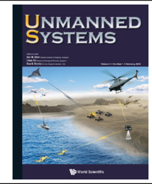
Unmanned Systems, Vol. 13, No. 5 (2025) 1395–1402

DOI: 10.1142/S230138502544008X

UNMANNED SYSTEMS
https://www.worldscientific.com/worldscinet/us



Adaptive Optimal Output Regulation for Unknown Linear Systems via Internal Model Principle and Policy Iteration

Liquan Lin, Jie Huang*

Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong, P. R. China

The data-driven output regulation problem via internal model principle has been studied by both policy-iteration method and value-iteration method. But the results were limited to single-input single-output linear systems with zero input-output transmission matrix. Recently, we have extended the existing results to multi-input multi-output linear systems with nonzero input-output transmission matrix and improved the algorithm by value-iteration method. Since the policy-iteration method is simpler and has a much faster convergence speed than the value-iteration method, in this paper, we further establish the results parallel to the value-iteration work by the policy-iteration method. Compared with the existing policy-iteration results, we are able to handle multi-input multi-output linear systems with nonzero input-output transmission matrix. Moreover, we further improve the existing policy-iteration algorithm by significantly reducing the computational cost and weakening the solvability conditions. A numerical example is used to illustrate the advantages of the improved algorithm.

Keywords: Data-driven control; output regulation; reinforcement learning; policy iteration; internal model principle.

US

1. Introduction

Reinforcement Learning (RL) or Adaptive Dynamic Programming (ADP) provides a way to iteratively solve a Ricatti equation associated with a continuous-time linear control system without knowing some or all matrices defining the system [1]. The approach was first developed to solve the Linear-Quadratic-Regulator (LQR) problem [2–5] and then the Output Regulation Problem (ORP) [6–10]. There are essentially two ADP methods: Policy-Iteration (PI) method [2, 3] and Value-Iteration (VI) method [4]. The PI method needs a stabilizing control gain to start the iteration while the VI method does not need it. On the other hand, the PI method is simpler than the VI method and has a much faster convergence speed.

Since the output regulation problem arises from numerous control problems such as aircraft landing and take-off, vehicle cruise control, motor speed regulation [11–13], it has been well studied since 1970s by two model-based methods: The feedforward design method and internal model approach. Recently, there has been significant interest in developing model-free approaches for solving the ORP. For example, [6] tackled the model-free ORP for linear systems with unknown state equation using the feedforward design approach by the PI method. Paper [7] employed the internal model principle to address the ORP for completely unknown Single-Input Single-Output (SISO) linear systems by the PI-based method. Paper [8] further adopted the VI method to solve the ORP with the feedforward design method. Later, [9] proposed both PI method and VI method to solve the cooperative optimal ORP for unknown linear multi-agent systems via the distributed internal model approach. However, the followers in [9] are assumed to be SISO linear systems and the input-output transmission matrix of each follower is assumed to be zero. Recently, using the VI method, [10] further investigated the model-free ORP for Multi-Input Multi-Output (MIMO) linear systems with nonzero input-output

Received 26 May 2025; Revised 30 July 2025; Accepted 30 July 2025. Published 28 August 2025. This paper was recommended for publication in its revised form by Special Issue Editors, Bin Xin and Hao Fang.
Email Address: *jhuang@mae.cuhk.edu.hk*
*Corresponding author.

transmission matrix, and improved the VI-based method given in [9] to reduce the computational cost and weaken the solvability conditions.

Since the PI method is simpler than the VI method and has a much faster convergence speed, it is interesting to establish the results parallel to those in [10] by PI method. More specifically, in this paper, we will extend the PI-based data-driven algorithm of [9] from SISO linear systems to MIMO linear systems. Besides, we allow the input-output transmission matrix to be nonzero. Like in [9], the PI algorithm comes down to iteratively solving a sequence of linear equations. For a linear system with m inputs, p outputs, n-dimensional state vector, and q-dimensional exosystem, each equation in the sequence typically contains $\frac{(n+pq)(n+pq+1)}{2} + (m+q)(n+pq)$ unknown variables. Thus, the computational cost can be quite high for a high-dimensional system. For this reason, like in [10], by defining a simplified dynamics for the learning phase, we are able to solve our problem by solving a linear equation with $\frac{(n+pq)(n+pq+1)}{2} + (n+pq)m + nq$ unknown variables and a sequence of linear equations with $\frac{(n+pq)(n+pq+1)}{2}$ unknown variables. As a result, this improved PI algorithm significantly reduces the computational cost and weakens the solvability conditions over the first PI algorithm. Moreover, if the input-output transmission matrix is equal to zero like in [6–9], the number of the unknown variables for the first equation is further reduced to $\frac{(n+pq)(n+pq+1)}{2} + n(m+q)$. In this scenario, our improved PI algorithm will relax the solvability condition more profoundly. It is also noted that, using the same example in [10] as a testbed, the PI method in this paper takes seven steps to converge while the VI method of [10] takes 293 steps to converge.

Notation. Throughout this paper, $\mathbb{R}$, $\mathbb{N}$, $\mathbb{N}_+$ and $\mathbb{C}_-$ represent the sets of real numbers, nonnegative integers, positive integers and the open left-half complex plane, respectively. $\|\cdot\|$ represents the Euclidean norm for vectors and the induced norm for matrices. $\otimes$ denotes the Kronecker product. For

$$b = [b_1, b_2, \dots, b_n]^T \in \mathbb{R}^n,$$
$$\mathrm{vecv}(b) = [b_1^2, b_1b_2, \dots, b_1b_n, b_2^2, b_2b_3, \dots, b_{n-1}b_n, b_n^2]^T \in \mathbb{R}^{\frac{n(n+1)}{2}}.$$

For a symmetric matrix

$$P = [p_{ij}]_{n\times n} \in \mathbb{R}^{n\times n},$$
$$\mathrm{vecs}(P) = [p_{11}, 2p_{12}, \dots, 2p_{1n}, p_{22}, 2p_{23}, \dots, 2p_{n-1,n}, p_{nn}]^T \in \mathbb{R}^{\frac{n(n+1)}{2}}.$$

For $v \in \mathbb{R}^n$, $|v|_P = v^TPv$. For column vectors $a_i, i = 1, \dots, s$, $\mathrm{col}(a_1, \dots, a_s) = [a_1^T, \dots, a_s^T]^T$, and, if $A = [a_1, \dots, a_s]$, then $\mathrm{vec}(A) = \mathrm{col}(a_1, \dots, a_s)$. For $A \in \mathbb{R}^{n\times n}$, $\sigma(A)$ denotes the set composed of all the eigenvalues of A. For matrices $A_i, i = 1, \dots, n$, $\mathrm{blockdiag}(A_1, \dots, A_n)$ is the block diagonal matrix $\begin{bmatrix} A_1 & & \\ & \ddots & \\ & & A_n \end{bmatrix}$. I_n denotes the identity matrix of dimension n.

2. Preliminaries

In this section, we summarize the internal model approach for solving the output regulation problem based on [13].

Consider a class of continuous-time linear systems in the following form:

$$\begin{aligned} \dot{x} &= Ax + Bu + Ev, \\ y &= Cx + Du, \\ e &= Cx + Du + Fv, \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$ and $y \in \mathbb{R}^p$ are the state, control input and output of the system, respectively, and $v \in \mathbb{R}^q$ is the exogenous signal generated by the following exosystem:

$$\dot{v} = Sv. \tag{2}$$

Let $y_0 = -Fv$ be the reference output generated by the exosystem. Then, $e \in \mathbb{R}^p$ represents the tracking error of the system.

The dynamic state feedback output regulation problem is defined as follows.

Problem 1. *Given plant* (1) *and exosystem* (2), *design a dynamic state feedback controller* u *of the following form*:

$$\begin{aligned} u &= K_x x + K_z z, \\ \dot{z} &= G_1 z + G_2 e, \end{aligned} \tag{3}$$

with $z \in \mathbb{R}^{n_z}$ *for some positive integer* n_z *such that the closed-loop system is exponentially stable with* v *set to zero and* $\lim_{t\to\infty} e(t) = 0$.

The solvability of Problem 1 needs the following assumptions.

Assumption 1. The pair (A, B) is stabilizable.

Assumption 2. S has no eigenvalues with negative real parts.

Assumption 3. For all $\lambda \in \sigma(S)$, $\mathrm{rank}\left(\begin{bmatrix} A-\lambda I & B \\ C & D \end{bmatrix}\right) = n + p$.

The concept of internal model is given as follows.

Definition 1. A pair of matrices (G_1, G_2) is said to be a minimum $p-$ copy internal model of the matrix S if

$$G_1 = \mathrm{blockdiag}\underbrace{(\beta, \dots, \beta)}_{p-\text{tuple}}, \quad G_2 = \mathrm{blockdiag}\underbrace{(\sigma, \dots, \sigma)}_{p-\text{tuple}},$$

where β is a constant square matrix whose characteristic polynomial equals the minimal polynomial of S, and σ is a constant column vector such that (β, σ) is controllable.

Let (G_1, G_2) be a minimum $p-$copy internal model of S. We call the following system the augmented system.

$$\begin{aligned}\dot{x} &= Ax + Bu + Ev,\\ \dot{z} &= G_1 z + G_2 e,\\ e &= Cx + Du + Fv.\end{aligned} \tag{4}$$

Let $\xi = \mathrm{col}(x, z)$, $Y = \begin{bmatrix} A & 0 \\ G_2 C & G_1 \end{bmatrix}$ and $J = \begin{bmatrix} B \\ G_2 D \end{bmatrix}$. Then the state equation of the augmented system (4) can be put into the following compact form:

$$\dot{\xi} = Y\xi + Ju + \begin{bmatrix} E \\ G_2 F \end{bmatrix} v. \tag{5}$$

Let $K = [K_x, K_z]$. Then, applying the static state feedback control law $u = K\xi$ to the augmented system (5) gives

$$\begin{aligned}\dot{\xi} &= Y\xi + Ju + \begin{bmatrix} E \\ G_2 F \end{bmatrix} v\\ &= (Y + JK)\xi + \begin{bmatrix} E \\ G_2 F \end{bmatrix} v.\end{aligned} \tag{6}$$

Remark 2.1. Generically, the minimal polynomial of S is equal to the characteristic polynomial of S. Thus, typically, $n_z = pq$ and hence the dimension of ξ is typically equal to $(n + pq)$.

By [13, Lemma 1.26] and [13, Remark 1.28], the solvability of Problem 1 is summarized as follows.

Theorem 2.1. *Under Assumptions* 1–3, *let* (G_1, G_2) *be a minimum* $p-$copy *internal model of* S*. Then, the pair* (Y, J) *is stabilizable. Let* K *be such that* $Y + JK$ *is Hurwitz. Then Problem* 1 *is solved by the dynamic state feedback controller* (3)*.*

Since by Theorem 2.1, (Y, J) is stabilizable, the following Algebraic Riccati Equation (ARE)

$$Y^T P^* + P^* Y + Q - P^* J R^{-1} J^T P^* = 0, \tag{7}$$

where $Q = Q^T \geq 0$, $R = R^T > 0$, with $(Y, \sqrt{Q})$ observable, has a unique positive definite solution P^* and an optimal stabilizing feedback gain is given by $K^* = -R^{-1} J^T P^*$ [14].

3. Adaptive Optimal Output Regulation via Internal Model Principle and Policy Iteration

When the matrices A, B, C, D, E, F are unknown, the output regulation problem has been studied by a few papers using various data-driven internal model approaches. For example, paper [7] presented a PI-based method for solving the output regulation problem of linear SISO systems with $D = 0$. Reference [9] presented both PI-based and VI-based methods for the cooperative output regulation problem for N SISO linear systems, which contains the results of [7] as a special case with $N = 1$. Reference [10] further considered the adaptive ORP for MIMO linear systems with nonzero input-output transmission matrix by an improved VI-based method with less computing cost and milder solvability condition compared with the VI method in [9]. Since the VI-based method typically requires many more iteration steps to converge, in this section, we will establish the results paralleled to those in [10] with PI-based method and make a comparison between these two methods. To this end, we need one more assumption as follows.

Assumption 4. The minimal polynomial of S is known.

Remark 3.1. Assumption 4 is made for two purposes. First, under Assumption 4, the minimum p-copy internal model (G_1, G_2) of the matrix S is known. Second, as pointed out in [6], under Assumption 4, there exist known vector $\hat{v}(t)$, known matrix $\hat{S}$ whose characteristic polynomial is equal to the minimal polynomial of S, and unknown matrix G depending on $v(0)$ such that

$$\dot{\hat{v}} = \hat{S}\hat{v}, \quad v = G\hat{v}.$$

As a result, (1) can be recast as follows:

$$\begin{aligned}\dot{x} &= Ax + Bu + \hat{E}\hat{v},\\ e &= Cx + Du + \hat{F}\hat{v},\end{aligned} \tag{8}$$

where $\hat{E} = EG$, $\hat{F} = FG$ are unknown matrices. In what follows, to save notation, we will still use (1) while assuming v is known.

3.1. *PI-based data-driven algorithm*

As (7) is nonlinear, an iterative approach for obtaining P^* is given by solving the following linear Lyapunov equations [15]:

$$0 = (Y + JK_k)^T P_k + P_k (Y + JK_k) + Q + K_k^T R K_k, \tag{9a}$$

$$K_{k+1} = -R^{-1} J^T P_k, \tag{9b}$$

where $k = 0, 1, \ldots$, and K_0 is such that $Y + JK_0$ is Hurwitz. Equations (9a) and (9b) guarantee the following property for $k \in \mathbb{N}$:

(1) $\sigma(Y + JK_k) \in \mathbb{C}_-$;
(2) $P^* \leq P_{k+1} \leq P_k$;
(3) $\lim_{k\to\infty} K_k = -R^{-1} J^T P^*$, $\lim_{k\to\infty} P_k = P^*$.

Then, from (6), we obtain

$$\begin{aligned}\dot{\xi} &= Y\xi + Ju + \begin{bmatrix} E \\ G_2 F \end{bmatrix} v\\ &= Y_k \xi + J(-K_k \xi + u) + \begin{bmatrix} E \\ G_2 F \end{bmatrix} v,\end{aligned} \tag{10}$$

where $Y_k = Y + JK_k$.

From (9) and (10), we further obtain

$$\begin{aligned}
&|\xi(t+\delta t)|_{P_k} - |\xi(t)|_{P_k} \\
&\quad = \int_t^{t+\delta t} \Bigg[|\xi|_{Y_k^T P_k + P_k Y_k} + 2(-K_k\xi + u)^T J^T P_k \xi \\
&\qquad + 2v^T \begin{bmatrix} E \\ G_2 F \end{bmatrix}^T P_k \xi \Bigg] d\tau \\
&\quad = \int_t^{t+\delta t} \Bigg[-|\xi|_{Q+K_k^T R K_k} + 2(K_k\xi - u)^T R K_{k+1} \xi \\
&\qquad + 2v^T \begin{bmatrix} E \\ G_2 F \end{bmatrix}^T P_k \xi \Bigg] d\tau. \qquad (11)
\end{aligned}$$

For convenience, for any vectors $a \in \mathbb{R}^n, b \in \mathbb{R}^m$ and any integer $s \in \mathbb{N}_+$, define

$$\begin{aligned}
\delta_a &= [\mathrm{vecv}(a(t_1)) - \mathrm{vecv}(a(t_0)), \ldots, \\
&\qquad \mathrm{vecv}(a(t_s)) - \mathrm{vecv}(a(t_{s-1}))]^T, \\
\Gamma_{ab} &= \left[\int_{t_0}^{t_1} a \otimes b\, d\tau, \int_{t_1}^{t_2} a \otimes b\, d\tau, \ldots, \int_{t_{s-1}}^{t_s} a \otimes b\, d\tau \right]^T. \qquad (12)
\end{aligned}$$

Then, (11) and (12) lead to

$$\Psi_k \begin{bmatrix} \mathrm{vecs}(P_k) \\ \mathrm{vec}(K_{k+1}) \\ \mathrm{vec}\left(\begin{bmatrix} E \\ G_2 F \end{bmatrix}^T P_k \right) \end{bmatrix} = \Phi_k, \qquad (13)$$

where

$$\begin{aligned}
\Psi_k &= [\Psi_k^1, \Psi_k^2, \Psi_k^3], \\
\Psi_k^1 &= \delta_\xi, \\
\Psi_k^2 &= -2\Gamma_{\xi\xi}(I_{n+n_z} \otimes K_k^T R) + 2\Gamma_{\xi u}(I_{n+n_z} \otimes R), \\
\Psi_k^3 &= -2\Gamma_{\xi v}, \\
\Phi_k &= -\Gamma_{\xi\xi}\mathrm{vec}(Q + K_k^T R K_k).
\end{aligned}$$

The solvability of (13) is summarized by the following lemma.

Lemma 3.1. *The matrix Ψ_k has full column rank for any $k \in \mathbb{N}$ if*

$$\mathrm{rank}([\Gamma_{\xi\xi}, \Gamma_{\xi u}, \Gamma_{\xi v}]) = \frac{(n+n_z)(n+n_z+1)}{2} + (n+n_z)(m+q). \qquad (14)$$

Proof. The proof of this lemma is similar to that of [3, Lemma 6] and is thus omitted. □

Remark 3.2. The difficulty of extending SISO systems to MIMO systems is overcome by our adoption of minimum p-copy internal model given in Definition 1. In contrast, [7, 9] used one-copy internal model which cannot handle MIMO systems. On the other hand, by defining the augmented system (5), we have made D part of the matrix J so that nonzero D can also be handled.

3.2. *An improved PI-based data-driven algorithm*

In this section, we will improve the PI-based data-driven algorithm presented in the last section.

First, like in [10], let us observe from (7) that the matrix P^* and hence the optimal stabilizing feedback gain K^* only depend on the matrices Y and J and they have nothing to do with the matrix $\left[\begin{smallmatrix} E \\ G_2F \end{smallmatrix}\right]$.

Thus, during the learning phase, we can replace the dynamic compensator $\dot{z} = G_1 z + G_2 e$ by the following dynamic compensator (15):

$$\dot{\hat{z}} = G_1 \hat{z} + G_2 y. \qquad (15)$$

Let $\hat{\xi} = \mathrm{col}(x, \hat{z})$. Then, we have

$$\begin{aligned}
\dot{\hat{\xi}} &= \begin{bmatrix} A & 0 \\ G_2 C & G_1 \end{bmatrix} \hat{\xi} + \begin{bmatrix} B \\ G_2 D \end{bmatrix} u + \begin{bmatrix} E \\ 0 \end{bmatrix} v \\
&= Y\hat{\xi} + Ju + \begin{bmatrix} E \\ 0 \end{bmatrix} v \\
&= Y_k \hat{\xi} + J(-K_k\hat{\xi} + u) + \begin{bmatrix} E \\ 0 \end{bmatrix} v. \qquad (16)
\end{aligned}$$

Thus, from (9) and (16), we have

$$\begin{aligned}
&|\hat{\xi}(t+\delta t)|_{P_k} - |\hat{\xi}(t)|_{P_k} \\
&\quad = \int_t^{t+\delta t} \Bigg[-|\hat{\xi}|_{Q+K_k^T R K_k} + 2(-K_k\hat{\xi} + u)^T J^T P_k \hat{\xi} \\
&\qquad + 2v^T \begin{bmatrix} E \\ 0 \end{bmatrix}^T P_k \hat{\xi} \Bigg] d\tau. \qquad (17)
\end{aligned}$$

Combining (12) and (17) gives

$$\hat{\Psi}_k \begin{bmatrix} \mathrm{vecs}(P_k) \\ \mathrm{vec}(J^T P_k) \\ \mathrm{vec}\left(\begin{bmatrix} E \\ 0 \end{bmatrix}^T P_k \right) \end{bmatrix} = \hat{\Phi}_k, \qquad (18)$$

where

$$\hat{\Psi}_k = [\hat{\Psi}_k^1, \hat{\Psi}_k^2, \hat{\Psi}_k^3],$$
$$\hat{\Psi}_k^1 = \delta_{\hat{\xi}},$$
$$\hat{\Psi}_k^2 = 2\Gamma_{\hat{\xi}\hat{\xi}}(I_{n+n_z} \otimes K_k^T) - 2\Gamma_{\hat{\xi}u},$$
$$\hat{\Psi}_k^3 = -2\Gamma_{\hat{\xi}v},$$
$$\hat{\Phi}_k = -\Gamma_{\hat{\xi}\hat{\xi}}\mathrm{vec}(Q + K_k^T R K_k).$$

It is noted that there are three unknown matrices P_k, J and E in (18). Among these three matrices, only P_k needs to be updated with respect to k. Thus, we can first identify J and E, and then simplify (18) using identified J and E.

For this purpose, letting $P_k = I_{n+n_z}$ in (17) gives

$$\begin{aligned} &|\hat{\xi}(t+\delta t)|_{I_{n+n_z}} - |\hat{\xi}(t)|_{I_{n+n_z}} \\ &= \int_t^{t+\delta t} \left[|\hat{\xi}|_{Y^T+Y} + 2u^T J^T \hat{\xi} + 2v^T \begin{bmatrix} E \\ 0 \end{bmatrix}^T \hat{\xi} \right] d\tau \\ &= \int_t^{t+\delta t} [|\hat{\xi}|_{Y^T+Y} + 2u^T J^T \hat{\xi} + 2v^T E^T x] d\tau. \qquad (19) \end{aligned}$$

Since for any symmetric matrix $H \in \mathbb{R}^{n\times n}$, there exists a constant matrix $M_n \in \mathbb{R}^{n^2 \times \frac{n(n+1)}{2}}$ with full column rank such that $M_n \mathrm{vecs}(H) = \mathrm{vec}(H)$, (19) leads to

$$\hat{\Psi}' \begin{bmatrix} \mathrm{vecs}(Y^T + Y) \\ \mathrm{vec}(J^T) \\ \mathrm{vec}(E^T) \end{bmatrix} = \hat{\Phi}', \qquad (20)$$

where

$$\hat{\Psi}' = [\Gamma_{\hat{\xi}\hat{\xi}} M_{n+n_z}, 2\Gamma_{\hat{\xi}u}, 2\Gamma_{xv}],$$
$$\hat{\Phi}' = \delta_{\hat{\xi}} \mathrm{vecs}(I_{n+n_z}).$$

The solvability of (20) is guaranteed by the following lemma.

Lemma 3.2. *The matrix* $\hat{\Psi}'$ *has full column rank if*

$$\mathrm{rank}([\Gamma_{\hat{\xi}\hat{\xi}}, \Gamma_{\hat{\xi}u}, \Gamma_{xv}]) = \frac{(n+n_z)(n+n_z+1)}{2} + (n+n_z)m + nq. \qquad (21)$$

Proof. Given any column vector $a \in \mathbb{R}^{n+n_z}$ and any symmetric matrix $H \in \mathbb{R}^{(n+n_z)\times(n+n_z)}$, it can be verified that $a^T H a = (a \otimes a)^T \mathrm{vec}(H) = \mathrm{vecv}(a)^T \mathrm{vecs}(H)$. Further, recalling that $\mathrm{vec}(H) = M_{n+n_z} \mathrm{vecs}(H)$ for some matrix M_{n+n_z} gives

$$(a \otimes a)^T M_{n+n_z} \mathrm{vecs}(H) = \mathrm{vecv}(a)^T \mathrm{vecs}(H).$$

Since H can be any symmetric matrix, we obtain $(a \otimes a)^T M_{n+n_z} = \mathrm{vecv}(a)^T$ for any column vector $a \in \mathbb{R}^{n+n_z}$. Thus, $\Gamma_{\hat{\xi}\hat{\xi}} M_{n+n_z} = \hat{\Gamma}_{\hat{\xi}}$, where

$$\hat{\Gamma}_{\hat{\xi}} = \left[\int_{t_0}^{t_1} \mathrm{vecv}(\hat{\xi}) d\tau, \int_{t_1}^{t_2} \mathrm{vecv}(\hat{\xi}) d\tau, \ldots, \int_{t_{s-1}}^{t_s} \mathrm{vecv}(\hat{\xi}) d\tau \right]^T.$$

Since M_{n+n_z} is of full column rank, $\mathrm{rank}([\Gamma_{\hat{\xi}\hat{\xi}}, \Gamma_{\hat{\xi}u}, \Gamma_{xv}]) = \mathrm{rank}([\hat{\Gamma}_{\hat{\xi}}, \Gamma_{\hat{\xi}u}, \Gamma_{xv}])$. The proof is complete. □

Remark 3.3. The rank condition (21) is milder than the original one (14) since the column dimension of the matrix to be tested decreases by $n_z q$. This improvement can be significant when $n_z q$ is large.

Remark 3.4. If $D = 0$ as in [6–9], the rank condition (21) can be further relaxed to

$$\mathrm{rank}([\Gamma_{\hat{\xi}\hat{\xi}}, \Gamma_{xu}, \Gamma_{xv}]) = \frac{(n+n_z)(n+n_z+1)}{2} + nm + nq. \qquad (22)$$

Since in this case, $J = \begin{bmatrix} B \\ 0 \end{bmatrix}$.

In contrast, even if $D = 0$, the rank condition in [9] is as follows:

$$\mathrm{rank}([\Gamma_{\xi\xi}, \Gamma_{\xi u}, \Gamma_{\xi v}]) = \frac{(n+n_z)(n+n_z+1)}{2} + (n+n_z)(m+q)$$

Thus, compared with [9], if $D = 0$, our method will relax the rank condition more profoundly.

Once we have obtained J and E by solving (20), the only unknown matrix in (18) is P_k. With the definition of M_{n+n_z}, we have

$$\mathrm{vec}(J^T P_k) = (I_{n+n_z} \otimes J^T) M_{n+n_z} \mathrm{vecs}(P_k),$$
$$\mathrm{vec}\left(\begin{bmatrix} E \\ 0 \end{bmatrix}^T P_k \right) = \left(I_{n+n_z} \otimes \begin{bmatrix} E \\ 0 \end{bmatrix}^T \right) M_{n+n_z} \mathrm{vecs}(P_k). \qquad (23)$$

From (18) and (23), we obtain

$$\hat{\Psi}_k' \mathrm{vecs}(P_k) = \hat{\Phi}_k', \qquad (24)$$

where

$$\begin{aligned} \hat{\Psi}_k' &= \delta_{\hat{\xi}} + 2\Gamma_{\hat{\xi}\hat{\xi}}(I_{n+n_z} \otimes K_k^T J^T) M_{n+n_z} \\ &\quad - 2\Gamma_{\hat{\xi}u}(I_{n+n_z} \otimes J^T) M_{n+n_z} \\ &\quad - 2\Gamma_{\hat{\xi}v}\left(I_{n+n_z} \otimes \begin{bmatrix} E \\ 0 \end{bmatrix}^T \right) M_{n+n_z}, \\ \hat{\Phi}_k' &= -\Gamma_{\hat{\xi}\hat{\xi}} \mathrm{vec}(Q + K_k^T R K_k). \end{aligned}$$

The solvability of (24) is guaranteed by the following lemma.

Lemma 3.3. *The matrix* $\hat{\Psi}_k'$ *has full column rank for any* $k \in \mathbb{N}$ *if* $\operatorname{rank}(\Gamma_{\hat{\xi}\hat{\xi}}) = \frac{(n+n_z)(n+n_z+1)}{2}$.

Proof. The proof of this lemma is similar to that of [3, Lemma 6] and is thus omitted. □

Remark 3.5. Since $\hat{\Psi}_k'$ has full column rank once (21) is satisfied, (21) is the only requirement to ensure the solvability of (20) and (24).

Remark 3.6. In our improved PI algorithm, for all $k \in \mathbb{N}$, we have reduced the problem of solving linear equation (13) to that of solving (24). Thus, the number of unknown variables decreases by $(n + n_z)(m + q)$ and the computational cost decreases significantly.

Our improved PI-based data-driven algorithm is summarized as Algorithm 1.

Remark 3.7. The results here are parallel to those in [10] obtained by VI algorithm. In comparison with the VI method in [10], the advantages of PI method are that the algorithm is simpler and the convergence rate is much faster. Nevertheless, these advantages are achieved at the cost of requiring a known initially stabilizing feedback gain. In contrast, the VI algorithm in [10] does not require an initially stabilizing feedback gain to start the iteration.

For comparison of the two algorithms in this section, let us consider a case with $n = 10$, $m = 8$, $p = 5$, $q = 20$, $n_z = 50$. Tables 1 and 2 show the numbers of unknown variables and rank conditions of the two algorithms, respectively. The contrast is stark.

Algorithm 1. Improved PI-based Algorithm for ORP.

1: Find a feedback gain K_0 such that $Y + JK_0$ is a Hurwitz matrix. Apply an initial input $u^0 = K_0\hat{\xi} + \delta$ with exploration noise δ. Collecting data starting from arbitrary t_0 until the rank condition (21) is satisfied.
2: Solve J and E from (20). $k \leftarrow 0$.
3: **repeat**
4: Solve P_k from (24). $K_{k+1} = -R^{-1}J^TP_k$. $k \leftarrow k + 1$.
5: **until** $\|P_k - P_{k-1}\| < \varepsilon$ with a sufficiently small constant $\varepsilon > 0$
6: $K^* = [K_x^*, K_z^*] = -R^{-1}J^TP_k$.
7: Obtain the following optimal controller

$$u^* = K_x^*x + K_z^*z. \tag{25}$$

Table 1. Comparison of computational complexity.

Algorithm	Equation label	Unknown variables
First algorithm	(13)	3510
Improved algorithm	(24)	1830

Table 2. Comparison of rank condition.

Algorithm	Rank condition for data collecting
First algorithm	$\operatorname{rank}([\Gamma_{\xi\xi}, \Gamma_{\xi u}, \Gamma_{\xi v}]) = 3510$
Improved algorithm	$\operatorname{rank}([\Gamma_{\hat{\xi}\hat{\xi}}, \Gamma_{\hat{\xi}u}, \Gamma_{xv}]) = 2510$

4. An Application

In this section, we use an example to illustrate the effectiveness of our proposed improved PI-based data-driven algorithm.

Consider a linear system in the form of (1) with the matrices A, B, C, D, E, F as follows.

$$A = \begin{bmatrix} -1 & 1 \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad E = \begin{bmatrix} 0 & 0 \\ 0 & 1.5 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad F = \begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix}. \tag{26}$$

The matrix S for the leader (2) is as follows:

$$S = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \tag{27}$$

It can be verified that Assumptions 1–3 are all satisfied. The minimal $p-$copy internal model of S used in this example is given as

$$G_1 = I_2 \otimes S, \quad G_2 = I_2 \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{28}$$

Solving ARE (7) with both Q and R being identity matrices gives $K^* = -R^{-1}J^TP^*$ as follows:

$$K^* = \begin{bmatrix} -0.2528 & -3.2586 & 0.1276 & -0.2507 & 0.3905 & -1.3298 \\ -1.1406 & -0.2528 & 0.4131 & -1.3230 & -0.0330 & 0.2793 \end{bmatrix}.$$

To implement Algorithm 1, we perform the learning process with the initial input $u^0 = K_0\hat{\xi} + \delta$, where

$$K_0 = \begin{bmatrix} -0.9721 & -7.0206 & -4.9093 & -3.8948 & -2.0526 & -11.0652 \\ -4.9794 & 0.0289 & -1.9473 & -10.9351 & 5.0930 & 4.1086 \end{bmatrix},$$

is a stabilizing initial gain and the exploration noise δ is as follows:

$$\delta = \begin{bmatrix} \sin(-3t) + \sin(-t) + \sin(2t) + \sin(5t) \\ \sin(-7t) + \sin(-3t) + \sin(2t) + \sin(4t) \end{bmatrix}.$$

Let $t_j = 0.2j\ sec, j = 0, 1, \ldots, s$ with $s = 40$. Then, with the initial condition $\hat{\xi}(0) = \mathbf{0}, v(0) = [3, 4]^T$, the trajectory data under the initial input u^0 between $[0, 8]$ s are collected to fit the rank condition (21).

Let $\varepsilon = 0.01$. Then it takes seven steps for the convergence criterion $|P_k - P_{k-1}| < \varepsilon$ to be satisfied. The comparison between P_k, K_k and their target values P^*, K^* for the improved PI-based data-driven Algorithm 1 is given in Fig. 1, and we obtain the learned optimal control gain K_7 after seven steps of iteration as follows:

$$K_7 = \begin{bmatrix} -0.2528 & -3.2586 & 0.1276 & -0.2507 & 0.3905 & -1.3298 \\ -1.1405 & -0.2528 & 0.4131 & -1.3229 & -0.0330 & 0.2792 \end{bmatrix}.$$

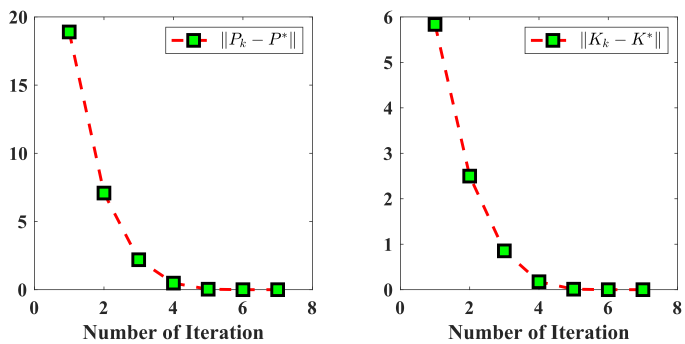


Fig. 1. Comparison of P_k, K_k and their target values P^*, K^* for PI-based method.

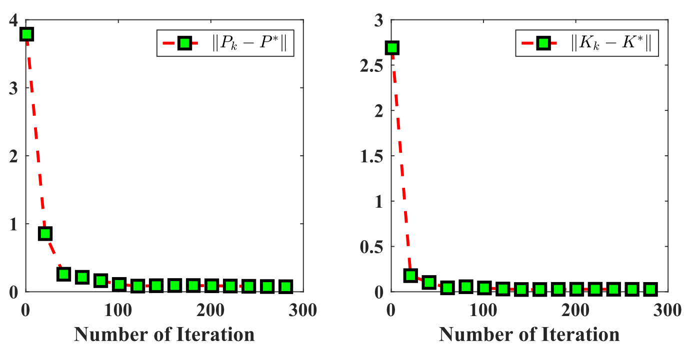


Fig. 2. Comparison of P_k, K_k and their target values P^*, K^* for VI-based method in [10].

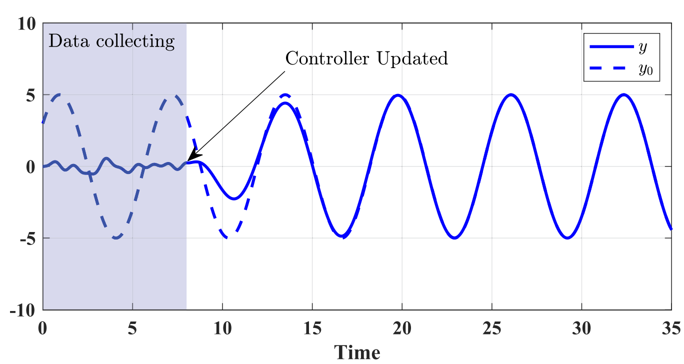


Fig. 3. Plots of the first component of y and y_0 for PI-based method.

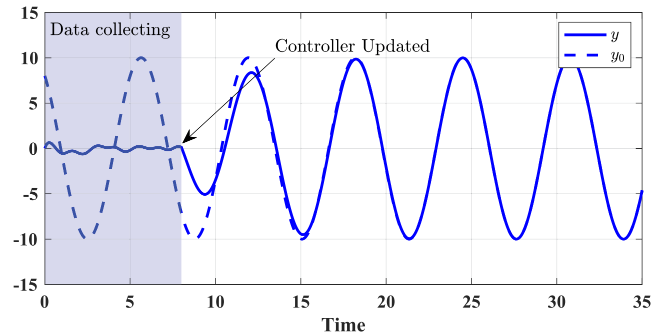


Fig. 4. Plots of the second component of y and y_0 for PI-based method.

Remark 4.1. The same example was solved in [10] by the improved VI algorithm, which takes 293 steps to converge as shown in Fig. 2, while the improved PI algorithm here only takes seven steps to converge.

The control policy is updated to the learned optimal controller (25) at 8 s and the evolution of the first and second components of y and y_0 is shown in Figs. 3 and 4, respectively. As expected, the improved PI-based algorithm works satisfactorily.

In this example, $n = 2, n_z = 4, m = 2, p = 2, q = 2$. Thus, by using (24) instead of (13), we reduce the unknown variables from 45 to 21.

5. Conclusions

In this paper, we have investigated the model-free ORP via the internal model principle and policy iteration method. We have first extended the PI-based data-driven algorithm of [9] from SISO linear systems to MIMO linear systems. Besides, we allow the input-output transmission matrix to be nonzero. Then, like in [10], by separating the dynamics between the learning phase and the control phase, we have further proposed an improved algorithm that has significantly reduced the computational cost of the first algorithm and hence weakened the solvability conditions of the first algorithm. The extension of our PI approach to the cooperative output regulation problem of unknown linear multi-agent systems is underway.

Acknowledgments

This paper was supported in part by the Research Grants Council of the Hong Kong Special Administration Region under Grant No. 14201621.

ORCID

Liquan Lin https://orcid.org/0000-0002-6013-4380
Jie Huang https://orcid.org/0000-0002-6753-8679

References

[1] F. L. Lewis, D. Vrabie and V. L. Syrmos, *Optimal Control* (John Wiley & Sons, 2012).

[2] D. Vrabie, O. Pastravanu, M. Abu-Khalaf and F. L. Lewis, Adaptive optimal control for continuous-time linear systems based on policy iteration, *Automatica* **45**(2) (2009) 477–484.

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

[4] T. Bian and Z. P. Jiang, Value iteration and adaptive dynamic programming for data-driven adaptive optimal control design, *Automatica* **71** (2016) 348360.

[5] S. A. A. Rizvi and Z. Lin, Reinforcement learning-based linear quadratic regulation of continuous-time systems using dynamic output feedback, *IEEE T. Cybern.* **50**(11) (2019) 4670–4679.

[6] W. Gao and Z. P. Jiang, Adaptive dynamic programming and adaptive optimal output regulation of linear systems, *IEEE Trans. Autom. Control* **61**(12) (2016) 4164–4169.

[7] Y. Liu and W. Gao, Adaptive optimal output regulation of continuous-time linear systems via internal model principle, in *Proc. 9th IEEE UEMCON* (IEEE, 2018), pp. 38–43.

[8] Y. Jiang, W. Gao, J. Na, D. Zhang, T. T. Hmlinen, V. Stojanovic and F. L. Lewis, Value iteration and adaptive optimal output regulation with assured convergence rate, *Control Eng. Practice* **121** (2022) 105042.

[9] W. Gao, M. Mynuddin, D. C. Wunsch and Z. P. Jiang, Reinforcement learning-based cooperative optimal output regulation via distributed adaptive internal model, *IEEE Trans. Neural Netw. Learn. Syst.* **33**(10) (2021) 5229–5240.

[10] L. Lin and J. Huang, Data-driven optimal output regulation for continuous-time linear systems via internal Model Principle, *IEEE Trans. Autom. Control* **70**(6) (2025) 4202–4208, doi:10.1109/TAC.2025.3535281.

[11] E. J. Davison, The robust control of a servomechanism problem for linear time-invariant multivariable systems, *IEEE Trans. Autom. Control* **21**(1) (1976) 25–34.

[12] B. A. Francis and W. M. Wonham, The internal model priciple of control theory, *Automatica* **12**(5) (1976) 457–465.

[13] J. Huang, *Nonlinear Output Regulation*: *Theory and Applications* (SIAM, Philadelphia, PA, 2004).

[14] V. Kucera, A contribution to matrix quadratic equations, *IEEE Trans. Autom. Control* **17**(3) (1972) 344–347.

[15] D. Kleinman, On an iterative technique for riccati equation computations, *IEEE Trans. Autom. Control* **13**(1) (1968) 114115.

Liquan Lin received his B. Eng. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2021. He is currently pursuing Ph.D. degree in the Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong SAR, China.

His current research interests include multi-agent systems, output regulation, data-driven control and reinforcement learning.

Jie Huang received his Ph.D. degree from the Johns Hopkins University, Baltimore, MD, USA in 1990. He is now Choh-Ming Li Research Professor of Mechanical and Automation Engineering, The Chinese University of Hong Kong (CUHK), and Associate Dean (Research) of Faculty of Engineering, CUHK. His research interests include nonlinear control theory and applications, multi-agent systems, game theory and flight guidance and control.

Dr. Huang is a life fellow of IEEE, Fellow of IFAC, CAA, HKIE and AAIA.