Convergence and Robustness of Value and Policy Iteration for the Linear Quadratic Regulator

Bowen Song, Chenxuan Wu, Andrea Iannelli

***Abstract*— This paper revisits and extends the convergence and robustness properties of value and policy iteration algorithms for discrete-time linear quadratic regulator problems. In the model-based case, we extend current results concerning the region of exponential convergence of both algorithms. In the case where there is uncertainty on the value of the system matrices, we provide input-to-state stability results capturing the effect of model parameter uncertainties. Our findings offer new insights into these algorithms at the heart of several approximate dynamic programming schemes, highlighting their convergence and robustness behaviors. Numerical examples illustrate the significance of some of the theoretical results.**

I. Introduction

Approximate dynamic programming (ADP) [1]–[3] is a powerful algorithmic approach designed to solve sequential decision-making problems across a broad range of applications. Two fundamental approaches in ADP are: value iteration (VI) and policy iteration (PI), which have been extensively analyzed in the literature [4], [5]. VI updates the value function of the underlying optimal control problem iteratively [6], [7], while PI evaluates and improves policies sequentially [8]. The convergence properties of these two algorithms have been studied in works such as [9] and [10]. In [9], the convergence rates of VI and PI are compared for a finite state and action Markov decision problem. In contrast, [10] investigates the conditions for asymptotic and exponential convergence of VI and PI in the context of the discrete-time linear quadratic regulator (LQR) problem. ADP's application to the LQR problem has received substantial attention due to its analytically tractable nature, making it an ideal benchmark for studying ADP in environments with continuous state and action spaces [11], [12]. Studying the performance of VI and PI for the LQR problem is an active area of research [10], [13]–[15].

Typically, performing VI or PI requires knowledge of the system model, which is where most existing theoretical results are established [10], [11], [16]. However, the system model is often unavailable in practice. To address this, data can be used to either identify a model and based on the estimate apply the algorithm (indirect data-driven control) or directly design the controller (direct data-driven Control). In [17], PI is combined with online model estimation, while [18] proposes a direct formulation of PI using online data, bypassing model estimation. For both approaches to VI and PI, analyzing the robustness of the algorithms is crucial due to unavoidable uncertainty associated with the use of finite and potentially noisy data, which can introduce estimation errors that affect the controller's design. In [19], [20], the robustness of PI applied to continuous-time LQR problems and stochastic LQR problems is analyzed, respectively. Inspired by [19], our previous work [17] extended this analysis to the robustness of PI for discrete-time LQR problems.

In this work, we investigate the nominal (i.e. with known model) exponential convergence and robustness of VI and PI applied to the discrete-time LQR problem. For the nominal case, we extend the standard conditions for exponential convergence of VI and PI algorithms as provided in [10]. Building on these results, we analyze the robustness to model uncertainties of VI and PI, an aspect that has been explored for PI under different uncertainty structures in [19], [20], but not for VI. Specifically, we study the performance of two algorithms when estimates of the true model are used, and we analyze the effect of uncertainty on the convergence properties. A motivating example for this analysis is the use of online identification routines providing at each iteration of the PI/VI algorithm a different estimate, which differs from the true one by a certain amount. We show that both VI and PI algorithms have inherent robustness to uncertainties within specific bounds. This property is crucial for the reliable deployment of indirect VI and PI algorithms, where handling uncertainties and estimation errors should be considered.

The paper is organized as follows. Section II introduces the problem setting and some preliminaries. Section III and Section IV detail the exponential convergence and robustness analysis for both VI and PI algorithms, respectively. Section V provides simulations to illustrate some theoretical examples. Section VI concludes the work.

Notations: We denote by $A \succeq 0$ and $A \succ 0$ a positive semidefinite and positive definite matrix A, respectively. For matrices, $\|\cdot\|_F$ and $\|\cdot\|$ denote respectively their Frobenius norm and induced 2-norm. Given positive definite matrix $P_\epsilon \preceq I$ [10, Lemma 6], we define the induced P_ϵ-norm for matrices of compatible dimension as: $\|\cdot\|_{P_\epsilon} := \sqrt{\lambda_{max}((\cdot)^\top P_\epsilon(\cdot))}$. For $X \in \mathbb{R}^{m\times n}$, we define $vec(X) := [X_1^\top, ..., X_n^\top]^\top$, where X_i is the i-th column of matrix X. Kronecker product is represented as $\otimes$. For $Y \in \mathbb{R}^{m\times n}$ and $r > 0$, we define $\mathcal{B}_r(Y) := \{X \in \mathbb{R}^{m\times n} | \|X - Y\|_F < r\}$. A sequence $\{Y_i\}$ is a map $\mathbb{Z}_+ \to \mathbb{R}^{n\times m}$. For bounded scalar sequences, we denote by $\|Y\|_\infty := \sup_{i\in\mathbb{Z}_+} \{Y_i\}$.

Bowen Song acknowledges the support of the International Max Planck Research School for Intelligent Systems (IMPRS-IS). Andrea Iannelli acknowledges the German Research Foundation (DFG) for support of this work under Germany's Excellence Strategy - EXC 2075 – 390740016.

The authors are with the University of Stuttgart, Institute for Systems Theory and Automatic Control, 70550 Stuttgart, Germany {bowen.song, andrea.iannelli}@ist.uni-stuttgart.de, {st176873@stud.uni-stuttgart.de}.

II. Preliminaries and Problem Setting

We consider discrete-time linear time-invariant systems of the form:

$$x_{t+1} = Ax_t + Bu_t, \tag{1}$$

where $x_t \in \mathbb{R}^{n_x}$ is the system state, $u_t \in \mathbb{R}^{n_u}$ is the control input, t is the timestep, and pair (A, B) is stabilizable. The objective is to design a state-feedback controller $u_t = Kx_t$ that minimizes the following infinite horizon cost:

$$J(x_t, K) = \sum_{k=t}^{+\infty} r(x_k, u_k) = \sum_{k=t}^{+\infty} x_k^\top Q x_k + u_k^\top R u_k, \tag{2}$$

where $R \succ 0$ and $Q \succeq 0$. Given a linear state-feedback gain K that is stabilizing (i.e. $A + BK$ is Schur stable), the corresponding cost $J(x_t, K)$ can be expressed as $x_t^\top P x_t$, where $P \succ 0$ is also called the quadratic kernel of the cost function associated with K [18]. Starting from (2) and using the principle of optimality, P can be calculated by solving the model-based Bellman equation [17]:

$$P = Q + K^\top R K + (A + BK)^\top P (A + BK). \tag{3}$$

We introduce the following definition related to the gain K and the quadratic kernel P:

Definition 1 (Stability of gain K and kernel P): A control gain K is said to be stabilizing if $(A + BK)$ is Schur stable. A positive semi-definite matrix P is said to be stabilizing if the gain $K = -(R + B^\top P B)^{-1} B^\top P A$ is stabilizing.

It is a well-known result [21] that the optimal controller solution to the LQR problem is a linear state-feedback, and the optimal feedback gain K^* is obtained via:

$$K^* = -(R + B^\top P^* B)^{-1} B^\top P^* A, \tag{4a}$$

$$P^* = Q + A^\top P^* A - A^\top P^* B (R + B^\top P^* B)^{-1} B^\top P^* A. \tag{4b}$$

Here, P^* is the quadratic kernel of the value function, i.e. of the cost associated with the optimal gain K^*, and is the unique solution of the discrete algebraic Riccati equation (DARE) (4b). The optimal gain K^* is stabilizing. Therefore, based on Definition 1, P^* is stabilizing.

Solving (4b) directly is challenging, especially when dealing with a high number of system states. Value iteration and policy iteration offer an effective iterative approach to find the optimal gain K^* and are introduced in the following subsections.

A. Value Iteration (VI)

The procedure of value iteration is summarized in Algorithm 1, which requires knowledge of the system matrices A and B and only uses matrix multiplication to update the cost function. In the value iteration, the kernel P_i is iteratively updated based on (4b), treating P_i as the kernel of value function.

Algorithm 1 Value Iteration

Require: A, B, a positive semidefinite kernel P_0
  for $i = 0, ..., +\infty$ **do**
    Update the kernel P_i
    $P_{i+1} = Q + A^\top P_i A - A^\top P_i B (R + B^\top P_i B)^{-1} B^\top P_i A$
  end for

The properties of Algorithm 1 are summarized in the following theorem.

Theorem 1: Properties of VI [10]

If the system dynamics (A, B) are stabilizable, then for all $P_0 \succeq 0$:

1) $\lim_{i \to \infty} P_i = P^*$, thus the sequence $\{P_i\}$ converges asymptotically to P^*;
2) if $P_0 \succeq P^*$, then $\|P_{i+1} - P^*\|_{P_\epsilon} \leq d \|P_i - P^*\|_{P_\epsilon}$ with $d \in (0, 1)$, $\forall i \in \mathbb{Z}_+$. Thus, the sequence $\{P_i\}$ converges exponentially to P^*.

B. Policy Iteration (PI)

The basic version of the policy iteration algorithm [16] is summarized in Algorithm 2. The PI algorithm is more complex than the VI algorithm, as the VI only requires the matrix multiplication and inversion, while PI involves solving the Lyapunov equation in the policy evaluation step. In the policy evaluation phase, the performance of K_i is evaluated by using (3). In the policy improvement phase, the policy is improved by treating the evaluation P_i as the kernel of value function and using (4a).

Algorithm 2 Policy Iteration

Require: A, B, a stabilizing policy gain K_0
  for $i = 0, ..., +\infty$ **do**
    Policy Evaluation: find P_i
    $P_i = Q + K_i^\top R K_i + (A + BK_i)^\top P_i (A + BK_i)$
    Policy Improvement: update gain K_{i+1}
    $K_{i+1} = -(R + B^\top P_i B)^{-1} B^\top P_i A$
  end for

The properties of Algorithm 2 are summarized below.

Theorem 2: Properties of PI [16] [17, Theorem 4]

If K_0 stabilizes (1), then

1) $P_0 \succeq P_1 \succeq ... \succeq P^*$;
2) K_i stabilizes the system (A, B), $\forall i \in \mathbb{Z}_+$;
3) $\lim_{i \to \infty} P_i = P^*$, $\lim_{i \to \infty} K_i = K^*$;
4) $\|P_{i+1} - P^*\|_F \leq c \|P_i - P^*\|_F$ with $c \in (0, 1)$, $\forall i \in \mathbb{Z}_+$. Thus, the sequence $\{P_i\}$ converges exponentially to P^*.

From Theorem 1, the asymptotic convergence of VI is guaranteed by $P_0 \succeq 0$, while the exponential convergence of VI can be only guaranteed under the condition $P_0 \succeq P^*$. From Theorem 2, the exponential convergence of PI is guaranteed by initializing with a stabilizing policy gain K_0.

III. Convergence Analysis of VI and PI

In this section, we relax the conditions for the exponential convergence properties of VI and PI algorithms. We begin by deriving the following lemma which combines the continuity

of matrix eigenvalues [22, Chapter 6] with the stability property of P^* discussed earlier:

Lemma 1 (Stability of P around P^*): There exists a $\delta_0 > 0$ such that for any $P \in \mathcal{B}_{\delta_0}(P^*)$, P is stabilizing.

In the following two subsections, we investigate the exponential convergence properties of VI and PI algorithms within the region $\mathcal{B}_{\delta_0}(P^*)$. To facilitate our analysis, we introduce the following notations:

$$L(P) := (R + B^\top P B)^{-1} B^\top P A, \tag{5a}$$

$$\mathcal{A}(P) := A - BL(P). \tag{5b}$$

A. Exponential Convergence of VI

From Theorem 1, the asymptotic convergence is guaranteed for any positive semidefinite P_0, and exponential convergence is achieved when $P_0 \succeq P^*$. We now introduce new requirements for the exponential convergence of VI.

Theorem 3 (Local exponential convergence of VI): For any $P_i \in \mathcal{B}_{\delta_0}(P^*)$, with δ_0 defined in Lemma 1, the following inequality holds:

$$\|P_{i+1} - P^*\|_{P_\epsilon} \leq \alpha \|P_i - P^*\|_{P_\epsilon}, \quad \forall i \in \mathbb{Z}_+, \tag{6}$$

where $\alpha \in (0, 1)$ is a constant. Thus, the sequence $\{P_i\}$ converges exponentially to P^* when $P_0 \in \mathcal{B}_{\delta_0}(P^*)$.

Proof: First, we define the Bellman operator $\mathcal{T}$ [10] as follows:

$$\mathcal{T}(P) := \mathcal{A}(P)^\top P \mathcal{A}(P) + L(P)^\top R L(P) + Q, \tag{7}$$

which is a fixed point iteration in VI, i.e. $P_{i+1} = \mathcal{T}(P_i)$. Using this operator, a sequence $\{P_i\}$ is constructed, where $P_{i+1} = \mathcal{T}(P_i)$ with initialization P_0. Then the proof of Theorem 3 follows by establishing upper and lower bounds on the operator $\mathcal{T}(P) - P^*$, and then showing the conditions under which exponential convergence is guaranteed. An upper bound of $\mathcal{T}(P) - P^*$ can be derived as:

$$\begin{aligned}
&\mathcal{T}(P) - P^* \\
&= \begin{bmatrix} I \\ -L(P) \end{bmatrix}^\top \underbrace{\begin{bmatrix} A^\top P A & A^\top P B \\ B^\top P A & R + B^\top P B \end{bmatrix}}_{=: M(P)} \begin{bmatrix} I \\ -L(P) \end{bmatrix} \\
&\quad - \begin{bmatrix} I \\ -L(P^*) \end{bmatrix}^\top M(P^*) \begin{bmatrix} I \\ -L(P^*) \end{bmatrix} \\
&\preceq \begin{bmatrix} I \\ -L(P^*) \end{bmatrix}^\top (M(P) - M(P^*)) \begin{bmatrix} I \\ -L(P^*) \end{bmatrix} \\
&= \mathcal{A}(P^*)^\top (P - P^*) \mathcal{A}(P^*),
\end{aligned} \tag{8}$$

the inequality is due to the definition of $L(P)$ in (5) and [23, Lemma 4]. Similarly, a lower bound can be derived by replacing $L(P^*)$ with $L(P)$ at the first equality in (8) and using [23, Lemma 4]:

$$\mathcal{T}(P) - P^* \succeq \mathcal{A}(P)^\top (P - P^*) \mathcal{A}(P).$$

From the upper bound and lower bound, we obtain the following for all $i \in \mathbb{Z}_+$:

$$\begin{aligned}
\mathcal{A}(P_i)^\top (P_i - P^*) \mathcal{A}(P_i) \preceq \mathcal{T}(P_i) - P^* = \\
P_{i+1} - P^* \preceq \mathcal{A}(P^*)^\top (P_i - P^*) \mathcal{A}(P^*).
\end{aligned} \tag{9}$$

Combining this with [10, Lemma 6], we conclude:

$$\begin{aligned}
\|P_{i+1} - P^*\|_{P_\epsilon} \leq \max\{\|\mathcal{A}(P_i)\|^2_{P_\epsilon}, \|\mathcal{A}(P^*)\|^2_{P_\epsilon}\} \\
\|P_i - P^*\|_{P_\epsilon}.
\end{aligned}$$

By asymptotic convergence of $\{P_i\}$ from Theorem 1 and the definition of δ_0, we have $\lim_{i \to \infty} \mathcal{A}(P_i) = \mathcal{A}(P^*)$ and $\mathcal{A}(P_i)$ is Schur stable for all $i \in \mathbb{Z}_+$. Then we know $\max\{\|\mathcal{A}(P_i)\|^2_{P_\epsilon}, \|\mathcal{A}(P^*)\|^2_{P_\epsilon}\} < 1$, $\forall i \in \mathbb{Z}_+$. We define $\alpha := \sup_i \{\|\mathcal{A}(P_i)\|^2_{P_\epsilon}\}$. Because $\lim_{i \to \infty} \|\mathcal{A}(P_i)\|_{P_\epsilon} = \|\mathcal{A}(P^*)\|_{P_\epsilon} < 1$ and $\|\mathcal{A}(P_i)\|_{P_\epsilon} < 1, \forall i \in \mathbb{Z}_+$, we have $\alpha \in (0, 1)$. Then we conclude the proof of Theorem 3. ■

Remark 1: Unlike Theorem 1, Theorem 3 does not require the condition $P_0 \succeq P^*$ for exponential convergence. Instead, exponential convergence is guaranteed when $P_0 \in \mathcal{B}_{\delta_0}(P^*)$.

Building on Theorem 1 and Theorem 3, the following corollary establishes a larger region for local exponential convergence than what is currently available:

Corollary 1 (Exponential Convergence of VI): Defining set $\mathcal{S} := \{P \succeq 0 | P \succeq P^* \cup P \in \mathcal{B}_{\delta_0}(P^*)\}$ with δ_0 defined in Lemma 1, for any $P_i \in \mathcal{S}$, we have:

$$\|P_{i+1} - P^*\|_{P_\epsilon} \leq v \|P_i - P^*\|_{P_\epsilon}, \quad \forall i \in \mathbb{Z}_+, \tag{10}$$

where $v \in (0, 1)$ is a constant. Thus, the sequence $\{P_i\}$ converges exponentially to P^* when $P_0 \in \mathcal{S}$.

The proof of Corollary 1 is a combination of Theorem 1 and Theorem 3, where $v := \max\{d, \alpha\}$.

B. Exponential Convergence of PI

As outlined in Algorithm 2, the PI procedure begins with an initial stabilizing control gain K_0 followed by the estimation of P_0 through the solution of a Lyapunov equation (3) and continues by iterating K_i and P_i. To facilitate the comparison with the VI algorithm and the robustness analysis in Section IV, we consider the PI algorithm initialized with P_0 instead. For any $P_i \succeq P^*$, K_{i+1} stabilizes the system (A, B). Then from Theorem 2, the sequence $\{P_i\}$ converges exponentially to P^*.

Similarly to the analysis conducted for the VI algorithm, we investigate the convergence properties of the PI algorithm when $P_i \in \mathcal{B}_{\delta_0}(P^*)$.

Theorem 4 (Local exponential convergence of PI):

1) For any $P_i \in \mathcal{B}_{\delta_0}(P^*)$, with δ_0 defined in Lemma 1, the following inequality holds:

$$\|P_{i+1} - P^*\|_F \leq \sigma_0 \|P_i - P^*\|_F, \quad \forall i \in \mathbb{Z}_{++}, \tag{11}$$

with $\sigma_0 = c \in (0, 1)$ defined in Theorem 2. Thus, if $P_0 \in \mathcal{B}_{\delta_0}(P^*)$, the sequence $\{P_i\}$ converges exponentially to P^*. The distance from P^* decreases monotonically starting from $i = 1$.

2) There exists a constant $\delta_1 \in (0, \delta_0]$, such that for any $P_i \in \mathcal{B}_{\delta_1}(P^*)$, the following inequality holds:

$$\|P_{i+1} - P^*\|_F \leq \sigma_1 \|P_i - P^*\|_F, \quad \forall i \in \mathbb{Z}_+, \tag{12}$$

where $\sigma_1 \in (0, 1)$. Thus, the sequence $\{P_i\}$ converges exponentially to P^* when $P_0 \in \mathcal{B}_{\delta_1}(P^*)$.

The proof of Theorem 4 is provided in [24, Appendix A].

Remark 2: In contrast to Theorem 2, Theorem 4 does not require the condition $P_0 \succeq P^*$ for the exponential convergence. Instead, exponential convergence is guaranteed when $P_0 \in \mathcal{B}_{\delta_0}(P^*)$. If $P_0 \in \mathcal{B}_{\delta_1}(P^*)$, the distance to P^* decreases monotonically from the initial step $i = 0$, which is necessary for the robustness analysis in Section IV.

By combining Theorem 2 and Theorem 4, we can derive the following corollary, which provides a larger region of initial conditions P_0 for which exponential convergence is guaranteed:

Corollary 2 (Exponential Convergence of PI): For any $P_i \in \mathcal{S}$, with $\mathcal{S}$ defined in Corollary 1, we have:

$$\|P_{i+1} - P^*\|_F \leq c\|P_i - P^*\|_F, \quad \forall i \in \mathbb{Z}_{++}, \tag{13}$$

where $c \in (0, 1)$ is defined in Theorem 2. Thus, the sequence $\{P_i\}$ converges exponentially to P^* when $P_0 \in \mathcal{S}$.

The proof is a combination of Theorem 2 and Theorem 4. We note that when K_0 is known a-priori to be stabilizing, the corresponding kernel P_0 automatically satisfies the condition $P_0 \succeq P^*$, and thus the enlargement of the exponential region provided by Theorem 4 is not of immediate use. However, it holds a significant value when the system matrices (A, B) are unknown and thus the condition on P_0 required by Theorem 2 cannot be easily established a-prior. Establishing Theorem 4 is crucial for effectively analyzing the robustness of PI algorithm, which is a central theme of Section IV.

C. Comparison between VI and PI

From the analysis in previous sections, we know that when $P_0 \succeq P^*$, the sequences $\{P_i\}$ generated by both VI or PI algorithms converges exponentially to the optimal P^*, as graphically shown in the shaded region in Figure 1. In Theorem 3 and Theorem 4, we identified the local region $\mathcal{B}_{\delta_0}(P^*)$ around P^* of exponential convergence for both VI and PI. Additionally, we introduced $\mathcal{B}_{\delta_1}(P^*)$ specifically for PI, within which the distance from P^* decreases monotonically for all $i \in \mathbb{Z}_+$, as illustrated in Figure 1.

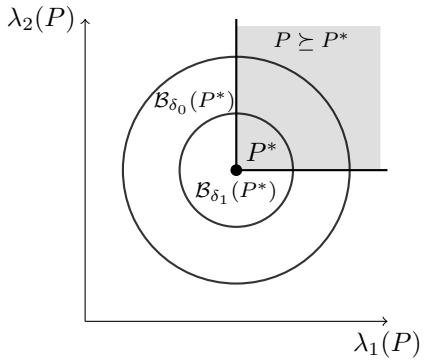

Fig. 1: 2-dimensional Graphic Representation

However, there is no explicit expression for δ_0 (defined in Lemma 1). Theorem 3 and Theorem 4 only prove the existence of the region $\mathcal{B}_{\delta_0}(P^*)$ and $\mathcal{B}_{\delta_1}(P^*)$. Nonetheless, in certain special cases as the following one, we can provide sufficient conditions that ensure the stability of P_0, and we can use them to provide verifiable conditions on the initialization of VI and PI, which ensure their exponential convergence beyond classic results from the literature.

Theorem 5 (Convergence of VI and PI with $P \succeq 0$): If the system matrix A is Schur stable, then any $P \succeq 0$ is stabilizing. Therefore, for any $P_0 \succeq 0$, the sequences $\{P_i\}$ generated by both VI and PI converge exponentially to P^*.

The proof of Theorem 5 is provided in [24, Theorem 5]. Theorem 5 enables the initialization of any $P_0 \succeq 0$ in both VI and PI algorithms when $\|A\| < 1$. This theorem provides the exponential convergence guarantee for such an initialization for VI. For PI, convergence is also guaranteed, eliminating the need for an initializing stabilizing policy gain K_0 and instead allowing for initialization with any $P_0 \succeq 0$.

IV. Robust Analysis of VI and PI

In the previous section, we analyzed the exponential convergence properties of VI and PI algorithms based on the assumption of perfect knowledge of the system matrices (A, B). However, in practical scenarios, the system matrices are often unknown or only partially known. Therefore, we consider the case where, at each iteration i of the algorithm, VI and PI employ estimates $\hat{A}_i$ and $\hat{B}_i$ in place of A and B, respectively. We denote the differences between them as:

$$\Delta A_i := \hat{A}_i - A, \ \Delta B_i := \hat{B}_i - B. \tag{14}$$

For the analysis in this section, we introduce two scalar sequences $\{a_i\}$ and $\{b_i\}$, whose entries are defined as:

$$a_i := \|\Delta A_i\|_F, \ b_i := \|\Delta B_i\|_F. \tag{15}$$

This setting captures the case where a fixed model estimate is used ($\hat{A}_i = \hat{A}$ and $\hat{B}_i = \hat{B}$ for all $i \in \mathbb{Z}_+$) but also the more interesting case where a running estimate of the model is updated throughout the design process. The latter scenario arises for example in model-based RL [25, Section 5], and indirect data-driven control [26], where a system identification algorithm uses collected data to update online an estimate of the model. Given the use of VI and PI algorithms as building blocks of complex learning-based schemes [17], it is crucial to analyze their robustness in the face of inexact estimates of the system matrices.

A. Robustness of Inexact VI

The procedure of inexact VI algorithm formulated by estimate system matrices is given in Algorithm 3.

Algorithm 3 Value Iteration with Estimates $(\hat{A}_i, \hat{B}_i)$

Require: $\{\hat{A}_i\}\{\hat{B}_i\}$, a stabilizing $\hat{P}_0$
for $i = 0, ..., +\infty$ **do**
$\hat{P}_{i+1} = \hat{A}_i^\top \hat{P}_i \hat{A}_i + Q - \hat{A}_i^\top \hat{P}_i \hat{B}_i (R + \hat{B}_i^\top \hat{P}_i \hat{B}_i)^{-1} \hat{B}_i^\top \hat{P}_i \hat{A}_i$
end for

Note that the initial matrix $\hat{P}_0$ must be stabilizing for the true system. The following theorem analyzes the convergence properties of Algorithm 3.

Theorem 6 (Robustness of VI): Given α and δ_0 as defined in Theorem 3 and Lemma 1, there always exist constants $\bar{a}_v(\delta_0, \alpha) \geq 0$ and $\bar{b}_v(\delta_0, \alpha) \geq 0$ such that if $\|a\|_\infty \leq \bar{a}_v$, $\|b\|_\infty \leq \bar{b}_v$ and $\hat{P}_0 \in \mathcal{B}_{\delta_0}(P^*)$, where sequences $\{a_i\}$ and $\{b_i\}$ are defined in (15), then:

1) $\hat{P}_i$ is stabilizing, $\forall i \in \mathbb{Z}_+$;
2) the following holds:

$$\begin{aligned} \|\hat{P}_i - P^*\|_{P_\epsilon} \leq \beta_1(\|\hat{P}_0 - P^*\|_{P_\epsilon}, i) \\ +\gamma_1(\|a\|_\infty) + \gamma_2(\|b\|_\infty), \quad \forall i \in \mathbb{Z}_+, \end{aligned} \tag{16}$$

where $\beta_1(x, i) := \alpha^i x$; $\gamma_1(x) := \frac{\bar{v}_a}{1-\alpha} x$; $\gamma_2(x) := \frac{\bar{v}_b}{1-\alpha} x$ with constants $\bar{v}_a,\ \bar{v}_b > 0$;
3) if $\lim_{i\to\infty} \|\Delta A_i\|_F = 0$ and $\lim_{i\to\infty} \|\Delta B_i\|_F = 0$, then $\lim_{i\to\infty} \|\hat{P}_i - P^*\|_{P_\epsilon} = 0$.

The proof of Theorem 6 is provided in [24, Theorem 6].

B. Robustness of Inexact PI

The procedure for the inexact policy iteration algorithm is outlined in Algorithm 4.

Algorithm 4 Policy Iteration with Estimates $(\hat{A}_i, \hat{B}_i)$

Require: $\{\hat{A}_i\}\{\hat{B}_i\}$, a stabilizing gain $\hat{K}_0$
for $i = 0, ..., +\infty$ **do**
$\hat{P}_i = Q + \hat{K}_i^\top R \hat{K}_i + (\hat{A}_i + \hat{B}_i \hat{K}_i)^\top \hat{P}_i (\hat{A}_i + \hat{B}_i \hat{K}_i)$
$\hat{K}_{i+1} = -(R + \hat{B}_{i+1}^\top \hat{P}_i \hat{B}_{i+1})^{-1} \hat{B}_{i+1}^\top \hat{P}_i \hat{A}_{i+1}$
end for

The following theorem analyzes the convergence properties of Algorithm 4.

Theorem 7 (Robustness of PI): Given σ_1 and δ_1 defined in Theorem 4, there always exist constants $\bar{a}_p(\delta_1, \sigma_1) \geq 0$ and $\bar{b}_p(\delta_1, \sigma_1) \geq 0$ such that if $\|a\|_\infty \leq \bar{a}_p$, $\|b\|_\infty \leq \bar{b}_p$ and $\hat{P}_0 \in \mathcal{B}_{\delta_1}(P^*)$, where sequences $\{a_i\}$ and $\{b_i\}$ are defined in (15), then:

1) $\hat{K}_i$ is stabilizing, $\forall i \in \mathbb{Z}_+$;
2) the following holds:

$$\begin{aligned} \|\hat{P}_i - P^*\|_F \leq \beta_2(\|\hat{P}_0 - P^*\|_F, i) \\ +\gamma_3(\|a\|_\infty) + \gamma_4(\|b\|_\infty), \quad \forall i \in \mathbb{Z}_+, \end{aligned} \tag{17}$$

where $\beta_2(x, i) := \sigma_1^i x$; $\gamma_3(x) := \frac{\bar{p}_a}{1-\sigma_1} x$; $\gamma_4(x) := \frac{\bar{p}_b}{1-\sigma_1} x$ with constants $\bar{p}_a, \bar{p}_b > 0$;
3) if $\lim_{i\to\infty} \|\Delta A_i\|_F = 0$ and $\lim_{i\to\infty} \|\Delta B_i\|_F = 0$, then $\lim_{i\to\infty} \|\hat{P}_i - P^*\|_F = 0$.

The proof of this result follows similar arguments to that of Theorem 6 and can be found in the [24, Appendix B].

Theorem 6 and Theorem 7 show that both VI and PI algorithms have an inherent robustness against uncertainties in the system matrices, when the uncertainties remain within the bounds specified in the theorems.

V. Simulation

In this section, we present some numerical results[1] to compare the convergence and robustness of VI and PI algorithms. We consider the following system which was already used in prior studies [17], [18]:

$$x_{t+1} = \underbrace{\begin{bmatrix} 1.01 & 0.01 & 0 \\ 0.01 & 1.01 & 0.01 \\ 0 & 0.01 & 1.01 \end{bmatrix}}_{A} x_t + \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{B} u_t. \tag{18}$$

The weight matrices Q and R are set to $0.001 I_3$ and I_3, respectively.

Figure 2 illustrates the convergence properties of the VI and PI algorithms assuming perfect knowledge of the system matrices (A, B).

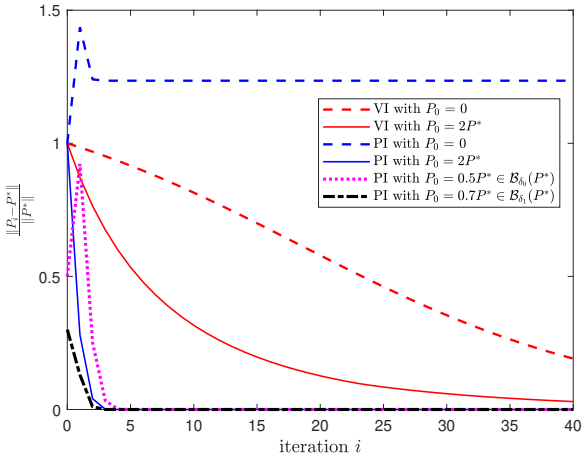

Fig. 2: Convergence of VI and PI

The figure above presents six curves illustrating the convergence behavior of the VI and PI algorithms under different initializations. The blue and red solid lines depict the convergence of the VI and PI algorithms, respectively, with initial condition $P_0 = 2P^* \succeq P^*$, corresponding to Theorem 1 and Theorem 2. When initialized with $P_0 = 0$, VI converges to the optimal solution (red dashed line), consistent with Theorem 3, whereas PI does not (blue dashed line). In the case of a closer initialization ($P_0 = 0.5P^*$) to P^* (magenta dotted line), the sequence $\{P_i\}$ converges to the optimal, and the distance between P_i and P^* decreases monotonicity after the first step, as described in item 1 of Theorem 4. Finally, when the initialization ($P_0 = 0.7P^*$) is even closer to P^*, PI converges monotonically to the optimal solution as shown by the black dash-dotted line, in alignment with item 2 in Theorem 4.

Next, we investigate the robustness properties of the VI and PI algorithms numerically.

[1]The Matlab codes used to generate these results are accessible from the repository: `https://github.com/col-tasas/2024-ConvergenceRobustness-VIPI`

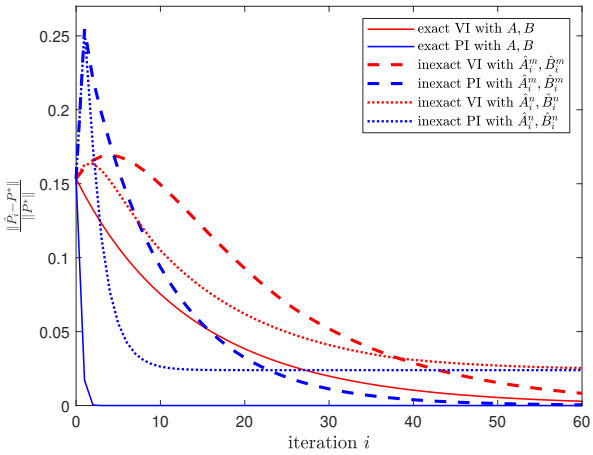

Fig. 3: Robustness of VI and PI

We consider two different scenarios for the estimate system matrices used by the algorithms. In the first case, we have:

$$\hat{A}_i^m = A + 0.9^i \times 0.01I, \ \hat{B}_i^m = B + 0.9^i \times 0.01I,$$

which satisfy the conditions $\lim_{i\to\infty} \hat{A}_i = A$ and $\lim_{i\to\infty} \hat{B}_i = B$. From Figure 3, it is evident that $\{\hat{P}_i\}$ converges to the optimal for both VI and PI, as established in item 3 of Theorem 6 and Theorem 7. In the second case, we have

$$\hat{A}_i^n = A + (0.6^i + 0.1) \times 0.01I,$$
$$\hat{B}_i^n = B + (0.6^i + 0.1) \times 0.01I.$$

As expected, the algorithms converge but do not recover the optimal kernel matrix P^* because of the non-vanishing mismatch in the estimate matrices.

VI. Conclusion

This study contributes a thorough analysis of the convergence and robustness properties of value and policy iteration algorithms within the framework of the linear quadratic regulator problem. We extend the conditions for the exponential convergence of both VI and PI algorithms, which is provided in [10] and, building on them, present input-to-state stability results to evaluate the robustness of the VI and PI algorithms against uncertainties in the system matrices. Additionally, we provide numerical examples to illustrate our analytical findings. In future work, we aim to integrate the robustness analysis with online system identification using noisy data to assess the performance of indirect data-driven VI and PI algorithms.

References

[1] Yu Jiang and Zhong-Ping Jiang. *Robust Adaptive Dynamic Programming*, chapter 5, pages 85–111. John Wiley & Sons, Ltd, 2017.

[2] D. Bertsekas. *Abstract Dynamic Programming: 3rd Edition*. Athena scientific optimization and computation series. Athena Scientific, 2022.

[3] F.L. Lewis and D. Liu. *Reinforcement Learning and Approximate Dynamic Programming for Feedback Control*. IEEE Press Series on Computational Intelligence. Wiley, 2013.

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

[5] Dimitri P. Bertsekas. Value and policy iterations in optimal control and adaptive dynamic programming. *IEEE Transactions on Neural Networks and Learning Systems*, 28(3):500–509, 2017.

[6] Dimitri P. Bertsekas. A new value iteration method for the average cost dynamic programming problem. *SIAM Journal on Control and Optimization*, 36(2):742–759, 1998.

[7] Dimitri Bertsekas. Multiagent value iteration algorithms in dynamic programming and reinforcement learning. *Results in Control and Optimization*, 1:100003, 2020.

[8] Manuel S. Santos and John Rust. Convergence properties of policy iteration. *SIAM Journal on Control and Optimization*, 42(6):2094–2115, 2004.

[9] M. Gargiani, A. Zanelli, D. Liao-McPherson, T. H. Summers, and J. Lygeros. Dynamic programming through the lens of semismooth Newton-type methods. *IEEE Control Systems Letters*, 6:2996–3001, 2022.

[10] Donghwan Lee. Convergence of dynamic programming on the semidefinite cone for discrete-time infinite-horizon LQR. *IEEE Transactions on Automatic Control*, 67(10):5661–5668, 2022.

[11] Youngsuk Park, Ryan Rossi, Zheng Wen, Gang Wu, and Handong Zhao. Structured policy iteration for linear quadratic regulator. In *Proceedings of the 37th International Conference on Machine Learning*. PMLR, 13–18 Jul 2020.

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

[13] Yongliang Yang, Bahare Kiumarsi, Hamidreza Modares, and Chengzhong Xu. Model-free λ-policy iteration for discrete-time linear quadratic regulation. *IEEE Transactions on Neural Networks and Learning Systems*, 34(2):635–649, 2023.

[14] Corrado Possieri and Mario Sassano. Value iteration for continuous-time linear time-invariant systems. *IEEE Transactions on Automatic Control*, 68(5):3070–3077, 2023.

[15] Wenwu Fan and Junlin Xiong. Value iteration for LQR control of unknown stochastic-parameter linear systems. *Systems & Control Letters*, 185:105731, 2024.

[16] G. Hewer. An iterative technique for the computation of the steady state gains for the discrete optimal regulator. *IEEE Transactions on Automatic Control*, 16(4):382–384, 1971.

[17] Bowen Song and Andrea Iannelli. The role of identification in data-driven policy iteration: A system theoretic study. *International Journal of Robust and Nonlinear Control*, 2024.

[18] Farnaz Adib Yaghmaie, Fredrik Gustafsson, and Lennart Ljung. Linear quadratic control using model-free reinforcement learning. *IEEE Transactions on Automatic Control*, 68(2):737–752, Feb 2023.

[19] Bo Pang, Tao Bian, and Zhong-Ping Jiang. Robust policy iteration for continuous-time linear quadratic regulation. *IEEE Transactions on Automatic Control*, 67(1):504–511, 2022.

[20] Bo Pang and Zhong-Ping Jiang. Robust reinforcement learning for stochastic linear quadratic control with multiplicative noise. *IFAC-PapersOnLine*, 54(7):240–243, 2021. 19th IFAC Symposium on System Identification SYSID 2021.

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

[22] C.D. Meyer. *Matrix Analysis and Applied Linear Algebra*. Society for Industrial and Applied Mathematics, 2023.

[23] Donghwan Lee and Jianghai Hu. Primal-dual Q-learning framework for LQR design. *IEEE Transactions on Automatic Control*, 64(9):3756–3763, 2019.

[24] Bowen Song, Chenxuan Wu, and Andrea Iannelli. Convergence and robustness of value and policy iteration for the linear quadratic regulator. arXiv preprint arXiv:2411.04548, 2024.

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

[26] Lorenzo Sforni, Guido Carnevale, Ivano Notarnicola, and Giuseppe 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.