---

**Article:**
Sanusi, I. orcid.org/0000-0002-3198-9048, Mills, A. orcid.org/0000-0002-6798-5284, Dodd, T. et al. (1 more author) (2020) Online optimal and adaptive integral tracking control for varying discrete-time systems using reinforcement learning. International Journal of Adaptive Control and Signal Processing. ISSN 0890-6327

https://doi.org/10.1002/acs.3115

---

Received: 4 June 2019 | Revised: 4 February 2020 | Accepted: 24 March 2020

DOI: 10.1002/acs.3115



RESEARCH ARTICLE

WILEY

# Online optimal and adaptive integral tracking control for varying discrete-time systems using reinforcement learning

**Ibrahim Sanusi** | **Andrew Mills** | **Tony Dodd** | **George Konstantopoulos**

Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, UK

**Correspondence**
Ibrahim Sanusi, Department of Automatic Control and Signal Processing, University of Sheffield, Sheffield, UK.
Email: iesanusi1@sheffield.ac.uk

**Summary**

Conventional closed-form solution to the optimal control problem using optimal control theory is only available under the assumption that there are known system dynamics/models described as differential equations. Without such models, reinforcement learning (RL) as a candidate technique has been successfully applied to iteratively solve the optimal control problem for unknown or varying systems. For the optimal tracking control problem, existing RL techniques in the literature assume either the use of a predetermined feedforward input for the tracking control, restrictive assumptions on the reference model dynamics, or discounted tracking costs. Furthermore, by using discounted tracking costs, zero steady-state error cannot be guaranteed by the existing RL methods. This article therefore presents an optimal online RL tracking control framework for discrete-time (DT) systems, which does not impose any restrictive assumptions of the existing methods and equally guarantees zero steady-state tracking error. This is achieved by augmenting the original system dynamics with the integral of the error between the reference inputs and the tracked outputs for use in the online RL framework. It is further shown that the resulting value function for the DT linear quadratic tracker using the augmented formulation with integral control is also quadratic. This enables the development of Bellman equations, which use only the system measurements to solve the corresponding DT algebraic Riccati equation and obtain the optimal tracking control inputs online. Two RL strategies are thereafter proposed based on both the value function approximation and the Q-learning along with bounds on excitation for the convergence of the parameter estimates. Simulation case studies show the effectiveness of the proposed approach.

**KEYWORDS**

adaptive control, adaptive dynamic programming, optimal tracking control, Q-function approximation, reinforcement learning

## 1 | INTRODUCTION

Reinforcement learning (RL) is a type of machine learning technique that has been used extensively in the area of computing and artificial intelligence to solve complex optimization problems.[1,2] Due to its successes, there have been concerted efforts by researchers in the control community to explore the overlap between RL and optimal control theory, which usually involves solving the general-purpose Hamilton-Jacobi Bellman (HJB) equations. The conventional approach to optimal control minimizes a weighted cost function composed of state and control minimization objectives. A closed-form solution (eg, Riccati equation) to this problem is available under the assumption that there are known system dynamics described as differential equations.[3] Without such models, this closed-form solution is not available. RL has been successfully applied to iteratively optimize these control cost functions for unknown or varying systems by providing solutions to the HJB equations online.[4,5] This article addresses the optimal online tracking control of varying systems under less restrictive assumptions than previously proposed solutions and builds upon the objectives of performance seeking and real-time optimization techniques.[6,7]

Variations occur in systems due to a number of factors including degradation and changing operating conditions, which can result in a reduction in system performance. From the traditional control perspective, adaptive control offers strategies to compensate for the system variations and can be indirect or direct. Indirect adaptive schemes use the system measurements to learn new system models, which are then used in a conventional model-based control design while direct schemes use the system measurements to adapt some parameterized controllers. In both of these schemes, optimality is not directly achieved in the sense of optimizing some user-defined cost function.[8] RL enables the development of both optimal and adaptive strategies that are able to cope with the system variations by using only the system measurements and has been linked to both optimal and adaptive control.[9-12] These enabling methods are therefore classed as intelligent, defined as self-diagnostic, prognostic, and optimizing, resulting in a through-life adaptation strategy and has been widely reported in many applications.[13-16]

Mathematical implementation of RL is enabled through approximate/adaptive dynamic programming (ADP)[17,18] and has been described by different other labels including neurodynamic programming and adaptive critic designs.[9,19,20] Through interaction with the systems, the RL-ADP strategies have been applied to incrementally improve the desired control behavior for the regulation of feedback systems involving unknown continuous and discrete-time (DT) dynamics.[21-28] For the tracking control problem, existing RL strategies are split between methods that employ dynamics inversion and those that use an augmented formulation.[29]

In Reference [30], a method employing the dynamics inversion for the infinite-time tracking control for the DT nonlinear systems has been proposed. The method assumes that the steady-state feedforward control input is known a priori and uses a new quadratic performance index to compute the feedback control input using RL techniques. A finite-horizon neurooptimal equivalent that minimizes the tracking error over a finite horizon but equally assumes a known steady-state feedforward input is proposed in Reference [31]. Likewise, the authors in Reference [32] have proposed an optimal tracking control for nonlinear DT systems that uses three online approximators and a heuristic tuning law for the feedforward portion of the control input but assumes bounded approximation errors with fixed model structures for the identification of the system parameters. A similar approach that uses three approximators and a generalized policy iteration ADP to include two iteration procedures for the tracking control has been proposed in Reference [33]. This approach learns a model of the system dynamics online and generally requires pretrained models while assuming fixed model structures for the identification. In contrast to these approaches, strategies that employ the augmented formulation obviate the need to have a predetermined feedforward control input by transforming the tracking control into a regulation problem using augmented system states. In Reference [34], an approach that transforms the optimal tracking control problem into a regulation problem has been provided by augmenting the system states with the reference model dynamics for linear DT systems. The approach assumes that the reference generator matrix is governed by dynamics that tend toward zero, thereby limiting any practical usage of the approach in the case of nonzero reference inputs. Consequently, an approach that relaxes the restriction on the reference dynamics by using a discounted tracking cost is given in Reference [35]. Extension of the approaches to DT nonlinear systems using neural networks and discounted tracking cost is given in Reference [36], while the continuous time equivalents of the strategies to include input constraints are given in Reference [37]. However, by introducing a discounted tracking cost, convergence of the tracking error to zero can no longer be guaranteed, thereby limiting the practicality of the approaches. As a result, none of the existing optimal online tracking RL strategies are able to guarantee a nonzero steady-state tracking error using the system dynamics inversion

or augmented formulations. Moreover, the restrictive assumptions involved in both formulations make the approaches less desirable for use in practical tracking applications.

This article therefore presents an optimal online reinforcement learning tracking control framework for DT systems, which uses an augmented formulation with integral control and transforms the DT optimal tracking control problem into one of regulation. In contrast to the approaches discussed in Reference [30,31,34-36], the proposed framework removes the need to have either a predetermined feedforward control input, any restrictive assumptions on the reference model dynamics, or a discounted tracking cost that limits the practical applications of existing online tracking RL approaches. Furthermore, the proposed RL framework eliminates steady-state tracking error and is able to cope with systems with unknown or varying dynamics leading to a through-life adaptation strategy. It is shown in this article that the resulting value function for the DT linear quadratic tracker (LQT) using the augmented formulation with integral control is also quadratic. This enables the development of Bellman equations, which use only the system measurements to solve the corresponding DT algebraic Riccati equation (ARE) and obtain the optimal control inputs online. Two RL strategies are proposed in this article based on both the value function approximation (VFA) and Q-learning along with bounds on excitation for the convergence of the parameter estimates.

The rest of the article is organized as follows. Section 2 presents the general optimal tracking control problem for DT systems along with the existing solution strategies and their limitations. Section 3 presents the proposed augmented formulation, while Section 4 provides the model-based control solution to the augmented LQT problem. In Section 5, the two RL strategies and an intelligent framework for the augmented LQT problem are developed and respective algorithms provided, while Section 6 gives two representative simulation case studies using the proposed algorithms.

## 2 | PROBLEM FORMULATION

Consider the control affine-in-input discrete-time system with the following dynamics:

$$\begin{aligned} \boldsymbol{x}_{k+1} &= f(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_k \\ \boldsymbol{y}_k &= h(\boldsymbol{x}_k), \end{aligned} \tag{1}$$

where $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{u} \in \mathbb{R}^m$, and $\boldsymbol{y} \in \mathbb{R}^p$ are, respectively, the system states, inputs, and outputs. The aim of the tracking control problem is to minimize a cost function:

$$J(\boldsymbol{x}_k, \boldsymbol{u}_k) = \sum_{i=k}^{\infty} \lambda^{i-k} L_i^1, \tag{2}$$

where $L_i^1 = (\boldsymbol{y}_i - \boldsymbol{r}_i)^\top Q_T (\boldsymbol{y}_i - \boldsymbol{r}_i) + \boldsymbol{u}_i^\top R \boldsymbol{u}_i$ is a quadratic energy function with $Q_T \geq 0$ and $R > 0$, $\boldsymbol{r}$ is a desired reference trajectory, and $0 < \lambda \leq 1$ is the discount factor.

It can be shown that for the special case where (1) is a linear time invariant (LTI) system:

$$\begin{aligned} \boldsymbol{x}_{k+1} &= A\boldsymbol{x}_k + B\boldsymbol{u}_k \\ \boldsymbol{y}_k &= C\boldsymbol{x}_k \end{aligned} \tag{3}$$

the standard solution to the given infinite horizon tracking problem using calculus of variation with $\lambda = 1$ is:[10]

$$\boldsymbol{u}_k = -K^x \boldsymbol{x}_k + K^v \boldsymbol{v}_{k+1}, \tag{4}$$

where:

$$\begin{aligned} &K^x = (B^\top P B + R)^{-1} B^\top P A; \quad K^v = (B^\top P B + R)^{-1} B^\top \\ &\boldsymbol{v}_k = (A - BK^x)^\top \boldsymbol{v}_{k+1} + C^\top Q_T \boldsymbol{r}_k \end{aligned} \tag{5}$$

and $P = P^\top > 0$ is the solution to the algebraic Riccati equation:

$$C^\top Q_T C - P + A^\top PA - A^\top PB(B^\top PB + R)^{-1}B^\top PA = 0.$$

**Assumption 1.** Sufficient conditions for a solution are that the pair $(A, B)$ and $(A, \sqrt{Q_T}C)$ are, respectively, controllable and observable.

It is noted that the control input (4) for the tracking problem consists of both a feedback term $K^x$ that stabilizes the system and a feedforward term $K^v$ for reference tracking. Furthermore, the given standard solution is noncausal,[35] as it is dependent on a *backward in-time* recursion of variable $\boldsymbol{v}_k$. An implication of this is that the standard solution to the tracking problem can only be obtained offline and with full knowledge of the system dynamics. Consequently, causal solution strategies that can be computed online have been proposed in the literature and will now be briefly presented.

Causal solution to the optimal tracking problem

Existing causal solution strategies to the online tracking control problem can be categorized into two and are briefly summarized as follows:

(1) Strategies using dynamics inversion:[30-32] These methods assume that the desired reference dynamics is given as:

$$\boldsymbol{r}_{k+1} = f(\boldsymbol{r}_k) + g(\boldsymbol{r}_k)\boldsymbol{u}_{d,k}, \tag{6}$$

where $\boldsymbol{u}_{d,k} = g^{-1}(\boldsymbol{r}_k)\,(\boldsymbol{r}_{k+1} - f(\boldsymbol{r}_k))$ is the feedforward tracking control input. The tracking error $\boldsymbol{e}_k = \boldsymbol{x}_k - \boldsymbol{r}_k$ is minimized by defining a cost function as:

$$J(\boldsymbol{e}_k, \boldsymbol{u}_{e,k}) = \sum_{i=k}^{\infty} \left( \boldsymbol{e}_i^\top Q_e \boldsymbol{e}_i + \boldsymbol{u}_{e,i}^\top R_e \boldsymbol{u}_{e,i} \right) \tag{7}$$

with $Q_e \geq 0, R_e > 0$ and where $\boldsymbol{u}^*_{e,k} = -\frac{1}{2}R_e^{-1}g^\top(\boldsymbol{x}_k)\frac{\partial J^*(\boldsymbol{e}_{k+1})}{\partial \boldsymbol{e}_{k+1}}$ is the feedback tracking control input. The overall control input is thus given as:

$$\boldsymbol{u}^*_k = \boldsymbol{u}^*_{e,k} + \boldsymbol{u}_{d,k}. \tag{8}$$

*Remarks*

- Complete knowledge of the system dynamics is needed to compute the feedforward term $\boldsymbol{u}_d$, with a further assumption that function $g(\boldsymbol{r})$ is invertible.
- Online implementation of this approach therefore assumes $\boldsymbol{u}_d$ is known a priori, and only the feedback term $\boldsymbol{u}_e$ is computed online. As a result, practical online adaptation strategies to cope with varying or unknown system dynamics are limited using this strategy.

(2) Strategies using augmented formulation:[34-36,38,39] These methods enable the simultaneous online computation of both the feedforward and feedback terms of the tracking control input. This approach assumes that the reference dynamics is governed by:

$$\boldsymbol{r}_{k+1} = \psi(\boldsymbol{r}_k), \tag{9}$$

where $\psi(\boldsymbol{r}_k)$ is some reference generator model with $\psi(0) = 0$. An augmented system is then formulated using the tracking error and the reference dynamics as:

$$\begin{aligned} \boldsymbol{X}^r_{k+1} &= \begin{bmatrix} f(\boldsymbol{e}_k + \boldsymbol{r}_k) - \psi(\boldsymbol{r}_k) \\ \psi(\boldsymbol{r}_k) \end{bmatrix} + \begin{bmatrix} g(\boldsymbol{e}_k + \boldsymbol{r}_k) \\ 0 \end{bmatrix} \boldsymbol{u}_k \\ &= F_r(\boldsymbol{X}^r_k) + G_r(\boldsymbol{X}^r_k)\boldsymbol{u}_k, \end{aligned} \tag{10}$$

where $\boldsymbol{X}_k^r = \begin{bmatrix} \boldsymbol{e}_k \\ \boldsymbol{r}_k \end{bmatrix}$ and with a new cost defined as:

$$J(\boldsymbol{x}_k, \boldsymbol{r}_k, \boldsymbol{u}_k) = \sum_{i=k}^{\infty} \lambda^{i-k} \left( \boldsymbol{e}_i^\top Q_e \boldsymbol{e}_i + \boldsymbol{u}_i^\top R_e \boldsymbol{u}_i \right) \tag{11}$$

with $\boldsymbol{u}_k^* = -\frac{\lambda}{2} R_e^{-1} G_r^\top (\boldsymbol{X}_k^r) \frac{\partial J^*(\boldsymbol{X}_{k+1}^r)}{\partial \boldsymbol{X}_{k+1}^r}$. This way, the tracking problem is recast as a regulation problem, the solution of which gives both the feedforward and feedback terms of the control input online.

*Remarks*

- It is assumed that $\psi(\boldsymbol{r}_k) \to 0$ as $k \to \infty$; where this is not, a discounted performance function with $0 < \lambda \leq 1$ must be used to ensure the value of the cost function remains finite.[35] This assumption poses a restriction on the class of reference generator that can be used with the approach.
- By using a discount factor in the cost function, this approach cannot guarantee zero steady-state tracking error as discussed in Reference [35]. This restrictive assumption on the reference dynamics and discounted cost makes the approach less desirable for use in practical tracking applications.

Consequently, existing RL techniques for the online optimal tracking control problem assume either the use of a predetermined feedforward input for the tracking control or use restrictive assumptions on the reference model dynamics and discounted tracking costs. In the following, a new augmented formulation for the online optimal tracking control problem that guarantees zero steady-state tracking error without imposing any restrictive assumptions on the reference dynamics or discounted performance cost is proposed to overcome the limitations of the existing strategies.

## 3 | AUGMENTED FORMULATION FOR THE OPTIMAL TRACKING PROBLEM WITH INTEGRAL CONTROL

Consider again the optimal tracking control problem for system (1) and let a new state $\dot{\boldsymbol{z}}$ be defined as the integral of the difference between the desired reference and the system output as:

$$\dot{\boldsymbol{z}}(t) = \int (\boldsymbol{r}(t) - \boldsymbol{y}(t))\, dt \in \mathbb{R}^p. \tag{12}$$

Using Euler's approximation, an equivalent discrete-time state with sampling time $t_s$ gives:

$$\boldsymbol{z}_{k+1} = \boldsymbol{z}_k + t_s(\boldsymbol{r}_k - h(\boldsymbol{x}_k)). \tag{13}$$

An augmented system can therefore be formed using the new state as:

$$\begin{bmatrix} \boldsymbol{x}_{k+1} \\ \boldsymbol{z}_{k+1} \end{bmatrix} = \begin{bmatrix} f(\boldsymbol{x}_k) \\ \boldsymbol{z}_k - t_s h(\boldsymbol{x}_k) \end{bmatrix} + \begin{bmatrix} g(\boldsymbol{x}_k) \\ 0 \end{bmatrix} \boldsymbol{u}_k + \begin{bmatrix} 0 \\ t_s I \end{bmatrix} \boldsymbol{r}_k. \tag{14}$$

At steady state, the augmented system (14) becomes:

$$\begin{bmatrix} \boldsymbol{x}_\infty \\ \boldsymbol{z}_\infty \end{bmatrix} = \begin{bmatrix} f(\boldsymbol{x}_\infty) \\ \boldsymbol{z}_\infty - t_s h(\boldsymbol{x}_\infty) \end{bmatrix} + \begin{bmatrix} g(\boldsymbol{x}_\infty) \\ 0 \end{bmatrix} \boldsymbol{u}_\infty + \begin{bmatrix} 0 \\ t_s I \end{bmatrix} \boldsymbol{r}_\infty. \tag{15}$$

For a constant reference signal, that is, $\boldsymbol{r}_\infty = \boldsymbol{r}_k$, subtracting (15) from (14) gives:

$$\begin{bmatrix} \boldsymbol{x}_{k+1} - \boldsymbol{x}_\infty \\ \boldsymbol{z}_{k+1} - \boldsymbol{z}_\infty \end{bmatrix} = \begin{bmatrix} f(\boldsymbol{x}_k) - f(\boldsymbol{x}_\infty) \\ \boldsymbol{z}_k - \boldsymbol{z}_\infty - t_s \left( h(\boldsymbol{x}_x) - h(\boldsymbol{x}_\infty) \right) \end{bmatrix} + \begin{bmatrix} g(\boldsymbol{x}_k)\boldsymbol{u}_k - g(\boldsymbol{x}_\infty)\boldsymbol{u}_\infty \\ 0 \end{bmatrix}. \tag{16}$$

Further simplification of (16) becomes:

$$\boldsymbol{X}_{k+1} = F(\boldsymbol{X}_k) + G(\boldsymbol{X}_k)\tilde{\boldsymbol{u}}_k \tag{17}$$

with $\boldsymbol{X}_k = \begin{bmatrix} \boldsymbol{x}_k - \boldsymbol{x}_\infty \\ \boldsymbol{z}_k - \boldsymbol{z}_\infty \end{bmatrix} \in \mathbb{R}^{n+p}$, $\tilde{\boldsymbol{u}}_k = (\boldsymbol{u}_k - \boldsymbol{u}_\infty) \in \mathbb{R}^m$ and where $F(\boldsymbol{X}_k) = \begin{bmatrix} f(\boldsymbol{x}_k) - f(\boldsymbol{x}_\infty) + g(\boldsymbol{x}_k)\boldsymbol{u}_\infty - g(\boldsymbol{x}_\infty)\boldsymbol{u}_\infty \\ \boldsymbol{z}_k - \boldsymbol{z}_\infty - t_s\left(h(\boldsymbol{x}_x) - h(\boldsymbol{x}_\infty)\right) \end{bmatrix}$ and $G(\boldsymbol{X}_k) = \begin{bmatrix} g(\boldsymbol{x}_k) \\ 0 \end{bmatrix}$.

The tracking cost (2) is therefore redefined as:

$$J(\boldsymbol{X}_k, \tilde{\boldsymbol{u}}_k) = \sum_{i=k}^{\infty} \lambda^{i-k}\left(\boldsymbol{X}_i^\top Q_1 \boldsymbol{X}_i + \tilde{\boldsymbol{u}}_i^\top R \tilde{\boldsymbol{u}}_i\right), \tag{18}$$

where $Q_1 \in \mathbb{R}^{(n+p)\times(n+p)}$. This way, the tracking problem is converted to that of regulation such that the control input for a minimum of (18) eliminates the steady-state error by ensuring that $\boldsymbol{x}_k \to \boldsymbol{x}_\infty$ and $\boldsymbol{z}_k \to \boldsymbol{z}_\infty$ as $\boldsymbol{X}_k \to 0$. Furthermore, as the new augmented system states are not dependent on the reference dynamics, this approach removes the restrictive assumptions of the existing methods. An equivalent difference equation to (18) for a given fixed policy is given by the value function and defined as:

$$\begin{aligned} V(\boldsymbol{X}_k) &= \sum_{i=k}^{\infty} \lambda^{i-k}\left(\boldsymbol{X}_i^\top Q_1 \boldsymbol{X}_i + \tilde{\boldsymbol{u}}_i^\top R \tilde{\boldsymbol{u}}_i\right) = \sum_{i=k}^{\infty} \lambda^{i-k} L_i \\ &= L_k + \lambda \sum_{i=k+1}^{\infty} \lambda^{i-(k+1)} L_i \\ \therefore V(\boldsymbol{X}_k) &= L_k + \lambda V(\boldsymbol{X}_{k+1}), \end{aligned} \tag{19}$$

where $L_k = \boldsymbol{X}_k^\top Q_1 \boldsymbol{X}_k + \tilde{\boldsymbol{u}}_k^\top R \tilde{\boldsymbol{u}}_k$ and $V(0) = 0$. Using the Bellman principle of optimality,[9] the optimum value becomes:

$$V^*(\boldsymbol{X}_k) = \min_u \left(L_k + \lambda V^*(\boldsymbol{X}_{k+1})\right). \tag{20}$$

Equation (20) gives the DT Hamilton-Jacobi-Bellman (HJB) equation for the augmented tracking formulation with integral control from which the optimal tracking control input is obtained as:

$$\tilde{\boldsymbol{u}}_k^* = \arg\min_{\tilde{u}} \left(L_k + \lambda V^*(\boldsymbol{X}_{k+1})\right). \tag{21}$$

## 4 | MODEL-BASED SOLUTION TO THE AUGMENTED LQT FORMULATION WITH INTEGRAL CONTROL

A model-based control solution to the optimal tracking problem using the augmented formulation with integral control for discrete-time (DT) linear systems is first presented to be used in comparison with the model-free RL approaches introduced in later sections. Using the system dynamics in (3), the augmented system of (17) becomes:

$$\begin{aligned} \boldsymbol{X}_{k+1} &= \begin{bmatrix} \boldsymbol{x}_{k+1} - \boldsymbol{x}_\infty \\ \boldsymbol{z}_{k+1} - \boldsymbol{z}_\infty \end{bmatrix} = \begin{bmatrix} A & 0 \\ -t_s C & I \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_k - \boldsymbol{x}_\infty \\ \boldsymbol{z}_k - \boldsymbol{z}_\infty \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} (\boldsymbol{u}_k - \boldsymbol{u}_\infty) \\ &= A_1 \boldsymbol{X}_k + B_1 \tilde{\boldsymbol{u}}_k. \end{aligned} \tag{22}$$

**Lemma 1.** *(Quadratic value function). Given the LQT cost of (18) and system with dynamics (22), for any stabilizing control law:*

$$\tilde{\boldsymbol{u}}_k = -\begin{bmatrix} K_x & -K_I \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_k - \boldsymbol{x}_\infty \\ \boldsymbol{z}_k - \boldsymbol{z}_\infty \end{bmatrix} = -K_1 \boldsymbol{X}_k, \tag{23}$$

*where $K_1 \in \mathbb{R}^{m\times(n+p)}$, $K_x \in \mathbb{R}^{m\times n}$, and $K_I \in \mathbb{R}^{m\times p}$; the value function for the augmented formulation with integral control is quadratic for some matrix $P_1 = P_1^\top > 0 \in \mathbb{R}^{(n+p)\times(n+p)}$ and given as:*

$$V(\boldsymbol{X}_k) = \boldsymbol{X}_k^\top P_1 \boldsymbol{X}_k. \tag{24}$$

For simplicity of notation in subsequent analysis, ($\boldsymbol{x}_\infty$ and $\boldsymbol{z}_\infty$) are dropped in the augmented states.

*Proof.* Change the lower limit for the summation in (19) and substitute for $\tilde{\boldsymbol{u}}_k$ to give:

$$V(\boldsymbol{X}_k) = \sum_{i=0}^{\infty} \lambda^i \left[\boldsymbol{X}_{i+k}^\top Q_1 \boldsymbol{X}_{i+k} + \boldsymbol{X}_{i+k}^\top K_1^\top R K_1 \boldsymbol{X}_{i+k}\right]. \tag{25}$$

Noting that $\boldsymbol{X}_{i+k} = (A_1 - B_1K_1)^i \boldsymbol{X}_k = \left(\begin{bmatrix} A & 0 \\ -t_sC & I \end{bmatrix} - \begin{bmatrix} BK_x & -BK_I \\ 0 & 0 \end{bmatrix}\right)^i \begin{bmatrix} \boldsymbol{x}_k \\ \boldsymbol{z}_k \end{bmatrix} = M \begin{bmatrix} \boldsymbol{x}_k \\ \boldsymbol{z}_k \end{bmatrix}$, where $M = \begin{bmatrix} A - BK_x & BK_I \\ -t_sC & I \end{bmatrix}^i = \begin{bmatrix} M_{11} \in \mathbb{R}^{n\times n} & M_{12} \in \mathbb{R}^{n\times p} \\ M_{21} \in \mathbb{R}^{p\times n} & M_{22} \in \mathbb{R}^{p\times p} \end{bmatrix}$ and $Q_1 = \begin{bmatrix} Q_{11} \in \mathbb{R}^{n\times n} & Q_{12} \in \mathbb{R}^{n\times p} \\ Q_{21} \in \mathbb{R}^{p\times n} & Q_{22} \in \mathbb{R}^{p\times p} \end{bmatrix}$

Equation (25) becomes:

$$\begin{aligned} V(\boldsymbol{X}_k) = \sum_{i=0}^{\infty} \lambda^i \Bigg[ & \begin{bmatrix} M_{11}\boldsymbol{x}_k + M_{12}\boldsymbol{z}_k \\ M_{21}\boldsymbol{x}_k + M_{22}\boldsymbol{z}_k \end{bmatrix}^\top \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \begin{bmatrix} M_{11}\boldsymbol{x}_k + M_{12}\boldsymbol{z}_k \\ M_{21}\boldsymbol{x}_k + M_{22}\boldsymbol{z}_k \end{bmatrix} \\ & + \begin{bmatrix} M_{11}\boldsymbol{x}_k + M_{12}\boldsymbol{z}_k \\ M_{21}\boldsymbol{x}_k + M_{22}\boldsymbol{z}_k \end{bmatrix}^\top \begin{bmatrix} K_x^\top R K_x & -K_x^\top R K_I \\ -K_I^\top R K_x & K_I^\top R K_I \end{bmatrix} \begin{bmatrix} M_{11}\boldsymbol{x}_k + M_{12}\boldsymbol{z}_k \\ M_{21}\boldsymbol{x}_k + M_{22}\boldsymbol{z}_k \end{bmatrix} \Bigg]. \end{aligned} \tag{26}$$

Therefore,

$$V(\boldsymbol{X}_k) = \boldsymbol{x}_k^\top P_1^{(11)} \boldsymbol{x}_k + \boldsymbol{x}_k^\top P_1^{(12)} \boldsymbol{z}_k + \boldsymbol{z}_k^\top P_1^{(21)} \boldsymbol{x}_k + \boldsymbol{z}_k^\top P_1^{(22)} \boldsymbol{z}_k = \boldsymbol{X}_k^\top P_1 \boldsymbol{X}_k, \tag{27}$$

where $P_1 = \begin{bmatrix} P_1^{(11)} & P_1^{(12)} \\ P_1^{(21)} & P_1^{(22)} \end{bmatrix}$ and $P_1^{(11)} = \sum_{i=0}^{\infty} \lambda^i [M_{11}^\top Q_{11} M_{11} + M_{12}^\top Q_{12} M_{11} + M_{11}^\top Q_{12} M_{21} + M_{12}^\top Q_{22} M_{21} + M_{11}^\top K_x^\top R K_x M_{11} - M_{21}^\top K_I^\top R K_x M_{11} - M_{11}^\top K_x^\top R K_I M_{12} + M_{21}^\top K_I^\top R K_I M_{12}]$

$P_1^{(12)} = \sum_{i=0}^{\infty} \lambda^i [M_{11}^\top Q_{11} M_{12} + M_{12}^\top Q_{21} M_{12} + M_{11}^\top Q_{12} M_{22} + M_{12}^\top Q_{22} M_{22} + M_{11}^\top K_x^\top R K_x M_{12} - M_{21}^\top K_I^\top R K_x M_{12} - M_{11}^\top K_x^\top R K_I M_{22} + M_{21}^\top K_I^\top R K_I M_{22}]$

$P_1^{(21)} = \sum_{i=0}^{\infty} \lambda^i [M_{12}^\top Q_{11} M_{11} + M_{22}^\top Q_{21} M_{11} + M_{12}^\top Q_{12} M_{21} + M_{22}^\top Q_{22} M_{21} + M_{12}^\top K_x^\top R K_x M_{11} - M_{22}^\top K_I^\top R K_x M_{11} - M_{12}^\top K_x^\top R K_I M_{12} + M_{22}^\top K_I^\top R K_I M_{12}]$

$P_1^{(22)} = \sum_{i=0}^{\infty} \lambda^i [M_{12}^\top Q_{11} M_{12} + M_{22}^\top Q_{21} M_{12} + M_{12}^\top Q_{12} M_{22} + M_{22}^\top Q_{22} M_{22} + M_{12}^\top K_x^\top R K_x M_{12} - M_{22}^\top K_I^\top R K_x M_{12} - M_{12}^\top K_x^\top R K_I M_{22} + M_{22}^\top K_I^\top R K_I M_{22}]$. ∎

From (20), the Bellman equation for the optimal value function is thus given as:

$$V^*(\boldsymbol{X}_k) = \boldsymbol{X}_k^\top P_1^* \boldsymbol{X}_k = \boldsymbol{X}_k^\top Q_1 \boldsymbol{X}_k + \tilde{\boldsymbol{u}}_k^\top R \tilde{\boldsymbol{u}}_k + \lambda \boldsymbol{X}_{k+1}^\top P_1^* \boldsymbol{X}_{k+1} \tag{28}$$

and the optimal control input of (21) with $\lambda = 1$ becomes:

$$\begin{aligned} \tilde{\boldsymbol{u}}_k &= \arg\min_{\tilde{u}} \left(\boldsymbol{X}_k^\top Q_1 \boldsymbol{X}_k + \tilde{\boldsymbol{u}}_k^\top R \tilde{\boldsymbol{u}}_k + \boldsymbol{X}_{k+1}^\top P_1 \boldsymbol{X}_{k+1}\right) \\ &= -(R + B_1^\top P_1 B_1)^{-1} B_1^\top P_1 A_1 \boldsymbol{X}_k \\ &= -K_1 \boldsymbol{X}_k, \end{aligned} \tag{29}$$

where $K_1 = \left((R + B_1^\top P_1 B_1)^{-1} B_1^\top P_1 A_1\right) = \begin{bmatrix} K_x & -K_I \end{bmatrix}$

Equation (29) gives the model-based control solution to the augmented DT LQT problem consisting of both the integral feedforward and feedback gains. Substituting for $\tilde{\boldsymbol{u}}_k$ in (28) and simplifying gives the corresponding algebraic Riccati

equation (ARE) as:

$$P_1 = Q_1 + A_1^\top P_1 A_1 - A_1^\top P_1 B_1 (R + B_1^\top P_1 B_1)^{-1} B_1^\top P_1 A_1. \tag{30}$$

Lyapunov stability can be shown for the LQT system by using the Lyapunov function:

$$\begin{aligned}\Delta V(\boldsymbol{X}_k) &= V(\boldsymbol{X}_{k+1}) - V(\boldsymbol{X}_k) = \boldsymbol{X}_k^\top P_1 \boldsymbol{X}_{k+1} - \boldsymbol{X}_k^\top P_1 \boldsymbol{X}_k < 0 \\ &= (A_1 \boldsymbol{X}_k + B_1 \tilde{\boldsymbol{u}}_k)^\top P_1 (A_1 \boldsymbol{X}_k + B_1 \tilde{\boldsymbol{u}}_k) - \boldsymbol{X}_k^\top P_1 \boldsymbol{X}_k < 0 \\ &= \boldsymbol{X}_k^\top A_1^\top P_1 A_1 \boldsymbol{X}_k + \boldsymbol{X}_k^\top A_1^\top P_1 B_1 \tilde{\boldsymbol{u}}_k + \tilde{\boldsymbol{u}}_k^\top B_1^\top P_1 A_1 \boldsymbol{X}_k + \tilde{\boldsymbol{u}}_k^\top B_1^\top P_1 B_1 \tilde{\boldsymbol{u}}_k - \boldsymbol{X}_k^\top P_1 \boldsymbol{X}_k < 0.\end{aligned} \tag{31}$$

Substitute for control input (29) as:

$$\begin{aligned}\Delta V(\boldsymbol{X}_k) &= \boldsymbol{X}_k^\top A_1^\top P_1 A_1 \boldsymbol{X}_k - \boldsymbol{X}_k^\top A_1^\top P_1 B_1 K_1 \boldsymbol{X}_k - \boldsymbol{X}_k^\top K_1^\top B_1^\top P_1 A_1 \boldsymbol{X}_k + \boldsymbol{X}_k^\top K_1^\top B_1^\top P_1 B_1 K_1 \boldsymbol{X}_k - \boldsymbol{X}_k^\top P_1 \boldsymbol{X}_k < 0 \\ &= \boldsymbol{X}_k^\top \left[A_1^\top P_1 A_1 - A_1^\top P_1 B_1 K_1 - K_1^\top B_1^\top P_1 A_1 + K_1^\top B_1^\top P_1 B_1 K_1 - P_1\right] \boldsymbol{X}_k < 0.\end{aligned} \tag{32}$$

Add and subtract $K_1^\top R K_1$, then simplify further to give:

$$\begin{aligned}\Delta V(\boldsymbol{X}_k) &= \boldsymbol{X}_k^\top \left[A_1^\top P_1 A_1 - A_1^\top P_1 B_1 K_1 - K_1^\top B_1^\top P_1 A_1 + K_1^\top B_1^\top P_1 B_1 K_1 - P_1 + K_1^\top R K_1 - K_1^\top R K_1\right] \boldsymbol{X}_k < 0 \\ &= \boldsymbol{X}_k^\top \left[A_1^\top P_1 A_1 - A_1^\top P_1 B_1 K_1 - K_1^\top B_1^\top P_1 A_1 + K_1^\top (R + B_1^\top P_1 B_1) K_1 - K_1^\top R K_1 - P_1\right] \boldsymbol{X}_k < 0.\end{aligned} \tag{33}$$

Finally, substitute for $K_1 = (R + B_1^\top P_1 B_1)^{-1} B_1^\top P_1 A_1$ in (33) to give:

$$\Delta V(\boldsymbol{X}_k) = \boldsymbol{X}_k^\top \left[A_1^\top P_1 A_1 - A_1^\top P_1 B_1 (R + B_1^\top P_1 B_1)^{-1} B_1^\top P_1 A_1 - K_1^\top R K_1 - P_1\right] \boldsymbol{X}_k < 0. \tag{34}$$

However, the ARE for the LQT system is given in terms of $P_1$ in (30); therefore, Lyapunov stability is guaranteed for the following condition:

$$\Delta V(\boldsymbol{X}_k) = \boldsymbol{X}_k \left[-Q_1 - K_1^\top R K_1\right] \boldsymbol{X}_k < 0, \tag{35}$$

if and only if $Q_1$ and $R$ are positive semidefinite.

Figure 1 shows the block diagram of the augmented tracking control framework with integral control consisting of both a feedforward integral gain $K_I$ and a feedback gain $K_x$. The given baseline integral-proportional (I-P) control structure is widely used in practice where the tracking error is fed into the feedforward integral term, while the proportional term is implemented in feedback.[40,41]

Therefore, using knowledge of the system dynamics, the above tracking framework with integral control can be used to achieve optimal tracking control online and does not impose restrictions on the reference model dynamics or use of discounted tracking costs. For systems with unknown or varying dynamics, an approximate online solution to the optimal tracking control framework with integral control is developed in the next section using reinforcement learning. This offers the advantage of not requiring the full knowledge of the system dynamics while converging to the optimum values.

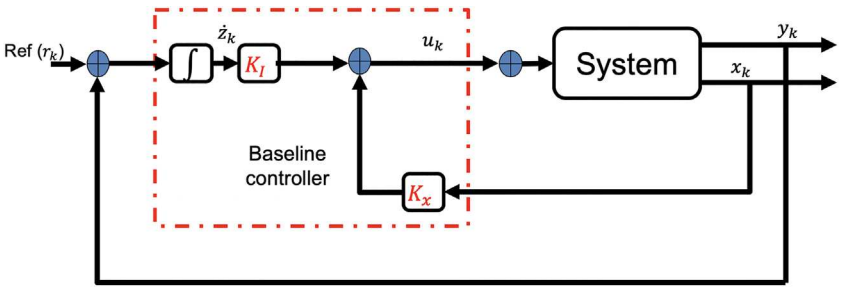


**FIGURE 1** Block diagram of an augmented tracking control framework with integral control consisting of both a feedforward integral gain $K_I$ and a feedback gain $K_x$ [Colour figure can be viewed at wileyonlinelibrary.com]

## 5 | REINFORCEMENT LEARNING FRAMEWORK FOR THE OPTIMAL TRACKING CONTROL USING AUGMENTED FORMULATION WITH INTEGRAL CONTROL

As discussed in Section 2, existing approaches for the optimal tracking control problem using RL either assume that the feedforward part of the control is known a priori or make restrictive assumptions on the reference model dynamics and use of discounted tracking costs. These restrictive assumptions are eliminated by using the augmented formulation with integral control as proposed in Section 3. Consequently, a novel optimal RL framework is proposed for the LQT problem that converges to the optimum solution for systems with varying or unknown system dynamics using the augmented formulation with integral control. Furthermore, unlike the previously proposed RL tracking approaches,[30-32,34-36,38,39] the proposed formulation is able to guarantee zero steady-state tracking error and provides adaptation for both the feedforward and feedback controller gains. The framework continually adapts the controller gains to optimum values and provides a through-life adaptation strategy.

Model-free RL approaches are enabled by iterative techniques that utilize the Bellman optimality equations to develop *forward-in-time* update equations, which are solved at each time step.[2,42] One of such iterative technique is the policy iteration (PI) method, which requires an initially admissible policy[8] (ie, stabilizing policy with a finite cost $V(\cdot)$) and successively alternates between the following update equations for (20) and (21) as follows:

$$V_{k+1}(\boldsymbol{X}_k) = \boldsymbol{X}_k^\top Q_1 \boldsymbol{X}_k + \tilde{\boldsymbol{u}}_k^\top R \tilde{\boldsymbol{u}}_k + V_{k+1}(\boldsymbol{X}_{k+1}) \tag{36}$$

$$\tilde{\boldsymbol{u}}_{k+1} = \arg\min_{\tilde{u}} \left( \boldsymbol{X}_k^\top Q_1 \boldsymbol{X}_k + \tilde{\boldsymbol{u}}_k^\top R \tilde{\boldsymbol{u}}_k + V_{k+1}(\boldsymbol{X}_{k+1}) \right). \tag{37}$$

Given an admissible policy $\pi(\boldsymbol{X})$, the value is evaluated by solving (36) till convergence while an improved policy is computed using (37). Both update equations, respectively, constitute the policy evaluation and policy update steps of the PI method. The PI method is justified in Reference [43] by showing that the improved policy ensures that $V_{k+1}(\boldsymbol{X}_k) \leq V_k(\boldsymbol{X}_k)$ and is associated with the monotonicity property of the update equations. This way the PI recursion computes a strictly improved policy, and convergence to the optimal policy and value under Assumption 1 has been shown in Reference [44].

Model-free approaches for the LQT problem are therefore enabled by approximating the value function of (20) as follows:

$$V^\pi(\boldsymbol{X}_k) \approx \theta_c^\top \Phi(\boldsymbol{X}_k) = \sum_{i=k}^{\infty} \lambda^{i-k} L_i, \tag{38}$$

where $\Phi(\boldsymbol{X})$ is a set of basis function and $\theta_c$ are the function weights. Equation (38) gives the value function approximation (VFA) and is defined as the sum of the discounted reward signal $L_k$ starting from state $\boldsymbol{X}_k$ under some fixed policy $\pi(\boldsymbol{X})$. Similarly, an approximation to the state-action value function with basis function $\Psi(\boldsymbol{X}, \tilde{\boldsymbol{u}})$ and weights $\beta$ is approximated as:

$$Q^\pi(\boldsymbol{X}_k, \tilde{\boldsymbol{u}}_k) \approx \beta^\top \Psi(\boldsymbol{X}_k, \tilde{\boldsymbol{u}}_k) = \sum_{i=k}^{\infty} \lambda^{i-k} L_i. \tag{39}$$

Equation (39) is the Q-function approximation (QFA), which is defined as the sum of the discounted reward signal $L_k$ starting from state $\boldsymbol{X}_k$ and taking action $\tilde{\boldsymbol{u}}_k$, then following policy $\pi(\boldsymbol{X})$ thereon. Depending on the function that is being approximated, two RL strategies are therefore proposed for the LQT problem.

### 5.1 | VFA-based RL algorithm

For the VFA approximation, the Bellman equation for the value function (38) becomes:

$$\theta_c^\top \Phi(\boldsymbol{X}_k) = \boldsymbol{X}_k^\top Q_1 \boldsymbol{X}_k + \tilde{\boldsymbol{u}}_k^\top R \tilde{\boldsymbol{u}}_k + \lambda \theta_c^\top \Phi(\boldsymbol{X}_{k+1}). \tag{40}$$

A second function approximation is used to adapt the controller gains and given as:

$$\tilde{\boldsymbol{u}}_k = \theta_a^\top \boldsymbol{X}_k = -\hat{K}_1 \boldsymbol{X}_k. \tag{41}$$

The RL adaptation utilizes the PI recursion (36), (37) consisting of both value and policy update steps. For the value update step, the policy is kept fixed while the value function parameters are updated using the system measurements at $N$ episodic intervals (ie, from some initial state $\boldsymbol{X}_0$ to a terminal state $\boldsymbol{X}_N$). After each episode, the controller parameters are adapted from (21) using a gradient descent tuning as:

$$\begin{aligned}
\theta_a^{i+1} &= \theta_a^i - l_a \frac{\partial \theta_c^\top \Phi(\boldsymbol{X}_i)}{\partial \theta_a} \\
&= \theta_a^i - l_a \frac{\partial \theta_c^\top \Phi(\boldsymbol{X}_i)}{\partial \tilde{\boldsymbol{u}}_i} \times \frac{\partial \tilde{\boldsymbol{u}}_i}{\partial \theta_a} \\
\therefore \theta_a^{i+1} &= \theta_a^i - l_a \boldsymbol{X}_i \left( 2R\tilde{\boldsymbol{u}}_i + \lambda B_1^\top \frac{\partial \theta_c^\top \Phi(\boldsymbol{X}_{i+1})}{\partial \boldsymbol{X}_{i+1}} \right),
\end{aligned} \tag{42}$$

where $l_a > 0 \in \mathbb{R}$ is a tuning step size. This is repeated till convergence of both the value function parameters and the controller gains. This way, the VFA based RL method solves the online LQT problem of Section 2 using the proposed augmented formulation with integral control and without requiring knowledge of the system dynamics. Algorithm 1 describes the VFA-based adaptation of the controller parameters using a policy iteration (PI) recursion.

**Algorithm 1.** VFA-based RL algorithm for the LQT problem

Initialize $V(\boldsymbol{X}) \approx \theta_{c,k}^\top \Phi(\boldsymbol{X})$ at $k = 0$ for some stabilizing policy $\pi(\boldsymbol{X}) = \theta_{a,k}^\top \boldsymbol{X}$, and do till convergence:

**Value function update step**

1: **for** $j = 0 : N$ **do**

2: At $\boldsymbol{X}_j$, compute the control input $\tilde{\boldsymbol{u}}_j$ with exploration signal $\epsilon$ as $\tilde{\boldsymbol{u}}_j = \pi(\boldsymbol{X}_j) + \epsilon$.

3: Compute the least squares solution for $\theta_{c,j+1}$ using measurements $L_j = \boldsymbol{X}_j^\top Q_1 \boldsymbol{X}_j + \tilde{\boldsymbol{u}}_j^\top R \tilde{\boldsymbol{u}}_j$, $\boldsymbol{X}_j$ and $\boldsymbol{X}_{j+1}$ as:

$$\theta_{c,j+1}^\top \left( \Phi(\boldsymbol{X}_j) - \lambda \Phi(\boldsymbol{X}_{j+1}) \right) = \boldsymbol{X}_j^\top Q_1 \boldsymbol{X}_j + \tilde{\boldsymbol{u}}_j^\top R \tilde{\boldsymbol{u}}_j$$

4: $j = j + 1$.

5: **end forPolicy update step**

**Require:** Set $\theta_{c,k+1} = \theta_{c,j+1} \mid_{j=N}$

6: Update the policy parameters using the gradient descent tuning as:

$$\theta_{a,k}^{i+1} = \theta_{a,k}^i - l_a \boldsymbol{X}_i \left( 2R\theta_{a,k}^{i\top} \boldsymbol{X}_i + \lambda B_1^\top \frac{\partial \theta_{c,k+1}^\top \Phi(\boldsymbol{X}_{i+1})}{\partial \boldsymbol{X}_{i+1}} \right)$$

7: At the end of the gradient tuning, set $\theta_{a,k+1} = \theta_{a,k}^{i+1}$ and update the policy as:

$$\pi(\boldsymbol{X}) = \theta_{a,k+1}^\top \boldsymbol{X} = -K_1 \boldsymbol{X}$$

8: Increment time step $k = k + 1$.

*Remarks on implementation of Algorithm 1*

- The gradient tuning update steps $i$ can be chosen as the number of episodic steps $j$ for the value function update.
- The VFA-based RL algorithm is not completely model free as knowledge of the input matrix $B_1$ is needed in computing the controller parameters update. Consequently, this approach is limited to systems with variations occurring only in the drift or dynamics matrix $A_1$ as this is assumed unknown.

- For convergence of the parameter estimates, a persistence of excitation (PE) condition on the regressor matrix given by References [45] and [46] is required. An exploration signal is therefore added to the algorithm to ensure that the regressor matrix satisfies:

$$aI \leq \sum_{i=k}^{k+M} \Gamma_i \Gamma_i^\top \leq bI \quad \forall i, \tag{43}$$

where $\Gamma_i = [\Phi(\boldsymbol{X}_i) - \lambda\Phi(\boldsymbol{X}_{i+1})]$ and with $M > 0$, $a > 0$, $b > 0$.

## 5.2 | Q-function-based RL algorithm

Similar to the VFA approximation method, the Bellman equation for the Q-function (39) becomes:

$$\beta^\top \Psi(\boldsymbol{X}_k, \tilde{\boldsymbol{u}}_k) = \boldsymbol{X}_k^\top Q_1 \boldsymbol{X}_k + \tilde{\boldsymbol{u}}_k^\top R \tilde{\boldsymbol{u}}_k + \lambda \beta^\top \Psi(\boldsymbol{X}_{k+1}, \tilde{\boldsymbol{u}}_{k+1}). \tag{44}$$

The RL adaptation equally utilizes the PI recursion (36), (37) and consists of both Q-function and policy update steps. In contrast to the VFA algorithm, the Q-function explicitly approximates the control inputs for each state from which the optimal control input can be obtained via a greedy optimization. This makes the QFA algorithm completely model free by using only the measurements observed along the system trajectories for the controller updates and is further described in Algorithm 2.

**Algorithm 2.** QFA-based RL algorithm for the LQT problem

Initialize $Q(\boldsymbol{X}, \tilde{\boldsymbol{u}}) \approx \beta_k^\top \Psi(\boldsymbol{X}, \tilde{\boldsymbol{u}})$ at $k = 0$ for some stabilizing policy $\pi(\boldsymbol{X}) = \arg\min_{\tilde{u}} \left(\beta_k^\top \Psi(\boldsymbol{X}, \tilde{\boldsymbol{u}})\right)$, and do till convergence:

**Q-function update step**

1: **for** $j = 0 : N$ **do**

2: At $\boldsymbol{X}_j$, compute the control input $\tilde{\boldsymbol{u}}_j$ with exploration signal $\epsilon$ as $\tilde{\boldsymbol{u}}_j = \pi(\boldsymbol{X}_j) + \epsilon$.

3: Compute the least squares solution for $\beta_{j+1}$ using measurements $L_j = \boldsymbol{X}_j^\top Q_1 \boldsymbol{X}_j + \tilde{\boldsymbol{u}}_j^\top R \tilde{\boldsymbol{u}}_j$, $\boldsymbol{X}_j$ and $\boldsymbol{X}_{j+1}$ as:

$$\beta_{j+1}^\top \left(\Psi(\boldsymbol{X}_j, \tilde{\boldsymbol{u}}_j) - \lambda \Psi(\boldsymbol{X}_{j+1}, \tilde{\boldsymbol{u}}_{j+1})\right) = \boldsymbol{X}_j^\top Q_1 \boldsymbol{X}_j + \tilde{\boldsymbol{u}}_j^\top R \tilde{\boldsymbol{u}}_j$$

where $\tilde{\boldsymbol{u}}_{j+1} = \pi(\boldsymbol{X}_{j+1})$

4: $j = j + 1$.

5: **end for Policy update step**

**Require:** Set $\beta_{k+1} = \beta_{j+1}\,|_{j=N}$

6: Update the policy parameters using a greedy optimization as:

$$\pi(\boldsymbol{X}) = \arg\min_{\tilde{u}} \left(\beta_{k+1}^\top \Psi(\boldsymbol{X}, \tilde{\boldsymbol{u}})\right) = -K_1 \boldsymbol{X}$$

7: Increment time step $k = k + 1$.

The Q-function parameters are updated in each episode while keeping the policy fixed and constitutes the Q-function update step. For the policy update, a greedy optimization is performed after each episode using the adapted Q-function parameters as:

$$\tilde{\boldsymbol{u}}_k = \arg\min_{\tilde{u}} \left(\beta^\top \Psi(\boldsymbol{X}_k, \tilde{\boldsymbol{u}}_k)\right) = -\hat{K}_1 \boldsymbol{X}_k. \tag{45}$$

Like Algorithm 1, an exploration signal is added to ensure PE and to satisfy (43), where $\Gamma_i = [\Psi(\boldsymbol{X}_i, \tilde{\boldsymbol{u}}_i) - \lambda\Psi(\boldsymbol{X}_{i+1}, \tilde{\boldsymbol{u}}_{i+1})]$.

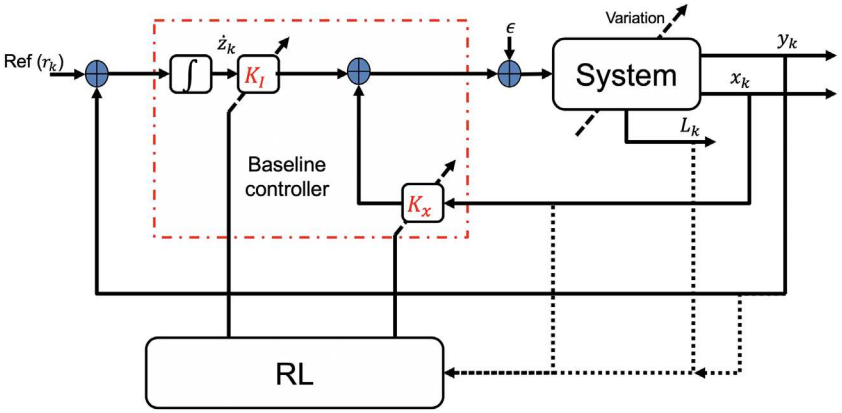


**FIGURE 2** Schematic of the proposed RL framework for the optimal tracking control using augmented formulation with integral control. The RL block represents either the VFA or QFA algorithm that continually uses the observed system measurements to adapt the tracking controller gains to optimum values subject to varying or unknown system dynamics [Colour figure can be viewed at wileyonlinelibrary.com]

The RL control strategies described above solve the online LQT problem without knowledge of the system dynamics or variations. Furthermore, by using the proposed augmented formulation with integral control, the RL frameworks do not require any predetermined feedforward tracking control input or restrictive assumptions on the reference generator dynamics and use of discounted tracking costs. The RL tracking control scheme is represented schematically in Figure 2, where the RL block represents either the VFA or QFA algorithm that continually uses the observed system measurements to adapt the tracking controller gains to optimum values subject to varying or unknown system dynamics.

## 6 | SIMULATION CASE STUDIES

The LQT RL approach is demonstrated on two simulation case studies. The first is a system with an initially unstable and unknown dynamics that shows convergence of the proposed RL tracking methods to the optimal tracking controller gains. The second case study addresses the optimal tracking control problem in a buck power converter system, which is subject to uncertain or varying component tolerances under different operating conditions.

### 6.1 | Case study 1

Consider a 2-state system with unstable dynamics given as:

$$\begin{aligned}\dot{\boldsymbol{x}}(t) &= \begin{bmatrix}-1 & 2\\ 2.2 & 1.7\end{bmatrix}\boldsymbol{x}(t) + \begin{bmatrix}2\\ 1.5\end{bmatrix}\boldsymbol{u}(t)\\ y(t) &= \begin{bmatrix}1 & 1\end{bmatrix}\boldsymbol{x}(t).\end{aligned} \tag{46}$$

Using a sampling time $t_s = 0.03$ seconds, an equivalent discrete-time system using Euler's discretization is formed as:

$$\begin{aligned}\boldsymbol{x}_{k+1} &= \underbrace{\begin{bmatrix}0.9724 & 0.0607\\ 0.0668 & 1.0544\end{bmatrix}}_{A_{(1)}}\boldsymbol{x}_k + \underbrace{\begin{bmatrix}0.0605\\ 0.0482\end{bmatrix}}_{B}\boldsymbol{u}_k\\ \boldsymbol{y}_k &= \underbrace{\begin{bmatrix}1 & 1\end{bmatrix}}_{C}\boldsymbol{x}_k.\end{aligned} \tag{47}$$

The tracking control problem is to track a time-varying step reference input from any finite initial condition $\boldsymbol{x}_0$ representative of step commands in a servo system or precision-tracking applications. Tracking cost parameters in (18) are considered as $Q_1 = 2 * \mathcal{I}(3)$, $R = 0.05$, and $\lambda = 1$.

(1) Existing online solution approach with the use of discounted cost: Existing online solution to the optimal tracking control problem as discussed in Section 2 requires knowledge of the reference dynamics and the use of discounted

tracking cost. For the given tracking problem, consider the reference dynamics of (9) to be given by the linear difference equation:

$$\boldsymbol{r}_{k+1} = F\boldsymbol{r}_k, \tag{48}$$

where $F = 1$. An augmented system with the reference dynamics can then be formulated according to (10). Furthermore, as a result of using a reference dynamics that does not tend to zero, a discounted cost must be used. Comparison of the performance of this approach using different discount factors to the proposed augmented formulation with integral control is shown in Figure 3. As observed in the simulation result, a discount factor of $\lambda = 0.8$ had a slower response but a reduced steady-state error, while a discount factor of $\lambda = 0.7$ had a faster response but larger steady-state error. Existing online tracking approaches with the use of a discount factor are therefore not only restrictive to the type of reference dynamics that can be used but also cannot guarantee zero steady-state tracking error. In the following, the proposed online solution approaches that do not require knowledge of the reference dynamics or the use of discounted cost will now be presented.

(2) Model-based solutions using the proposed augmented formulation with integral control: Baseline solution for the augmented formulation with integral control using the system models is first presented. An augmented system with integral control is formed according to (22) as:

$$\boldsymbol{X}_{k+1} = \underbrace{\begin{bmatrix} 0.9724 & 0.0607 & 0 \\ 0.0668 & 1.0544 & 0 \\ -0.03 & -0.03 & 1 \end{bmatrix}}_{A_{1(1)}} \boldsymbol{X}_k + \underbrace{\begin{bmatrix} 0.0605 \\ 0.0482 \\ 0 \end{bmatrix}}_{B_1} \tilde{\boldsymbol{u}}_k. \tag{49}$$

Using the given system models $(A_{1(1)}, B_1)$, the optimal solution to the corresponding ARE (30) is given as:

$$P^*_{1(1)} = \begin{bmatrix} 10.1584 & -6.9476 & -8.8170 \\ -6.9476 & 18.5835 & 4.5047 \\ -8.8170 & 4.5047 & 68.4224 \end{bmatrix} \tag{50}$$

with the optimal tracking controller gains as:

$$K^*_{1(1)} = \left((R + B_1^\top P_1 B_1)^{-1} B_1^\top P_1 A_{1(1)}\right) = \begin{bmatrix} K_x & -K_I \end{bmatrix} = \begin{bmatrix} 3.6277 & 5.7644 & -4.6873 \end{bmatrix}. \tag{51}$$

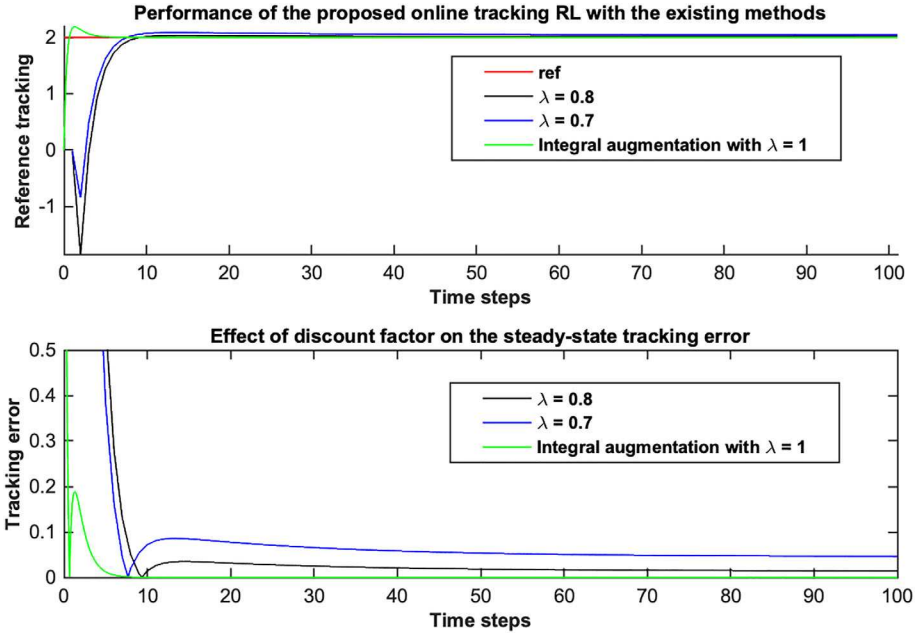


**FIGURE 3** Comparison of the existing online tracking methods with the use of discount factors with the proposed integral augmentation approach [Colour figure can be viewed at wileyonlinelibrary.com]

However, in practice, the system dynamics may be unknown or time varying therefore motivating the use of online RL methods.

(3) Model-free RL solutions: The proposed model-free RL approaches can be used to obtain the optimal tracking controller gains online subject to the unknown or varying system dynamics.

### 6.1.1 | VFA-based RL adaptation

From Lemma 1, the value function for the augmented formulation with integral control is quadratic, thus the value function approximation for the given 2-state system in Algorithm 1 is chosen to be the quadratic function:

$$V(\boldsymbol{X}) \approx \theta_c^\top \Phi(\boldsymbol{X}) = \theta_c^\top \begin{bmatrix} x_1^2 \\ x_1 x_2 \\ x_1 z \\ x_2^2 \\ x_2 z \\ z^2 \end{bmatrix}. \tag{52}$$

From Algorithm 1, an initially suboptimal but stabilizing policy is arbitrarily selected as:

$$\pi(\boldsymbol{X}) = \underbrace{\begin{bmatrix} 0.4112 & -2.0412 & 2.5011 \end{bmatrix}}_{-\hat{K}_{1(0)}}. \tag{53}$$

The rest of Algorithm 1 is then run online till convergence of the tracking controller parameters using only the observed system measurements. The VFA parameters converged to the following optimal values, $\theta^*_{c(1)} = [9.9155; -14.9830; -16.7696; 18.0048; 10.1510; 68.8826]$ with:

$$\begin{bmatrix} P_1^{11} & P_1^{12} & P_1^{13} \\ P_1^{21} & P_1^{22} & P_1^{23} \\ P_1^{31} & P_1^{32} & P_1^{33} \end{bmatrix} = \begin{bmatrix} \theta_c^{(1)} & 0.5\theta_c^{(2)} & 0.5\theta_c^{(3)} \\ 0.5\theta_c^{(2)} & \theta_c^{(4)} & 0.5\theta_c^{(5)} \\ 0.5\theta_c^{(3)} & 0.5\theta_c^{(5)} & \theta_c^{(6)} \end{bmatrix} \tag{54}$$

and $\theta^*_{a(1)} = [-3.4202; -5.5650; 4.6468] = -\hat{K}^*_{1(1)}$.

To demonstrate the adaptation of the tracking controller gains to optimal values using the proposed RL tracking control framework, the system drift matrix $A$ is changed instantaneously during simulation to:

$$A_{(2)} = \begin{bmatrix} 0.8706 & 0.1672 \\ -0.0395 & 1.1654 \end{bmatrix} \tag{55}$$

with a new baseline model-based solution from using the system $A_{(2)}$ matrix given as:

$$\begin{aligned} P^*_{1(2)} &= \begin{bmatrix} 23.3462 & -27.4260 & -34.6172 \\ -27.4260 & 49.7383 & 38.9057 \\ -34.6172 & 38.9057 & 127.0261 \end{bmatrix} \\ K^*_{1(2)} &= \begin{bmatrix} 1.2757 & 8.8839 & -4.6907 \end{bmatrix}. \end{aligned} \tag{56}$$

Following this system variation, the tracking controller gains are no longer optimal resulting in a decline in the system performance. This can be detected in practice by using a threshold on standard step response parameters like percentage overshoot (P.O.), rise time, and so on and used as an enable signal to reinitiate the RL learning process. The VFA parameters after the system variation converged to $\theta^*_{c(2)} = [23.3958; -56.2854; -69.6663; 49.9611; 78.9013; 127.1470]$ and $\theta^*_{a(2)} = [-0.9668; -8.7688; -4.6794] = -\hat{K}^*_{1(2)}$. Figure 4 shows the parameter convergence using the VFA-based RL adaptation to the optimal but assumed unknown values before and after the system variation.

Figure 5 shows the overall system response to time-varying step reference inputs at the various stages of the RL adaptation. The region with $\theta_{a,c(0)}$ in the figure corresponds to the system response using the initial suboptimal controller gains, while the region with $\theta_{a,c(1)}$ shows the system response after convergence to the optimal controller values from the RL

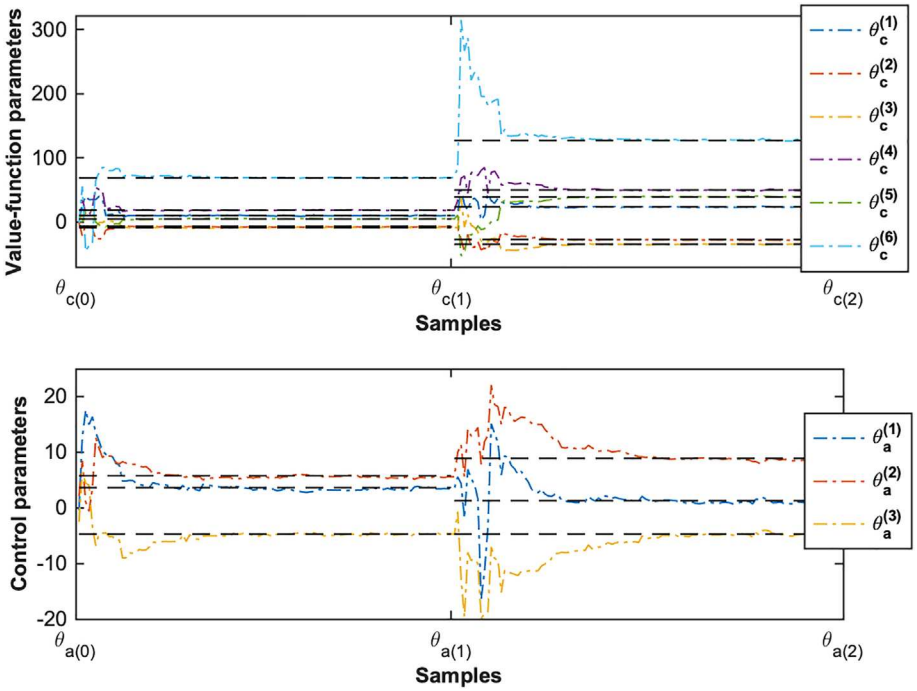


**FIGURE 4** Online adaptation and convergence of both the value function and controller parameters to the optimal values (in black dashed lines) using Algorithm 1. $\theta_{a,c(0)}$ are the initial suboptimal controller parameters, while $\theta_{a,c(1)}$ and $\theta_{a,c(2)}$ are, respectively, the identified optimal controller parameters before and after the system variation [Colour figure can be viewed at wileyonlinelibrary.com]

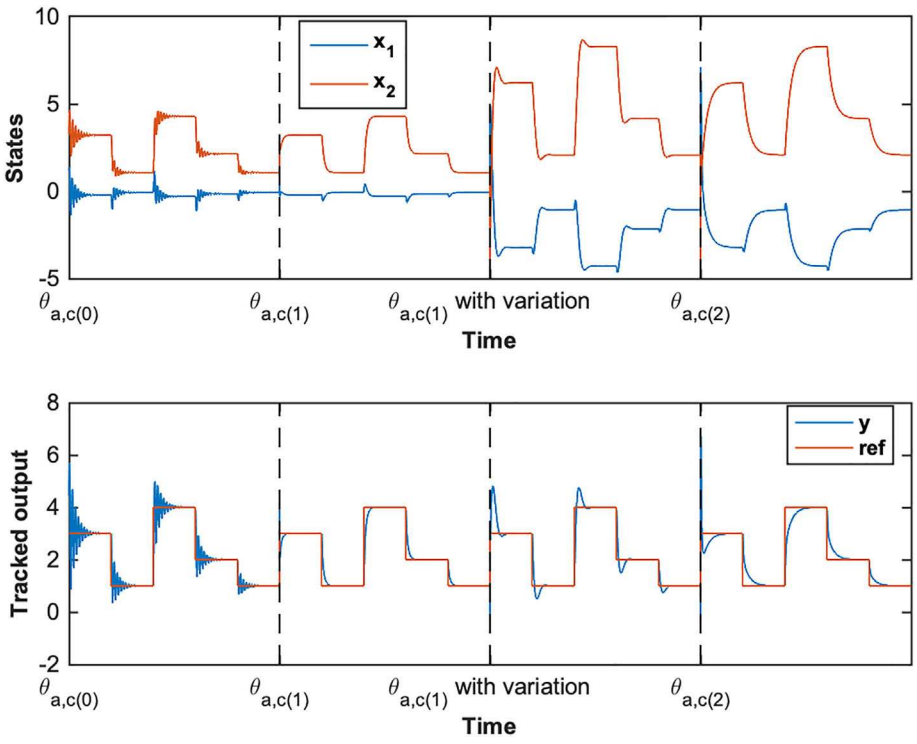


**FIGURE 5** System response showing the system states and tracked output at the various stages of the RL adaptations. Region with $\theta_{a,c(0)}$ shows the response using the initial suboptimal controller gains, while region with $\theta_{a,c(1)}$ shows the response from the adapted controller gains to the optimal values using the proposed Algorithms. Region with $\theta_{a,c(1)}$*with variation* shows the decline in the system performance following variations in the system dynamics while keeping the controller values fixed, while region with $\theta_{a,c(2)}$ onward shows the response after adaptation to the new optimal control gains [Colour figure can be viewed at wileyonlinelibrary.com]

adaptation. After the system variation and keeping the controller values fixed, the region with $\theta_{a,c(1)}$*with variation* shows the decline in system performance following which the RL adaptation is reenabled. The new system performance after convergence to the new optimal control gains is then shown in the region with $\theta_{a,c(2)}$.

### 6.1.2 | QFA-based RL adaptation

The QFA provides a completely model-free approach to the LQT problem and similar to the VFA, the Q-functions from Algorithm 2 are approximated for the 2-state system using a quadratic basis set as:

$$Q(\boldsymbol{X},\tilde{\boldsymbol{u}}) \approx \beta^{\top}\Psi(\boldsymbol{X},\tilde{\boldsymbol{u}}) = \beta^{\top}\begin{bmatrix} x_1^2 \\ x_1x_2 \\ x_1z \\ x_1\tilde{u} \\ x_2^2 \\ x_2z \\ x_2\tilde{u} \\ z^2 \\ z\tilde{u} \\ \tilde{u}^2 \end{bmatrix}. \tag{57}$$

Using Algorithm 2, the Q-function parameters converged to $\beta^*_{(1)} = [11.3564; -10.0880; -20.7299; 0.6605; 21.6084; 4.0903; 1.0495; 70.4224; -0.8534; 0.0910]$ with:

$$Q^* = \begin{bmatrix} Q_1 + \lambda A_1^{\top}P_1^*A_1 & \lambda A_1^{\top}P_1^*B_1 \\ \lambda B_1^{\top}P_1^*A_1 & R + \lambda B_1^{\top}P_1^*B_1 \end{bmatrix} = \begin{bmatrix} \beta^{(1)} & 0.5\beta^{(2)} & 0.5\beta^{(3)} & 0.5\beta^{(4)} \\ 0.5\beta^{(2)} & \beta^{(5)} & 0.5\beta^{(6)} & 0.5\beta^{(7)} \\ 0.5\beta^{(3)} & 0.5\beta^{(6)} & \beta^{(8)} & 0.5\beta^{(9)} \\ 0.5\beta^{(4)} & 0.5\beta^{(7)} & 0.5\beta^{(9)} & \beta^{(10)} \end{bmatrix}. \tag{58}$$

Corresponding controller gains are then derived according to (45) as:

$$\begin{aligned} \pi(\boldsymbol{X}) &= \underset{\tilde{u}}{\operatorname{argmin}} \left(\beta^{\top}\Psi(\boldsymbol{X},\tilde{\boldsymbol{u}})\right) \\ &= -0.5{*}\beta^{(10)^{-1}}\left(\beta^{(4)}x_1 + \beta^{(7)}x_2 + \beta^{(9)}z\right) = \theta_a^{\top}\boldsymbol{X}. \end{aligned} \tag{59}$$

Therefore, the optimal controller gains with $\beta_1^*$ are computed as $\theta^*_{a(1)} = [-3.6277; -5.7644; 4.6873] = -\hat{K}^*_{1(1)}$.

After variation of the system drift matrix to $A_{(2)}$ during simulation, the parameters reconverged to new optimal values as $\beta^*_{(2)} = [23.4941; -52.7916; -70.3222; 0.2319; 56.9123; 70.2357; 1.6150; 129.0261; -0.8527; 0.0909]$ and $\theta^*_{a(2)} = [-1.2757; -8.8839; 4.6907] = -\hat{K}^*_{1(2)}$. Figure 6 shows the online adaptation and convergence of the Q-function parameters before and after the system variation, respectively. After convergence to the optimal values, the system response using the QFA-based RL adaptations is as shown in Figure 5. The QFA RL approach therefore provides a completely model-free online tracking control solutions.

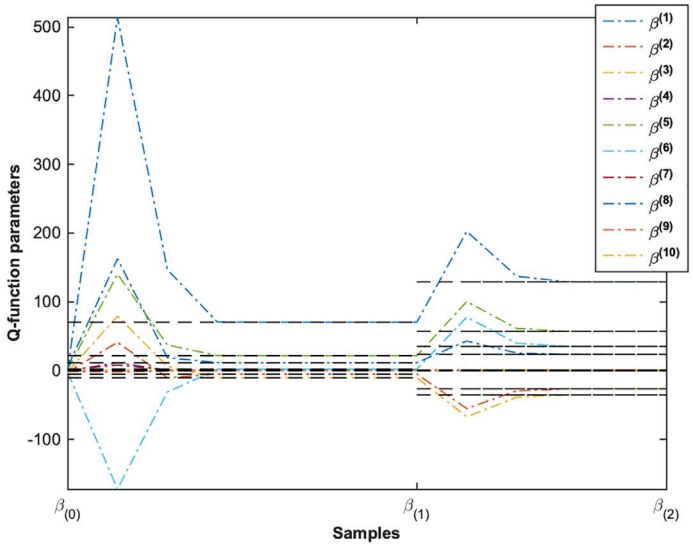


**FIGURE 6** Online adaptation and convergence of the Q-function parameters to the optimal values (in black dashed lines) using Algorithm 2. $\beta_{(0)}$ are the initial suboptimal controller parameters, while $\beta_{(1)}$ and $\beta_{(2)}$ are, respectively, the identified optimal controller parameters before and after the system variation [Colour figure can be viewed at wileyonlinelibrary.com]

## 6.2 | Case study 2

This case study addresses the optimal tracking control problem in a buck power converter system, which is subject to uncertain or varying component tolerances under different operating conditions. Consider a buck power converter with a switching element and consisting of an inductor $L_p$ with a small series resistance $r$, a capacitor $C_p$, and a diode. The voltage drop in the diode can be neglected as the value is typically small.[47] For a continuous conduction mode operation (CCM), the control input is defined as the duty-ratio $u \in [0, 1]$, and the buck converter dynamics are given as:[47]

$$\begin{aligned} L_p \frac{di(t)}{dt} &= -ri(t) - v(t) + Eu(t) \\ C_p \frac{dv(t)}{dt} &= i(t) - i_{\mathrm{L}}, \end{aligned} \tag{60}$$

where $E$ is the dc input voltage, $i$ is the inductor current, $v$ is the output voltage, $i_{\mathrm{L}} = \frac{v}{R_{\mathrm{L}}}$ is the load current, and $R_{\mathrm{L}}$ is the load resistor.

The aim of the controller is to regulate the output voltage to a given $v_{\mathrm{ref}}$. With the states chosen as the inductor current $i$ and output voltage $v$, a corresponding state-space dynamics is formulated as:

$$\begin{aligned} \dot{\boldsymbol{x}}(t) = \begin{bmatrix} \dot{i}(t) \\ \dot{v}(t) \end{bmatrix} &= \begin{bmatrix} \frac{-r}{L_p} & \frac{1}{L_p} \\ \frac{1}{C_p} & \frac{-1}{C_p R_{\mathrm{L}}} \end{bmatrix} \begin{bmatrix} i(t) \\ v(t) \end{bmatrix} + \begin{bmatrix} \frac{E}{L_p} \\ 0 \end{bmatrix} u(t) \\ &= A\boldsymbol{x}(t) + Bu(t), \end{aligned} \tag{61}$$

$$\begin{aligned} y(t) &= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} i(t) \\ v(t) \end{bmatrix} \\ y(t) &= C\boldsymbol{x}(t), \end{aligned} \tag{62}$$

The system component parameters are given as $r = 0.5\,\Omega$, $L_p = 1$ mH, $C_p = 50\,\mu$F, and $E = 48$ V. Variations can occur due to modeling uncertainties and component tolerances under different operating conditions. For this example, the load resistor is changed instantaneously during simulation from $R_{\mathrm{L}} = 200\,\Omega$ to $100\,\Omega$ and is assumed unknown. To demonstrate the proposed online tracking RL approach, an augmented system as given in (22) is formed with sampling time $t_s = 100\,\mu$s while the tracking cost parameters (18) are considered as $Q_1 = \mathcal{I}(3)$, $R = 0.5$, and $\lambda = 1$.

Using Algorithm 2, an initially suboptimal I-P tracking controller is selected as $K_{1(0)} = [0.3086\ 0.1856\ -0.0810]$, while the corresponding Q-functions are approximated as in (57). Algorithm 2 is thereafter run till convergence as this does not require any knowledge of the system dynamics. With the initially unknown $R_{\mathrm{L}} = 200\,\Omega$, the Q-function parameters converged to $\beta^*_{(1)} = [8.9375; 5.3019; -6.7680; 50.0148; 2.1108; -3.5540; 14.5264; 10003.7412; -18.3036; 84.5461]$, while the adapted optimal control gains are computed from (59) as $\theta_{a(1)} = [-0.2958; -0.0859; 0.1082] = -\hat{K}^*_{1(1)}$. Figure 7 shows the

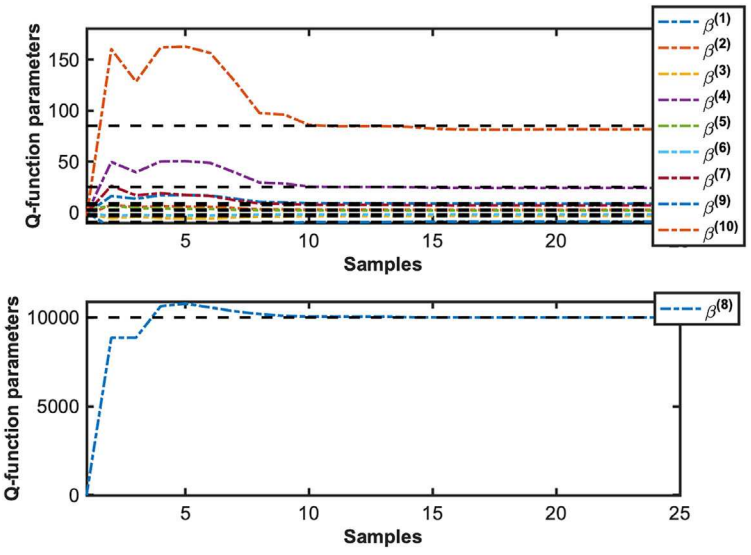


**FIGURE 7** Online adaptation and convergence of the Q-function parameters of the buck power converter to the optimal values (in black dashed lines) using Algorithm 2 [Colour figure can be viewed at wileyonlinelibrary.com]

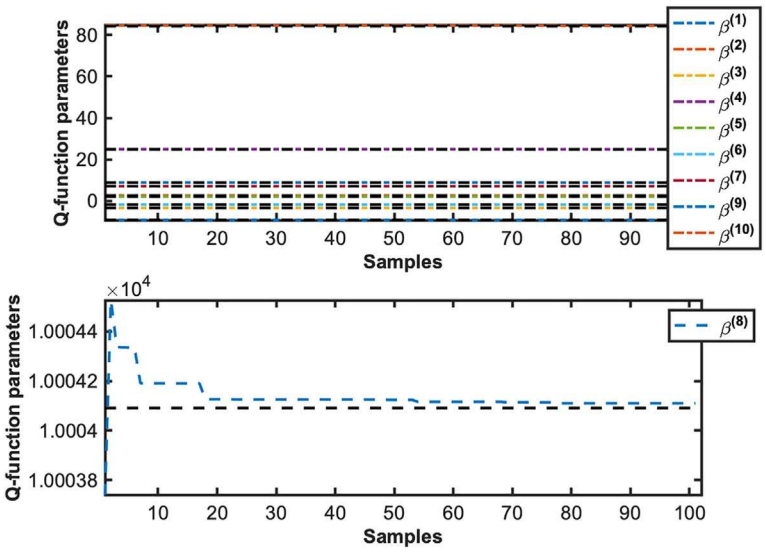


**FIGURE 8** Online adaptation and convergence of the Q-function parameters of the buck power converter to the optimal values (in black dashed lines) after variation in the load resistor $R_L$ using Algorithm 2 [Colour figure can be viewed at wileyonlinelibrary.com]

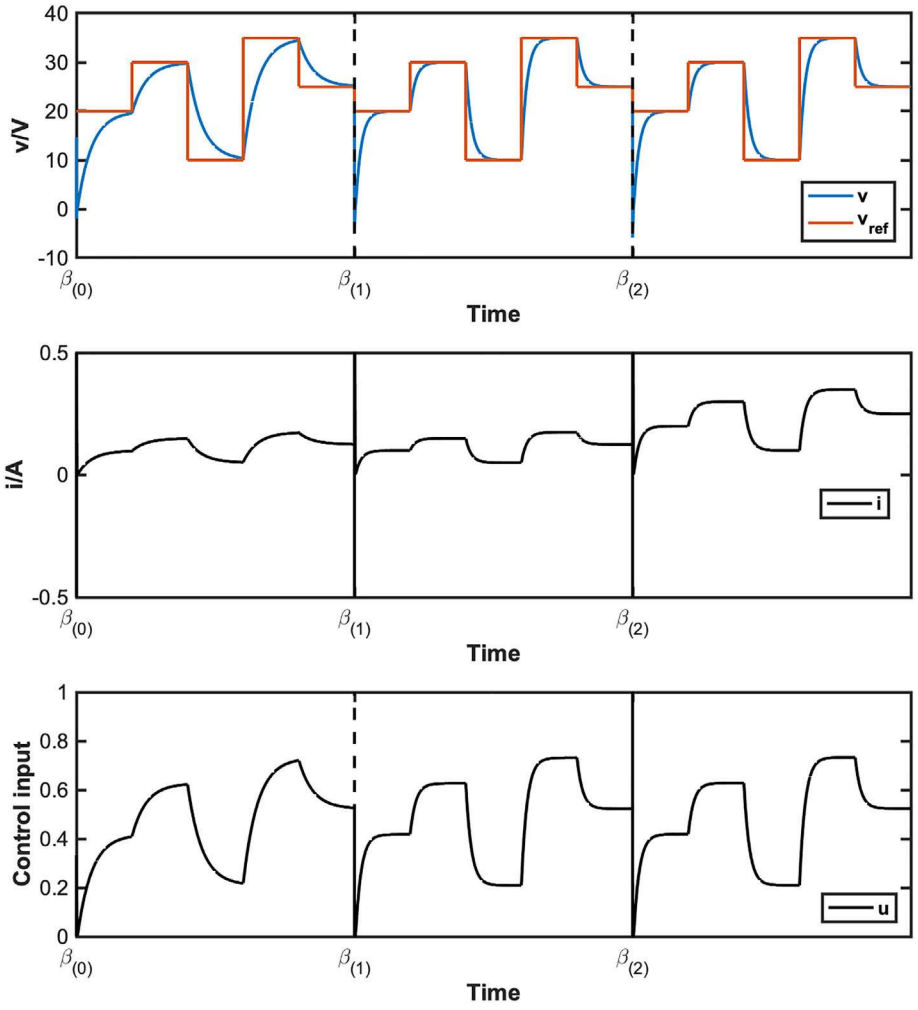


**FIGURE 9** Buck power converter response showing the system states and control input at the various stages of the RL adaptations. Region with $\beta_{(0)}$ shows the response using the initial suboptimal controller gains, while region with $\beta_{(1)}$ shows the response from the adapted controller gains to the optimal values using the proposed Algorithms. Following variations in the load resistor $R_L$, region with $\beta_{(2)}$ onward shows the response after adaptation to the new optimal control gains [Colour figure can be viewed at wileyonlinelibrary.com]

convergence of the online adaptation of the Q-function parameters compared with the optimal but assumed unknown values.

With a variation in the load resistor to $R_L = 100\,\Omega$, the Q-function parameters reconverged to $\beta^*_{(2)} = [8.8961; 5.2366; -6.7850; 49.8444; 2.0918; -3.5425; 14.3679; 10004.1098; -18.3944; 84.3797]$ as shown in Figure 8 and to optimal control gains $\theta_{a(2)} = [-0.2954; -0.0851; 0.1090] = -\hat{K}^*_{1(2)}$.

Figure 9 shows the overall buck power converter system response at the various stages of the online RL adaptation. The region with $\beta_{(0)}$ in the figure corresponds to the system response using the initially suboptimal tracking controller gains, while the region with $\beta_{(1)}$ shows the system response after convergence to the optimal controller values from the RL adaptation. Following variation in the load resistor $R_L$, the system performance after convergence to the new optimal control gains is then shown in the region with $\beta_{(2)}$. This way the proposed online optimal and adaptive tracking RL framework is able to maintain the desired level of system performance subject to gradual variations in the system parameters.

## 7 | CONCLUSIONS

This article has proposed and demonstrated an online optimal and adaptive reinforcement learning (RL) tracking controller using an augmented formulation with integral control for unknown or varying discrete-time (DT) systems. Existing online tracking methods either assume a predetermined feedforward input for the tracking control or use restrictive assumptions on the reference model dynamics and discounted tracking costs. Moreover, the existing online tracking methods are unable to guarantee zero steady-state tracking error. By contrast, the proposed method transforms the DT optimal tracking control into a regulation problem and solves a resulting DT algebraic Riccati equations online without knowledge of the system dynamics or any restrictive assumptions of the existing methods and eliminates steady-state tracking error. Two RL strategies are proposed for the LQT based on both the value function approximation and Q-learning. The approaches offer a through-life adaptation strategy for the controller gains and guarantee zero steady-state tracking error as shown in the case study examples.

## ORCID

*Ibrahim Sanusi* https://orcid.org/0000-0002-3198-9048
*George Konstantopoulos* https://orcid.org/0000-0003-3339-6921

## REFERENCES

1. Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning. *Nature*. 2015;518(7540):529.
2. Sutton RS, Barto AG. *Reinforcement Learning: An Introduction*. Vol 1. Cambridge, MA: MIT Press; 1998.
3. Levine WS. *The Control Systems Handbook: Control System Advanced Methods*. Cambridge, MA: CRC Press; 2010.
4. Si J. *Handbook of Learning and Approximate Dynamic Programming*. Vol 2. Hoboken, NJ: John Wiley & Sons; 2004.
5. Wang F-Y, Zhang H, Liu D. Adaptive dynamic programming: an introduction. *IEEE Comput Intell Mag*. 2009;4(2):39-47.
6. Åström KJ, Wittenmark B. *Adaptive Control*. Mineola, NY: Dover Publications, INC; 2013.
7. Adetola VA. Integrated Real-Time Optimization and Model Predictive Control Under Parametric Uncertainties (Ph.D. thesis). Queen's University; 2008.
8. Lewis FL, Vrabie D, Vamvoudakis KG. Reinforcement learning and feedback control: using natural decision methods to design optimal adaptive controllers. *IEEE Control Syst*. 2012;32(6):76-105.
9. Bertsekas DP, Tsitsiklis JN. Neuro-dynamic programming: an overview. Paper presented at: Proceedings of the 34th IEEE Conference on Decision and Control, 1995; Vol. 1, 1995:560-564; IEEE.
10. Lewis FL, Syrmos VL. *Optimal Control*. Hoboken, NJ: John Wiley & Sons; 1995.
11. Khan SG, Herrmann G, Lewis FL, Pipe T, Melhuish C. Reinforcement learning and optimal adaptive control: an overview and implementation examples. *Ann Rev Control*. 2012;36(1):42-59.
12. Xu X, Wang C, Lewis FL. Some recent advances in learning and adaptation for uncertain feedback control systems. *Int J Adapt Control Signal Processing*. 2014;28(3-5):201-204.
13. Czubenko M, Kowalczuk Z, Ordys A. Autonomous driver based on an intelligent system of decision-making. *Cognit Comput*. 2015;7(5):569-581.
14. Li F-D, Wu M, He Y, Chen X. Optimal control in microgrid using multi-agent reinforcement learning. *ISA Trans*. 2012;51(6):743-751.
15. Garg S. Aircraft turbine engine control research at NASA Glenn research center. *J Aerospace Eng*. 2013;26(2):422-438.

16. Zhu Y, Zhao D, He H, Ji J. Event-triggered optimal control for partially unknown constrained-input systems via adaptive dynamic programming. *IEEE Trans Ind Electron*. 2016;64(5):4101-4109.
17. Werbos PJ. Neural networks for control and system identification. Paper presented at: Proceedings of the 28th IEEE Conference on Decision and Control, 1989; 1989:260-265; IEEE.
18. Werbos PJ. Approximate dynamic programming for real-time control and neural modeling. *Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches*. Vol 15. New York, NY: Van Nostrand Reinhold; 1992:493-525.
19. Si J, Wang Y-T. Online learning control by association and reinforcement. *IEEE Trans Neural Netw*. 2001;12(2):264-276.
20. Ferrari S, Stengel R. F. An adaptive critic global controller. Paper presented at: Proceedings of the 2002 American Control Conference (IEEE Cat. No. CH37301); Vol. 4; 2002:2665-2670; IEEE.
21. Vrabie D, Pastravanu O, Abu-Khalaf M, Lewis FL. Adaptive optimal control for continuous-time linear systems based on policy iteration. *Automatica*. 2009;45(2):477-484.
22. Lewis FL, Vamvoudakis KG. Optimal adaptive control for unknown systems using output feedback by reinforcement learning methods. Paper presented at: Proceedings of the 2010 8th IEEE International Conference on Control and Automation (ICCA); IEEE; 2010:2138-2145.
23. Dierks T, Jagannathan S. Online optimal control of affine nonlinear discrete-time systems with unknown internal dynamics by using time-based policy update. *IEEE Trans Neural Netw Learn Syst*. 2012;23(7):1118-1129.
24. Modares H, Lewis FL, Naghibi-Sistani M-B. Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems. *Automatica*. 2014;50(1):193-202.
25. Zhu Y, Zhao D, Li X. Using reinforcement learning techniques to solve continuous-time non-linear optimal tracking problem without system dynamics. *IET Control Theory Appl*. 2016;10(12):1339-1347.
26. Zhao Q, Xu H, Sarangapani J. Finite-horizon near optimal adaptive control of uncertain linear discrete-time systems. *Opt Control Appl Methods*. 2015;36(6):853-872.
27. Lv Y, Na J, Yang Q, Wu X, Guo Y. Online adaptive optimal control for continuous-time nonlinear systems with completely unknown dynamics. *Int J Control*. 2016;89(1):99-112.
28. Kiumarsi B, Vamvoudakis KG, Modares H, Lewis FL. Optimal and autonomous control using reinforcement learning: a survey. *IEEE Trans Neural Netw Learn Syst*. 2018;29(6):2042-2062.
29. Kiumarsi B, Modares H, Lewis FL. Optimal tracking control of uncertain systems: on-policy and off-policy reinforcement learning approaches. *Control of Complex Systems*. Oxford, UK: Butterworth-Heinemann; 2016:165-186.
30. Zhang H, Wei Q, Luo Y. A novel infinite-time optimal tracking control scheme for a class of discrete-time nonlinear systems via the greedy hdp iteration algorithm. *IEEE Trans Syst Man Cybern Part B (Cybern)*. 2008;38(4):937-942.
31. Wang D, Liu D, Wei Q. Finite-horizon neuro-optimal tracking control for a class of discrete-time nonlinear systems using adaptive dynamic programming approach. *Neurocomputing*. 2012;78(1):14-22.
32. Dierks T, Jagannathan S. Optimal tracking control of affine nonlinear discrete-time systems with unknown internal dynamics.Paper presented at: Proceedings of the 48th IEEE Conference on Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009; 2009:6750-6755; IEEE.
33. Lin Q, Wei Q, Liu D. A novel optimal tracking control scheme for a class of discrete-time nonlinear systems using generalised policy iteration adaptive dynamic programming algorithm. *Int J Syst Sci*. 2017;48(3):525-534.
34. Kiumarsi-Khomartash B, Lewis FL, Naghibi-Sistani MB, Karimpour A. Optimal tracking control for linear discrete-time systems using reinforcement learning. Paper presented at: Proceedings of the 52nd IEEE Conference on Decision and Control; 2013:3845-3850; IEEE.
35. Kiumarsi B, Lewis FL, Modares H, Karimpour A, Naghibi-Sistani M-B. Reinforcement q-learning for optimal tracking control of linear discrete-time systems with unknown dynamics. *Automatica*. 2014;50(4):1167-1175.
36. Kiumarsi B, Lewis FL. Actor–critic-based optimal tracking for partially unknown nonlinear discrete-time systems. *IEEE Trans Neural Netw Learn Syst*. 2014;26(1):140-151.
37. Modares H. Optimal Tracking Control of Uncertain Systems: On-Policy and Off-policy Reinforcement Learning Approaches (Ph.D. thesis). The University of Texas at Arlington; August 2015.
38. Modares H, Lewis FL. Linear quadratic tracking control of partially-unknown continuous-time systems using reinforcement learning. *IEEE Trans Automat Control*. 2014;59(11):3051-3056.
39. Modares H, Lewis FL. Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning. *Automatica*. 2014;50(7):1780-1792.
40. Precup R-E, Preitl S, Rudas IJ, Tomescu ML, Tar JK. Design and experiments for a class of fuzzy controlled servo systems. *IEEE/ASME Trans Mechatron*. 2008;13(1):22-35.
41. Krommydas KF, Alexandridis AT. Nonlinear stability analysis for ac/dc voltage source converters driven by pi current-mode controllers. Paper presented at: Proceedings of the 2014 European Control Conference (ECC); 2014:2774-2779; IEEE.
42. Busoniu L, Babuska R, De Schutter B, Ernst D. *Reinforcement Learning and Dynamic Programming Using Function Approximators*. Vol 39. Boca Raton, FL: CRC Press; 2010.
43. Bertsekas DP. *Dynamic Programming and Optimal Control*. Vol 2. Belmont, MA: Athena Scientific; 1995.
44. Hewer G. An iterative technique for the computation of the steady state gains for the discrete optimal regulator. *IEEE Trans Automat Control*. 1971;16(4):382-384.
45. Ljung L. System identification. *Encyclopedia of Electrical and Electronics Engineering*. Hoboken, NJ: Wiley; 1999:1-19.

46. Grzedzinski K, Trodden P. Learning MPC: System stability and convergent identification under bounded modelling error.Paper presented at: Proceedings of the 2018 Australian & New Zealand Control Conference (ANZCC); 2018:125-130. doi:https://doi.org/10.1109/anzcc.2018.8606603.
47. Konstantopoulos GC, Zhong Q-C. Current-limiting dc/dc power converters. *IEEE Trans Control Syst Tech*. 2018;27(2):855-863.

**How to cite this article:** Sanusi I, Mills A, Dodd T, Konstantopoulos G. Online optimal and adaptive integral tracking control for varying discrete-time systems using reinforcement learning. *Int J Adapt Control Signal Process*. 2020;1–21. https://doi.org/10.1002/acs.3115