Abstract—In this paper, we investigate the mathematical properties of generalized policy iteration (GPI) applied to a class of continuous-time linear systems with unknown internal dynamics. GPI is a class of dynamic programming method to solve an optimal control problem by using two consecutive steps—policy evaluation and policy improvement. We first provide several formula equivalent to GPI, and as a result, reveal its relations to linear quadratic optimal control problems and the fact that the computational complexity due to back-up operations in policy evaluation steps can be lessened by increasing the time horizon of GPI. A variety of local stability and convergence criteria is also provided with the connection to the convergence speed. Finally, several numerical simulations are performed to verify the results.

I. INTRODUCTION

In the field of computational intelligence, generalized policy iteration (GPI), together with policy iteration (PI) and value iteration (VI), are well-known dynamic programming (DP) algorithms, with extensive practical applications [1]–[4], for computing optimal policies for a finite Markov decision process (MDP) iteratively [1]. These algorithms are closely related to reinforcement learning (RL) [1] and consist of two consecutive interactive steps, the one called policy evaluation making the value function approximately consistent with the current policy, and the other called policy improvement making the policy greedyly in terms of the approximate value function given by policy evaluation step [1]–[3]. The mathematical theories on the convergence and monotonicity of these ones are already well-presented in the literatures [1]–[3], and according to the theories, if these two steps are repeated again and again, one obtains the optimal policy and its corresponding optimal value function.

The key difference among the algorithms lies in the policy evaluation step—PI evaluates the exact value function with respect to the current policy, which requires the infinite number of iterations; VI executes only one-step recursion in policy evaluation, which decreases the computational burden commonly arising in PI, but introduces approximation error of the evaluated value function in return [1]–[4]. Meanwhile, GPI lies between PI and VI—it takes a number $k$ of iterations ($1 \leq k < \infty$) in policy evaluation step to evaluate an approximate value function, making a tradeoff between the accuracy and computational complexity depending on how large $k$ is. Here, we refer to $k$ as the iteration horizon of GPI. For the special case $k=1$, GPI equals to the VI by which only one-step recursion is carried out; for $k=\infty$, the approximate value function goes to the exact one, and GPI technique exactly becomes PI.

Based on the spirit of these algorithms in a finite MDP, extensive researches have been carried out on extending the results in finite MDP frameworks to the dynamic systems in discrete-time (DT) domain [4]–[8] at first, and later, in continuous-time (CT) framework [9]–[16] (see [6] and [12] for a survey). Unfortunately, to the best authors’ knowledge, there are no works on GPI algorithms for DT dynamic systems, similar to those for a finite MDP. In CT framework, only one GPI technique was presented by Vrabie et al. [14], which belongs to a class of algorithms named as interval (or integral) RL (I-RL) [12], [14]. These I-RL methods iteratively perform policy evaluation and improvement steps with the reinforcement signal made by observing the cost during the finite time horizon $T$, to solve a class of infinite-horizon optimal control problems regarding CT input-affine nonlinear systems with unknown internal dynamics [12], [14].

According to the spirit of the algorithms in a finite MDP above, these I-RL methods can be classified into PI [12], [13], VI [11], [12], and GPI [14], where we call these algorithms in this paper integral PI (I-PI), integral VI (I-VI), and integral GPI (I-GPI), respectively.

In addition to the applicability to CT systems with unknown internal dynamics, the advantage of these I-RLs over the others is that the stability, monotonicity, and convergence analysis is well-established. For I-PI, the stability and convergence were proven in general case [12], and in case of linear quadratic regulation (LQR), it was proven in [13] that I-PI is actually equivalent to the Kleinman’s Newton method [17]. This Kleinman’s technique monotonically improves the policy by iterations and guarantees the global stability and convergence to the optimal solution, with local 2nd-order convergence [17]. In case of I-VI for LQR, the local stability and convergence conditions were investigated by Lee et al. [16], with a generalized framework. For I-GPI, with the connection to Banach fixed point theorem, it was proven in [14] that under the admissible policy assumption, the value function approximated by $k$-number of iterations in the policy evaluation step converges to the exact one as $k \to \infty$. However, to the best authors’ knowledge, despite the advantage of GPI mentioned above, mathematical characteristics about I-GPI regarding the stability, monotonicity, convergence, and the relations to the target optimal control problems were not explored even for LQR case. Moreover, over the all I-RL algorithms, there exists no analysis about how the iteration horizon $k$ and time horizon $T$ affect the accuracy.
and computational complexity of the policy evaluations.

This paper deeply focuses on the I-GPI algorithms applied to LQR problems and provides various mathematical results. The main contributions of this paper can be summarized as follows:

- the relationships between the iteration and time horizon $k$ and $T$ are investigated with the connection to the convergence, accuracy, and computational complexity of the policy evaluation steps of I-GPI;
- various local stability, monotonicity, and convergence criteria are suggested for I-GPI algorithm; especially, we explored the speed of convergence of I-GPI, and as a result, gives linear and quadratic convergence criteria;
- we provide several useful equivalent formula for the I-GPI algorithm and their strong connections to LQR; these formula give some valuable insight to understand the I-GPI method for LQR.

Notations: The mathematical symbols used in this paper are summarized as follows:

- $(\cdot)'$: transpose operator;
- $\mathbb{N} = \{1, 2, \cdots\}$: the set of natural numbers;
- $\mathbb{Z}_+ = \{0, 1, 2, \cdots\}$: the set of nonnegative integers;
- $M_{m \times n}^\mathbb{R}$: the set of all $m \times n$ matrices;
- $M_{n \times n}^\mathbb{R}$: the set of all $n \times n$ matrices;
- $M_{n \times n}^+$: the set of all $n \times n$ positive definite matrices;
- $M_{p \times n}^+$: the set of all $p \times n$ positive semidefinite matrices;
- $|A|$: spectral norm for a matrix $A \in M_{m \times n}^\mathbb{R}$;
- $\|x\|$: Euclidean norm for a vector $x \in \mathbb{R}^n$.

Here, all the matrices are assumed to be constant, and the spectral norm and Euclidean norm are defined by a maximum singular value of $M$ and $\|x\| := (x^TXx)^{1/2}$, respectively.

II. RELATED TOPICS ON LQR

In this section, we focus on the topics on LQR, which is closely related to the GPI algorithm—the one concerning the value function $V_\pi(x,t)$ with a stabilizing policy $\pi$, and the other concerning Bellman’s optimality principle with the DP operator. In the first place, we state the following lemma, which will be extensively employed throughout the paper:

**Lemma 1:** For any matrices $X \in M_{m \times n}^\mathbb{R}$ and $Y \in M_{n \times n}^\mathbb{R}$, the following integral formula holds for all $T > 0$:

$$
e^{X^TY}e^{XT} - Y = \int_0^T e^{\tau}(X'Y + YX)e^{X\tau}d\tau.

(1)

**Proof:** Noting that $\frac{d}{dt}e^{Xt} = Xe^{Xt} = e^{Xt}X$, we have

$$
\int_0^T e^{X\tau}(X'Y + YX)e^{X\tau}d\tau = \int_0^T \frac{d}{d\tau}e^{X\tau}Ye^{X\tau}d\tau

= e^{X'T}Ye^{X'T} - Y,

$$
which completes the proof.

In addition, if $X$ is Hurwitz, then (1) can be simplified as

$$
-Y = \int_0^\infty e^{X\tau}(X'Y + YX)e^{X\tau}d\tau

(2)

by letting $T \to \infty$. Together with Lemma 1, this equation will be used to explain the connection of LQR and GPI algorithm.

A. Value Function with a Stabilizing Policy

Now, consider the following CT linear system ($t \geq 0$):

$$
x_t = Ax_t + Bu_t,

(3)

for a state $x_t \in \mathbb{R}^n$, a control input $u_t \in \mathbb{R}^m$, and the matrices $A \in M_{m \times n}^\mathbb{R}$ and $B \in M_{m \times m}^\mathbb{R}$, with the infinite-horizon quadratic value function

$$
V_\pi(x_t,t) = \int_{t}^{\infty} x'_tSx_t + u'_tRu_t d\tau

(4)

where $S \in M_{p \times p}^\mathbb{R}$ and $R \in M_{n \times n}^\mathbb{R}$. Here, throughout the paper, $u(t), u\cdot$, and simply $u$ will be used interchangeably for the input of the system (3) and the triple $(A,B,S^{1/2})$ is assumed to be stabilizable and detectable.

Let $u = -Kx$ be any policy for the system (3) and $A_K$ its corresponding closed loop matrix $A - BK$. Defining $Q_K$ for a policy $K$ as $Q_K := S + K'RK$ for simplicity, then, we can represent $V_\pi(x_t,t)$ in terms of $Q_K$ as

$$
V_\pi(x_t,t) = x'_t\left(\int_{t}^{\infty} e^{K(t-\tau)}Q_Ke^{K(\tau-\tau)}d\tau\right)x_t = x'_tP_Kx_t

(5)

where $P_K$ is defined as

$$
P_K := \int_0^\infty e^{K\tau}Q_Ke^{K\tau}d\tau.

(6)

Here, we have used $x_t = e^{K(t-\tau)x_t}$ and change of variables. If $u = -Kx$ is a stabilizing policy, then, the value function (4) is finite [17], making the problem feasible. Now, by applying (2) to $P_K$ with $X = Ax$, (5) can be rewritten as

$$
\int_0^\infty e^{K\tau}Ric(K)Ke^{K\tau}d\tau = 0

(7)

for a stabilizing $u = -Kx$, where $Ric(K)(P_K)$ is defined as

$$
Ric(K)(P_K) := A_K'P_K + P_KA_K + Q_K.

(8)

Note that (6) always holds for all stabilizing $K$ and $P_K$. This implies the pair $(K,P_K)$ always satisfies the Lyapunov equation $Ric(K)(P_K) = 0$. Such $P_K$ always exists uniquely for any given stabilizing $K$ and $Q_K \in M_{n \times n}^+$ [18]. Therefore, for any stabilizing $K$, one can always find the corresponding value function $V_\pi(x) = x'_TP_Kx$ by solving the Lyapunov equation $Ric(K)(P_K) = 0$.

B. DP Operator & Optimality Principle

Regarding the system dynamics (3), we define the dynamic programming operator $\mathcal{K}_F : X \rightarrow X$ on the space $X$ of the continuous functionals $V(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ at fixed time $t \geq 0$ as

$$
\mathcal{K}_F V(x) := \int_t^{t+T} x'_tQ_Kx_t d\tau + V(x_{t+T}).

(9)

where the trajectories of $x_t$ are generated by the system (3) with a given control $u = -Kx$. We also define $\mathcal{K}_F^2$ as $(\mathcal{K}_F^2) V(x) := \mathcal{K}_F[\mathcal{K}_F V(x)]$, and so does $\mathcal{K}_F^k$ at fixed time $t \geq 0$ for any $k \in \mathbb{N}$. This operator simplifies mathematical statements related with the optimality principle and I-GPI algorithm. Moreover, it also possesses the following useful mathematical properties:
Lemma 2: Consider the system dynamics (3) with \( u = -Kx \) and a continuous functional of the form \( V(x) = x^TPx \). Then, we have

\[
\mathcal{K}V(x) = x_i^T \left( P + \int_0^T e^{\kappa \tau} Ric_K(P) e^{\kappa \tau} \, d\tau \right) x_i,
\]
\[
\frac{d}{dt} \mathcal{K}V(x) = x_i^T \left( e^{\kappa \tau} Ric_K(P) e^{\kappa \tau} - Ric_K(P) \right) x_i + x_i^T (A_kP + PA_k)x_i.
\]  

Proof: First, consider the following of (7):

\[
\mathcal{K}V(x) = \int_0^T x_i^T QKx_i + x_i^T P_{t+T} x_i + x_i^T P_{t+T} x_i - x_i^T Ric_K(P) x_i + x_i^T (A_kP + PA_k)x_i.
\]

Now, applying Lemma 1 to \( e^{\kappa \tau} P e^{\kappa \tau} - P \) yields (8), the proof of first part. Next, note that the direct differentiation of (7) leads to

\[
\frac{d}{dt} \mathcal{K}V(x) = x_i^T QKx_i + x_i^T (A_kP + PA_k)x_i.
\]

Using the operator \( \mathcal{K} \), the exact value function \( V(x) = x^TPKx \) for a stabilizing \( u = -Kx \) can be expressed as

\[
V_i(x_i) = \int_0^T x_i^T QKx_i + x_i^T (A_kP + PA_k)x_i.
\]

Similar expression is also possible for the equation on the Bellman’s optimality principle [19]:

\[
V^*(x_i) = \min_K \mathcal{K}V^*(x)
\]  

III. GENERALIZED POLICY ITERATION

In this section we briefly discuss the ordinary GPI algorithm, and gives some mathematical properties for I-GPI algorithm including stability, monotonicity, and convergence. Generally, GPI consists of two successive steps—policy evaluation and policy improvement, and can be described as follows:

--- Algorithm 1: GPI ---

1: \( i \leftarrow 0 \)
2: Initialize \( P_0 \in \mathbb{M}_P^{n \times n} \) and let \( K_0 \leftarrow -R^{-1}B'P_0 \).
3: \( \text{do} \)
4: **Policy Evaluation:**
5: For a policy \( K_i \), find \( P_{i+1} \) which is an approximate of \( P_K \) satisfying \( Ric_{K_i}(P_K) = 0 \).
6: **Policy Improvement:**
7: \( K_{i+1} \leftarrow -R^{-1}B'P_{i+1} \)
8: \( i \leftarrow i + 1 \)
9: \( \text{until } \|P_{i+1} - P_i\| < \varepsilon \).

This GPI is highly related to the equations \( Ric_{K_i}(P_K) = 0 \) and (12). In policy evaluation step (line 14), it tries to minimize the norm \( \|Ric_{K}(P_{i+1})\| \) and as a result, gives an approximate solution \( P_{i+1} \) of \( P_K \) satisfying \( Ric_{K_i}(P_K) = 0 \). In policy improvement step (line 5), it updates \( K_{i+1} \) based on \( P_{i+1} \) to improve the policy \( K_{i+1} \) over \( K_i \), that is, to achieve, for example, \( \|Ric_{K_{i+1}}(P_{i+1})\| < \|Ric_{K_i}(P_i)\| \). This achievement indeed implies the improvement of the policy since the pair \( (P_i, K_i) \) always satisfies \( Ric(P_i) = Ric_{K_i}(P_i) \) by (12) and \( Ric(P_i) = 0 \) holds whenever \( P_i = P_K \). In line 7, some exploration signal is injected to the system (3) through \( u \) to hold the excitation condition which is necessary for the computation of \( P_i \) [7], [11]–[16].

A. Integral GPI with DP Operator

The I-GPI is a class of the GPI methods, given in [14], to solve a given optimal control problem without knowing the system internal dynamics. In this paper, we only pay attention to the application to LQR. The basic operation of this algorithm is the one-step recursion at time \( t \geq 0 \), represented by

\[
V_{i+1}(x_i) = \mathcal{K}V_{ij}(x_i)
\]  

where \( i \in \mathbb{Z}_+ \) is the iteration number, \( j \in \mathbb{Z}_+ \) is the recursion index at \( i \)-th iteration, and \( V_{ij}(x_i) \) is a functional defined as \( V_{ij}(x_i) := x_i^TP_{ij}x \) for a matrix \( P_{ij} \in \mathbb{M}_P^{n \times n} \) indexed by \( (i, j) \). This one-step recursion (13) actually comes from the approximation of optimality principle (11), where \( V_{ij}(x_i) \) on the right hand side of (11) is replaced by \( V_{ij}(x_i) \), assumed to be the most accurate approximate of \( V_{ij} \) at \( (i, j) \)-th iteration. Now, the policy evaluation and improvement step of I-GPI applied to LQR can be derived from (13) as follows:
where $V_i(x)$ is defined as $V_i(x) := x^TP_i x$ for a indexed matrix $P_i \in \mathbb{M}^{n \times n}$ at $i$-th iteration. Here, the iteration horizon $k$ represents the number of recursions (13) executed at each policy evaluation step. In [14], the authors mentioned that the policy evaluation (14) is a fixed point iteration, and proved the convergence to the exact value function $V_\infty$ as $k \rightarrow \infty$, provided that the policy $K_i$ is admissible. If $k = 1$, this I-GPI is actually the same as I-VI method [11], and as $k \rightarrow \infty$, I-GPI algorithm becomes the well-known I-PI [12], whenever (14) converges to a fixed point. This I-PI guarantees global stability and convergence [12] and is shown below:

In this I-PI, policy evaluation step (16) exactly evaluates the value function $V_{i+1}(x_i)$ for the current policy $K_i$, which is same to the exact formula (10). In case of I-GPI with finite $k$, $V_{i+1}(x_i)$ can be an approximate of $V_K(x_i)$. Note that the error $|V_{i+1}(x_i) - V_K(x_i)|$ can be made arbitrarily small by adjusting $k$ if $V_i(x_i)$ converges to $V_K(x_i)$ as $k \rightarrow \infty$. However, the large $k$ introduces heavy computational burdens and hence, make the algorithm hard to implement in practice.

B. Policy Evaluation Step of I-GPI

We now mathematically explores policy evaluation step of I-GPI, and as a result, provides useful equivalent formula and convergence property, with the connection to the update horizon $\gamma > 0$ defined as a product of the iteration horizon $k$ and time horizon $T$, that is, $\gamma := kT$. By using (13), the policy evaluation (14) of I-GPI can be represented as

Polcy Evaluation: $V_{i+1}(x_i) = (\mathcal{K}_i) (\mathcal{K}_i) V_{i+1}(x_i)$

where $V_{i=0}(x_i) := V_{i+1}(x_i)$ and $V_{i=0}(x_i) := V_{i+1}(x_i)$. For notational convenience, we define $A_i$ as the matrix of $i$-th closed-loop system $A_i := A_K$ and $M_{ij}$ as

\[ M_{ij} := \int_0^T e^{A_j t} RicK_i (P_{ij}) e^{A_i T} dt. \]

Consider the general $k$-th order recursive mapping

\[ V_{i+k+1}(x_i) = (\mathcal{K}_i)^k V_{i+k}(x_i). \]

If some properties regarding (19) are proven, then, they also holds for $V_{i+k}(x_i)$ and $V_{i+k}^T(x_i)$ satisfying (18) as a special case. Here is the main theorem concerning $k$-th order recursive mapping (19) and its convergence:

**Theorem 1:** Consider the mapping (19) with the system (3). Then, for any $k \in \mathbb{N}$, $j \in \mathbb{Z}_+$, and $T > 0$, the mapping

\[ V_{i+j+k}(x_i) = (\mathcal{K}_i)^j V_{i+k}(x_i) \]

is equivalent to the followings:

1. $RicK_i (P_{ij+k}) = e^{A_k T} RicK_i (P_{ij+k}) e^{A_j T}$
2. $P_{ij+k} = P_{ij} + \int_0^T e^{AT} RicK_i (P_{ij}) e^{A^T} d\tau$
3. $P_{ij+k} = P_{ij} - \left( RicK_i (P_{ij}) \right)^{-1} \times \left[ RicK_i (P_{ij}) - e^{A_k T} RicK_i (P_{ij}) e^{A_j T} \right]

where $RicK_i (P_{ij})$ denotes the Frechet derivative of $RicK_i (P_{ij})$ taken with respect to $P_{ij}$. Moreover, if $A_i$ is Hurwitz, then $V_{i+j+k}(x_i)$ converges to the exact value function $V_\infty(x_i)$ as $\gamma := kT \rightarrow \infty$.

**Proof:** First, consider the one-step mapping $V_{i+j+1}(x_i) = (\mathcal{K}_i) V_{i+j}(x_i)$. Then, by (8) in Lemma 2, we have

\[ P_{ij+1} = P_{ij} + \int_0^T e^{AT} RicK_i (P_{ij}) e^{A^T} d\tau. \]

That is, $P_{ij+1} = P_{ij} + M_{ij}$ in short. Now, by using this matrix equation, $RicK_i (P_{ij} + M_{ij})$ can be expressed in terms of $RicK_i (P_{ij})$ as follows:

\[ RicK_i (P_{ij} + M_{ij}) = A_i P_{ij+1} + P_{ij+1} A_i + K_i R K_i + Q = RicK_i (P_{ij}) + A_i M_{ij} + M_{ij} A_i. \]

where $A_i M_{ij} + M_{ij} A_i$ can be rewritten as

\[ A_i M_{ij} + M_{ij} A_i = e^{AT} RicK_i (P_{ij}) e^{A^T} - RicK_i (P_{ij}). \]

Here, we used $A_i e^{AT} = e^{AT} A_i$ and Lemma 1. Substituting this into (22), we have

\[ RicK_i (P_{ij+1}) = e^{AT} RicK_i (P_{ij}) e^{A^T}, \]

which is equivalent to the one-step mapping $V_{ij+1}(x_i) = (\mathcal{K}_i) V_{ij}(x_i)$. Therefore, (20) can be easily derived by recursively applying this relation as

\[ RicK_i (P_{ij+k}) = e^{AT} RicK_i (P_{ij+k-1}) e^{A^T} = (e^{AT})^2 RicK_i (P_{ij+k-2}) (e^{A^T})^2 \]

\[ \vdots \]

\[ = (e^{AT})^k RicK_i (P_{ij}) (e^{A^T})^k. \]

Next, we prove the equivalence between (21) and the mapping $V_{ij+k}(x_i) = (\mathcal{K}_i)^j V_{ij+k}(x_i)$. By employing $P_{ij+1} = P_{ij} + M_{ij}$ to $P_{ij+k}$ repetitively, one has

\[ P_{ij+k} = P_{ij+k-1} + M_{ij+k} = P_{ij+k-2} + M_{ij+k-2} + M_{ij+k-1} \]

\[ \vdots \]

\[ = P_{ij} + \sum_{l=0}^{k-1} M_{ij+l}. \]
Here, by (24), $M_{i,j+l}$ is
\[
M_{i,j+l} = \int_{0}^{\tau} e^{\Delta t \cdot \delta_T} R_{i,j}(P_{i,j+l-1}) e^{\Delta_t \cdot \delta_T} d\tau \\
= \frac{\delta_T}{\gamma} e^{\Delta t \cdot \delta_T} R_{i,j}(P_{i,j+l-1}) e^{\Delta_t \cdot \delta_T} d\tau \\
= \frac{\delta_T}{\gamma} e^{\Delta t \cdot \delta_T} R_{i,j}(P_{i,j+l-1}) e^{\Delta t \cdot \delta_T} d\tau \\
\vdots \\
= \int_{0}^{\theta(T+1)^T} e^{\Delta t \cdot \delta_T} R_{i,j}(P_{i,j}) e^{\Delta t \cdot \delta_T} d\tau.
\]
Therefore, we have
\[
\sum_{l=0}^{k-1} M_{i,j+l} = \int_{0}^{\theta(T+1)^T} e^{\Delta t \cdot \delta_T} R_{i,j}(P_{i,j}) e^{\Delta t \cdot \delta_T} d\tau,
\]
which implies the equivalence between (21) and $V_{i,j+l}(x) = (\mathcal{R}_k)^l V_{i,j}(x)$. For the proof of (22), take the time derivative of the one-step mapping $V_{i,j+l}(x) = \mathcal{R}_k V_{i,j}(x)$ and employ (9) in Theorem 2. Then, one obtains
\[
A_i^j P_{i,j+l+1} + P_{i,j+l+1} A_i = e^{\Delta t \cdot \delta_T} R_{i,j}(P_{i,j}) e^{\Delta t \cdot \delta_T} - R_{i,j}(P_{i,j}) \\
= A_i^j P_{i,j+l} + P_{i,j+l} A_i \\
(25)
\]
which holds for all $j \in \mathbb{Z}_+$. By iteratively applying (24)–(25) to $A_i^{j+l+j+k} + P_{i,j+k} A_i$, one obtains
\[
A_i^j P_{i,j+k} + P_{i,j+k} A_i \\
= e^{\Delta t \cdot \delta_T} R_{i,j}(P_{i,j+k-1}) e^{\Delta t \cdot \delta_T} - R_{i,j}(P_{i,j+k-1}) \\
= A_i^j P_{i,j+k-1} + P_{i,j+k-1} A_i \\
= e^{\Delta_t \cdot \delta_T} R_{i,j}(P_{i,j+k-2}) (e^{\Delta t \cdot \delta_T})^2 - R_{i,j}(P_{i,j+k-2}) \\
= A_i^j P_{i,j+k-2} + P_{i,j+k-2} A_i \\
\vdots \\
= (e^{\Delta t \cdot \delta_T})^k R_{i,j}(P_{i,j}) (e^{\Delta t \cdot \delta_T})^k - R_{i,j}(P_{i,j}) + A_i^j P_{i,j+k-1} + P_{i,j+k} A_i \\
(26)
\]
which is exactly same to the k-order recursive mapping (19) since we only employed (24)–(25) equivalent to the one step mapping $V_{i,j+l}(x) = \mathcal{R}_k V_{i,j}(x)$. Finally, rewriting (26) as
\[
A_i^j (P_{i,j+k} - P_{i,j}) + (P_{i,j+k} - P_{i,j}) A_i \\
= e^{\Delta t \cdot \delta_T} R_{i,j}(P_{i,j}) e^{\Delta t \cdot \delta_T} - R_{i,j}(P_{i,j}) \\
(27)
\]
yields (22) which is just another expression of (27) [11].

Now, we prove the convergence of the mapping (19). Note that if the term $e^{\Delta t \cdot \delta_T} R_{i,j}(P_{i,j}) e^{\Delta t \cdot \delta_T}$ converges to zero, (25) goes to the iteration
\[
P_{i,j+k} = P_{i,j} - (R_{i,j+k})^{-1} R_{i,j}(P_{i,j}), \\
(28)
\]
which is exactly the Kleinman’s Newton method [13], [17]. This $e^{\Delta t \cdot \delta_T} R_{i,j}(P_{i,j}) e^{\Delta t \cdot \delta_T} \rightarrow 0$ happens exactly when $kT$ ($= \tau$) goes to infinity and $A_i$ is Hurwitz. Therefore, whenever $\gamma \rightarrow \infty$, (25) becomes (28) which is equivalent to the PI algorithm [13]. Since policy evaluation step of PI algorithm exactly calculates the value function $V_{i,j+k}(x) = x^T P_{i,j+k} x$, the function $V_{i,j+k}(x) = x^T P_{i,j+k} x$ goes to $V_n(x)$ as $\gamma \rightarrow \infty$, which completes the proof.

Note that the update term in (21) is the same as the term in (6) except the update horizon $\gamma (= kT)$ is finite and $P_{i,j+k}$ is replaced with $P$. Obviously, this update term turns out to be zero when $P_i$ equals to the exact value function $P_k$. By increasing $\gamma$, one can increase the update horizon of the integral. Furthermore, the increase of the update horizon $\gamma$ can actually reduce the error $||P_{i,j+k} - P_k||$ when $A_i$ is Hurwitz. Note that by the convergence argument, with Hurwitz matrix $A_i$, $\forall k > 0$, $\exists \gamma^* \in \mathbb{N}$ such that $\forall \gamma \geq \gamma^*$, $||P_{i,j+k} - P_k|| < \varepsilon$ holds. This can be also seen from (20) where sufficiently large $\gamma$ makes the term $e^{\Delta t \cdot \delta_T}$ and thus $R_{i,j+k}$ arbitrarily small when $A_i$ is Hurwitz. The Hurwitzness of $A_i$ is certainly required for both the existence of $P_k$ and the convergence of $P_{i,j+k}$ to $P_k$.

Remark 1: According to (20)–(22) in Theorem 1, all of the mappings $V_{i,j+k}(x) = (\mathcal{R}_k)^l V_{i,j}(x)$ with the same update horizon $\gamma \in \mathbb{R}$ are actually all equivalent and have the same convergence speed. This means that the computational complexity due to the large iteration horizon $k$ can be lessened by increasing the time horizon $T$ for the same convergence speed.

Now, we present a statement concerned with the policy evaluation (18) and its convergence. The proof can be directly verified by Theorem 1.

Corollary 1: Consider the policy evaluation step (18)—$V_{i,j+k}(x) = (\mathcal{R}_k)^l V_{i,j}(x)$. Then, it satisfies (20)–(22) with $j = 0$. Moreover, if $A_i$ is Hurwitz, then, $V_{i,j+k}(x)$ converges to the exact value function $V_n(x)$ as $\gamma \rightarrow \infty$.

Remark 2: In [14], the convergence of $V_{i,j+k}(x)$ to $V_n(x)$ was proven with respect to $k \in \mathbb{N}$, based on Banach’s fixed point theorem. On the other hand, Corollary 1 shows that the convergence result can be extended with respect to the update horizon $\gamma \in \mathbb{R}$, without employing such Banach’s theorem.

C. Stability & Convergence of I-GPI Algorithm

Based on the results from Section III-B, we derive the local stability and convergence of the I-GPI algorithm (14)–(15). First, for notational convenience, define $\Phi(i,k)$ and $M(i,k)$ as
\[
\Phi(i,k) := \int_{0}^{\theta(T+1)^T} \|e^{\Delta t \cdot \delta_T}\|^2 d\tau, \\
M(i,k) := \int_{0}^{\theta(T+1)^T} e^{\Delta t \cdot \delta_T} R_{i,j}(P_{i,j}) e^{\Delta t \cdot \delta_T} d\tau.
\]

In the policy improvement step (15), $K_i$ is determined by $K_i = -R_i^{-1} B_i P_i$. Therefore, by incorporating this into the results in Section III-B, one obtains the following equivalent formulas:

...
Proposition 1: Consider the I-GPI algorithm (14)–(15). Then, it is equivalent to the following iterative forms:

1) \( \text{Ric}(P_{i+1}) = e^{A_{i}(kT)P} e^{A_{i}(kT)} - M_{i,k}BR^{-1}B'M_{i,k} \) \( \quad (31) \)

2) \( P_{i+1} = P_i + \int_0^{kT} e^{A_{i}^*}\text{Ric}(P_i)e^{A_{i}\tau} d\tau \) \( \quad (32) \)

3) \( P_{i+1} = P_i - \frac{1}{\text{Ric}(P_i) - e^{A_{i}(kT)P}} e^{A_{i}(kT)P} \) \( \quad (33) \)

Proof: The equivalence to (32) and (33) can be easily derived by substituting \( K_i = -R^{-1}B'P_i \) into (21) and (22), respectively. For the proof of (31), consider \( \text{Ric}(P_{i+1}) \) with its expansion:

\[
\text{Ric}(P_{i+1}) = \text{Ric}(P_i) + A_{i}^*M_{i,k} + M_{i,k}A_{i} - M_{i,k}BR^{-1}B'M_{i,k}
\]

(34)

where \( A_{i}^*M_{i,k} + M_{i,k}A_{i} \) can be represented as \( A_{i}^*M_{i,k} + M_{i,k}A_{i} \) = \( e^{A_{i}(kT)P}e^{A_{i}(kT)} - \text{Ric}(P_i) \) by Lemma 1 and \( A_i e^{A_{i}T} = e^{A_{i}T}A_i \). By substituting this into (34), we have (31), which completes the proof.

In comparison to (20), the term \(-M_{i,k}BR^{-1}B'M_{i,k}\) appears in the equation (31) which is caused by the policy improvement (15). The equation (31) plays a central role in the proof of convergence, and so do the two lemmas presented below:

Lemma 3: For any \( (i,k) \in \mathbb{Z}_+^2, \Phi_{i,k} \) and \( M_{i,k} \), defined by (29) and (30) respectively, satisfy the following inequality:

\[
\|M_{i,k}\| \leq \Phi_{i,k}\|\text{Ric}(P_i)\|
\]

(35)

Proof: This can be directly proven by the use of the property of matrix norm and integral as follows:

\[
\|M_{i,k}\| \leq \int_0^{kT}\|e^{A_{i}^*Ric(K_i)}e^{A_{i}\tau}\| d\tau 
\leq \left(\int_0^{kT}\|e^{A_{i}\tau}\|^2 d\tau\right)\|\text{Ric}(K_i)\|
\]

(36)

Lemma 4: Let \( P_{i+1} \) obtained by I-GPI (14)–(15) algorithm converges to \( P^* \). Then, \( P^* = P_K \) holds. That is, \( P^* \) is the LQ optimal solution satisfying \( \text{Ric}(P^*) = 0 \).

Proof: Since \( \{P_i\} \) is assumed to converge to \( P^* \), taking limit of (32) yields

\[
0 = \lim_{P_i \to P^*} (P_{i+1} - P_i) = \lim_{P_i \to P^*} \int_0^{kT} e^{A_{i}^*\text{Ric}(P_i)e^{A_{i}\tau} d\tau}, \quad (36)
\]

which implies \( \text{Ric}(P^*) = 0 \). Since \( (A,B,Q^{1/2}) \) is stabilizable and detectable, \( \text{Ric}(P^*) = 0 \) has a unique solution. Therefore, \( P^* = P_K \) holds.

Now, we state the local stability and convergence of I-GPI. For a precise statement, we define linear and quadratic convergence as follows:

Definition 1: A sequence of matrix \( \{P_i\} \) is said to converge to the solution \( P_K \), linearly (resp. quadratically) in a set \( \Omega \subset \Omega \) if it locally converges to \( P_K \) in \( \Omega \), and have the property \( \|\text{Ric}(P_{i+1})\| \leq \|\text{Ric}(P_i)\| \) (resp. \( \|\text{Ric}(P_{i+1})\| < \|\text{Ric}(P_i)\| \)) whenever \( P_i \in \Omega \). We also say that \( \{P_i\} \) linearly (resp. quadratically) converges to \( P_K \) if \( \Omega = \Omega \).

Theorem 2: Consider the I-GPI algorithm (14)–(15) with the system (3) and define the bounds \( C_i, D_i, \) and \( E_i \) as \( C_i := \langle 2\|BR^{-1}B'\|\|Y\|\Phi_{i,k}\rangle^{-1}\),

\[
D_i := 1 - \frac{\|e^{A_{i}(kT)}\|^2}{\|BR^{-1}B'\|^2\Phi_{i,k}^2}, \quad E_i := \frac{\|e^{A_{i}(kT)}\|^2}{1 - \|BR^{-1}B'\|^2\Phi_{i,k}^2},
\]

respectively. Suppose \( \Omega_i \) and \( \Omega_i \) be the sets defined as

\[
\Omega_i := \{P \in \mathbb{R}^{n \times n} : \|\text{Ric}(P)\| \leq C_i \}, \quad \Omega_i := \{P \in \mathbb{R}^{n \times n} : E_i < \|\text{Ric}(P)\| < 1 \},
\]

respectively. Then, for all \( i \in \mathbb{Z}_+ \),

1. \( (stability) A_{i+1} \) is Hurwitz when \( \|\text{Ric}(P_i)\| \leq C_i \) holds;

2. \( (1^{\text{st}}\text{-order monotonicity}) \) if \( P_i \in \Omega_i \) at \( i \)-th iteration, then \( P_{i+1} \) satisfies \( \|\text{Ric}(P_{i+1})\| < \|\text{Ric}(P_i)\| \);  

3. \( (2^{\text{nd}}\text{-order monotonicity}) \) if \( \|BR^{-1}B'\|^2\Phi_{i,k}^2 \neq 1 \) and \( E_i < \|\text{Ric}(P_i)\| \) is satisfied at \( i \)-th iteration, then \( P_{i+1} \) satisfies \( \|\text{Ric}(P_{i+1})\| < \|\text{Ric}(P_i)\|^2 \);  

4. \( (\text{linear convergence & quadratic decreasing}) \) if \( P_i \in \Omega_i \) for all \( i \in \mathbb{Z}_+ \), then \( P_i \) linearly converges to \( P_K \); moreover, if \( D_i \) is larger than \( 1 \) (\( D_i \)), then, \( \Omega_i \notin \emptyset \) and \( \Omega_i \subset \Omega_i \) holds; in this case, \( P_i \) converges to \( P_K \), quadratically in \( \Omega_i \).

5. \( (\text{quadratic convergence of policy iteration}) \) if \( \{P_i\} \) is generated by policy iteration (16)–(17), then it quadratically converges to \( P_K \) whenever \( P_0 \in \Omega_0 \) (\( \Omega_0 = \Omega_E \)) and \( \text{A}_0 \) is Hurwitz.

Proof: For the proof of the stability part, follow the same procedure given in (16), with \( M_{i,k} \) defined in (30) (see the proof of Theorem 1 [16]). For the proof of the monotonicity, take the matrix norm \( || \cdot || \) to (31) in Proposition 1, and employ Lemma 3 and the properties of the norm as follows:

\[
\|\text{Ric}(P_{i+1})\| 
\leq \|e^{A_{i}(kT)P} e^{A_{i}(kT)}\| + \|M_{i,k}BR^{-1}B'M_{i,k}\| 
\leq \|e^{A_{i}(kT)}\|^2 \|\text{Ric}(P_i)\| + \|M_{i,k}\|^2 \|BR^{-1}B'\| 
\leq \|e^{A_{i}(kT)}\|^2 \|\text{Ric}(P_i)\| + \Phi_{i,k}^2 \|BR^{-1}B'\| \|\text{Ric}(P_i)\| \| \quad (37)
\]

By applying \( \|\text{Ric}(P_i)\| < D_i \) to (37), we prove the 1\text{-}\text{st-order monotonicity} \( \|\text{Ric}(P_{i+1})\| < \|\text{Ric}(P_i)\| \). Next, assume \( \|\text{Ric}(P_i)\| < E_i \) and consider the following inequality from (37):

\[
\|\text{Ric}(P_{i+1})\| \leq \left(\frac{\|e^{A_{i}(kT)}\|^2}{\|\text{Ric}(P_i)\|} + \Phi_{i,k}^2 \|BR^{-1}B'\| \right) \|\text{Ric}(P_i)\| \| \quad (38)
\]
Then, we have $\|\text{Ric}(P_{i+1})\| < \|\text{Ric}(P_i)\|^2$ by inverting the inequality $\|\text{Ric}(P_i)\| < E_i$ and applying it to (38), the proof of the 2nd-order monotonicity.

In the sequel, we will focus on the convergence of $P_i$ if $\|\text{Ric}(P_i)\| < D_i$ holds for all $i \in \mathbb{Z}_+$, then by 1st-order monotonicity $\|\text{Ric}(P_{i+1})\| < \|\text{Ric}(P_i)\|$ and lower boundedness of $\|\text{Ric}(P_i)\|$ by zero, $\|\text{Ric}(P_i)\|$ converges, and so does $\{\text{Ric}(P_i)\}$ with this topology. Therefore, by Lemma 4, we conclude that $P_i$ linearly converges to $P_K$, whenever $P_i \in \Omega_{D_i}$ for all $i \in \mathbb{Z}_+$.

For the proof of quadratic decreasing, suppose $D_i > 1$ and rearrange the inequality. Then, one obtains $1 - \|BR^{-1}B^t\| \cdot P_i > 0$ and $E_i < 1$. Therefore, $\Omega_{E_i} \neq \emptyset$ is valid, and thus, by 2nd-order monotonicity, $\|\text{Ric}(P_{i+1})\| < \|\text{Ric}(P_i)\|^2$ holds whenever $P_i \in \Omega_{E_i}$. In this case, $\Omega_{E_i} \subseteq \Omega_{D_i}$ is obvious by $E_i < 1 < D_i$. Note that $P_i \rightarrow P_K$ if $P_i \in \Omega_{D_i}, \forall i \in \mathbb{Z}_+$ by linear convergence argument. Then, it is obvious that $P_i$ converges to $P_K$, quadratically in $\Omega_{E_i}$ ($\subseteq \Omega_{D_i}$).

Now, note that if $A_i$ is Hurwitz and $\gamma \rightarrow \infty$, $E_i$ goes to zero and I-GPI (14)–(15) becomes the I-PI (16)–(17). Since I-PI yields Hurwitz $A_i$, when $A_0$ is Hurwitz, one can assume $A_0$ is Hurwitz without loss of generality. Now, considering a topology on $\{\Omega_{E_i}: 0 \leq E < 1\}$ generated by the metric $d(\Omega_{E_i}, \Omega_{E_j}) = |E_i - E_j|$, one can see $\lim_{t \rightarrow \infty} \Omega_{E_i} = \Omega_0$. Therefore, by the argument of quadratic decreasing, $\{P_i\}$ generated by (16)–(17) quadratically converges to $P_K$ whenever $P_0 \in \Omega_0$, which completes the proof.

Remark 3: Although the monotonicity and convergence were proven independently of the stability of $A_i$, it is actually related to the existence of $P \in \mathbb{R}^{n \times n}$ such that $0 < \|\text{Ric}(P)\| < D_i$ ($P \in \Omega_{D_i}$) holds. Note that to satisfy $0 < D_i$ for any $i \in \mathbb{N}$ and any $T > 0$, $A_i$ should be at least Hurwitz, which is the connection between $C_i$ and $D_i$.

Remark 4: As is well-known in the literature [17], I-PI (16)–(17) guarantees quadratic convergence in the vicinity of $\text{Ric}(P_K) = 0$. In this article, a concrete set $\Omega_0$ around $P_K$ is provided in which the I-PI (16)–(17) achieves quadratic convergence.

IV. NUMERICAL RESULTS

To verify the claims raised in Section III, we simulated the I-GPI (14)–(15) applied to a step-down converter for various $k \in \mathbb{N}$. The step-down converter dynamics is as follows [20]:

$$ A = \begin{bmatrix} 0 & -1/L \\ 1/C & -1/R_C \end{bmatrix}, \quad B = \begin{bmatrix} V_o/L \\ 0 \end{bmatrix}. $$

where the parameters represent the electric inductance $L$ and capacitance $C$, the output load resistance $R_o$, and the input voltage $V_o$ of the step-down converter. In this simulation, these parameters were set to $L = 200$ [\mu F], $C = 200$ [\mu F], $R_o = 25$ [\Omega], and $V_o = 24$ [V], and we considered the performance index (4) with $S = \text{diag}(5,1)$ and $R = 300$. With these settings, $P_K$, associated with the optimal value function $V_o(x)$ can be evaluated as

$$ P_K = \begin{bmatrix} 0.3418 & 0.0606 \\ 0.0606 & 0.5431 \end{bmatrix} \times 10^{-3}. $$

Furthermore, for the implementation of I-GPI algorithm (14)–(15), the batch least squares, already used in [11]–[16], is adopted for the calculation of $P_{i+1}$ in policy evaluation. For $k = \infty$, $P_{i+1}$ is updated by the least squares method presented in [13]. In the case of the converter (39), the batch least squares solver should gather at least 3 number of data samples $(x_i, u_i)$ to calculate $P_{i+1}$ at each $i$-th policy evaluation (line 4 of Algorithm 1), and hence we collect 5 data samples per each policy evaluation step to obtain $P_{i+1}$. After each policy improvement step is performed by using $P_{i+1}$ (line 5 of Algorithm 1), an exploration signal $w(t) = 10^{-2} \sin 2\pi f t$ with $f = 50$ [kHz] is applied for one period $T$ through $u$ (line 7 of Algorithm 1). This exploration helps prevent $x_i$ from being stationary and hence maintain the excitation condition.

Fig. 1 illustrates the trajectories of the converged parameters $P_{11}, P_{12},$ and $P_{22}$ for various $k \in \mathbb{N}$ with the time horizon $T = 40$ [ms] and initial policy $u_0 \equiv 0$. As can be seen from the figures, the convergence speeds of $P_i$ are lowest when $k = 1$ (I-VI case). On the other hand, as $k \in \mathbb{N}$ is increased, the convergence to the solution $P_K$ tends to be achieved more rapidly than the case $k = 1$, but introduces much higher overshoots at $i = 1$ especially when $k \geq 10$. Note that the largest overshoots appear when $k = \infty$ (I-PI case), and in this...
case, the convergence speeds do not seem to be significantly improved in comparison to the case $k \geq 10$. Therefore, the choice of suitable $k$ can be a main issue which achieves an appropriate trade-off between the convergence speeds and the degree of the overshoots.

Fig. 2 describes the evolutions of $P_i$ when $u_0 \equiv 0$, $k = 1$, and $T = 40 \times 5$ [ms]. Comparing Fig. 2 with the case $k = 5$ and $T = 40$ [ms] in Fig. 1 in iteration domain, one can see that both $P_i$’s evaluated by (14) with the same update horizon $\gamma = 1 \times 5 \times 40 = 200$ [ms] are exactly same to each other. This verifies the claims from Theorem 1 and Proposition 1—I-GPI algorithms (14)–(15) with the same update horizon $\gamma$ are all equivalent, and thus, computational complexity due to the large $k$ can be lessened by increasing $T$.

To investigate how much the bounds $C_i$, $D_i$, and $E_i$ are conservative, additional simulations are carried out for the different $\gamma$. Since Theorem 2 provides the local stability and convergence in the vicinity of Ric($P_{k\gamma}$) = 0, we employ $K_0 = -R^{-1}B^TP_0$ as the initial policy $u_0$, where $P_0$ is the solution of ARE with $A$ replaced by the nominal matrix $A_{nom}$.

Table I shows some bounds for different $k$, where minimum is taken over the whole iterations, and $C^*$, $D^*$, and $E^*$ denote the values of $C_i$, $D_i$, and $E_i$ after $P_i$ converges. Through the result, one can see that $E_i$ converges to 0 as $k \to \infty$, enlarging the region of quadratic convergence. Also note that the larger $k$, the larger areas of stability and convergence bounds $\Omega_D$ can be achieved.

### TABLE I

<table>
<thead>
<tr>
<th>$\gamma$</th>
<th>$C_i$</th>
<th>$D_i$</th>
<th>$E_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\min C_i$</td>
<td>$C^*$</td>
<td>$\min D_i$</td>
<td>$D^*$</td>
</tr>
<tr>
<td>1</td>
<td>0.94</td>
<td>0.94</td>
<td>0.06</td>
</tr>
<tr>
<td>2</td>
<td>0.51</td>
<td>0.23</td>
<td>0.20</td>
</tr>
<tr>
<td>5</td>
<td>0.67</td>
<td>0.67</td>
<td>2.54</td>
</tr>
<tr>
<td>10</td>
<td>1.34</td>
<td>1.34</td>
<td>21.27</td>
</tr>
<tr>
<td>20</td>
<td>2.68</td>
<td>2.68</td>
<td>104.39</td>
</tr>
<tr>
<td>$\infty$</td>
<td>13.40</td>
<td>13.40</td>
<td>2651.1</td>
</tr>
</tbody>
</table>

### V. CONCLUDING REMARKS

In summary, we have provided the various equivalent formula with respect to the update horizon $\gamma (= kT)$, which revealed the relationships between the time horizon $T$ and the computational complexity due to $k$. The criteria regarding local stability and convergence were also provided for I-GPI. However, I-PI, the special case of I-GPI, actually guarantees the global stability and convergence [13], which implies there would be less conservative bounds than those in Theorem 2. Therefore, the future works would be to find such bounds or global stability conditions of I-GPI, make research on the relations of discount factor and learning rate [16] to I-GPI, and extend the results to the general nonlinear systems.

### REFERENCES