n+1)}{2}}$ denotes a row vector consisting of $n$ diagonal entries and $n(n - 1)/2$ non-diagonal terms $M_{ij}$, $i = 1, \ldots, n$, $i < j \leq n$. Specifically, $\text{vech}(M) = [M_{11}, \ldots, M_{1n}, M_{12}, \ldots, M_{1n}, M_{23}, \ldots, M_{2n}, \ldots, M_{n-1,n}]$. For a symmetric matrix $M \in \mathbb{R}^{n \times n}$, $\text{vecs}(M) \in \mathbb{R}^{n(n+1)/2}$ denotes a column vector composed of $n$ diagonal entries of...
For any stabilizing state-feedback gains for the matrix pair \((1)\) evolving along the dynamical system \(u\) feedback control law, that is, 
\[
N_u \succeq (1)
\]
the initial state is denoted by \(z_0 = z\), which is assumed to be a random variable. In this paper, we consider the state-feedback control law, that is, \(u_k = F x_k\), where \(F\) is the feedback gain. \(x_k(F, z)\) denotes the solution to the dynamical system (1) under the control input \(u_k = F x_k\) while starting from \(x_0 = z\). It is assumed that the noise \(w_k\) is independent of the initial state as well as the noise \(w_c\) for \(k \neq k'\).

Under the state-feedback control law \(u_k = F x_k\), the cost function is denoted by
\[
J(F, z) = \sum_{k=0}^{\infty} \gamma^k \mathbb{E} \left[ x_k(F, z)' \begin{bmatrix} A_x & B u_k & \gamma \end{bmatrix} x_k(F, z) \right]\end{equation}

where \(\gamma \in (0, 1)\) is the discount factor, and \(A \doteq \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \succeq 0\).

Define \(T \doteq \{ F \in \mathbb{R}^{m \times n} : \rho(A + BF) \leq 1 \}\), which is the set of stabilizing state-feedback gains for the matrix pair \((A, B)\). Now, we are ready to formally give the problem of interest, which is called the stochastic LQR problem.

**Problem 1 (Stochastic LQR Problem)** Suppose that the initial state \(z\) satisfying \(z \doteq \mathbb{E}(zz') > 0\). The stochastic LQR problem aims to find the optimal feedback gain \(F^*\) minimizing the quadratic cost function (2) with the system state evolving along the dynamical system (1). That is,
\[
\min_{F \in T} J(F, z) \quad \text{s.t.} \quad x_{k+1} = A x_k + B u_k + w_k.
\]

The optimal feedback gain \(F^*\) is defined as \(F^* \doteq \arg \min_{F \in T} J(F, z)\). The corresponding optimal cost for a given initial state \(z \in \mathbb{R}^n\) is denoted by \(J^*(z) \doteq J(F^*, z)\).

The followings are some assumptions that will be used in the remaining contents.

**Assumption 1** Assume that
1. \((A, B)\) is not known;
2. An initial stabilizing feedback gain \(F_{stab} \in T\) is known;
3. \(Q \succeq 0\) and \(R > 0\);
4. The matrix pair \((A, B)\) is stabilizable;
5. The matrix pair \((A, C)\) is detectable, where \(C\) is a matrix such that \(CC' = Q\).

For an stabilizing feedback gain \(F \in T\), define the value function as
\[
V(F, x_k) \doteq \sum_{k=0}^{\infty} \gamma^k \mathbb{E} \left[ x_k'(Q + FRF)x_k \right].
\]

The optimal value function is given by
\[
V(F^*, x_k) \doteq \min_{F \in T} V(F, x_k).
\]

**Lemma 1 (26, Lemma 3)** For any stabilizing state-feedback gain \(F \in T\), the stochastic LQR problem, i.e., Problem 1, is well-posed and the corresponding value function is
\[
V(F, x_k) = \mathbb{E}[x_k'X x_k] + \frac{\gamma}{1-\gamma} \text{Tr}(X \sigma_x'^2),
\]
where \(X \in \mathbb{S}^n_+\) is the unique solution to the Lyapunov equation
\[
X = \gamma(A + BF)'X(A + BF) + F'RF + Q.
\]

From the definition of the value function (3), one has
\[
V(F, x_k) = \mathbb{E}[x_k'(Q + F'RF)x_k] + \gamma V(F, x_k+1)
\]
which is called the Bellman equation. Define the Hamiltonian as
\[
H(F, x_k) \doteq \mathbb{E}[x_k'(Q + F'RF)x_k] + \gamma \mathbb{E}(x_k'(Q + F'RF)x_k)
\]
- \(\mathbb{E}(x_k'X x_k) - \gamma \text{Tr}(X \sigma_x'^2).

The first-order necessary condition for optimality gives
\[
\frac{\partial H(F, x_k)}{\partial F} = 2(R + \gamma B'X B)F \mathbb{E}[x_k x_k'] + 2\gamma B'X A \mathbb{E}[x_k x_k'] = 0.
\]

It follows that the unique optimal control gain is given by
\[
F^* = -\gamma(R + \gamma B'X B)^{-1}B'X A
\]
with \(X^* = Q + \gamma A'X^* A - \gamma^2 A'X^* B(R + \gamma B'X B)^{-1}B'X^* A\).

By substituting the optimal feedback gain (8) into the value function (7), one can obtain that
\[
V(F^*, x_k) = \mathbb{E}\left( \begin{bmatrix} x_k \\ F' x_k \end{bmatrix} x_k \right) + \frac{\gamma}{1-\gamma} \text{Tr}(X^* \sigma_x'^2),
\]
where
\[
P^* \doteq \begin{bmatrix} Q + \gamma A'X^* A & \gamma A'X^* B \\ \gamma B'X^* A & R + \gamma B'X B \end{bmatrix}.
\]

In the following analysis, let \(P_{11}^* \doteq Q + \gamma A'X^* A\), \(P_{12}^* \doteq (P_{11}^*)' \doteq \gamma A'X^* B\), and \(P_{22}^* \doteq R + \gamma B'X B\).

Based on the above analysis, we know that calculating the optimal feedback gain \(F^*\) requires both the system matrix \(A\) and the control input matrix \(B\). In the following two sections, we provide two approaches from a RL-based view and an optimization-based view, respectively, to compute the optimal feedback gain without using the matrices \(A\) and \(B\).

**3 RL-Based Method**

The goal of this section is to solve Problem 1 by designing RL algorithms.
Theorem 1

**Algorithm 1 Classical PI Algorithm**

1: **Initialization:** set a stabilizing control gain $F^0$, $s = 0$, and $\epsilon > 0$;
2: **Repeat**
3: **policy evaluation:** solve $X^s$ from
   \[
   X^s = \gamma(A + BF^s)'X'(A + BF^s) + (F^s)'RF^s + Q; \tag{12}
   \]
4: **policy improvement:** update control gain from
   \[
   F^{s+1} = -\gamma(R + \gamma B'X'B)^{-1}B'X'A; \tag{13}
   \]
5: $s \leftarrow s + 1$;
6: **Until** $\|F^s - F^{s-1}\| \leq \epsilon$;
7: **Return** $F^s$.

3.1 Classical PI Algorithm

Note that by replacing $(A, B)$ with $(\gamma^{1/2}A, \gamma^{1/2}B)$, the ARE (9) is identical to that obtained from a standard LQR problem [27, Chapter 2.1.4], in which the dynamical system is not influenced by the stochastic noise. There have been various algorithms to seek the solutions to the ARE, one of which is the policy iteration algorithm. A classical PI algorithm proposed in [27, Chapter 2.2] to solve (9) is given below, cf. Algorithm 1.

The iterations (12) and (13) in Algorithm 1 rely upon repeatedly solving the Lyapunov equation (6). Hence, Algorithm 1 is an offline algorithm that requires complete knowledge of the system matrices $A$ and $B$. Note that Algorithm 1 is also called Hewer’s algorithm, and it has been shown to converge to the solution of the ARE (9) in [28].

3.2 MF-OPPI Algorithm

In this subsection, we intend to propose an off-policy algorithm that can calculate the optimal feedback gain (8) without using the system matrices.

To this end, we first rewrite the original system (1) as

\[
x_{k+1} = (A + BF^s)x_k + B(u_k - F^s x_k) + w_k, \tag{14}
\]

where $u_k^* \not\equiv F^s x_k$ is the target policy being learned and updated by iterations, and $u_k$ is the behavior policy that is exerted on the system to generate data for learning.

By using (14) and the Lyapunov equation (6), one can obtain that

\[
\mathbb{E}[x_k'X'x_k] - \gamma\mathbb{E}[x_{k+1}'X'x_{k+1}] = \mathbb{E}[x_k'((Q + (F^s)'RF^s)x_k + w_k)] - 2\gamma\mathbb{E}[(u_k - F^s x_k)'B'X'Ax_k] - \gamma\mathbb{E}[(u_k - F^s x_k)'B'X'B(F^s x_k + u_k)],
\]

which is termed the off-policy Bellman equation.

Then, we use (15) for policy evaluation, and propose an MF-OPPI algorithm, i.e., Algorithm 2.

**Theorem 1** In Algorithm 2, it holds that $\lim_{s \to \infty} X^s = X^*$ and $\lim_{s \to \infty} F^s = F^*$, where $X^*$ and $F^*$ are defined in (9) and (8), respectively.

\footnote{The stochastic LQR problem is said to be well-posed if the optimal value function satisfies $-\infty \leq V(F^*, x_k) \leq +\infty$.}

**Algorithm 2 MF-OPPI Algorithm**

1: **Initialization:** set a stabilizing control gain $F^0$, $s = 0$, and $\epsilon > 0$;
2: **Repeat**
3: **policy evaluation:** solve $X^s$, $X_1^s$, and $X_2^s$ from
   \[
   \mathbb{E}[x_k'X_k'x_k] - \gamma\mathbb{E}[x_{k+1}'X'x_{k+1}] = \mathbb{E}[x_k'(Q + (F^s)'RF^s)x_k] - 2\gamma\mathbb{E}[(u_k - F^s x_k)'X_1'[x_k'; x_k' + u_k];
   \]
   \[
   - \gamma\mathbb{E}[(u_k - F^s x_k)'X_2'[F^s x_k + u_k]'] - \gamma\mathbb{E}[X'x_k + u_k]'] - \gamma\mathbb{E}[X'x_k + u_k]); \tag{16}
   \]
4: **policy improvement:** update control gain from
   \[
   F^{s+1} = -\gamma(R + \gamma X_2')'X_1'; \tag{17}
   \]
5: $s \leftarrow s + 1$;
6: **Until** $\|F^s - F^{s-1}\| \leq \epsilon$;
7: **Return** $F^s$.

**Proof.** Substituting (14) into the left-hand side of (16), it follows that

\[
\mathbb{E}[x_k'X_k'x_k] - \gamma\mathbb{E}[x_{k+1}'X'x_{k+1}] = \mathbb{E}[x_k'(Q + (F^s)'RF^s)x_k] - 2\gamma\mathbb{E}[(u_k - F^s x_k)'B'X'Ax_k] - \gamma\mathbb{E}[(u_k - F^s x_k)'B'X'B(F^s x_k + u_k)].
\]

Therefore, it holds that

\[
\mathbb{E}[x_k'X_k'x_k] - \gamma\mathbb{E}[(A + BF^s)x_k + w_k)'X'(A + BF^s)x_k + w_k]) = \mathbb{E}[x_k'(Q + (F^s)'RF^s)x_k] - \gamma\mathbb{E}[X'x_k'] - \gamma\mathbb{E}[X'x_k'] - \gamma\mathbb{E}[(u_k - F^s x_k)'B'X'B(F^s x_k + u_k)].
\]

From the dynamics (1), it holds that for $k \geq 1$,

\[
\mathbb{E}[x_k'X_k'x_k] - \gamma\mathbb{E}[x_{k+1}'X'x_{k+1}] = \mathbb{E}[x_k'(Q + (F^s)'RF^s)x_k]. \tag{18}
\]

From (15) and (16), we have $X_1^s = B'X'^*A$ and $X_2^s = B'X'^*B$, which further imply that (17) is identical to (13).

3.3 Implementation of Algorithm 2

To implement Algorithm 2, a natural question is how to solve the matrices $X^s$, $X_1^s$, and $X_2^s$ from (16). In this subsection, we provide an effective way for the implementation of Algorithm 2, where a numerical average is adopted to approximate the mathematical expectation and a batch least square (BLS) method [29] is employed to estimate the unknown matrices.

Specifically, by vectorization, the policy evaluation step (16) in Algorithm 2 becomes
\[ \mathbb{E} \left[ \text{vech}(x_k'X_k') - \gamma \text{vech}(x_{k+1}'X_{k+1}') \right] + \gamma \text{vech}(\sigma^2_{u_k}) \text{vecs}(X^s) + 2\gamma \mathbb{E} \left[ \text{vecn}(u_k - F^s x_k')) \right] \text{vecn}(X^s) + \gamma \mathbb{E} \left[ \text{vech}[\{(u_k - F^s x_k')(u_k + F^s x_k')\}] \right] \text{vecs}(X^s) \]
\[ \equiv \mathbb{E} \left[ x_k'Q + (F^s)'RF^s x_k \right] . \quad (20) \]

The equation (20) has \( \frac{(n+1)m}{n+1} + mn + \frac{(m+1)}{m+1} \) unknown parameters. Thus, at least \( K \geq \frac{(n+1)m}{n+1} + mn + \frac{(m+1)}{m+1} \) sample points are required to solve \( X^s, \tilde{X}^1, \) and \( \tilde{X}^2 \) from (20). Sample the data generated under the behavior policy \( u_k \) for \( K \) time steps, and define
\[ \Phi^s \triangleq \begin{bmatrix} \mathbb{E}[x_k'(Q + (F^s)'RF^s)x_k] \\ \mathbb{E}[x_{k+1}'(Q + (F^s)'RF^s)x_{k+1}] \\ \vdots \\ \mathbb{E}[x_{k-K}'(Q + (F^s)'RF^s)x_{k-K}] \end{bmatrix} , \quad (21) \]
\[ \Psi^s \triangleq \begin{bmatrix} \mathbb{E}[H_{s,1}] & \mathbb{E}[H_{u,1}] & \mathbb{E}[H_{w,1}] \\ \vdots & \vdots & \vdots \\ \mathbb{E}[H_{s,K}] & \mathbb{E}[H_{u,K}] & \mathbb{E}[H_{w,K}] \end{bmatrix} , \quad (22) \]

with
\[ H_{s,i} \triangleq \text{vech}(x_{i+1}'x_{i+1}') - \gamma \text{vech}(x_i'x_i'), \]
\[ H_{u,i} \triangleq 2\gamma \text{vech} \left[ (u_{i+1} - F^s x_{i+1}')(x_{i+1}'), \right] , \]
\[ H_{w,i} \triangleq \gamma \text{vech} \left[ (u_{i+1} - F^s x_{i+1}')(u_{i+1} + F^s x_{i+1}') \right] . \]

From (20), (21), and (22), one can obtain
\[ \Psi^s \left[ \left( \text{vecs}(X^s) \right)' \left( \text{vecs}(X^s) \right)' \left( \text{vecs}(X^s) \right)' \right] = \Phi^s , \]
which can be solved by
\[ \left[ \left( \text{vecs}(X^s) \right)' \left( \text{vecs}(X^s) \right)' \left( \text{vecs}(X^s) \right)' \right] \equiv \left[ \left( \Psi^s \right)' \right]^{-1} \left( \Psi^s \right)^\prime \Phi^s . \quad (23) \]

Note that to guarantee the solvability of the BLS estimator (23), a probing noise is required to add into the behavior policy \( u_k \) to guarantee the invertibility of \((\Psi^s)'\Psi^s\). It is worthwhile to point out that adding a probing noise does not influence the iteration results of Algorithm 2, which can be regarded as an advantage of the off-policy algorithm compared with the on-policy one [24]. The detailed implementation of Algorithm 2 is given in Algorithm 3.

Algorithm 3 includes two main phases. In the first phase, we exert the behavior policy on the real system for \( K \) time steps, and sample the resulting system states and control actions. Then, a numerical average of \( N \) trajectories is used to calculate the mathematical expectation. In the second phase, we implement policy evaluation and policy improvement steps by reusing the data sampled in the first phase. The reuse of data is an important advantage of the proposed MF-OPPI algorithm when compared with the on-policy RL algorithms, since the latter are required to collect the state-input data when applying every iterated control policies [5, 30]. Therefore, the proposed MF-OPPI algorithm is able to alleviate the data-hungry problem to some extent.

Algorithm 3 Implementation of MF-OPPI Algorithm

1: **Phase 1 (Data Collection):**
2: **Initialization:** set a stabilizing control gain \( F^0, s = 0, K > 0, \) and \( \epsilon > 0; \) let \( E_{x_k,u_k} = 0, E_{u_k,u_k} = 0, \) and \( E_{u_k,u_k} = 0 \) for \( k = 0, \ldots, K; \)
3: **For** \( q = 1 : N \)
4: apply the behavior policy \( u_k \) for \( K \) time steps;
5: sample \( \{x_k, u_k\}_{t=0}^{k-1} ; \)
6: \( E_{x_k} = E_{x_k,u_k} = x_k', k = 0, \ldots, K; \)
7: \( E_{u_k} = E_{u_k,u_k} = u_k', k = 0, \ldots, K; \)
8: \( E_{u_k,u_k} = E_{u_k,u_k} = u_k'u_k', k = 0, \ldots, K; \)
9: **End**
10: \( E_{x_k} = E_{x_k,n}/N, E_{u_k} = E_{u_k,n}/N, E_{u_k,u_k} = E_{u_k,u_k}/N, k = 0, \ldots, K; \)
11: **Phase 2 (Reuse of Collected Data for Iteration):**
12: **Initialization:** set a stabilizable control gain \( F^0, s = 0, \) and \( \epsilon > 0; \)
13: **Repeat**
14: calculate \( \Phi^s \) and \( \Psi^s \) using \( F^s \) and collected data;
15: **policy evaluation:** solve \( X^s, \tilde{X}^1, \) and \( \tilde{X}^2 \) from
\[ \left[ \left( \text{vecs}(X^s) \right)' \left( \text{vecs}(X^s) \right)' \left( \text{vecs}(X^s) \right)' \right] = \left[ \left( \Psi^s \right)' \right]^{-1} \left( \Psi^s \right)^\prime \Phi^s , \]
16: **policy improvement:** update control gain from
\[ F^{s+1} = -\gamma (R + \gamma D^s)' A'; \]
17: \( s \leftarrow s + 1; \)
18: **Until** \( \|F^s - F^{s-1}\| \leq \epsilon \)
19: Return \( F^s; \)

4 Optimization-Based Method

In this section, we reformulate Problem 1 as a constrained optimization problem, and resolve it from the primal-dual optimization perspective.

4.1 Problem Reformulation

For technical reasons, we need to make some modifications on Problem 1. To do this, some new notations are required.

We introduce the augmented state vector \( v_k \triangleq \left[ x_k' u_k' \right] \), and the resulting augmented system is described by
\[ v_{k+1} = A_F v_k + F w_k , \quad (24) \]
where \( A_F \triangleq \begin{bmatrix} A & B \\ FA & FB \end{bmatrix} \in \mathbb{R}^{(n+m)(n+m)} \) and \( F \triangleq \begin{bmatrix} I_n \\ F \end{bmatrix} \in \mathbb{R}^{(n+m)\times n} . \)

The following is a useful property that will be used in the remaining part of this paper.

**Lemma 2 (Property of Spectral Radius [21, Lemma 1])**

It holds that \( p(A + BF) = p(AF) \).

Let \( z_i \in \mathbb{R}^m \) for \( i \in \{1, \ldots, r\} \) be different samples from the random variable \( z \). Let \( u_i \in \mathbb{R}^m \) for \( i \in \{1, \ldots, r\} \) be different initial stabilizing control laws. Different initial states for the augmented system (24) are denoted by \( v_i = \left[ x_i' u_i' \right] \) for \( i \in \{1, \ldots, r\} \). Denote by \( v_k(F, v_i) \) the solution
of (24) starting from the initial state \( v(i) \) and evolving under the control \( u(t) = F \cdot x_k, k \geq 1 \).

A new cost function is defined as

\[
\hat{J}(F,v(i)) = \sum_{k=0}^{\infty} \gamma^k \mathbb{E} \left[ v_k(F,v(i))' \Lambda v_k(F,v(i)) \right].
\]  

(25)

Note that different from the cost function \( J(F,z) \) defined in (2), where the initial state-feedback control law \( u(0) = Fz \) is considered, the initial control \( u(0) \) used in \( \hat{J}(F,v(i)) \) is arbitrary, as long as it is stabilizable.

The new problem is formally formulated as below.

**Problem 2 (Modified Stochastic LQR Problem)** Suppose that \( z(i) \in \mathbb{R}^n, u(i) \in \mathbb{R}^m, \) and \( v(i) = [z'_i, u'_i]' \) for \( i \in \{1, \ldots, r\} \), are chosen such that \( \frac{1}{r} \sum_{i=1}^{r} v(i)' = \Gamma > 0 \), with \( r \) being a positive constant. The following problem,

\[
\begin{align*}
\min_{F \in \mathcal{F}} & \quad \frac{1}{r} \sum_{i=1}^{r} J(F,v(i)) \\
\text{s.t.} & \quad x_{k+1} = Ax_k + Bu_k + w_k,
\end{align*}
\]

is called a modified stochastic LQR problem. The corresponding optimal feedback gain is defined as \( \hat{F}^* = \arg\min_{F \in \mathcal{F}} \frac{1}{r} \sum_{i=1}^{r} J(F,v(i)) \). The resulting optimal value is denoted by \( \hat{J}^* = \frac{1}{r} \sum_{i=1}^{r} J(F,v(i)) \).

**Proposition 1** The optimal solution \( \hat{F}^* \) obtained by solving Problem 2 is unique, and it is identical to \( F^* \).

**Proof.** From the results in Section 2, we know that although \( J^*(F,z) \) has different values under different initial state \( z \), the optimal control gain \( F^* = \arg\min_{F \in \mathcal{F}} J(F,z) \), whose explicit form is given in (8), does not rely upon \( z \). It follows that \( \arg\min_{F \in \mathcal{F}} J(F,z) = \arg\min_{F \in \mathcal{F}} \frac{1}{r} \sum_{i=1}^{r} J(F,v(i)) \) for any initial states \( z \) and \( i \in \{1, \ldots, r\} \), satisfying \( z_i > 0 \).

Based on the definitions of \( J(\cdot,\cdot) \) and \( \hat{J}(\cdot,\cdot) \), an algebraic manipulation leads to

\[
\frac{1}{r} \sum_{i=1}^{r} J(F,v(i)) = \frac{1}{r} \sum_{i=1}^{r} J(F,z(i)) + \frac{1}{r} \sum_{i=1}^{r} \sum_{k=1}^{\infty} \gamma^k \mathbb{E} \left[ v_k(F,v(i))' \Lambda v_k(F,v(i)) \right]
\]

where \( z_i = [A B]v(i) \). Since the last two terms on the right-hand side of the above equation are constants, it holds that \( \arg\min_{F \in \mathcal{F}} \frac{1}{r} \sum_{i=1}^{r} J(F,z) = \arg\min_{F \in \mathcal{F}} \frac{1}{r} \sum_{i=1}^{r} J(F,v(i)) \).

From the above, it follows that \( F^* = \arg\min_{F \in \mathcal{F}} \frac{1}{r} \sum_{i=1}^{r} J(F,z) = \arg\min_{F \in \mathcal{F}} \frac{1}{r} \sum_{i=1}^{r} J(F,v(i)) = \hat{F}^* \). Moreover, the uniqueness of \( F^* \) guarantees that \( \hat{F}^* \) is unique.

The followings are two conclusions that will be frequently used in the following contents.

**Lemma 3 (Lyapunov Stability Theorems** [31, Chapter 3]) Let \( A \in \mathbb{R}^{nxn} \). Then,

1. If \( \rho(A) < 1 \), then for each given matrix \( M \in \mathbb{S}^n_+ \), there exists a unique matrix \( P \in \mathbb{S}^n_+ \) satisfying \( A'PA + M = P \);
2. If \( \rho(A) < 1 \) and only if each given matrix \( M \in \mathbb{S}^n_+ \), there exists a unique matrix \( P \in \mathbb{S}^n_+ \) such that \( A'PA + M = P \).

4.2 Primal-Dual Formulation

From the previous subsection, we know that Problem 2 is a non-convex optimization problem with the objective function \( \frac{1}{r} \sum_{i=1}^{r} \hat{J}(F,v(i)) \), the static constraint \( F \in \mathcal{F} \), and the dynamic constraint (1). In this subsection, we propose an equivalent non-convex optimization formulation of Problem 2, and show that its dual gap is zero.

We first give the definition of the non-convex optimization formulation.

**Primal Problem 1** Solve

\[
\begin{align*}
J_p & = \inf_{S \in \mathbb{R}^{mn}, F \in \mathcal{F}, \sigma^2} \text{Tr}(AS) \\
\text{s.t.} & \quad S > 0,
\end{align*}
\]

(26)

\[
\gamma A_F S A_F' + \Gamma + \gamma \hat{F}^* = S.
\]

(27)

It is obvious to see that Primal Problem 1 is a single-objective multi-variable optimization problem consisting of a linear objective function, a linear inequality constraint, and a quadratic equality constraint [32]. Hence, it is a non-convex optimization problem.

The following proposition shows the equivalence between Problem 2 and Primal Problem 1.

**Proposition 2** The optimal solution of Primal Problem 1, which is denoted by \( (S_p, F_p) \), is unique. Furthermore, Primal Problem 1 is equivalent to Problem 2 in the sense that \( J_p = \hat{J}^* \) and \( F_p = \hat{F}^* \).

**Proof.** By the properties of trace, we rewrite the objective function of Problem 2 as

\[
\frac{1}{r} \sum_{i=1}^{r} \hat{J}(F,v(i)) = \text{Tr}(AS)
\]

with

\[
S = S(F) = \frac{1}{r} \sum_{i=1}^{r} \sum_{k=1}^{\infty} \gamma^k \mathbb{E}[v_k(F,v(i))'v_k(F,v(i))].
\]

(28)

From (24), we have

\[
v_k(F,v(i)) = A_F v(i) + \sum_{j=0}^{k-1} A'_F \hat{F} w_{k-j-1}, k \geq 1.
\]

(29)

Using the equation (29) and the fact that \( w_k \sim N(0_n, \sigma_k^2) \) is the i.i.d. random noise for each \( k \geq 0 \), we can obtain that for any \( F \in \mathcal{F} \) and \( k \geq 1 \), it holds

\[
\mathbb{E}[v_k(F,v(i))'v_k(F,v(i))'] = \sum_{j=0}^{k-1} A'_F \hat{F} \sigma_k^2 \hat{F}^* (A'_F)^{'}. \]

Thus, \( S \) in (28) becomes

\[
S = \frac{1}{r} \sum_{i=1}^{r} \sum_{k=1}^{\infty} \gamma^k \left[ A'_F v(i)'v(i) + \sum_{j=0}^{k-1} A'_F \hat{F} \sigma_k^2 \hat{F}^* (A'_F)^{'}. \right]
\]

(30)

By an algebraic manipulation, one can obtain that
\[ S - \gamma A_F S A_F' = \Gamma + \sum_{k=1}^{\infty} \gamma^k \bar{F} \sigma_w^2 \bar{F}' = \Gamma + \frac{\gamma}{1-\gamma} \bar{F} \sigma_w^2 \bar{F}', \]

which implies that \( S \) satisfies the Lyapunov equation (27).

Since \( \Gamma + \frac{\gamma}{1-\gamma} \bar{F} \sigma_w^2 \bar{F}' > 0 \), it follows from Lemma 2 and the second statement of Lemma 3 that \( F \in \mathcal{F} \) if and only if there exists a unique \( S > 0 \) such that the constraint (27) holds. Therefore, we can replace the constraint \( F \in \mathcal{F} \) in Problem 2 by \( S > 0 \) without changing its optimal solution.

From the above discussions, we know that if \((S_p, F_p)\) is the optimal solution of Primal Problem 1 and the corresponding optimal cost is \( J_p \), then it holds that \( S_p > 0 \) and \( \rho(A_{F_p}) < 1 \). From Lemma 2, \( F_p \in \mathcal{F} \) and \( F_p \) is a feasible point of Problem 2. Therefore, \( J_p \) is lower bounded by the optimal value of Problem 2, i.e., \( J_p \geq \hat{J} \). In addition, if \( S_p \) is the unique solution of (27) with \( F_p = \hat{F} \in \mathcal{F} \), then the resulting objective function of Primal Problem 1 is \( J_p = \hat{J} \). Thus, we can conclude that \((S_p, F_p)\) is the solution of Primal Problem 1 and \( J_p = \hat{J} \). The uniqueness of \((S_p, F_p)\) can be shown by the uniqueness of \( \hat{F} \).

For any fixed \( P \in \mathbb{S}^{n+m}, \) \( P_0 \in \mathbb{S}^{n+m} \), define the Lagrangian function as

\[
L(P, P_0, F, S) = \text{Tr}(AS) + \text{Tr}(-SP_0) + \text{Tr}\left( \left[ A \frac{\gamma}{1-\gamma} \bar{F} \sigma_w^2 \bar{F}' - S \right] P_0 \right),
\]

where \( P \) and \( P_0 \) are Lagrangian multipliers. The Lagrangian dual function is defined as

\[
d(P, P_0) = \inf_{S \in \mathbb{S}^{n+m}, F \in \mathcal{F}} L(P, P_0, F, S).
\]

Then, the dual problem corresponding to Primal Problem 1 is defined as

\[
J_d = \sup_{P \in \mathbb{S}^{n+m}, P_0 \in \mathbb{S}^{n+m}} d(P, P_0) = \sup_{P \in \mathbb{S}^{n+m}, P_0 \in \mathbb{S}^{n+m}} \inf_{S \in \mathbb{S}^{n+m}, F \in \mathcal{F}} L(P, P_0, F, S),
\]

where \( L(P, P_0, F, S) \) is as defined in (31).

The weak duality [32, Chapter 5] guarantees that \( J_d \leq J_p \) holds, where \( J_p - J_d \geq 0 \) is called the duality gap. When the duality gap is zero, we say that the optimization problem has strong duality [32, Chapter 5]. The following theorem shows that Primal Problem 1 has strong duality.

**Theorem 2 (Strong Duality)** The strong duality holds for Primal Problem 1, i.e., \( J_d = J_p \).

**Proof.** See Appendix A.

### 4.3 MB-PD Algorithm

It is well-known that for a non-convex optimization problem, if its objective and constraints functions are differentiable and it has strong duality, then any primal and dual optimal points must satisfy the corresponding KKT conditions, but not vice versa [32, Chapter 5.5.3]. However, if the solution to the KKT conditions is unique, then this solution must be the primal and dual optimal points. In view of this, we first derive the KKT conditions and show the uniqueness of their solutions in this subsection, and then design an algorithm to seek the solution to Problem 2 according to the solutions to the KKT conditions.

**Lemma 4** Suppose that \((S_p, F_p)\) is the primal optimal point and \((P_{0p}, P_{0p})\) is the dual optimal point of Primal Problem 1. Then, \((S_p, F_p, P_{0p}, P_{0p})\) satisfies the KKT conditions for \((S, F, F, P_0)\)

\[
\gamma A_F S A_F' + \Gamma + \frac{\gamma}{1-\gamma} \bar{F} \sigma_w^2 \bar{F}' - S = 0, \quad S > 0;
\]

\[
P_0 = 0_{m+n},
\]

\[
\gamma A_F' P A_F + \Lambda = P.
\]

**Proof.** From the conclusions in [32, Chapter 5.9.2], the KKT conditions of Primal Problem 1 are as follows:

1. **primal feasibility condition:**
   \[
   \gamma A_F S A_F' + \Gamma + \frac{\gamma}{1-\gamma} \bar{F} \sigma_w^2 \bar{F}' - S = 0, \quad S > 0;
   \]

2. **dual feasibility condition:**
   \[
P_0 \geq 0;
   \]

3. **complementary slackness condition:**
   \[
   \text{Tr}(SP_0) = 0;
   \]

4. **stationary conditions:**
   \[
   \frac{\partial L}{\partial S} = \Lambda + \gamma A_F S A_F' - P - P_0 = 0,
   
   \frac{\partial L}{\partial F} = 2(P'_{12} + P_{22} F)(\gamma[A B] S [A B]' + \frac{\gamma}{1-\gamma} \sigma_w^2) = 0.
   \]

By the second statement of Lemma 3, \( \Gamma + \frac{\gamma}{1-\gamma} \sigma_w^2 > 0 \) and \( F \in \mathcal{F} \) guarantee that \( S > 0 \), which further implies that the only solution to the equation \( \text{Tr}(SP_0) = 0 \) is \( P_0 = 0_{m+n} \). Hence, the KKT conditions in (32)-(36) can be obtained. According to [32, Chapter 5.5.3], the strong duality guarantees that the solutions to the KKT conditions must be the primal and dual optimal points.

The following lemma is given to show that the uniqueness of the solutions to the KKT conditions (32)-(36), which guarantees that the solutions satisfying the KKT conditions must be the primal and dual optimal points.

**Lemma 5** Suppose that \((S, F, P, P_0)\) satisfies the KKT conditions (32)-(36). Then, \((S, F, P, P_0)\) is unique. Furthermore, it holds that \( F = F_p, P = P_p, P_0 = 0_{m+n} \), and \( S_p \) is the unique solution to (32) with \( F = F_p \) and \( P = P_p \).

**Proof.** By observation, we can know that the sufficient condition for (36) is \( F = -(P_{22})^{-1} P_{12}' \). To show this lemma, we first show that \((F, P)\) satisfying \( \gamma A_F' P A_F + \Lambda = P \), \( F = -(P_{22})^{-1} P_{12}' \), and \( P \geq 0 \) is exactly \( (F_p, P_p) \) with \( F_p \in \mathcal{F} \). Using the same argument as in the proof of Lemma 7, we have \( P = P^* \). From Proposition 1 and Proposition 2, it holds that \( F_p = F^* \). In view of this, we only need to show that \((F, P)\) satisfying \( \gamma A_F' P A_F + \Lambda = P \), \( F = -(P_{22})^{-1} P_{12}' \), and \( P \geq 0 \) is exactly \( (F^*, P^*) \).

By substituting \( F = -(P_{22})^{-1} P_{12}' \) into (35), it follows that

\[
\gamma[A B]'(P_{11} - P_{12} P_{22}^{-1} P_{12}')[A B] + \Lambda = P.
\]

Define \( \tilde{X} = P_{11} - P_{12} P_{22}^{-1} P_{12}' \). The above equation becomes

\[
\gamma[A B]' \tilde{X}[A B] + \Lambda = P,
\]

the expansion of which can be written as
Algorithm 4 MB-PD Algorithm

1: Initialization: $F_p^0 \in \mathcal{F}$, $\varepsilon > 0$, and set $s = 0$;
2: Repeat
3:  dual update: solve $P^s$ from
4:  \[
        \gamma (A_F) P^s A_F + \Lambda = P^s;
        \]
5:  primal update: $F_p^{s+1} = -(P_{p,22}^s)^{-1}(P_{p,12}^s)';$
6:  $s \leftarrow s + 1$;
7: Until $\|F_p^s - F_p^{s-1}\| \leq \varepsilon$;
8: Return $F_p^s$.

\[
\begin{bmatrix}
\gamma A'X + Q & \gamma A'XB \\
\gamma B'X + \gamma B'XB + R & P_{11} P_{12}
\end{bmatrix} = \begin{bmatrix}
P_{11} & P_{12} \\
P_{12}' & P_{22}
\end{bmatrix},
\]

i.e., $P_{11} = \gamma A'X + Q$, $P_{12} = \gamma A'XB$, and $P_{22} = \gamma B'XB + R$. Substituting these expressions into the definition $\bar{X} \approx P_{11} - P_{12}P_{22}^{-1}P_{12}'$ yields

\[
Q + \gamma A'X - \gamma A'XB(R + \gamma B'XB)^{-1}\gamma B'X = \bar{X}.
\]

This is exactly the ARE (9), which has a unique solution. Therefore, it holds that $\bar{X} = X'$. From the definitions (8) and (11), we have $F = F'$ and $P = P'$.

Then, based on the fact that $F_p \approx F^*$ in $\mathcal{F}$ and $\Lambda \geq 0$, the first statement of Lemma 3 shows that there exists a unique $S = S^*$ satisfying (32) with $F = F_p$ and $P = P_p$. ■

Note that to calculate the primal optimal point $F_p$, one needs to know $P_p$, which is uniquely determined by solving (35). However, (35) is dependent upon the primal optimal point $F_p$, which means that the stationary conditions (35) and (36) are coupled with each other, so that $F_p$ and $P_p$ cannot be obtained independently. In what follows, we propose an algorithm, cf. Algorithm 4, to iteratively solve (35) and (36). Note that once $F_p$ and $P_p$ are attained, one can uniquely determine the primal variable $S_p$ by solving (32).

We have the following convergence result:

**Lemma 6** Given an initial stabilizing control gain $F_p^0$. Consider the two sequences \{$P^s\}_{s=0}^\infty$ and \{$F^s\}_{s=0}^\infty$ obtained from Algorithm 4. We have the following properties:

1. \{$P^s\}_{s=0}^\infty$ is a non-increasing sequence, that is, $P^s \geq P^{s+1}$;
2. $F^s$ obtained at each iteration step is stabilizing, that is, $F^s \in \mathcal{F}$ for each $s$;
3. $\lim_{s \to \infty} P^s = P^* = F^*$ and $\lim_{s \to \infty} F^s = F^*_p = F^*$, where $F^*$ and $F^*_p$ are defined in (11) and (8), respectively.

**Proof.** Note that Algorithm 4 can be transformed into Algorithm 1 developed in [21] by replacing $(A, B)$ with $(\gamma^{1/2}A, \gamma^{1/2}B)$. Therefore, this lemma can be shown directly by using the proof of Proposition 8 in [21] by some algebraic manipulations. ■

**4.4 Model-Free Implementation of Algorithm 4**

From the equation (28) that is defined in the proof of Proposition 2, $S(F^s_p)$ can be equivalently written as

\[
S(F^s_p) = \frac{1}{r} \sum_{i=1}^r \sum_{k=0}^\infty \gamma^k \mathbb{E}\left[ v_k(F^s_p, v_i) v_k(F^s_p, v_i)' \right].
\]

Algorithm 5 Model-Free Implementation of MB-PD Algorithm

1: Initialization: set a stabilizing control gain $F_p^0$, $s = 0$, $K > 0$, and $\varepsilon > 0$;
2: Repeat
3:  primal update: $\tilde{S} = 0$, $\tilde{W} = 0$;
4: for $i = 1 : r$
5:  $\tilde{S} = 0$, $\tilde{W} = 0$;
6: for $q = 1 : N$
7:  apply $u_k = F^*_p v_k$ for $K$ time steps;
8:  sample \{$v_k, u_k\}_{k=0}^\infty$;
9:  calculate $S = \sum_{k=0}^K \gamma^k v_k v_k'$ with $v_k = [x_k u_k]'$;
10: calculate $\tilde{W} = \sum_{k=0}^K \gamma^k v_{k+1} v_k'$;
11: $\tilde{S} = \tilde{S} + \tilde{S} + \tilde{S} + \tilde{W} = \tilde{W} + \tilde{W}$;
12: End
13: $\tilde{S} = \tilde{S}/N$, $\tilde{W} = \tilde{W}/N$;
14: $S = S + \tilde{S}$, $\tilde{W} = \tilde{W} + \tilde{W}$;
15: End
16: dual update: calculate $P^s_p$ by solving
17: $S^* P^s_p S - \gamma \tilde{W} P^s_p \tilde{W} = S^* S^*$;
18: $s \leftarrow s + 1$;
19: Until $\|F_p^s - F_p^{s-1}\| \leq \varepsilon$;
20: Return $F^s_p$.

We truncate the time horizon by $K > 0$, and define

\[
\tilde{S}(F^s_p) = \frac{1}{r} \sum_{i=1}^r \sum_{k=0}^K \gamma^k \mathbb{E}\left[ v_k(F^s_p, v_i) v_k(F^s_p, v_i)' \right].
\]

Note that for any $K > 0$, it holds that $\tilde{S}(F^s_p) > 0$ due to the positive definiteness of $\frac{1}{r} \sum_{i=1}^r \gamma^k \mathbb{E}[v_k(F^s_p, v_i)v_k(F^s_p, v_i)']$. In view of this, the stationary condition (35) holds if and only if $\tilde{S}(F^s_p)' (\gamma A_F F_p^s A_F + \Lambda - P^s_p) \tilde{S}(F^s_p) = 0$, which can be rewritten as the following linear matrix equation

\[
\tilde{S}(F^s_p)^\prime P^s_p \tilde{S}(F^s_p) - \gamma \tilde{W}(F^s_p)^\prime P^s_p \tilde{W}(F^p) = \tilde{S}(F^s_p)' \Lambda \tilde{S}(F^s_p),
\]

where

\[
\tilde{W}(F^s_p) = \frac{1}{r} \sum_{i=1}^r \sum_{k=0}^K \gamma^k \mathbb{E}\left[ v_{k+1}(F^s_p, v_i) v_k(F^s_p, v_i)' \right].
\]

Therefore, for a fixed and stabilizing control gain $F^*_p$, the corresponding dual variable $P^*_p$ can be obtained by solving the linear matrix equation (39).

Based on the above analysis, we give a detailed model-
free implementation of the proposed MB-PD algorithm, and summarise it in Algorithm 5. Similar to Algorithm 3, the mathematical expectations are approximated by the numerical averages herein.

Note that different from the Algorithm 3, the primal-dual optimization-based algorithm developed in this section has the advantage of averting the requirement of the excitation signal, which may cause undesirable system oscillation. However, this is at the cost of the requirement of more data.
Specifically, it is assumed that we can sample the state-input trajectories generated under a certain set of initial vectors which are linearly independent in $\mathbb{R}^n$.

5 Equivalency Between PI and MB-PD Algorithms

In the previous two sections, we solve the stochastic LQR problem from the perspective of RL and primal-dual optimization, respectively. The goal of this section is to show the equivalency of the algorithms developed before. Specifically, we show that classical PI and MB-PD algorithms are equivalent in the sense that there is a corresponding relation in the involved iterations. The specific relationship is described in the following theorem.

**Theorem 3** The dual and primal update steps in Algorithm 4 can be interpreted as the policy evaluation and policy improvement steps in Algorithm 1, respectively, in the sense that $X^s = (F_p^s)\gamma P_p^sF_p^s$ and $F^s = F_s$ with $F_p^s = [I_n (F_p^s)]^\gamma$.

**Proof.** By pre- and post-multiplying $(A_F)^\gamma P_{p1}\gamma A_F^\gamma + \Lambda = P_p^s$ by $[I_n (F_p^s)]^\gamma$ and its transpose, one can obtain that

$$\gamma (A + BF_p^s) (F_p^s)\gamma P_{p1} (A + BF_p^s) + Q + (F_p^s)^\gamma RF_p^s = (F_p^s)^\gamma P_p^s F_p^s$$

Define $X_p^s = (F_p^s)\gamma P_p^s F_p^s$. Then, (41) becomes

$$\gamma (A + BF_p^s) X_p^s (A + BF_p^s) + Q + (F_p^s)^\gamma RF_p^s = X_p^s,$$

which has a unique solution since $F_p^s \in \mathcal{F}$ and $Q + (F_p^s)^\gamma RF_p^s$ is positive semidefinite. By observation, we know that the dual update step in Algorithm 4 is equivalent to the policy evaluation step in Algorithm 1 in the sense that $X^s = X_p^s = (F_p^s)\gamma P_p^s F_p^s$.

Let $P_p^s = \begin{bmatrix} p_{p11}^s & p_{p12}^s \\ p_{p21}^s & p_{p22}^s \end{bmatrix}$ with $p_{p11}^s, p_{p21}^s \in \mathbb{R}^{nxn}, p_{p12}^s, p_{p22}^s \in \mathbb{R}^{nxm}$, and $p_{p21}^s = (p_{p12}^s)^\gamma$. Expanding $(F_p^s)^\gamma P_p^s F_p^s$ yields that

$$p_{p11}^s + (F_p^s)\gamma p_{p21}^s + p_{p12}^s + (F_p^s)^\gamma p_{p22}^s F_p^s = X^s.$$  (42)

Expanding the left-hand side of (12) yields that

$$Q + \gamma A'X'A + (F_p^s)^\gamma (yB'X'A) + \gamma A'X'B F_p^s + (F_p^s)^\gamma (R + \gamma B'X'B)F_p^s = X^s.$$  (43)

Making the corresponding terms in (42) and (43) identical, it follows that $p_{p11}^s = Q + \gamma A'X'A$, $p_{p12}^s = \gamma A'X'B$, and $p_{p22}^s = R + \gamma B'X'B$, which further imply that $F_p^{s+1} = F^{s+1}$. 

**Example 1** Considering the following discrete-time linear system:

$$x_{k+1} = 2x_k + u_k + w_k.$$  

The weight matrices and discount factor are selected as $Q = 1, R = 1$, and $\gamma = 0.7$. The system noise is assumed to be generated according to the Gaussian distribution $\mathcal{N}(0, 1)$. We choose $F^0 = F^p = -1$ as the initial control gain matrix. Suppose that the system matrices $A$ and $B$ are known.

By implementing the classical PI algorithm, i.e., Algorithm 1, for 4 steps, one can obtain that

$$X^1 = 6.6666, X^2 = 4.0675, X^3 = 3.9353, X^4 = 3.9345;$$

$$F^1 = -1.6471, F^2 = -1.4801, F^3 = -1.4673, F^4 = -1.4673.$$  

Then we implement the MB-PD algorithm, i.e., Algorithm 4, for 4 steps, producing that

$$P_p^1 = \begin{bmatrix} 19.6666 & 9.3333 \\ 9.3333 & 5.6667 \end{bmatrix},$$

$$P_p^2 = \begin{bmatrix} 12.3889 & 5.6945 \\ 5.6945 & 3.8472 \end{bmatrix};$$

$$P_p^3 = \begin{bmatrix} 12.0188 & 5.5094 \\ 5.5094 & 3.7547 \end{bmatrix},$$

$$P_p^4 = \begin{bmatrix} 12.0116 & 5.5083 \\ 5.5083 & 3.7542 \end{bmatrix};$$

$$F_p^1 = -1.6471, F_p^2 = -1.4801, F_p^3 = -1.4673, F_p^4 = -1.4673.$$  

From $X_p^s = (F_p^s)^\gamma P_p^s F_p^s$, it follows that

$$X_p^1 = 6.6667, X_p^2 = 4.0675, X_p^3 = 3.9353, X_p^4 = 3.9345.$$  

By simple observation and calculation, we can verify the correctness of the conclusion given in Theorem 3.

6 Simulations

In this section, we provide a numerical example to evaluate the performance of the proposed model-free algorithms. Considering the following discrete-time linear system:

$$x_{k+1} = \begin{bmatrix} 0.5 & 1 \\ 0.25 & 0.5 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u_k + w_k.$$  

The weight matrices and discount factor are selected as $Q = I_2, R = 1$, and $\gamma = 0.7$. The system noise $w_k$ is assumed to be generated according to the Gaussian distribution $\mathcal{N}(0, I_2)$. When one knows the exact system matrices $A$ and $B$, the optimal control gain $F^*$ can be obtained by solving the ARE (9). The result is $F^* = [-0.2446 - 0.4892]$.

6.1 Performance of Algorithm 3

When implement Algorithm 3, the initial state $z$ is randomly generated. Inspired by [24], the probing noise is considered as

$$e_k = 0.2 \sin(1.009k) + \cos^2(0.538k) + \sin(0.9k) + \cos(100k).$$

Initial stabilizing control gain is chosen as $F^0 = [-1 0]$. We use the numerical average of $N = 15$ trajectories to approximate the mathematical expectation, and sample $K = 20$ data points at each trajectory for the BLS method. The tolerant error is $\varepsilon = 10^{-3}$. We can observe from Fig. 1 that Algorithm 3 stops after 6 iterations and returns the estimated control gain $F^0 = [-0.2464 - 0.4960]$.  

![Fig. 1. Values of $F^s$ at each iteration step. The black dotted lines are baselines for two components of the optimal control gain $F^s$.](image-url)

Considering that the system noise is randomly generated, we use Monte-Carlo method to validate the effectiveness of Algorithm 3, as well as to show the influence of the variance.
of the system noise. We implement Algorithm 3 for \( Y = 10 \) times, and each experiment is stopped when \( s = 10 \). The average value of the resulting control gain from \( Y \) experiments is denoted by \( \bar{F}_s^* = \frac{1}{Y} \sum_{i=1}^{Y} F_i^* \), where \( F_i^* \) is the control gain obtained at the \( s \)-th step from the \( i \)-th experiment. We use \( E' = \frac{1}{Y} \sum_{i=1}^{Y} \|F_i^* - F^*\| \) to denote the average approximation error of the \( s \)-th iteration step from \( Y \) experiments. Fig. 2 shows the value of the control gain \( F_s^* \) at each experiment as well as the resulting average gain \( \bar{F}_s^* \), when the variance of the system noise is set as \( \sigma_n^2 = \alpha I_2 \) with \( \alpha \) being 0, 0.1, 0.5, and 1, respectively. Note that \( E^{10} = \frac{1}{Y} \sum_{i=1}^{Y} \|F_i^{10} - F^*\| \) denotes the average approximation error on convergence, that is, the average deviation of the obtained control gain at the \( i \)-th experiment from the optimal control gain \( F^* \). Fig. 3 shows the variation of the average approximation error \( E^{10} \), when \( \alpha \) is increased from 0 to 1. From Fig. 2 and Fig. 3, we conclude that roughly speaking, the resulting approximation errors are increased as the increase of the variance of the system noise. That is, a stronger system noise may lead to a higher control gain approximation error. Moreover, when \( \sigma_n^2 = 0 \), \( F^* \) obtained from Algorithm 3 converges to the optimal feedback gain \( F^* \).

Fig. 2. Values of \( F^* \) at each experiment and the resulting average values \( \bar{F}_s^* \). The black dotted lines are baselines for two components of the optimal control gain \( F^* \). (a) \( \sigma_n^2 = 0 \), (b) \( \sigma_n^2 = 0.1I_2 \), (c) \( \sigma_n^2 = 0.5I_2 \), (d) \( \sigma_n^2 = I_2 \).

Fig. 3. Average approximation errors of \( Y = 10 \) experiments at different values of \( \alpha \), where the variance of the system noise is \( \sigma_n^2 = \alpha I_2 \).

### 6.2 Performance of Algorithm 5

When implement Algorithm 5, we truncate the time horizon by \( K = 10 \) and let \( r = 3 \). Three pairs of initial states and initial control inputs are \( z(1) = [-1 \ 3], u(1) = -2, z(2) = [2 \ -1], u(2) = -5, z(3) = [-3 \ 3], u(3) = -8 \). The initial stabilizing control gain is chosen as \( F_p^{0} = [-1 \ 0] \). We use the numerical average of \( N = 15 \) trajectories to approximate the mathematical expectation. The desired convergence precision is set as \( \epsilon = 5 \times 10^{-3} \). We can observe from Fig. 4 that Algorithm 5 stops after 7 iterations and returns the estimated control gain \( F_p^7 = [-0.2443 \ -0.4884] \).

Fig. 4. Values of \( F_p^s \) at each iteration step. The black dotted lines are baselines for two components of the optimal control gain \( F^* \).

Similar to the previous subsection, we use Monte-Carlo method to discuss the influence of the system noise herein. We implement Algorithm 5 for \( Y = 10 \) times, and stop each experiment when \( s = 15 \). We use \( \bar{F}_s^* = \frac{1}{Y} \sum_{i=1}^{Y} F_{pi}^* \) to denote the average value of the resulting control gain from \( Y \) experiments, where \( F_{pi}^* \) is the control gain obtained at the \( s \)-th step from the \( i \)-th experiment. Correspondingly, the average deviation of the approximation error at the \( s \)-th iteration step obtained from \( Y \) experiments is denoted by \( E'_s = \frac{1}{Y} \sum_{i=1}^{Y} \|F_{pi}^* - F^*\| \). In Fig. 2, we show the value of \( F_p^s \) at each experiment as well as the resulting average value \( \bar{F}_s^* \), when the variance of the system noises is set as \( \sigma_n^2 = 0.2 \), \( \sigma_n^2 = 0.1I_2 \), \( \sigma_n^2 = 0.5I_2 \), and \( \sigma_n^2 = I_2 \), respectively. By observation, we know that in these four cases, \( F_p \) is in a small neighborhood of the optimal feedback gain after about 3 iterations, and the range of the neighborhood is increased as the increase of the variance of the system noise. This implies that when the variance of the system noise is large, Algorithm 5 may not converge under a small convergence precision. In spite of this, we can also see from Fig. 2 that the average control gain \( \bar{F}_s^* \) is close to the optimal one. Let \( E_{15} = \frac{1}{Y} \sum_{i=1}^{Y} \|F_{pi}^{15} - F^*\| \) which is the average derivation of the control gain obtained at the 15-th iteration step of the \( i \)-th experiment from the optimal control gain \( F^* \). Fig. 6 shows the values of \( E_{15} \) when \( \alpha \) is increased from 0 to 1. Roughly speaking, the resulting approximation errors are increased as the increase of the variance of the system noise.

### 7 Conclusion

In this paper, we have revisited the stochastic LQR problem, and have developed a MF-OPPI algorithm and a MB-PD algorithm from the perspective of RL and primal-dual optimization, respectively. Both algorithms can be implemented without using the information of system matrices, such that one can circumvent model identification, which may cause undesirable behaviors due to the propagation and accumulation of modeling errors. Furthermore, we have shown that the dual and primal update steps in the MB-PD algorithm can be interpreted as the policy evaluation and policy improvement steps in the classical PI algorithm. This advances our understanding of common RL algorithms and might contribute to develop various RL algorithms based on optimization formulations.
We take the constraint \( F \in \mathcal{F} \) as the domain of the variable \( F \) in the Lagrangian formulation, instead of directly writing it in the Lagrangian function, since the constraint \( F \in \mathcal{F} \) is not an explicit equality or inequality constraint.

Then, the dual problem of Auxiliary Problem 1 is given as follows:

\[
J_d \triangleq \sup_{P \in \mathbb{S}_+^{n+m}, F \in \mathcal{F}} \inf_{S \in \mathbb{S}_+^{m}} \mathcal{L}(P, F, S),
\]

where \( \mathcal{L}(P, F, S) \) is as defined in (A.3).

Since \( J_p = J^∗ \) and \( J_d = J^∗ \), it holds that \( J_d = J_p \). By weak duality, we have \( J_d \geq J_p \). In view of this, we complete the proof of Theorem 2 by the following two parts:

1. Auxiliary Problem 1 has strong duality, i.e., \( J_p = J_d \) (Subsection A.1);
2. The optimal point of the dual problem of Primal Problem 1 is not larger than that of the dual problem of Auxiliary Problem 1, i.e., \( J_d \leq \tilde{J}_d \) (Subsection A.2).

### A.1 Proof for \( J_p = J_d \)

We first introduce another auxiliary optimization problem which can be shown to be equivalent to the modified stochastic LQR problem, i.e., Problem 2.

**Auxiliary Problem 2** Solve

\[
J_p \triangleq \inf_{P \in \mathbb{S}_+^{n+m}, F \in \mathcal{F}} \mathcal{L}(P, F, S),
\]

s.t. \( F \in \mathcal{F} \),

\[
\gamma A_F S A_F^T + \Gamma + \frac{\gamma}{1-\gamma} F \sigma_w^2 F^T = S. \tag{A.2}
\]

This is a non-convex optimization problem since the set \( \mathcal{F} \) is not convex and the equality constraint (A.2) is a quadratic function of \( F \). Note that we can easily deduce from the proof of Proposition 2 that the optimal value of Auxiliary Problem 1 is equivalent to that of Problem 2, i.e., \( J_p = J^∗ \).

The Lagrangian function of Auxiliary Problem 1 is defined as

\[
\mathcal{L}(P, F, S) = \text{Tr}(\Lambda S) + \frac{1}{\gamma} \text{Tr}(\gamma A_F S A_F^T + \Gamma + \frac{\gamma}{1-\gamma} F \sigma_w^2 F^T - \Lambda P),
\]

where \( P \in \mathbb{S}_+^{n+m} \) is the Lagrange multiplier. The corresponding Lagrangian dual function is defined as

\[
\tilde{d}(P) \triangleq \inf_{S \in \mathbb{S}_+^{m}} \mathcal{L}(P, F, S),
\]

We take the constraint \( F \in \mathcal{F} \) as the domain of the variable \( F \) in the Lagrangian formulation, instead of directly writing it in the Lagrangian function, since the constraint \( F \in \mathcal{F} \) is not an explicit equality or inequality constraint.

Then, the dual problem of Auxiliary Problem 1 is given as follows:

\[
J_d \triangleq \sup_{P \in \mathbb{S}_+^{n+m}, S \in \mathbb{S}_+^{m}, F \in \mathcal{F}} \inf_{P \in \mathcal{F}} \mathcal{L}(P, F, S),
\]

where \( \mathcal{L}(P, F, S) \) is as defined in (A.3).

Since \( J_p = J^∗ \) and \( J_d = J^∗ \), it holds that \( J_d = J_p \). By weak duality, we have \( J_d \geq J_p \). In view of this, we complete the proof of Theorem 2 by the following two parts:

1. Auxiliary Problem 1 has strong duality, i.e., \( J_p = J_d \) (Subsection A.1);
2. The optimal point of the dual problem of Primal Problem 1 is not larger than that of the dual problem of Auxiliary Problem 1, i.e., \( J_d \leq \tilde{J}_d \) (Subsection A.2).

### Proposition 3

The Auxiliary Problem 2 has a unique optimal solution denoted by \((\tilde{P}_p, \tilde{F}_p)\), and it is equivalent to Problem 2 in the sense that \( J_p = J^∗ \) and \( \tilde{F}_p = \tilde{F}^∗ \).

**Proof.** By the equation (29) and the fact that \( w_k \sim \mathcal{N}(0, \sigma_w^2) \) is the i.i.d. random noise for each \( k \geq 0 \), we can obtain that for any \( F \in \mathcal{F} \), it holds that

\[
\mathbb{E}[v_k(F, v_0)' \Lambda v_k(F, v_0)] \leq \mathbb{E}[v_0(F, v_0)'(A_F^j)' \Lambda A_F^j] + \text{Tr}\left( \sum_{j=0}^{k-1} F \sigma_w^2 F' (A_F^j)' \Lambda A_F^j \right).
\]

Therefore, the objective function of Problem 2 becomes

\[
\frac{1}{r} \sum_{i=1}^{r} J(F, v_i) = \text{Tr}(\Gamma P_1) + \frac{\gamma}{1-\gamma} \text{Tr}(F \sigma_w^2 F' P_2),
\]

with

\[
P_1 = \sum_{k=0}^{\infty} \gamma^k (A_F^j)' \Lambda A_F^j,
\]

\[
P_2 = \frac{1-\gamma}{\gamma} \sum_{k=0}^{\infty} \sum_{j=0}^{k-1} (A_F^j)' \Lambda A_F^j.
\]

We can easily verify that both \( P_1 \) and \( P_2 \) satisfy the Lyapunov equation (A.5). By the first statement of Lemma 3, there exists a unique \( P \in \mathbb{S}_+^n \) such that \( \gamma A_F P A_F + \Lambda = P \), which implies that \( P_1 = P_2 \).

Since the optimal solution \((\tilde{P}_p, \tilde{F}_p)\) of Auxiliary Problem 2 is one feasible point of Problem 2, it holds that \( \tilde{J}_p \geq \tilde{J}^∗ \). Furthermore, if \( \tilde{F}_p = \tilde{F}^∗ \in \mathcal{F} \), and \( \tilde{P}_p \) is the unique solution to the constraint (A.5), then the resulting objective function...
of Auxiliary Problem 2 is exactly $\hat{J}$. Therefore, we can con-
clude that $(\hat{S}_p, \hat{F}_p)$ is the solution of Auxiliary Problem 2
and $J_p = \hat{J}$. The uniqueness of $(\hat{S}_p, \hat{F}_p)$ can be obtained by
the uniqueness of $\hat{P}^*$.

To show the strong duality of Auxiliary Problem 1, the
following lemmas will be used.

**Lemma 7** The optimal point $(\hat{P}_p, \hat{F}_p)$ of Auxiliary
Problem 2 satisfies the following:

1. $\hat{P}_p = P^*$;
2. $\hat{P}_p \geq 0$, $\hat{P}_{p,22} > 0$, and $F_p = -(P_{p,22})^{-1}(P_{p,12})' \in \mathcal{F}$.

**Proof.** Based on the definitions in (8) and (11), direct
calculation yields that

\[
\begin{bmatrix}
I_n & \quad P_n & \\
F^* & \quad I_n
\end{bmatrix}
= X^*.
\]

Then, from (11), it follows that $P^* = \gamma [A B] X^* [A B] + \Lambda =
\gamma [A B] \left[
\begin{bmatrix}
I_n & \\
F^*
\end{bmatrix}
\begin{bmatrix}
P_n & \\
P_{p,12}^{-1} P_{p,22} P_{p,12}^{-1}
\end{bmatrix}
\begin{bmatrix}
I_n & \\
F^*
\end{bmatrix}
\right] + \Lambda = \gamma A_p^* P^* A_{F*} + \Lambda.
\]

Since $F^* \in \mathcal{F}$, i.e., $p(A_{F*}) < 1$, there exists a unique $P^*$
satisfying the above equation.

In addition, we have $\hat{F}_p = F^* = \hat{F}$ by Proposition 1
and Proposition 2. Therefore, when $F = \hat{F}_p$, the equality
constraint (A.5) has a unique solution $\hat{P}_p = P^*$.

The second statement can be directly shown via the def-
initions of $P^*$ and $F^*$ given in (11) and (8), respectively.

**Lemma 8 (21, Lemma 4)** If $P \geq 0$ and $P_{22} \geq 0$, then for
\(\forall F \in \mathbb{R}^{m \times n} \), it holds that

\[
\bar{F}^* P \bar{F}^* \geq P_{11} - P_{12} P_{22}^{-1} P_{12}',
\]

and the equality holds if and only if $F = -P_{22}^{-1} P_{12}'$.

Now, we are ready to show the strong duality of Auxiliary
Problem 1.

**Lemma 9** The dual gap for Auxiliary Problem 1 is zero, that is, $\hat{J} = J_d$.

**Proof.** For a given matrix $P \in \mathbb{S}^{n \times m}$, the Lagrangian dual
function is

\[
\hat{d}(P) = \inf_{S \in \mathbb{S}^{m \times n}, F \in \mathcal{F}} \hat{L}(P, F, S) = \begin{cases}
\inf_{F \in \mathcal{F}} \text{Tr} (\Gamma P) + \frac{\gamma}{1 - \gamma} \text{Tr} (\sigma^2 P \bar{F} \bar{P}^*), & \text{if } P \in \hat{\mathcal{P}} \\
-\infty, & \text{otherwise}
\end{cases}
\]

where $\hat{\mathcal{P}} = \{ P \in \mathbb{S}^{n \times m} : \gamma A_p^* P A_F - P + \Lambda \geq 0, \forall F \in \mathcal{F} \}$.

Next, we show that the optimal solution $\hat{P}_p$ of Aux-
iliary Problem 2 is an element of $\hat{\mathcal{P}}$, and thus the set $\hat{\mathcal{P}}$
is nonempty. By Lemma 7, $\gamma A_p^* P A_{F*} + \Lambda = \hat{P}_p$, where $F_p = -(P_{p,22})^{-1}(P_{p,12})'$. Then, by Lemma 8, it holds that

\[
\gamma A_p^* P A_F + \Lambda \geq \gamma A_p^* P A_{F*} + \Lambda = \hat{P}_p \text{ for all } F \in \mathbb{R}^{m \times n}.
\]

Therefore, the dual problem is equivalent to

\[
\hat{J}_d = \sup_{P \in \mathbb{S}^{n \times m}} \hat{d}(P) = \sup_{P \in \mathbb{S}^{n \times m}} \inf_{F \in \mathcal{F}} \text{Tr}(\Gamma P) + \frac{\gamma}{1 - \gamma} \text{Tr}(\sigma^2 P \bar{F} \bar{P}^*).
\]

For $\hat{P}_p \in \mathcal{P}$, we have

\[
\hat{d}(\hat{P}_p) = \inf_{F \in \mathcal{F}} \text{Tr}(\Gamma \hat{P}_p) + \frac{\gamma}{1 - \gamma} \text{Tr}(\sigma^2 \hat{F} \hat{P}_p \hat{F}).
\]

Obviously, $\hat{d}(\hat{P}_p) \leq \hat{J}_d$. Since $\hat{P}_p \geq 0$ is fixed and the objective
function in (A.7) is a quadratic function of $F$, the infimum in (A.7) is reached at $F = -(\hat{P}_{p,22})^{-1}(\hat{P}_{p,12})' \in \mathcal{F}$.

Therefore, $\hat{J}_p = \hat{d}(\hat{P}_p)$, which implies that $\hat{J}_p \leq \hat{J}_d$. By the weak
duality, there holds that $\hat{J}_d \leq \hat{J}_p$. Since $J_p = J^* = J_p$, we have $\hat{J}_d = \hat{J}_d$.

**A.2 Proof for $J_d \leq \hat{J}_d$**

**Lemma 10** There holds $J_d \leq \hat{J}_d$.

**Proof.** For any given $P \in \mathbb{S}^{n \times m}$, $P_0 \in \mathbb{S}^{n \times m}$, the Lagrangian
dual function $d(P, P_0)$ is

\[
d(P, P_0) = \inf_{S \in \mathbb{S}^{m \times n}, F \in \mathcal{F}} L(P, P_0, F, S) = \begin{cases}
\inf_{F \in \mathcal{F}} \text{Tr}(\Gamma P) + \frac{\gamma}{1 - \gamma} \text{Tr}(\sigma^2 P \bar{F} \bar{F}^*), & \text{if } P \in \hat{\mathcal{P}} \\
-\infty, & \text{otherwise}
\end{cases}
\]

where $\mathcal{P} = \{ P \in \mathbb{S}^{n \times m} : \gamma A_p^* P A_F - P + \Lambda - P_0 \geq 0, \forall F \in \mathcal{F} \}$.

Therefore, the dual problem corresponding to Primal
Problem 1 is equivalent to

\[
J_d = \sup_{P \in \mathcal{P}} d(P, P_0)
= \sup_{P \in \mathcal{P}, P_0 \in \mathbb{S}^{n \times m}} \inf_{F \in \mathcal{F}} \text{Tr}(\Gamma P) + \frac{\gamma}{1 - \gamma} \text{Tr}(\sigma^2 P \bar{F} \bar{F}^*).
\]

We can observe from (A.8) that $P_0$ affects the dual optimal
value $J_d$ by adjusting the size of the set $\mathcal{P}$. To obtain
the supremum of the Lagrangian dual function, the constraint
$P \in \mathcal{P}$ should be relaxed as much as possible. We can easily
observe from the definition of the set $\mathcal{P}$ that its maximum
is obtained when $P_0 = 0_{m \times m}$. Furthermore, if $P_0 = 0_{m \times m}$, then it holds that $\mathcal{P} = \{ P \in \mathbb{S}^{n \times m} : \gamma A_p^* P A_F - P + \Lambda \geq 0, \forall F \in \mathcal{F} \} = \hat{\mathcal{P}}$. As a result, it yields that

\[
J_d = \sup_{P \in \mathcal{P}} \inf_{F \in \mathcal{F}} \text{Tr}(\Gamma P) + \frac{\gamma}{1 - \gamma} \text{Tr}(\sigma^2 P \bar{F} \bar{F}^*).
\]

Since $\mathcal{F} \subseteq \mathbb{R}^{m \times n}$, it holds that $\inf_{F \in \mathcal{F}} \text{Tr}(\Gamma P) + \frac{\gamma}{1 - \gamma} \text{Tr}(\sigma^2 P \bar{F} \bar{F}^*) \leq \text{Tr}(\sigma^2 \hat{F} \hat{F}^*)$, which
implies that $J_d \leq \hat{J}_d$.

References

[1] R. S. Sutton, A. G. Barto. (1998). Reinforcement Learning: An
introduction, MIT Press, Cambridge, MA, USA.

[2] D. P. Bertsekas. (2019). Reinforcement Learning and Optimal
Control, Athena Scientific, Nashua, USA.

[3] K. Arulkumar, M. P. Deisenroth, M. Brundage, A. A. Bharath.
(2017). Deep reinforcement learning: A brief survey, IEEE Signal
Processing Magazine, 34 (6), 26–38.

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

[5] M. Li, J. Qin, N. M. Freris, D. W. C. Ho, Multiplayer Stackelberg-
Nash game for nonlinear system via value iteration-based integral
reinforcement learning, IEEE Transactions on Neural Networks and
Learning Systems, DOI: 10.1109/TNNLS.2020.3042331.
[6] F. Li, J. Qin, W. X. Zheng. (2019). Distributed Q-learning-based online optimization algorithm for unit commitment and dispatch in smart grid, IEEE Transactions on Cybernetics, 50 (9), 4146–4156.

[7] B. Wang, Z. Liu, Q. Li, A. Proorok. (2020). Mobile robot path planning in dynamic environments through globally guided reinforcement learning, IEEE Robotics and Automation Letters, 5 (4), 6932–6939.

[8] A. Haydari, Y. Yilmaz, Deep reinforcement learning for intelligent transportation systems: A survey, IEEE Transactions on Intelligent Transportation Systems, DOI: 10.1109/TITS.2020.3008612.

[9] A. S. Leong, A. Ramaswamy, D. E. Quevedo, H. Karl, L. Shi, Deep reinforcement learning for wireless sensor scheduling in cyber–physical systems, Automatica, DOI: https://doi.org/10.1016/j.automatica.2019.108759.

[10] A. Naik, R. Shariat, N. Yasui, R. S. Sutton, Discounted reinforcement learning is not an optimization problem, arXiv preprint arXiv:1910.02140.

[11] R. Zhang, T. Yu, Y. Shen, H. Jin, C. Chen. (2019). Text-based interactive recommendation via constraint-augmented reinforcement learning, in: Advances in Neural Information Processing Systems, Vancouver, Canada, pp. 15214–15224.

[12] Z. Qin, W. Li, F. Janoos. (2014). Sparse reinforcement learning via convex optimization, in: International Conference on Machine Learning, Beijing, China, pp. 424–432.

[13] N. Vieillard, O. Pietquin, M. Geist, On connections between constrained optimization and reinforcement learning, arXiv preprint arXiv:1910.08476.

[14] H. Schlüter, F. Solowjow, S. Trimpe, Event-triggered learning for linear quadratic control, IEEE Transactions on Automatic Control, DOI: 10.1109/TAC.2020.3030877.

[15] M. Fazel, R. Ge, S. M. Kakade, M. Mesbahi. (2018). Global convergence of policy gradient methods for the linear quadratic regulator, in: International Conference on Machine Learning, Stockholm, Sweden, pp. 1467–1476.

[16] D. Malik, A. Pananjady, K. Bhatia, K. Khamaru, B. L. Peter, W. J. Martin. (2019). Derivative-free methods for policy optimization: Guarantees for linear quadratic systems, in: The 22nd International Conference on Artificial Intelligence and Statistics, Stockholm, Sweden, pp. 2916–2925.

[17] Y. Li, Y. Tang, R. Zhang, N. Li, Distributed reinforcement learning for decentralized linear quadratic control: A derivative-free policy optimization approach, arXiv preprint [arXiv:2008.08734] arXiv preprint arXiv:1912.09135.

[18] K. Karl, S. Tu. (2019). Finite-time analysis of approximate policy iteration for the linear quadratic regulator, in: International Conference on Machine Learning, Vancouver, Canada, pp. 8514–8524.

[19] Z. Yang, Y. Chen, M. Hong, Z. Wang. (2019). Provably global convergence of actor-critic: A case for linear quadratic regulator with ergodic cost, in: Advances in Neural Information Processing Systems, Vancouver, Canada, pp. 8353–8365.

[20] R. Mitze, M. Mönningmann, A dynamic programming approach to solving constrained linear–quadratic optimal control problems, Automatica, DOI: https://doi.org/10.1016/j.automatica.2020.109132.

[21] D. Lee, J. Hu. (2018). Primal-dual Q-learning framework for LQR design, IEEE Transactions on Automatic Control, 64 (9), 3756–3763.

[22] D. D. Yao, S. Zhang, X. Y. Zhou. (2001). Stochastic linear-quadratic control via semidefinite programming, SIAM Journal on Control and Optimization, 40 (3), 801–823.

[23] Y. Jiang, Z.-P. Jiang. (2012). Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics, Automatica, 48 (10), 2699–2704.

[24] B. Kiumarsi, F. L. Lewis, Z.-P. Jiang. (2017). $H_\infty$ control of linear discrete-time systems: Off-policy reinforcement learning, Automatica, 78, 144–152.

[25] J. Qin, M. Li, Y. Shi, Q. Ma, W. X. Zheng. (2019). Optimal synchronization control of multiagent systems with input saturation via off-policy reinforcement learning, IEEE Transactions on Neural Networks and Learning Systems, 30 (1), 85–96.