Model-Based Reinforcement Learning for Infinite-Horizon Approximate Optimal Tracking

Rushikesh Kamalapurkar, Lindsey Andrews, Patrick Walters, and Warren E. Dixon

***Abstract*—This paper provides an approximate online adaptive solution to the infinite-horizon optimal tracking problem for control-affine continuous-time nonlinear systems with unknown drift dynamics. To relax the persistence of excitation condition, model-based reinforcement learning is implemented using a concurrent learning-based system identifier to simulate experience by evaluating the Bellman error over unexplored areas of the state space. Tracking of the desired trajectory and convergence of the developed policy to a neighborhood of the optimal policy are established via Lyapunov-based stability analysis. Simulation results demonstrate the effectiveness of the developed technique.**

***Index Terms*—reinforcement learning, optimal control, data-driven control, nonlinear control, system identification**

I. Introduction

REINFORCEMENT learning (RL)-based techniques have been effectively utilized to obtain online approximate solutions to optimal control problems for systems with finite state-action spaces, and stationary environments (cf. [1], [2]). Various implementations of RL-based learning strategies to solve deterministic optimal control problems in continuous state-spaces can be found in results such as [3]–[11] for set-point regulation, and [12]–[17] for trajectory tracking. Results such as [13], [16]–[18] solve optimal tracking problems for linear and nonlinear systems online, where persistence of excitation (PE) of the error states is used to establish convergence. In general, it is impossible to guarantee PE a priori; hence, a probing signal designed using trial and error is added to the controller to ensure PE. However, the probing signal is not considered in the stability analysis. In this paper, the objective is to employ data-driven model-based RL to design an online approximate optimal tracking controller for continuous-time uncertain nonlinear systems under a relaxed finite excitation condition.

RL in systems with continuous state and action spaces is realized via value function approximation, where the value function corresponding to the optimal control problem is approximated using a parametric universal approximator. The control policy is generally derived from the approximate value function; hence, obtaining a good approximation of the value function is critical to the stability of the closed-loop system. In trajectory tracking problems, the value function depends explicitly on time. Since universal function approximators can approximate functions with arbitrary accuracy only on compact domains, value functions for infinite-horizon optimal tracking problems can not be approximated with arbitrary accuracy [12], [18].

The technical challenges associated with the nonautonomous nature of the trajectory tracking problem are addressed in the authors' previous work in [18], where it is established that under a matching condition on the desired trajectory, the optimal trajectory tracking problem can be reformulated as a stationary optimal control problem. Since the value function associated with a stationary optimal control problem is time-invariant, it can be approximated using traditional function approximation techniques.

The aforementioned reformulation in [18] requires computation of the steady-state tracking controller, which depends on the system model; hence, the development in [18] requires exact model knowledge. Obtaining an accurate estimate of the desired steady-state controller, and injecting the resulting estimation error in the stability analysis are the major technical challenges in extending the work in [18] to uncertain systems.

Concurrent learning (CL)-based system identifiers are used in results such as [19] and [20] to solve optimal regulation problems for uncertain systems. Extension of the techniques in [19] and [20] to solve the optimal tracking problem is not trivial due to the fact that the optimal tracking problem requires knowledge of the steady-state controller. An estimate of the steady-state controller can be generated using CL-based system identifiers. The use of an estimate instead of the true steady-state controller results in additional approximation errors that can potentially cause instability during the learning phase.

A primary contribution of this paper and our preliminary work in [21] is to analyze the stability of the closed-loop system in the presence of the aforementioned approximation error. The error between the actual steady-state controller and its estimate is included in the stability analysis by examining the trajectories of the concatenated system under the implemented control signal. In addition to estimating the desired steady-state controller, the the CL-based system identifier is also used to simulate experience by evaluating the Bellman error (BE) over unexplored areas of the state space [21]–[23]. To illustrate the effectiveness of the developed technique, simulation results are presented that demonstrate approximation of the optimal policy without an added exploration signal.

Rushikesh Kamalapurkar, Lindsey Andrews, Patrick Walters, and Warren E. Dixon are with the Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL, USA. Email: {rkamalapurkar, landr010, walters8, wdixon}@ufl.edu.

This research is supported in part by NSF award number 1509516 and ONR grant number N00014-13-1-0151. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsoring agency.

II. Problem formulation and exact solution

Consider a control affine system described by the differential equation $\dot{x} = f(x) + g(x)u$, where $x \in \mathbb{R}^n$ denotes the state, $u \in \mathbb{R}^m$ denotes the control input, and $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ are locally Lipschitz continuous functions that denote the drift dynamics, and the control effectiveness, respectively.[1] The control objective is to optimally track a time-varying desired trajectory $x_d \in \mathbb{R}^n$. To facilitate the subsequent control development, an error signal $e \in \mathbb{R}^n$ is defined as $e \triangleq x - x_d$. Since the steady-state control input that is required for the system to track a desired trajectory is, in general, not identically zero, an infinite-horizon total-cost optimal control problem formulated in terms of a quadratic cost function containing e and u always results in an infinite cost. To address this issue, an alternative cost function is formulated in terms of the tracking error and the mismatch between the actual control signal and the desired steady-state control [12], [16]–[18]. The following assumptions facilitate the determination of the desired steady-state control.

Assumption 1. [18] The function g is bounded, the matrix $g(x)$ has full column rank for all $x \in \mathbb{R}^n$, and the function $g^+ : \mathbb{R}^n \to \mathbb{R}^{m\times n}$ defined as $g^+ \triangleq \left(g^T g\right)^{-1} g^T$ is bounded and locally Lipschitz.

Assumption 2. [18] The desired trajectory is bounded by a known positive constant $d \in \mathbb{R}$ such that $\|x_d\| \leq d$, and there exists a locally Lipschitz function $h_d : \mathbb{R}^n \to \mathbb{R}^n$ such that $\dot{x}_d = h_d(x_d)$ and $g(x_d)\, g^+(x_d)\,(h_d(x_d) - f(x_d)) = h_d(x_d) - f(x_d)$, $\forall t \in \mathbb{R}_{\geq t_0}$.

Based on Assumptions 1 and 2, the steady-state control policy $u_d : \mathbb{R}^n \to \mathbb{R}^m$ required for the system to track the desired trajectory x_d can be expressed as $u_d(x_d) = g_d^+ (h_d(x_d) - f_d)$, where $f_d \triangleq f(x_d)$ and $g_d^+ \triangleq g^+(x_d)$. The error between the actual control signal and the desired steady-state control signal is defined as $\mu \triangleq u - u_d(x_d)$. Using μ, the system dynamics can be expressed in the autonomous form

$$\dot{\zeta} = F(\zeta) + G(\zeta)\mu, \tag{1}$$

where the concatenated state $\zeta \in \mathbb{R}^{2n}$ is defined as $\zeta \triangleq \left[e^T,\ x_d^T\right]^T$, and the functions $F : \mathbb{R}^{2n} \to \mathbb{R}^{2n}$ and $G : \mathbb{R}^{2n} \to \mathbb{R}^{2n\times m}$ are defined as $F(\zeta) \triangleq \left[f^T(e + x_d) - h_d^T + u_d^T(x_d)\, g^T(e + x_d),\ \ h_d^T\right]^T$, $G(\zeta) \triangleq \left[g^T(e + x_d),\ \ \mathbf{0}_{m\times n}\right]^T$, where $\mathbf{0}_{n\times m}$ denotes an $n \times m$ matrix of zeros. The control error μ is treated hereafter as the design variable. The control objective is to solve the infinite-horizon optimal regulation problem online, i.e., to simultaneously synthesize and utilize a control signal μ online to minimize the cost functional $J(\zeta, \mu) \triangleq \int_{t_0}^{\infty} r(\zeta(\tau), \mu(\tau))\, d\tau$, under the dynamic constraint $\dot{\zeta} = F(\zeta) + G(\zeta)\mu$, while tracking the desired trajectory, where $r : \mathbb{R}^{2n} \times \mathbb{R}^m \to \mathbb{R}$ is the local cost defined as $r(\zeta, \mu) \triangleq Q(e) + \mu^T R \mu$, $R \in \mathbb{R}^{m\times m}$ is a positive definite symmetric matrix of constants, and $Q : \mathbb{R}^n \to \mathbb{R}$ is a continuous positive definite function.

Assuming that an optimal policy exists, the optimal policy can be characterized in terms of the value function $V^* : \mathbb{R}^{2n} \to \mathbb{R}$ defined as $V^*(\zeta) \triangleq \min_{\mu(\tau)\in U | \tau\in\mathbb{R}_{\geq t}} \int_t^{\infty} r(\phi^{\mu}(\tau, t, \zeta), \mu(\tau))\, d\tau$, where $U \in \mathbb{R}^m$ is the action space and the notation $\phi^{\mu}(t; t_0, \zeta_0)$ denotes the trajectory of $\dot{\zeta} = F(\zeta) + G(\zeta)\mu$, under the control signal $\mu : \mathbb{R}_{\geq 0} \to \mathbb{R}^m$ with the initial condition $\zeta_0 \in \mathbb{R}^{2n}$ and initial time $t_0 \in \mathbb{R}_{\geq 0}$. Assuming that a minimizing policy exists and that V^* is continuously differentiable, a closed-form solution for the optimal policy can be obtained as [24] $\mu^*(\zeta) = -\frac{1}{2} R^{-1} G^T(\zeta) \left(\nabla_\zeta V^*(\zeta)\right)^T$, where $\nabla_\zeta(\cdot) \triangleq \frac{\partial(\cdot)}{\partial \zeta}$. The optimal policy and the optimal value function satisfy the Hamilton-Jacobi-Bellman (HJB) equation [24]

$$\nabla_\zeta V^*(\zeta)(F(\zeta) + G(\zeta)\,\mu^*(\zeta)) + \overline{Q}(\zeta) + \mu^{*T}(\zeta) R \mu^*(\zeta) = 0, \tag{2}$$

with the initial condition $V^*(0) = 0$, where the function $\overline{Q} : \mathbb{R}^{2n} \to \mathbb{R}$ is defined as $\overline{Q}\left(\left[e^T,\ \ x_d^T\right]^T\right) = Q(e)$, $\forall e,\ x_d \in \mathbb{R}^n$.

Remark 1. Assumptions 1 and 2 can be eliminated if a discounted cost optimal tracking problem is considered instead of the total cost problem considered in this article. The discounted cost tracking problem considers a value function of the form $V^*(\zeta) \triangleq \min_{u(\tau)\in U | \tau\in\mathbb{R}_{\geq t}} \int_t^{\infty} e^{\kappa(t-\tau)} r(\phi^u(\tau, t, \zeta), u(\tau))\, d\tau$, where $\kappa \in \mathbb{R}_{>0}$ is a constant discount factor, and the control effort u is minimized instead of the control error μ. The control effort required for a system to perfectly track a desired trajectory is generally nonzero even if the initial system state is on the desired trajectory. Hence, in general, the optimal value function for a discounted cost problem does not satisfy $V^*(0) = 0$. Online continuous-time RL techniques are generally analyzed using the optimal value function as a candidate Lyapunov function. Since the optimal value function for a discounted cost problem does not evaluate to zero at the origin, it can not be used as a candidate Lyapunov function, leading to complications in the stability analysis of a discounted cost optimal controller during the learning phase. Hence, to make the stability analysis tractable, a total-cost optimal control problem is considered in this paper.

III. Bellman Error

Since a closed-form solution of the HJB equation is generally infeasible to obtain, an approximate solution is sought. In an actor-critic-based solution, the optimal value function V^* is replaced by a parametric estimate $\hat{V}\left(\zeta, \hat{W}_c\right)$ and the optimal policy μ^* by a parametric estimate $\hat{\mu}\left(\zeta, \hat{W}_a\right)$, where $\hat{W}_c \in \mathbb{R}^L$ and $\hat{W}_a \in \mathbb{R}^L$ denote vectors of estimates of the ideal parameters. The objective of the critic is to learn the parameters $\hat{W}_c$, and the objective of the actor is to learn the parameters $\hat{W}_a$. Substituting the estimates $\hat{V}$ and $\hat{\mu}$ for V^* and μ^* in the HJB equation, respectively, yields a residual error $\delta : \mathbb{R}^{2n} \times \mathbb{R}^L \times \mathbb{R}^L \to \mathbb{R}$, called the BE, is defined as

$$\delta\left(\zeta, \hat{W}_c, \hat{W}_a\right) = \overline{Q}(\zeta) + \hat{\mu}^T\left(\zeta, \hat{W}_a\right) R \hat{\mu}\left(\zeta, \hat{W}_a\right)$$

[1] For notational brevity, unless otherwise specified, the domain of all the functions is assumed to be $\mathbb{R}_{\geq 0}$. Furthermore, time-dependence is suppressed in equations and definitions. For example, the trajectory $x : \mathbb{R}_{\geq 0} \to \mathbb{R}^n$ is defined by abuse of notation as $x \in \mathbb{R}^n$ and unless otherwise specified, an equation of the form $f + h(y, t) = g(x)$ is interpreted as $f(t) + h(y(t), t) = g(x(t))$ for all $t \in \mathbb{R}_{\geq 0}$.

$$+\nabla_\zeta \hat{V}\left(\zeta,\hat{W}_c\right)\left(F\left(\zeta\right)+G\left(\zeta\right)\hat{\mu}\left(\zeta,\hat{W}_a\right)\right). \quad (3)$$

Specifically, to solve the optimal control problem, the critic aims to find a set of parameters $\hat{W}_c$ and the actor aims to find a set of parameters $\hat{W}_a$ such that $\delta\left(\zeta,\hat{W}_c,\hat{W}_a\right)=0$, and $\hat{u}\left(\zeta,\hat{W}_a\right)=-\frac{1}{2}R^{-1}G^T(\zeta)\left(\nabla_\zeta \hat{V}\left(\zeta,\hat{W}_a\right)\right)^T$, $\forall \zeta\in\mathbb{R}^{2n}$. Since an exact basis for value function approximation is generally not available, an approximate set of parameters that minimizes the BE is sought. In particular, to ensure uniform approximation of the value function and the policy over a compact operating domain $\mathcal{C}\subset\mathbb{R}^{2n}$, it is desirable to find parameters that minimize the error $E_s:\mathbb{R}^L\times\mathbb{R}^L\to\mathbb{R}$ defined as $E_s\left(\hat{W}_c,\hat{W}_a\right)\triangleq\sup_{\zeta\in\mathcal{C}}\left|\delta\left(\zeta,\hat{W}_c,\hat{W}_a\right)\right|$. Computation of the error E_s, and computation of the control signal u require knowledge of the system drift dynamics f. Two prevalent approaches employed to render the control design robust to uncertainties in the system drift dynamics are integral RL (cf. [10] and [25]) and state derivative estimation (cf. [7] and [18]). However, in techniques such as [7], [10], [18], [25] the BE can only be evaluated along the system trajectory. Thus, instead of E_s, the instantaneous integral error $\hat{E}\left(t\right)\triangleq\int_{t_0}^{t}\delta^2\left(\phi^{\hat{\mu}}\left(\tau,t_0,\zeta_0\right),\hat{W}_c\left(t\right),\hat{W}_a\left(t\right)\right)d\tau$ is used to facilitate learning.

Intuitively, for $\hat{E}$ to approximate E_s over an operating domain, the state trajectory $\phi^{\hat{\mu}}\left(t,t_0,\zeta_0\right)$ needs to visit as many points in the operating domain as possible. This intuition is formalized by the fact that techniques such as [7], [10], [18], [25], [26] require PE to achieve convergence. The PE condition is relaxed in [10] to a finite excitation condition by using integral RL along with experience replay, where each evaluation of the BE is interpreted as gained experience, and these experiences are stored in a history stack and are repeatedly used in the learning algorithm to improve data efficiency.

In this paper, a different approach is used to improve data efficiency. A dynamic system identifier is developed to generate a parametric estimate $\hat{F}\left(\zeta,\hat{\theta}\right)$ of the drift dynamics F, where $\hat{\theta}$ denotes the estimate of the matrix of unknown parameters. Given $\hat{F}$, $\hat{V}$, and $\hat{\mu}$, an estimate of the BE can be evaluated at any $\zeta\in\mathbb{R}^{2n}$. That is, using $\hat{F}$, experience can be simulated by extrapolating the BE over unexplored off-trajectory points in the operating domain. Hence, if an identifier can be developed such that $\hat{F}$ approaches F exponentially fast, learning laws for the optimal policy can utilize simulated experience along with experience gained and stored along the state trajectory.

If parametric approximators are used to approximate F, convergence of $\hat{F}$ to F is implied by convergence of the parameters to their unknown ideal values. It is well known that adaptive system identifiers require PE to achieve parameter convergence. To relax the PE condition, a CL-based (cf. [21]–[23], [27]) system identifier that uses recorded data for learning is developed in the following section.

IV. System Identification

On any compact set $\mathcal{C}\subset\mathbb{R}^n$ the function f can be represented using a neural network (NN) as $f\left(x\right)=\theta^T\sigma_f\left(Y^Tx_1\right)+\epsilon_\theta\left(x\right)$, where $x_1\triangleq\begin{bmatrix}1 & x^T\end{bmatrix}^T\in\mathbb{R}^{n+1}$, $\theta\in\mathbb{R}^{p+1\times n}$ and $Y\in\mathbb{R}^{n+1\times p}$ denote the constant unknown output-layer and hidden-layer NN weights, $\sigma_f:\mathbb{R}^p\to\mathbb{R}^{p+1}$ denotes a bounded NN basis function, $\epsilon_\theta:\mathbb{R}^n\to\mathbb{R}^n$ denotes the function reconstruction error, and $p\in\mathbb{N}$ denotes the number of NN neurons. Using the universal function approximation property of single layer NNs, given a constant matrix Y such that the rows of $\sigma_f\left(Y^Tx_1\right)$ form a proper basis, there exist constant ideal weights θ and known constants $\overline{\theta}$, $\overline{\epsilon_\theta}$, and $\overline{\epsilon'_\theta}\in\mathbb{R}$ such that $\|\theta\|\leq\overline{\theta}<\infty$, $\sup_{x\in\mathcal{C}}\|\epsilon_\theta\left(x\right)\|\leq\overline{\epsilon_\theta}$, and $\sup_{x\in\mathcal{C}}\|\nabla_x\epsilon_\theta\left(x\right)\|\leq\overline{\epsilon'_\theta}$, where $\|\cdot\|$ denotes the Euclidean norm for vectors and the Frobenius norm for matrices [28].

Using an estimate $\hat{\theta}\in\mathbb{R}^{p+1\times n}$ of the weight matrix θ, the function f can be approximated by the function $\hat{f}:\mathbb{R}^{2n}\times\mathbb{R}^{p+1\times n}\to\mathbb{R}^n$ defined as $\hat{f}\left(\zeta,\hat{\theta}\right)\triangleq\hat{\theta}^T\sigma_\theta\left(\zeta\right)$, where $\sigma_\theta:\mathbb{R}^{2n}\to\mathbb{R}^{p+1}$ is defined as $\sigma_\theta\left(\zeta\right)=\sigma_f\left(Y^T\begin{bmatrix}1 & e^T+x_d^T\end{bmatrix}^T\right)$. An estimator for online identification of the drift dynamics is developed as

$$\dot{\hat{x}}=\hat{\theta}^T\sigma_\theta\left(\zeta\right)+g\left(x\right)u+k\tilde{x}, \quad (4)$$

where $\tilde{x}\triangleq x-\hat{x}$, and $k\in\mathbb{R}$ is a positive constant learning gain.

Assumption 3. [23] A history stack containing recorded state-action pairs $\{x_j,u_j\}_{j=1}^M$ along with numerically computed state derivatives $\{\dot{\bar{x}}_j\}_{j=1}^M$ that satisfies $\lambda_{\min}\left(\sum_{j=1}^M\sigma_{fj}\sigma_{fj}^T\right)=\underline{\sigma_\theta}>0$, $\|\dot{\bar{x}}_j-\dot{x}_j\|<\overline{d}$, $\forall j$ is available a priori, where $\sigma_{fj}\triangleq\sigma_f\left(Y^T\begin{bmatrix}1 & x_j^T\end{bmatrix}^T\right)$, $\overline{d}\in\mathbb{R}$ is a known positive constant, $\dot{x}_j=f\left(x_j\right)+g\left(x_j\right)u_j$, and $\lambda_{\min}\left(\cdot\right)$ denotes the minimum eigenvalue.[2]

The weight estimates $\hat{\theta}$ are updated using the following CL-based update law:

$$\dot{\hat{\theta}}=\Gamma_\theta\sigma_f\left(Y^Tx_1\right)\tilde{x}^T+k_\theta\Gamma_\theta\sum_{j=1}^{M}\sigma_{fj}\left(\dot{\bar{x}}_j-g_ju_j-\hat{\theta}^T\sigma_{fj}\right)^T, \quad (5)$$

where $k_\theta\in\mathbb{R}$ is a constant positive CL gain, and $\Gamma_\theta\in\mathbb{R}^{p+1\times p+1}$ is a constant, diagonal, and positive definite adaptation gain matrix. Using the identifier, the BE in (3) can be approximated as

$$\hat{\delta}\left(\zeta,\hat{\theta},\hat{W}_c,\hat{W}_a\right)=\overline{Q}\left(\zeta\right)+\hat{\mu}^T\left(\zeta,\hat{W}_a\right)R\hat{\mu}\left(\zeta,\hat{W}_a\right)+\nabla_\zeta\hat{V}\left(\zeta,\hat{W}_a\right)\left(F_\theta\left(\zeta,\hat{\theta}\right)+F_1(\zeta)+G(\zeta)\,\hat{\mu}\left(\zeta,\hat{W}_a\right)\right). \quad (6)$$

In (6),

$$F_\theta\left(\zeta,\hat{\theta}\right)\triangleq\begin{bmatrix}\hat{\theta}^T\sigma_\theta\left(\zeta\right)-g\left(x\right)g^+\left(x_d\right)\hat{\theta}^T\sigma_\theta\left(\begin{bmatrix}\mathbf{0}_{n\times1}\\ x_d\end{bmatrix}\right)\\ \mathbf{0}_{n\times1}\end{bmatrix},$$

[2]A priori availability of the history stack is used for ease of exposition, and is not necessary. Provided the system states are exciting over a finite time interval $t\in\left[t_0,t_0+\overline{t}\right]$ (versus $t\in\left[t_0,\infty\right)$ as in traditional PE-based approaches) the history stack can also be recorded online. The controller developed in [18] can be used over the time interval $t\in\left[t_0,t_0+\overline{t}\right]$ while the history stack is being recorded, and the controller developed in this result can be used thereafter. The use of two different controllers results in a switched system with one switching event. Since there is only one switching event, the stability of the switched system follows from the stability of the individual subsystems.

and $F_1(\zeta) \triangleq \left[\left(-h_d + g(e+x_d) g^+(x_d) h_d\right)^T,\ h_d^T\right]^T$.

V. Value function approximation

Since V^* and μ^* are functions of the state ζ, the minimization problem stated in Section II is intractable. To obtain a finite-dimensional minimization problem, the optimal value function is represented over any compact operating domain $\mathcal{C} \subset \mathbb{R}^{2n}$ using a NN as $V^*(\zeta) = W^T \sigma(\zeta) + \epsilon(\zeta)$, where $W \in \mathbb{R}^L$ denotes a vector of unknown NN weights, $\sigma : \mathbb{R}^{2n} \to \mathbb{R}^L$ denotes a bounded NN basis function, $\epsilon : \mathbb{R}^{2n} \to \mathbb{R}$ denotes the function reconstruction error, and $L \in \mathbb{N}$ denotes the number of NN neurons. Using the universal function approximation property of single layer NNs, for any compact set $\mathcal{C} \subset \mathbb{R}^{2n}$, there exist constant ideal weights W and known positive constants $\overline{W}$, $\overline{\epsilon}$, and $\overline{\epsilon'} \in \mathbb{R}$ such that $\|W\| \leq \overline{W} < \infty$, $\sup_{\zeta \in \mathcal{C}} \|\epsilon(\zeta)\| \leq \overline{\epsilon}$, and $\sup_{\zeta \in \mathcal{C}} \|\nabla_\zeta \epsilon(\zeta)\| \leq \overline{\epsilon'}$ [28].

A NN representation of the optimal policy is obtained as $\mu^*(\zeta) = -\frac{1}{2} R^{-1} G^T(\zeta) \left(\nabla_\zeta \sigma^T(\zeta) W + \nabla_\zeta \epsilon^T(\zeta)\right)$. Using estimates $\hat{W}_c$ and $\hat{W}_a$ for the ideal weights W, the optimal value function and the optimal policy are approximated as

$$\hat{V}\left(\zeta, \hat{W}_c\right) \triangleq \hat{W}_c^T \sigma(\zeta),\ \hat{\mu}\left(\zeta, \hat{W}_a\right) \triangleq -\frac{1}{2} R^{-1} G^T(\zeta) \nabla_\zeta \sigma^T(\zeta) \hat{W}_a. \tag{7}$$

The optimal control problem is thus reformulated as the need to find a set of weights $\hat{W}_c$ and $\hat{W}_a$ online, to minimize the error $\hat{E}_{\hat{\theta}}\left(\hat{W}_c, \hat{W}_a\right) \triangleq \sup_{\zeta \in \chi} \left|\hat{\delta}\left(\zeta, \hat{\theta}, \hat{W}_c, \hat{W}_a\right)\right|$, for a given $\hat{\theta}$, while simultaneously improving $\hat{\theta}$ using (5), and ensuring stability of the system using the control law

$$u = \hat{\mu}\left(\zeta, \hat{W}_a\right) + \hat{u}_d\left(\zeta, \hat{\theta}\right), \tag{8}$$

where $\hat{u}_d\left(\zeta, \hat{\theta}\right) \triangleq g_d^+\left(h_d - \hat{\theta}^T \sigma_{\theta d}\right)$, and $\sigma_{\theta d} \triangleq \sigma_\theta\left(\begin{bmatrix} \mathbf{0}_{1\times n} & x_d^T \end{bmatrix}^T\right)$. The error between u_d and $\hat{u}_d$ is included in the stability analysis based on the fact that the error trajectories generated by the system $\dot{e} = f(x) + g(x) u - \dot{x}_d$ under the controller in (8) are identical to the error trajectories generated by the system $\dot{\zeta} = F(\zeta) + G(\zeta) \mu$ under the control law $\mu = \hat{\mu}\left(\zeta, \hat{W}_a\right) + g_d^+ \tilde{\theta}^T \sigma_{\theta d} + g_d^+ \epsilon_{\theta d}$, where $\epsilon_{\theta d} \triangleq \epsilon_\theta(x_d)$.

VI. Simulation of experience

Since computation of the supremum in $\hat{E}_{\hat{\theta}}$ is intractable in general, simulation of experience is implemented by minimizing a squared sum of BEs over finitely many points in the state space. The following assumption facilitates the aforementioned approximation.

Assumption 4. [21] There exists a finite set of points $\{\zeta_i \in \mathcal{C} \mid i = 1, \cdots, N\}$ and a constant $\underline{c} \in \mathbb{R}$ such that $0 < \underline{c} \triangleq \frac{1}{N}\left(\inf_{t \in \mathbb{R}_{\geq t_0}}\left(\lambda_{min}\left\{\sum_{i=1}^N \frac{\omega_i \omega_i^T}{\rho_i}\right\}\right)\right)$, where $\rho_i \triangleq 1 + \nu \omega_i^T \Gamma \omega_i \in \mathbb{R}$, and $\omega_i \triangleq \nabla_\zeta \sigma(\zeta_i)\left(F_\theta\left(\zeta_i, \hat{\theta}\right) + F_1(\zeta_i) + G(\zeta_i) \hat{\mu}\left(\zeta_i, \hat{W}_a\right)\right)$.

Using Assumption 4, simulation of experience is implemented by the weight update laws

$$\dot{\hat{W}}_c = -\eta_{c1} \Gamma \frac{\omega}{\rho} \hat{\delta}_t - \frac{\eta_{c2}}{N} \Gamma \sum_{i=1}^N \frac{\omega_i}{\rho_i} \hat{\delta}_{ti}, \tag{9}$$

$$\dot{\Gamma} = \left(\beta \Gamma - \eta_{c1} \Gamma \frac{\omega \omega^T}{\rho^2} \Gamma\right) \mathbf{1}_{\{\|\Gamma\| \leq \overline{\Gamma}\}},\ \|\Gamma(t_0)\| \leq \overline{\Gamma}, \tag{10}$$

$$\dot{\hat{W}}_a = -\eta_{a1}\left(\hat{W}_a - \hat{W}_c\right) - \eta_{a2} \hat{W}_a + \left(\frac{\eta_{c1} G_\sigma^T \hat{W}_a \omega^T}{4\rho} + \sum_{i=1}^N \frac{\eta_{c2} G_{\sigma i}^T \hat{W}_a \omega_i^T}{4N\rho_i}\right) \hat{W}_c, \tag{11}$$

where $\omega \triangleq \nabla_\zeta \sigma(\zeta)\left(F_\theta\left(\zeta, \hat{\theta}\right) + F_1(\zeta) + G(\zeta) \hat{\mu}\left(\zeta, \hat{W}_a\right)\right)$, $\Gamma \in \mathbb{R}^{L \times L}$ is the least-squares gain matrix, $\overline{\Gamma} \in \mathbb{R}$ denotes a positive saturation constant, $\beta \in \mathbb{R}$ denotes a constant forgetting factor, $\eta_{c1}, \eta_{c2}, \eta_{a1}, \eta_{a2} \in \mathbb{R}$ denote constant positive adaptation gains, $\mathbf{1}_{\{\cdot\}}$ denotes the indicator function of the set $\{\cdot\}$, $G_\sigma \triangleq \nabla_\zeta \sigma(\zeta) G(\zeta) R^{-1} G^T(\zeta) \nabla_\zeta \sigma^T(\zeta)$, and $\rho \triangleq 1 + \nu \omega^T \Gamma \omega$, where $\nu \in \mathbb{R}$ is a positive normalization constant. In (9)-(11) and in the subsequent development, for any function $\xi(\zeta, \cdot)$, the notation ξ_i, is defined as $\xi_i \triangleq \xi(\zeta_i, \cdot)$, and the instantaneous BEs $\hat{\delta}_t$ and $\hat{\delta}_{ti}$ are given by $\hat{\delta}_t = \hat{\delta}\left(\zeta, \hat{W}_c, \hat{W}_a, \hat{\theta}\right)$ and $\hat{\delta}_{ti} = \hat{\delta}\left(\zeta_i, \hat{W}_c, \hat{W}_a, \hat{\theta}\right)$.

Remark 2. To facilitate the stability analysis, the terms ω and δ are defined so that the update laws (9) - (11) have a similar form as the update laws in results such as [19] and [20]. However, the definitions of ω and δ, when expanded, are different in this paper on account of the optimal control problem being different. Hence, even though the update laws (9) - (11) look similar to the update laws in results such as [19] and [20], their content, and hence the resulting stability analysis, are different in this paper.

VII. Stability analysis

If the state penalty function $\overline{Q}$ is positive definite, then the optimal value function V^* is positive definite, and serves as a Lyapunov function for the concatenated system under the optimal control policy μ^*; hence, V^* is used (cf. [6], [7], [25]) as a candidate Lyapunov function for the closed-loop system under the policy $\hat{\mu}$. The function $\overline{Q}$, and hence, the function V^* are positive semidefinite; hence, the function V^* is not a valid candidate Lyapunov function. However, the results in [18] can be used to show that a nonautonomous form of the optimal value function denoted by $V_t^* : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$, defined as $V_t^*(e, t) = V^*\left(\left[e^T,\ x_d^T(t)\right]^T\right)$, $\forall e \in \mathbb{R}^n$, $t \in \mathbb{R}$, is positive definite and decrescent. Hence, $V_t^*(0, t) = 0$, $\forall t \in \mathbb{R}$ and there exist class $\mathcal{K}$ functions $\underline{v} : \mathbb{R} \to \mathbb{R}$ and $\overline{v} : \mathbb{R} \to \mathbb{R}$ such that $\underline{v}(\|e\|) \leq V_t^*(e, t) \leq \overline{v}(\|e\|)$, for all $e \in \mathbb{R}^n$ and for all $t \in \mathbb{R}$.

To facilitate the stability analysis, a concatenated state $Z \in \mathbb{R}^{2n+2L+n(p+1)}$ is defined as

$$Z \triangleq \left[e^T \quad \tilde{W}_c^T \quad \tilde{W}_a^T \quad \tilde{x}^T \quad \left(\text{vec}\left(\tilde{\theta}\right)\right)^T\right]^T,$$

and a candidate Lyapunov function is defined as

$$V_L(Z, t) \triangleq V_t^*(e, t) + \frac{1}{2} \tilde{W}_c^T \Gamma^{-1} \tilde{W}_c + \frac{1}{2} \tilde{W}_a^T \tilde{W}_a + \frac{1}{2} \tilde{x}^T \tilde{x} + \frac{1}{2} \text{tr}\left(\tilde{\theta}^T \Gamma_\theta^{-1} \tilde{\theta}\right) \tag{12}$$

where vec $(\cdot)$ denotes the vectorization operator. The saturated least-squares update law in (10) ensures that there exist

positive constants $\underline{\gamma}, \overline{\gamma} \in \mathbb{R}$ such that $\underline{\gamma} \leq \left\|\Gamma^{-1}(t)\right\| \leq \overline{\gamma}$, $\forall t \in \mathbb{R}$. Using the bounds on Γ and V_t^* and the fact that $\text{tr}\left(\tilde{\theta}^T \Gamma_\theta^{-1} \tilde{\theta}\right) = \left(\text{vec}\left(\tilde{\theta}\right)\right)^T \left(\Gamma_\theta^{-1} \otimes \mathbb{I}_{p+1}\right) \left(\text{vec}\left(\tilde{\theta}\right)\right)$, the candidate Lyapunov function in (12) can be bounded as

$$\underline{v_l}\left(\|Z\|\right) \leq V_L\left(Z, t\right) \leq \overline{v_l}\left(\|Z\|\right), \tag{13}$$

for all $Z \in \mathbb{R}^{2n+2L+n(p+1)}$ and for all $t \in \mathbb{R}$, where $\underline{v_l} : \mathbb{R} \to \mathbb{R}$ and $\overline{v_l} : \mathbb{R} \to \mathbb{R}$ are class $\mathcal{K}$ functions.

Theorem 1. *Provided Assumptions 2-4 hold, and the number of NN neurons, and the minimum singular values $\underline{c}$ and $\underline{\sigma_\theta}$ are large enough,*[3] *the controller in (8), along with the weight update laws (9)-(11), and the identifier in (4) along with the weight update law (5) ensure that the system states remain bounded, the tracking error is ultimately bounded, and that the control policy $\hat{\mu}$ converges to a neighborhood around the optimal control policy μ^*.*

Proof: Using (1) and the fact that $\dot{V}_t^*\left(e\left(t\right), t\right) = \dot{V}^*\left(\zeta\left(t\right)\right)$, $\forall t \in \mathbb{R}$, the time-derivative of the candidate Lyapunov function in (12) is

$$\dot{V}_L = \nabla_\zeta V^*\left(F + G\mu^*\right) - \tilde{W}_c^T \Gamma^{-1} \dot{\hat{W}}_c - \frac{1}{2} \tilde{W}_c^T \Gamma^{-1} \dot{\Gamma} \Gamma^{-1} \tilde{W}_c - \tilde{W}_a^T \dot{\hat{W}}_a + \dot{V}_0 + \nabla_\zeta V^* G \mu - \nabla_\zeta V^* G \mu^*. \tag{14}$$

Under sufficient gain conditions (cf. [29]), using (2), (4)-(7), and the update laws in (9)-(11) the expression in (14) can be bounded as

$$\dot{V}_L \leq -v_l\left(\|Z\|\right), \ \forall \|Z\| \geq v_l^{-1}\left(\iota\right), \ \forall Z \in \chi, \tag{15}$$

where ι is a positive constant, and $\chi \subset \mathbb{R}^{2n+2L+n(p+1)}$ is a compact set. Using (13) and (15), Theorem 4.18 in [30] can be invoked to conclude that every trajectory $Z\left(t\right)$ satisfying $\|Z\left(t_0\right)\| \leq \overline{v_l}^{-1}\left(\underline{v_l}\left(\rho\right)\right)$, where ρ is a positive constant, is bounded for all $t \in \mathbb{R}$ and satisfies $\limsup_{t \to \infty} \|Z\left(t\right)\| \leq \underline{v_l}^{-1}\left(\overline{v_l}\left(v_l^{-1}\left(\iota\right)\right)\right)$.[4] ∎

VIII. Simulation

In the following, the developed technique is applied to solve a linear quadratic tracking (LQT) problem. A linear system is selected because the optimal solution to the LQT problem can be computed analytically and compared against the solution generated by the developed technique. For simulation results on nonlinear systems, see [29]. To demonstrate convergence to the ideal weights, the following linear system is simulated: $\dot{x} = \begin{bmatrix} -1 & 1 \\ -0.5 & -0.5 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$. The control objective is to follow a desired trajectory, which is the solution of the initial value problem $\dot{x}_d = \begin{bmatrix} -1 & 1 \\ -2 & 1 \end{bmatrix} x_d, \quad x_d\left(0\right) = \begin{bmatrix} 0 \\ 2 \end{bmatrix}$. while ensuring convergence of the estimated policy $\hat{\mu}$ to a neighborhood of the policy μ^*, such that the control law $\mu\left(t\right) = \mu^*\left(\zeta\left(t\right)\right)$ minimizes the cost $\int_0^\infty \left(e^T\left(t\right) \text{diag}\left(\left[10, \ 10\right]\right) e\left(t\right) + \mu^2\left(t\right)\right) dt$.

[3] For a detailed description of the sufficient gain conditions, see [29].

[4] For detailed definitions of ι, ρ, and v_l, see [29]. The ultimate bound can be decreased by increasing learning gains and by increasing the number of neurons in the NNs provided the points in the history stack and the points for BE extrapolation can be selected to increase $\underline{\sigma_\theta}$ and $\underline{c}$.

Since the system is linear, the optimal value function is known to be quadratic. Hence, the value function is approximated using the quadratic basis $\sigma\left(\zeta\right) = \left[e_1^2, \ e_2^2, \ e_1 e_2, \ e_1 x_{d1}, \ e_2 x_{d2}, \ e_1 x_{d2}, \ e_2 x_{d1}\right]^T$, and the unknown drift dynamics is approximated using the linear basis $\sigma_\theta\left(x\right) = \left[x_1, x_2\right]^T$.[5]

The linear system and the linear desired dynamics result in the linear time-invariant concatenated error system

$$\dot{\zeta} = \begin{bmatrix} -1 & 1 & 0 & 0 \\ -0.5 & -0.5 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & -2 & 1 \end{bmatrix} \zeta + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \mu.$$

Since the system is linear, the optimal tracking problem reduces to an optimal regulation problem, which can be solved using the resulting Algebraic Riccati Equation. The optimal value function is given by $V^*\left(\zeta\right) = \zeta^T P_\zeta \zeta$, where the matrix P_ζ is given by

$$P_\zeta = \begin{bmatrix} 4.43 & 0.67 & \mathbf{0}_{2\times 2} \\ 0.67 & 2.91 & \\ \mathbf{0}_{2\times 2} & & \mathbf{0}_{2\times 2} \end{bmatrix}.$$

Using the matrix P_ζ, the ideal weighs corresponding to the selected basis can be computed as $W = [4.43, \ 1.35, \ 0, \ 0, \ 2.91, \ 0, \ 0]$.

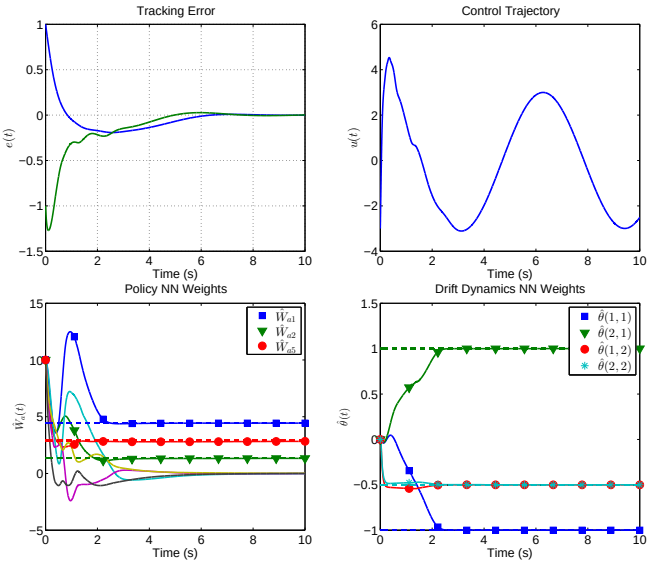

Fig. 1. System trajectories and policy weight trajectories and the unknown parameters in the system drift dynamics generated using the proposed method for the linear system. Dashed lines denote the ideal values for the policy and drift weights.

Figure 1 demonstrates that the controller remains bounded, the tracking error goes to zero, and the weight estimates $\hat{W}_c$, $\hat{W}_a$ and $\hat{\theta}$ go to their true values, establishing convergence of the approximate policy to the optimal policy. Figure 2 demonstrates satisfaction of the rank conditions in Assumptions 3 and 4.

[5] The learning gains, the basis functions for the NNs, and the points for BE extrapolation are selected using a trial and error approach. Alternatively, global optimization methods such as a genetic algorithm, or simulation-based methods such as a Monte-Carlo simulation can be used to tune the gains.

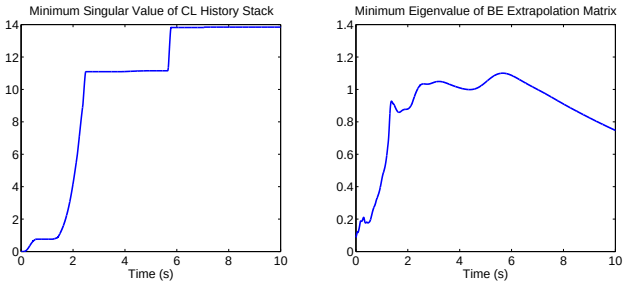

Fig. 2. Satisfaction of Assumptions 3 and 4 for the linear system.

IX. Conclusion

A concurrent-learning based implementation of model-based RL is developed to obtain an approximate online solution to infinite horizon optimal tracking problems for nonlinear continuous-time control-affine systems. The desired steady-state controller is used to facilitate the formulation of a feasible optimal control problem, and the system state is augmented with the desired trajectory to facilitate the formulation of a stationary optimal control problem. A CL-based system identifier is developed to remove the dependence of the desired steady-state controller on the system drift dynamics, and to facilitate simulation of experience via BE extrapolation. Simulation results are provided to demonstrate the effectiveness of the developed technique.

Similar to the PE condition in RL-based online optimal control literature, Assumption 4 can not, in general, be guaranteed a priori. However, Assumption 4 can be heuristically met by oversampling, i.e., by selecting $N \gg L$. Furthermore, unlike PE, the satisfaction of Assumption 4 can be monitored online; hence, threshold-based algorithms can be employed to preserve rank by selecting new points if the minimum singular value falls below a certain threshold. Provided the minimum singular value does not decrease during a switch, the trajectories of the resulting switched system can be shown to be uniformly bounded using a common Lyapunov function. Formulation of sufficient conditions for Assumption 4 that can be verified a priori is a topic for future research.

References

[1] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 1998.

[2] D. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed. Belmont, MA: Athena Scientific, 2007, vol. 2.

[3] K. Doya, "Reinforcement learning in continuous time and space," *Neural Comput.*, vol. 12, no. 1, pp. 219–245, 2000.

[4] Z. Chen and S. Jagannathan, "Generalized Hamilton-Jacobi-Bellman formulation -based neural network control of affine nonlinear discrete-time systems," *IEEE Trans. Neural Netw.*, vol. 19, no. 1, pp. 90–106, Jan. 2008.

[5] T. Dierks, B. Thumati, and S. Jagannathan, "Optimal control of unknown affine nonlinear discrete-time systems using offline-trained neural networks with proof of convergence," *Neural Netw.*, vol. 22, no. 5-6, pp. 851–860, 2009.

[6] K. Vamvoudakis and F. Lewis, "Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, no. 5, pp. 878–888, 2010.

[7] S. Bhasin, R. Kamalapurkar, M. Johnson, K. G. Vamvoudakis, F. L. Lewis, and W. E. Dixon, "A novel actor-critic-identifier architecture for approximate optimal control of uncertain nonlinear systems," *Automatica*, vol. 49, no. 1, pp. 89–92, Jan. 2013.

[8] H. Zhang, D. Liu, Y. Luo, and D. Wang, *Adaptive Dynamic Programming for Control Algorithms and Stability*, ser. Communications and Control Engineering. London: Springer-Verlag, 2013.

[9] D. Liu and Q. Wei, "Policy iteration adaptive dynamic programming algorithm for discrete-time nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 3, pp. 621–634, Mar. 2014.

[10] H. Modares, F. L. Lewis, and M.-B. Naghibi-Sistani, "Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems," *Automatica*, vol. 50, no. 1, pp. 193–202, 2014.

[11] X. Yang, D. Liu, and D. Wang, "Reinforcement learning for adaptive optimal control of unknown continuous-time nonlinear systems with input constraints," *Int. J. Control*, vol. 87, no. 3, pp. 553–566, 2014.

[12] H. Zhang, Q. Wei, and Y. Luo, "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.*, vol. 38, no. 4, pp. 937–942, 2008.

[13] T. Dierks and S. Jagannathan, "Optimal tracking control of affine nonlinear discrete-time systems with unknown internal dynamics," in *Proc. IEEE Conf. Decis. Control*, Shanghai, CN, Dec. 2009, pp. 6750–6755.

[14] H. Zhang, L. Cui, X. Zhang, and Y. Luo, "Data-driven robust approximate optimal tracking control for unknown general nonlinear systems using adaptive dynamic programming method," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2226–2236, Dec. 2011.

[15] Q. Wei and D. Liu, "Optimal tracking control scheme for discrete-time nonlinear systems with approximation errors," in *Advances in Neural Networks - ISNN 2013*, ser. Lecture Notes in Computer Science, C. Guo, Z.-G. Hou, and Z. Zeng, Eds. Springer Berlin Heidelberg, 2013, vol. 7952, pp. 1–10.

[16] B. Kiumarsi, F. L. Lewis, H. Modares, A. Karimpour, and M.-B. Naghibi-Sistani, "Reinforcement Q-learning for optimal tracking control of linear discrete-time systems with unknown dynamics," *Automatica*, vol. 50, no. 4, pp. 1167–1175, Apr. 2014.

[17] C. Qin, H. Zhang, and Y. Luo, "Online optimal tracking control of continuous-time linear systems with unknown dynamics by using adaptive dynamic programming," *Int. J. Control*, vol. 87, no. 5, pp. 1000–1009, 2014.

[18] R. Kamalapurkar, H. Dinh, S. Bhasin, and W. E. Dixon, "Approximate optimal trajectory tracking for continuous-time nonlinear systems," *Automatica*, vol. 51, pp. 40–48, Jan. 2015.

[19] R. Kamalapurkar, P. Walters, and W. E. Dixon, "Model-based reinforcement learning for approximate optimal regulation," *Automatica*, vol. 64, pp. 94–104, Feb. 2016.

[20] R. Kamalapurkar, J. Klotz, and W. E. Dixon, "Concurrent learning-based online approximate feedback Nash equilibrium solution of N-player nonzero-sum differential games," *IEEE/CAA J. Autom. Sin.*, vol. 1, no. 3, pp. 239–247, Jul. 2014.

[21] R. Kamalapurkar, L. Andrews, P. Walters, and W. E. Dixon, "Model-based reinforcement learning for infinite-horizon approximate optimal tracking," in *Proc. IEEE Conf. Decis. Control*, Los Angeles, CA, Dec. 2014, pp. 5083–5088.

[22] G. Chowdhary, "Concurrent learning adaptive control for convergence without persistencey of excitation," Ph.D. dissertation, Georgia Institute of Technology, Dec. 2010.

[23] G. V. Chowdhary and E. N. Johnson, "Theory and flight-test validation of a concurrent-learning adaptive controller," *J. Guid. Control Dynam.*, vol. 34, no. 2, pp. 592–607, Mar. 2011.

[24] D. Kirk, *Optimal Control Theory: An Introduction*. Mineola, NY: Dover, 2004.

[25] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal Control*, 3rd ed. Hoboken, NJ: Wiley, 2012.

[26] H. Modares and F. L. Lewis, "Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning," *Automatica*, vol. 50, no. 7, pp. 1780 – 1792, 2014.

[27] G. Chowdhary, T. Yucelen, M. Mühlegg, and E. N. Johnson, "Concurrent learning adaptive control of linear systems with exponentially convergent bounds," *Int. J. Adapt. Control Signal Process.*, vol. 27, no. 4, pp. 280–301, 2013.

[28] F. L. Lewis, S. Jagannathan, and A. Yesildirak, *Neural network control of robot manipulators and nonlinear systems*. Philadelphia, PA: CRC Press, 1998.

[29] R. Kamalapurkar, "Model-based reinforcement learning for online approximate optimal control," Ph.D. dissertation, University of Florida, 2014.

[30] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, NJ: Prentice Hall, 2002.