Adaptive Observer-Based Optimal Control via Dual Integral Reinforcement Learning

Jie Ren
Department of Electrical and Computer Engineering
Northeastern University
Boston, MA, USA

Bahram Shafai
Department of Electrical and Computer Engineering
Northeastern University
Boston, MA, USA

***Abstract*—Integral reinforcement learning (IRL) solves the linear quadratic regulator (LQR) problem without requiring the system dynamics matrix, but needs full state measurements. In the absence of states, a dedicated PI observer structure can be used in the learning process, which avoids the limitation of P observer. This paper presents a two-step framework for output-feedback control. The first step applies dual IRL to learn proportional-integral observer gains that achieve loop transfer recovery, preserving the robustness margins of LQR under output feedback. The second step applies primal IRL with the learned observer to obtain the optimal controller from output measurements. The dual IRL algorithm is derived from Newton–Kleinman iteration on the observer algebraic Riccati equation, with convergence guaranteed under standard stabilizability and detectability conditions. Simulation results demonstrate near-optimal performance for both minimum and non-minimum phase systems.**

***Index Terms*—reinforcement learning, observer design, loop transfer recovery, output feedback, optimal control**

I. Introduction

Integral Reinforcement Learning (IRL) solves the linear quadratic regulator (LQR) problem without requiring the system dynamics matrix A [1]. The algorithm learns the optimal gain from measured state trajectories, where A is embedded. However, IRL requires full state measurements. Systems that provide only output measurements cannot use IRL directly.

The standard solution for output-only systems is to estimate states with an observer. The LQG/LTR methodology designs observers to recover the robustness margins of full-state feedback [2]. However, asymptotic loop transfer recovery requires observer gains approaching infinity for minimum-phase systems, and is impossible for non-minimum-phase systems. The proportional-integral (PI) observer achieves time recovery for both minimum-phase and non-minimum-phase systems with finite gains [3]. Classical PI observer design requires the system matrix A. This reintroduces the model dependency that IRL avoids.

Existing output-feedback extensions of IRL handle inaccessible states without learning observer gains. Lewis and Vamvoudakis learn the optimal controller directly from input-output history, bypassing state reconstruction [4]. Modares, Lewis, and Jiang reconstruct states from output history [5]. Rizvi and Lin reconstruct states using embedded filters with user-defined poles [6]. None learn observer gains from data.

Several frameworks exist for data-driven observer design from state measurements. Mishra, van Waarde, and Bajcinca formulate LMI conditions for observer gains directly from state data [7]. Turan and Ferrari-Trecate design unknown-input observers using Willems' Fundamental Lemma with historical state trajectories [8]. Adachi and Wakasa derive dual system representations from state data and apply LMI-based design [9]. These methods design observers for state estimation. None address loop transfer recovery.

The classical Luenburger observer has limitation to recover robustness of LQR. It has been recognized that the PI observer is the remedy to achieve robustness in the framework of this paper. Consequently, we apply dual IRL to learn PI observer gains from state measurements. The design uses the PI observer LTR formulation of Niemann et al. [3], providing time recovery for both minimum-phase and non-minimum-phase systems, and asymptotic loop transfer recovery for minimum-phase systems. The learned observer provides state estimates for primal IRL that learns the controller from output measurements.

Section II reviews IRL and PI observer LTR design. Section III presents the two-step output-feedback learning framework. Section IV provides simulation results, followed by concluding remarks in V.

II. Background

A. LQR Problem

Consider the linear time-invariant system

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{1}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, and (A, B) is stabilizable [1]. The optimal control problem minimizes the infinite horizon quadratic cost

$$V(x(t_0), t_0) = \int_{t_0}^{\infty} \left(x^T(\tau)Qx(\tau) + u^T(\tau)Ru(\tau)\right) d\tau \tag{2}$$

with $Q \geq 0$, $R > 0$, and $(Q^{1/2}, A)$ detectable [1]. The optimal controller is $u(t) = -Kx(t)$ with

$$K = R^{-1}B^T P \tag{3}$$

where P is the unique positive definite solution of the algebraic Riccati equation (ARE)

$$A^T P + PA - PBR^{-1}B^T P + Q = 0. \tag{4}$$

Solving the ARE directly requires both the system matrix A and full state measurements $x(t)$.

B. Integral Reinforcement Learning

Integral reinforcement learning (IRL) solves for the optimal gain K without knowing A [1]. The algorithm uses policy iteration, which alternates between policy evaluation and policy improvement.

For any stabilizing gain K, the cost (2) under policy $u = -Kx$ takes the quadratic form

$$V(x(t)) = x^T(t)Px(t) \tag{5}$$

where P is the positive definite solution of the Lyapunov equation

$$(A - BK)^T P + P(A - BK) = -(K^T RK + Q). \tag{6}$$

When $K = R^{-1}B^T P$, the Lyapunov equation (III-B) reduces to the ARE (4). The classical approach to policy evaluation solves (III-B) for P given K, which requires A.

Vrabie et al. [1] observed that integrating (5) over a finite interval $[t, t+T]$ yields

$$x_t^T P_i x_t = \int_t^{t+T} x_\tau^T (Q + K_i^T R K_i) x_\tau \, d\tau + x_{t+T}^T P_i x_{t+T}. \tag{7}$$

This equation yields the same P_i as solving (III-B) with $K = K_i$, but can be solved without knowing A because A is embedded in the measured state trajectories x_τ [1].

To solve (7) from data, the quadratic form $x^T P x$ is parameterized as

$$x^T(t) P_i x(t) = \bar{p}_i^T \bar{x}(t) \tag{8}$$

where $\bar{x}(t)$ is the Kronecker product quadratic polynomial basis vector with elements $\{x_j(t)x_k(t)\}_{j=1,\ldots,n;\, k=j,\ldots,n}$, and $\bar{p}_i = \nu(P_i)$ contains the diagonal elements P_{jj} and twice the upper off-diagonal elements $2P_{jk}$ for $k > j$ [1]. Using (8), the policy evaluation (7) becomes

$$\bar{p}_i^T \left(\bar{x}(t) - \bar{x}(t+T) \right) = \int_t^{t+T} x^T(\tau)(Q + K_i^T R K_i) x(\tau) \, d\tau. \tag{9}$$

The right-hand side is the observed cost over the interval $[t, t+T]$, denoted

$$d(\bar{x}(t), K_i) \equiv \int_t^{t+T} x^T(\tau)(Q + K_i^T R K_i) x(\tau) \, d\tau. \tag{10}$$

Collecting $N \geq n(n+1)/2$ measurements along a state trajectory under sufficient excitation, the least squares solution is

$$\bar{p}_i = (XX^T)^{-1} XY \tag{11}$$

where $X = [\bar{x}_\Delta^1 \;\; \bar{x}_\Delta^2 \;\; \cdots \;\; \bar{x}_\Delta^N]$ with $\bar{x}_\Delta^j = \bar{x}^j(t) - \bar{x}^j(t+T)$, and $Y = [d(\bar{x}^1, K_i) \;\; \cdots \;\; d(\bar{x}^N, K_i)]^T$ [1].

The policy improvement step updates the gain using

$$K_{i+1} = R^{-1} B^T P_i. \tag{12}$$

Lemma 1 (IRL Convergence [1])**.** *Under persistence of excitation and given an initial stabilizing gain K_1, the policy iteration* (7)–(12) *converges to the optimal solution: $P_i \to P^*$ and $K_i \to K^* = R^{-1}B^T P^*$.*

IRL removes the need for the system matrix A but still requires full state measurements $x(t)$.

C. PI Observer with Loop Transfer Recovery

The full-state feedback $u = -Kx$ achieves optimal performance and provides robustness properties characterized by the target loop transfer function [3]

$$L_{TFL}(s) = -K(sI - A)^{-1} B \tag{13}$$

When states are estimated by an observer, the output-feedback controller $u = -K\hat{x}$ generally fails to recover these properties. Loop transfer recovery (LTR) designs the observer to minimize this gap, measured by the recovery matrix $M(s)$ which relates the sensitivity error to the target via

$$E_S(s) = S_{TFL}(s) M(s) \tag{14}$$

Consider the output

$$y(t) = Cx(t) \tag{15}$$

where $y(t) \in \mathbb{R}^p$ and $C \in \mathbb{R}^{p \times n}$. For a proportional (P) observer

$$\dot{\hat{x}} = A\hat{x} + L(C\hat{x} - y) + Bu \tag{16}$$

the recovery matrix is

$$M_P(s) = -K(sI - A - LC)^{-1} B \tag{17}$$

For minimum phase systems, asymptotic LTR (ALTRI) achieves $M_P(s) \to 0$ as $q \to \infty$, but requires observer gains $\|L(q)\| \to \infty$ [3]. For non-minimum phase systems, ALTRI is impossible—a finite recovery error remains regardless of gain.

The limitation of P observer in robust asymptotic recovery can be resolved using the PI observer. For a general exposition of PI observer and its advantages, see [3], [10], [11].

The proportional-integral (PI) observer augments the P observer with an integral term and is defined by

$$\dot{\hat{x}} = A\hat{x} + L_P(C\hat{x} - y) + Bu + Bv \tag{18}$$

$$\dot{v} = L_I(C\hat{x} - y) \tag{19}$$

where $L_P \in \mathbb{R}^{n \times p}$ is the proportional gain, $L_I \in \mathbb{R}^{m \times p}$ is the integral gain, and $v \in \mathbb{R}^m$ is the integral state that feeds back through the input matrix B. When $L_I = 0$, this reduces to a P observer.

The PI observer admits a compact form as a standard observer on an extended state space. Defining $z = [\hat{x}^T \; v^T]^T \in \mathbb{R}^{n+m}$, the dynamics (18)–(19) become

$$\dot{z} = A_x z + L_x(C_x z - y) + B_x u \tag{20}$$

with extended matrices

$$A_x = \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}, \quad B_x = \begin{bmatrix} B \\ 0 \end{bmatrix}, \tag{21}$$

$$C_x = \begin{bmatrix} C & 0 \end{bmatrix}, \quad L_x = \begin{bmatrix} L_P \\ L_I \end{bmatrix} \tag{22}$$

The PI observer achieves time recovery: $M_I(0) = 0$, meaning exact recovery in steady state as $t \to \infty$ [3]. Unlike ALTRI, time recovery does not require high observer gains. Moreover, time recovery holds even for non-minimum phase systems.

The extended formulation enables LQG design, giving the observer gain

$$L_x = -P_o C_x^T \Sigma^{-1} \tag{23}$$

where P_o is the positive definite solution to the observer algebraic Riccati equation

$$A_x P_o + P_o A_x^T + \Gamma - P_o C_x^T \Sigma^{-1} C_x P_o = 0 \tag{24}$$

with design weights $\Gamma \geq 0$ and $\Sigma > 0$.

Solving (24) requires knowledge of the system matrix A.

III. Two-Step Output-Feedback Learning

A. Problem Setup

Consider the system (1) with output (15). The PI observer (18)–(19) estimates states from output measurements. Let $e = \hat{x} - x \in \mathbb{R}^n$ denote the estimation error and $v \in \mathbb{R}^m$ the integral state. Define the augmented error-integral state

$$\xi = \begin{bmatrix} e \\ v \end{bmatrix} \in \mathbb{R}^{n+m}. \tag{25}$$

The error dynamics follow from (18)–(19) and (1):

$$\dot{\xi} = (A_x + L_x C_x)\xi \tag{26}$$

where A_x, C_x, L_x are defined in (21)–(22).

The observer design problem is to find L_x such that (26) is stable and achieves loop transfer recovery. Niemann et al. [3] show that LQG design on the extended system yields the observer ARE (24) with gain (23). Define

$$M := C_x^T \Sigma^{-1} C_x \succeq 0. \tag{27}$$

Here $A_x \in \mathbb{R}^{(n+m)\times(n+m)}$, $C_x \in \mathbb{R}^{p\times(n+m)}$, $\Gamma \succeq 0$, $\Sigma \succ 0$. Then the closed-loop matrix is

$$\mathcal{A}(P_o) := A_x + L_x C_x = A_x - P_o M. \tag{28}$$

B. Dual IRL for Output-Feedback Control

A simple parameter substitution approach of dual IRL would use observer error trajectories $\xi(t)$ from (26) and attempt the quadratic identity $\xi^T P_o \xi$. To have the same form of , this would require

$$(A_x + L_x C_x)^T P_o + P_o(A_x + L_x C_x) = -(\Gamma + L_x \Sigma L_x^T). \tag{29}$$

However, the observer ARE (24) implies a different Lyapunov equation. Substituting

$$L_x = -P_o C_x^T \Sigma^{-1}$$

into (24) and rearranging:

$$(A_x + L_x C_x)P_o + P_o(A_x + L_x C_x)^T = -(\Gamma + L_x \Sigma L_x^T). \tag{30}$$

Comparing (29) and (30): the transpose appears on different terms. These equations are equal only if $(A_x + L_x C_x)$ is symmetric, which does not hold in general. Therefore, the simple parameter substitution would require the trajectory of system $(A_x + L_x C_x)^T$. This is complex without knowing the system matrix A.

We choose a different approach to formulate the observer ARE as a Newton iteration. Define the Riccati operator

$$\mathcal{R}(P) := A_x P + P A_x^T + \Gamma - PMP. \tag{31}$$

The observer ARE is $\mathcal{R}(P_o) = 0$.

The Fréchet derivative of $\mathcal{R}$ at P in direction Δ is

$$D\mathcal{R}(P)[\Delta] = A_x \Delta + \Delta A_x^T - \Delta MP - PM\Delta. \tag{32}$$

Given iterate P_i, the Newton step solves

$$D\mathcal{R}(P_i)[\Delta_i] = -\mathcal{R}(P_i)$$

for Δ_i, then sets $P_{i+1} = P_i + \Delta_i$.

Lemma 2 (Newton–Kleinman Sylvester Equation)**.** *Define*

$$\mathcal{A}_i := A_x - P_i M, \qquad Q_i := \Gamma + P_i M P_i \tag{33}$$

with $L_i := -P_i C_x^T \Sigma^{-1}$, so that $Q_i = \Gamma + L_i \Sigma L_i^T$. Then P_{i+1} satisfies the Newton equation if and only if it solves the Sylvester equation

$$\mathcal{A}_i P_{i+1} + P_{i+1} \mathcal{A}_i^T = -Q_i. \tag{34}$$

Proof. Starting from

$$D\mathcal{R}(P_i)[\Delta_i] = -\mathcal{R}(P_i)$$

with $\Delta_i = P_{i+1} - P_i$:

$$A_x\Delta_i + \Delta_i A_x^T - \Delta_i M P_i - P_i M \Delta_i = -(A_x P_i + P_i A_x^T + \Gamma - P_i M P_i).$$

Add $A_x P_i + P_i A_x^T$ to both sides:

$$A_x P_{i+1} + P_{i+1} A_x^T - \Delta_i M P_i - P_i M \Delta_i = -\Gamma + P_i M P_i.$$

Substitute $\Delta_i = P_{i+1} - P_i$ and Rearrange:

$$(A_x - P_i M)P_{i+1} + P_{i+1}(A_x^T - M P_i) = -\Gamma - P_i M P_i.$$

Since $M = M^T$:

$$A_x^T - M P_i = (A_x - P_i M)^T = \mathcal{A}_i^T,$$

and thus $\mathcal{A}_i P_{i+1} + P_{i+1} \mathcal{A}_i^T = -Q_i$. □

The Sylvester equation (34) admits a data-driven solution using a bilinear identity.

Lemma 3 (Bilinear Integral Identity)**.** *Let $\eta(t)$ and $\zeta(t)$ be solutions of the adjoint system*

$$\dot{\eta} = \mathcal{A}_i^T \eta, \qquad \dot{\zeta} = \mathcal{A}_i^T \zeta \tag{35}$$

where $\mathcal{A}_i^T = (A_x - P_i M)^T = A_x^T - M P_i$*. Then* P_{i+1} *satisfies* (34) *if and only if for all* $t \geq 0$ *and* $T > 0$*:*

$$\zeta(t)^T P_{i+1}\eta(t) - \zeta(t+T)^T P_{i+1}\eta(t+T) = \int_t^{t+T} \zeta(\tau)^T Q_i \eta(\tau)\, d\tau. \quad (36)$$

Proof. For any constant matrix X:

$$\frac{d}{dt}(\zeta^T X \eta) = \dot{\zeta}^T X \eta + \zeta^T X \dot{\eta} = \zeta^T(\mathcal{A}_i X + X \mathcal{A}_i^T)\eta. \quad (37)$$

If $X = P_{i+1}$ satisfies (34):

$$\frac{d}{dt}(\zeta^T P_{i+1}\eta) = -\zeta^T Q_i \eta.$$

Integrating over $[t, t+T]$ yields (36).

Conversely, if (36) holds for all trajectories, differentiating at $T = 0$ recovers (34). □

The bilinear identity (36) enables least-squares solution of P_{i+1} from data.

a) Vectorization.: Using

$$\zeta^T P_{i+1}\eta = (\eta^T \otimes \zeta^T)\mathrm{vec}(P_{i+1}),$$

define for sampling windows $[t_j, t_j + T]$:

$$\Delta\phi_j := (\eta(t_j)^T \otimes \zeta(t_j)^T) - (\eta(t_j+T)^T \otimes \zeta(t_j+T)^T) \quad (38)$$

$$d_j := \int_{t_j}^{t_j+T} \zeta(\tau)^T Q_i \eta(\tau)\, d\tau. \quad (39)$$

Stacking N samples: $\Phi = [\Delta\phi_1^T \ \cdots \ \Delta\phi_N^T]^T$ and $d = [d_1 \ \cdots \ d_N]^T$, where $\Phi \in \mathbb{R}^{N \times (n+m)^2}$ and $d \in \mathbb{R}^N$. The identity (36) becomes

$$\Phi\, \mathrm{vec}(P_{i+1}) = d. \quad (40)$$

Persistence of excitation (PE) means

$$\mathrm{rank}(\Phi) = (n+m)^2, \quad \text{equivalently} \quad \Phi^T\Phi \succ 0.$$

Under PE, the least-squares solution is

$$\mathrm{vec}(P_{i+1}) = (\Phi^T\Phi)^{-1}\Phi^T d. \quad (41)$$

b) Obtaining adjoint trajectories.: During training, physical error trajectories evolve under $\dot{\xi} = \mathcal{A}_i \xi$. From measured $(\xi, \dot{\xi})$ pairs, identify $\mathcal{A}_i$ via least squares:

$$\widehat{\mathcal{A}}_i = \arg\min_{\mathcal{A}} \sum_k \|\dot{\xi}(t_k) - \mathcal{A}\xi(t_k)\|^2. \quad (42)$$

Then simulate

$$\dot{\eta} = \widehat{\mathcal{A}}_i^T \eta, \qquad \dot{\zeta} = \widehat{\mathcal{A}}_i^T \zeta$$

to obtain adjoint trajectories for (36). This explicit identification of $\mathcal{A}_i$ is analogous to how Vrabie's method embeds A in measured trajectories.

Algorithm 1 Dual IRL for PI Observer Gains

Require: Design weights (Γ, Σ); initial stabilizing gain L_1

1: **for** $i = 1, 2, \ldots$ **do**
2: **Forward rollout:** Apply L_i; collect measured trajectories $\xi(t)$ from (26) over windows $[t_j, t_j + T]$
3: **System identification:** Identify $\widehat{\mathcal{A}}_i$ from $(\xi, \dot{\xi})$ via (42)
4: **Adjoint simulation:** Simulate $\dot{\eta} = \widehat{\mathcal{A}}_i^T \eta$, $\dot{\zeta} = \widehat{\mathcal{A}}_i^T \zeta$
5: **Policy evaluation:** Form (Φ, d) from (38)–(39), solve (41) for P_{i+1}
6: **Policy improvement:** $L_{i+1} = -P_{i+1} C_x^T \Sigma^{-1}$
7: **Stability check:** Verify $V(\xi) = \xi^T P_{i+1} \xi$ decreases along trajectories under L_{i+1}. If not, set $L_{i+1} \leftarrow (1-\alpha)L_i + \alpha L_{i+1}$ with $\alpha \in (0,1)$ and recheck.
8: **end for**

Assumption 1. (i) $(A_x, \Gamma^{1/2})$ *is stabilizable,* (C_x, A_x) *is detectable, and a stabilizing solution* $P_o^* \succ 0$ *of* (24) *exists.*

(ii) L_1 *is stabilizing and the regressions satisfy persistence of excitation.*

(iii) $\mathcal{A}_i$ *is Hurwitz at each iteration (enforced by the stability check).*

Lemma 4 (Exact Stability Identity)**.** *Let* P_{i+1} *solve the Sylvester equation* (34)*, and define*

$$P := P_{i+1}, \qquad \Delta := P_{i+1} - P_i, \qquad S := A_x^T - A_x.$$

Then

$$\mathcal{A}_{i+1}^T P + P\mathcal{A}_{i+1} = -\Gamma - \Delta M \Delta + [S, P] - PMP - [[M, P], P] \quad (43)$$

where $[X, Y] := XY - YX$.

Proof. Expand $\mathcal{A}_{i+1}^T P + P\mathcal{A}_{i+1}$ using $\mathcal{A}_{i+1} = A_x - PM$. Apply the identity $A_x^T P + P A_x = (A_x P + P A_x^T) + [S, P]$ and substitute (34) for $A_x P + P A_x^T$. Expressing $P_i = P - \Delta$ and $Q_i = \Gamma + L_i \Sigma L_i^T$, then simplifying $PMP - MP^2 - P^2 M = -PMP - [[M, P], P]$ yields (43). □

Lemma 5 (Compensated Storage)**.** *Assume* $\mathcal{A}_i$ *is Hurwitz. Let* $J_i = J_i^T$ *be the unique solution of*

$$\mathcal{A}_i^T J_i + J_i \mathcal{A}_i = -[S, P_{i+1}]. \quad (44)$$

Then

$$\mathcal{A}_{i+1}^T(P + J_i) + (P + J_i)\mathcal{A}_{i+1} = -\Gamma - PMP - \Delta M \Delta - (M\Delta J_i + J_i \Delta M) - [[M, P], P]. \quad (45)$$

Proof. From Lemma 4:

$$\mathcal{A}_{i+1}^T P + P\mathcal{A}_{i+1} = -\Gamma - \Delta M \Delta + [S, P] - PMP - [[M, P], P].$$

Since $\mathcal{A}_{i+1} - \mathcal{A}_i = -\Delta M$:

$$\mathcal{A}_{i+1}^T J_i + J_i \mathcal{A}_{i+1} = \mathcal{A}_i^T J_i + J_i \mathcal{A}_i - (M\Delta J_i + J_i \Delta M) = -[S, P] - (M\Delta J_i + J_i \Delta M)$$

by (44). Summing cancels $[S, P]$ and gives (45). □

Lemma 6 (Stability Propagation)**.** *Let J_i be defined as in Lemma 5 and assume $P_{i+1} + J_i \succ 0$. Define*

$$W(x) := x^T(P_{i+1} + J_i)x, \qquad L_{i+1} := -P_{i+1}C_x^T\Sigma^{-1}.$$

If the smallest eigenvalue $\lambda_{\min}(\Gamma + L_{i+1}\Sigma L_{i+1}^T)$ satisfies

$$\lambda_{\min}(\Gamma + L_{i+1}\Sigma L_{i+1}^T) > 2\|M\|\|\Delta_i\|\|J_i\| + \|[[M, P_{i+1}], P_{i+1}]\| \quad (46)$$

then $\mathcal{A}_{i+1}$ is Hurwitz.

Proof. Take quadratic forms of (45) and use $PMP = L_{i+1}\Sigma L_{i+1}^T$:

$$\dot{W}(x) = -x^T(\Gamma + L_{i+1}\Sigma L_{i+1}^T)x - x^T\Delta M\Delta x - x^T(M\Delta J_i + J_i\Delta M)x - x^T[[M, P], P]x.$$

Bounding in the induced 2-norm,

$$|x^T(M\Delta J_i + J_i\Delta M)x| \le 2\|M\|\|\Delta\|\|J_i\|\|x\|^2,$$
$$|x^T[[M, P], P]x| \le \|[[M, P], P]\|\|x\|^2.$$

Dropping $-x^T\Delta M\Delta x \le 0$:

$$\dot{W}(x) \le -x^T(\Gamma + L_{i+1}\Sigma L_{i+1}^T)x + (2\|M\|\|\Delta\|\|J_i\| + \|[[M, P], P]\|)\|x\|^2.$$

Under (46) with $P + J_i \succ 0$, $\dot{W}(x) < 0$ for $x \neq 0$, so $\mathcal{A}_{i+1}$ is Hurwitz. □

Theorem 1 (Convergence to Observer ARE)**.** *Under Assumption 1, Algorithm 1 implements Newton–Kleinman iteration for the observer ARE* (24)*. The iterates satisfy*

$$P_i \to P_o^*, \qquad L_i \to L_x^* = -P_o^*C_x^T\Sigma^{-1}$$

with local quadratic convergence.

Proof. By Lemma 2, each iteration solves the Newton step for the Riccati operator (31). By Lemma 3, the least-squares solution (41) recovers this Newton step from data. By Lemma 6, stability propagates. The result follows from classical Newton–Kleinman convergence theory [12]. □

Corollary 1 (LTR and Time Recovery)**.** *If (Γ, Σ) are LQG/LTR weights satisfying $\Gamma_{22} \succ 0$ in the partition corresponding to the integral state, then the converged gain L_x^* achieves time recovery: $M_I(0) = 0$ [3].*

C. Controller Learning with Estimated States

The dual IRL procedure (Algorithm 1) solves the observer ARE (24) for P_o^*, yielding the PI observer gain $L_x^* = -P_o^*C_x^T\Sigma^{-1}$. This gain is partitioned as $L_x^* = [L_P^{*T}\ L_I^{*T}]^T$ corresponding to proportional and integral components.

In the idealized case where A is available for observer implementation, the PI observer

$$\dot{\hat{x}} = A\hat{x} + Bu + L_P^*(C\hat{x} - y) + Bv \quad (47)$$
$$\dot{v} = L_I^*(C\hat{x} - y) \quad (48)$$

provides state estimates $\hat{x}(t)$ from output measurements $y(t) = Cx(t)$. The estimation error $e = \hat{x} - x$ evolves as $\dot{e} = (A + L_P^*C)e + Bv$ with $\dot{v} = L_I^*Ce$. These dynamics are exponentially stable since P_o^* is the stabilizing solution of (24) by Assumption 1(i) and Theorem 1.

Standard IRL solves the controller ARE

$$A^TP_c + P_cA + Q - P_cBR^{-1}B^TP_c = 0 \quad (49)$$

using state measurements along trajectories. Given an initial stabilizing gain K_1, the algorithm iterates policy evaluation and improvement using estimated states. With the observer providing $\hat{x}(t)$, the integral Bellman equation becomes

$$\hat{x}(t)^TP_{c,i}\hat{x}(t) - \hat{x}(t{+}T)^TP_{c,i}\hat{x}(t{+}T) = \int_t^{t+T} \hat{x}(\tau)^TQ_{K,i}\,\hat{x}(\tau)\,d\tau \quad (50)$$

where $Q_{K,i} = Q + K_i^TRK_i$ and $u = -K_i\hat{x}$. With $N \ge n(n+1)/2$ windows satisfying persistence of excitation, least squares yields $P_{c,i}$, and policy improvement gives $K_{i+1} = R^{-1}B^TP_{c,i}$.

Theorem 2 (Two-Step Output-Feedback Learning)**.** *Consider system* (1)–(15) *with (A, B) stabilizable and (C, A) detectable. Under Assumption 1, persistence of excitation in both steps, an initial stabilizing controller K_1, and exact model knowledge for observer implementation* (47)*, the two-step procedure:*

(i) *Dual IRL (Algorithm 1) yields P_o^* solving the observer ARE* (24)*, with gain $L_x^* = -P_o^*C_x^T\Sigma^{-1}$;*
(ii) *Standard IRL with estimated states $\hat{x}$ yields P_c^* solving the controller ARE* (49)*, with gain $K^* = R^{-1}B^TP_c^*$;*

produces a stabilizing output-feedback controller.

Proof. Step 1 convergence follows from Theorem 1.

For Step 2, in the matched case the estimation error satisfies $e(t) \to 0$ exponentially. After transients decay, $\hat{x}(t) \approx x(t)$ and (50) approximates the true Bellman equation arbitrarily well. Given the initial stabilizing K_1, convergence $P_{c,i} \to P_c^*$ follows from Lemma 1.

For closed-loop stability, the control $u = -K^*\hat{x} = -K^*(x+e)$ applied to $\dot{x} = Ax+Bu$ gives $\dot{x} = (A-BK^*)x - BK^*e$. Combined with the error dynamics, the closed-loop system is

$$\begin{bmatrix}\dot{x}\\ \dot{e}\\ \dot{v}\end{bmatrix} = \underbrace{\begin{bmatrix}A - BK^* & -BK^* & 0\\ 0 & A + L_P^*C & B\\ 0 & L_I^*C & 0\end{bmatrix}}_{=:\bar{A}} \begin{bmatrix}x\\ e\\ v\end{bmatrix}. \quad (51)$$

The matrix $\bar{A}$ is block upper triangular. Its eigenvalues are those of the diagonal blocks: $A - BK^*$ and $\begin{bmatrix}A + L_P^*C & B\\ L_I^*C & 0\end{bmatrix}$. The first is Hurwitz since P_c^* is the stabilizing solution of (49); the second is Hurwitz by Assumption 1(i). Hence $\bar{A}$ is Hurwitz. □

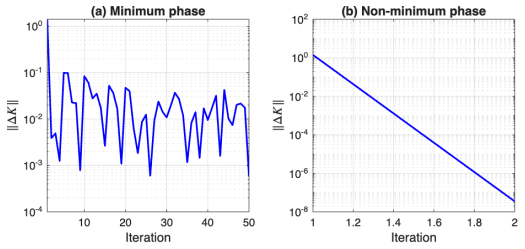

Fig. 1: Controller learning convergence: (a) minimum phase, (b) non-minimum phase.

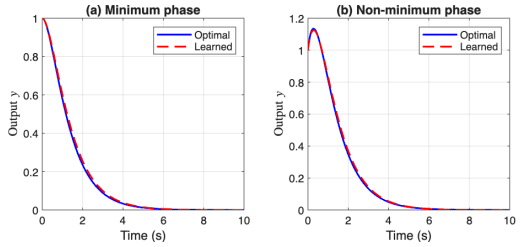

Fig. 2: Closed-loop output response: optimal controller (solid) vs. learned controller (dashed). (a) Minimum phase, (b) non-minimum phase.

IV. Simulation Results

Consider the second-order system with

$$A = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{52}$$

To test both minimum phase and non-minimum phase systems, we use $C_1 = \begin{bmatrix} 1 & 0 \end{bmatrix}$ and $C_2 = \begin{bmatrix} 1 & -0.5 \end{bmatrix}$ respectively. We set the LQR cost matrices to $Q = I$ and $R = 1$, giving optimal gain $K^* = \begin{bmatrix} 0.236 & 0.236 \end{bmatrix}$.

We first learn the observer gains using Algorithm 1 with the true system matrix A. The dual IRL procedure converges within 4–6 iterations for both minimum and non-minimum phase cases.

We then learn the controller gains using IRL with estimated states $\hat{x}$ from the learned PI observer. Convergence occurs within 40–50 iterations for the minimum phase case, as shown in Fig. 1. The controller gain update $\|\Delta K\|$ decreases over iterations, indicating convergence of the policy iteration.

We compare the closed-loop output response of the learned controller against the optimal LQR controller in Fig. 2. The cost ratios are within 2% of optimal for both minimum and non-minimum phase systems. The PI observer structure achieves near-optimal performance for both minimum and non-minimum phase systems.

V. Conclusion

We presented a dual IRL framework that learns PI observer gains for loop transfer recovery without knowing the system matrix A. Dual IRL learns the observer gains from state measurements during an offline training phase. Primal IRL then learns controller gains from output measurements using the learned observer. The PI observer achieves time recovery for both minimum and non-minimum phase systems.

The two-step framework provides a complete output-feedback learning solution. Theorem 1 establishes convergence of the dual IRL algorithm to the optimal PI observer gains. Corollary 1 guarantees time recovery under appropriate LQG/LTR weight selection. Theorem 2 shows that the combined two-step procedure yields a stabilizing output-feedback controller.

Simulations on minimum and non-minimum phase systems demonstrate that the learned controller achieves near-optimal performance, with cost ratios within 2% of the optimal LQR solution. The framework successfully handles both system types using the same algorithmic structure.

Future work includes analysis of deployment with approximate system models, extension to time-varying or nonlinear systems, and integration with modern data-driven techniques.

References

[1] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, and F. L. Lewis, "Adaptive optimal control for continuous-time linear systems based on policy iteration," *Automatica*, vol. 45, no. 2, pp. 477–484, 2009.

[2] G. Stein and M. Athans, "The LQG/LTR procedure for multivariable feedback control design," *IEEE Transactions on Automatic Control*, vol. 32, no. 2, pp. 105–114, 1987.

[3] H. H. Niemann, J. Stoustrup, B. Shafai, and S. Beale, "LTR design of proportional-integral observers," *International Journal of Robust and Nonlinear Control*, vol. 5, no. 7, pp. 671–693, 1995.

[4] F. L. Lewis and K. G. Vamvoudakis, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE Circuits and Systems Magazine*, vol. 11, no. 3, pp. 32–50, 2011.

[5] H. Modares, F. L. Lewis, and Z.-P. Jiang, "Optimal output-feedback control of unknown continuous-time linear systems using off-policy reinforcement learning," *IEEE Transactions on Cybernetics*, vol. 46, no. 11, pp. 2401–2410, 2016.

[6] S. A. A. Rizvi and Z. Lin, "Output feedback reinforcement learning control for the continuous-time linear quadratic regulator problem," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 7, pp. 2257–2268, 2020.

[7] V. K. Mishra, H. J. van Waarde, and N. Bajcinca, "Data-driven criteria for detectability and observer design for LTI systems," in *Proceedings of the 61st IEEE Conference on Decision and Control (CDC)*. IEEE, 2022, pp. 4846–4852.

[8] E. C. Turan and G. Ferrari-Trecate, "Data-driven unknown-input observers and state estimation," *IEEE Control Systems Letters*, vol. 6, pp. 1424–1429, 2022.

[9] R. Adachi and Y. Wakasa, "Design of full state observer based on data-driven dual system representation," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E106.A, no. 5, pp. 736–743, 2023.

[10] S. Beale and B. Shafai, "Robust control system design with a proportional integral observer," *International Journal of Control*, vol. 50, no. 1, pp. 97–111, 1989.

[11] B. Shafai and M. Saif, "Proportional-integral observer in robust control, fault detection, and decentralized control of dynamic systems," in *Control and Systems Engineering: A Report on Four Decades of Contributions*. Springer, 2015, pp. 13–43.

[12] D. L. Kleinman, "An easy way to stabilize a linear constant system," *IEEE Transactions on Automatic Control*, vol. 15, no. 6, p. 692, 1970.